ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN THỊ THÙY ANH

NGHIÊN CỨU PHÁT TRIỂN CÁC KỸ THUẬT HỌC MÁY HOÀN THIỆN ĐỒ THỊ TRI THỨC

Ngành đào tạo : Hệ thống thông tin
Mã số : 9480104

LUẬN ÁN TIẾN SĨ HỆ THỐNG THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC:
1. PGS. TS. HÀ QUANG THỤY
2. PGS. TS. PHAN XUÂN HIẾU

Hà Nội - 2025

Lời cam đoan

Tôi xin cam đoan luận án này là công trình nghiên cứu của riêng tôi. Các kết quả được viết chung với các tác giả khác đều được sự đồng ý của các đồng tác giả trước khi đưa vào luận án. Các kết quả nêu trong luận án là trung thực và chưa từng được công bố trong các công trình nào khác.

Nghiên cứu sinh

Nguyễn Thị Thùy Anh

Lời cảm ơn

Thời gian học nghiên cứu sinh và thực hiện luận án tại Bộ môn Các Hệ thống thông tin - Khoa Công nghệ thông tin - Trường Đại học Công nghệ - Đại học Quốc gia Hà Nội dưới sự hướng dẫn khoa học của PGS.TS. Hà Quang Thụy và PGS.TS. Phan Xuân Hiếu là khoảng thời gian vô cùng quý báu và ý nghĩa đối với tôi.

Trước tiên tôi xin bày tỏ lòng biết ơn sâu sắc tới PGS.TS. Hà Quang Thụy và PGS.TS. Phan Xuân Hiếu, những người Thầy đã dẫn dắt tôi một cách nghiêm túc, truyền đạt các kinh nghiệm nghiên cứu trong quá trình thực hiện luận án và nhờ có sự động viên, chỉ bảo của các thầy đã tạo động lực cho tôi vượt qua được những khó khăn trong quá trình nghiên cứu để hoàn thành bản luận án này, bên cạnh đó cũng giúp tôi trưởng thành và tự tin hơn trên con đường nghiên cứu khoa học của mình.

Tôi xin bày tỏ lòng biết ơn tới các Thầy đã cho chúng tôi một môi trường làm việc hiệu quả tại Phòng thí nghiệm Khoa học dữ liệu và Công nghệ tri thức - DS&KTLab. Những buổi sinh hoạt chuyên môn của các thành viên trong DS&KTLab dưới sự dẫn dắt của các Thầy đã không chỉ giúp chúng tôi có được sự kết nối tri thức phong phú giữa các chủ đề, lĩnh vực nghiên cứu khác nhau, mà còn giúp chúng tôi tạo được những nhóm làm việc kết hợp với nhau để hình thành ra các ý tưởng nghiên cứu mới.

Tôi xin bày tỏ lòng cảm ơn chân thành tới các cộng sự NCS Vương Thị Hồng, Sinh viên Tạ Viết Phương, Sinh viên Lê Thị Hạnh, Sinh viên Lê Thị Thảo, Sinh viên Lê Công Hoàng của khoa CNTT trường Đại học Công nghệ, Đai học Quốc gia Hà Nội, sinh viên Nguyễn Công Hiếu Đại học Bách khoa Hà Nội đã hỗ trợ tôi thực hiện các công trình nghiên cứu. Tôi cũng không bao giờ quên sự sẻ chia kinh nghiệm và động viên từ những người anh chị của tôi TS. Bùi Thị Hồng Nhung, TS. Nguyễn Thị Chăm, TS. Nguyễn Thị Hồng Khánh, TS. Nguyễn Thị Ngân, NCS. Nguyễn Khánh Tùng và các bạn giảng viên trẻ tại DS&KTLab TS. Vương Thị Hải Yến, NCS. Nguyễn Thị Cẩm Vân, NCS. Phạm Thị Quỳnh Trang…

Tôi xin chân thành cảm ơn tới Ban lãnh đạo, tập thể các Thầy Cô giáo, các Nhà khoa học thuộc Trường Đại học Công nghệ - Đại học Quốc gia Hà Nội, các Thầy Cô trong hội đồng chuyên môn bộ môn đã đóng góp các ý kiến chuyên môn vô cùng xác đáng và quý báu để tôi có thể hiểu rõ hơn về nội dung nghiên cứu và hoàn thiện

tốt nhất bản luận án của mình cũng như tạo điều kiện thuận lợi cho tôi trong quá trình nghiên cứu. Cảm ơn các chuyên viên của Phòng đào tạo và các chuyên viên của khoa CNTT như các anh chị Dương Đình Thiệu, Nguyễn Thị Minh Thanh, Nguyễn Thị Lan Hương, Phạm Thị Mai Bảo đã luôn hỗ trợ tôi trong quá trình hoàn thiện hồ sơ bảo vệ.

Tôi cũng bày tỏ lòng cảm ơn sâu sắc tới Ban giám đốc Học viện Ngân hàng; Ban lãnh đạo Khoa Công nghệ thông tin và Kinh tế số đã tạo mọi điều kiện thuận lợi cho tôi trong quá trình nghiên cứu; Cảm ơn các đồng nghiệp trong Khoa đã luôn ủng hộ, quan tâm và động viên tôi.

Tôi xin được gửi lời cảm ơn tới người bạn đời TS. Đoàn Duy Trung đã luôn chia sẻ, động viên và tạo mọi điều kiện thuận lợi cho tôi tập trung thời gian nghiên cứu. Cảm ơn hai con Đoàn Thùy Dương và Đoàn Gia Linh đã luôn tự ý thức và tự giác trong công việc cá nhân để tôi yên tâm nghiên cứu. Bên cạnh đó, tôi cảm thấy may mắn và luôn biết ơn đại gia đình Nội, Ngoại hai bên đã luôn hỗ trợ tôi trong mọi công việc, động viên tôi cả tinh thần và vật chất để tôi có được thành quả như ngày hôm nay.

Mục lục

Danh mục thuật ngữ và viết tắt

Thuật ngữ tiếng Anh	*Thuật ngữ tiếng Việt*	*Viết tắt*
Thuật ngữ và viết tắt tiếng Việt		
Artificial Intelligence (AI)	Trí tuệ nhân tạo	TTNT
Knowledge Graph (KG)	Đồ thị tri thức	ĐTTT
Thuật ngữ và viết tắt tiếng Anh		
Adaptive Multi-modal Fusion	Hợp nhất đa phương thức thích ứng (mô đun thuộc AdaMF-MAT)	AdaMF
Adaptive Multi-modal Fusion and Modality Adversarial Training	Huấn luyện đối nghịch đa phương thức và hợp nhất thích ứng	AdaMF-MAT
Artificial Neural Network	Mạng nơ ron nhân tạo	ANN
Convolutional Neural Network	Mạng nơ ron tích chập	CNN
Collaborative Modality Adversarial Training	Mô hình huấn luyện phương thức cộng tác	CoMAT
Diachronic Embedding	Nhúng lịch đại	DE
Generative Adversarial Network	Mạng đối nghịch tạo sinh	GAN
Gated Recurrent Unit	Đơn vị hồi quy cổng	GRU
Image-embodied Knowledge Representation Learning	Học biểu diễn tri thức thể hiện ảnh (mô hình)	IKRL
Knowledge-guided Cross-modal Attention	Chú ý xuyên phương thức hướng tri thức (mô hình)	KCA
Knowledge Graph Embedding	Nhúng ĐTTT	KGE
Knowledge Graph Completion	Hoàn thiện ĐTTT	KGC

Long Short-Term Memory	Bộ nhớ ngắn-dài hạn	LSTM
Modality-Aware Negative Sampling	Lấy mẫu âm theo nhận thức phương thức (phương pháp)	MANS
Modality Adversarial Training	Huấn luyện đối nghịch phương thức (mô đun thuộc AdaMF-MAT)	MAT
Masked Language Model	Mô hình ngôn ngữ bị che	MLM
Multilayer Percepton	Percepton đa lớp (mạng nơ ron)	MLP
Multi-Modal Knowledge Graph	ĐTTT đa phương thức	MMKG
MultiModal Relation-enhanced Negative Sampling	Lấy mẫu âm tăng cường quan hệ đa phương thức (phương pháp)	MMRNS
Mean Rank	Hạng trung bình (độ đo)	MR
Mean Reciprocal Rank	Hạng tương hỗ trung bình (độ đo)	MRR
Numeric, Audio, Text, Image, Video, and morE	Số, Âm thanh, Văn bản, Hình ảnh, Video, v.v. (mô hình)	NativE
Natural Language Processing	Xử lý ngôn ngữ tự nhiên	NLP
Next Sentence Prediction	Dự đoán câu kế tiếp	NSP
Open Knowledge Graph	ĐTTT mở	OKG
Positive Point-wise Mutual Information	Thông tin tương quan từng điểm dương (phương pháp nhúng từ)	PPMI
Recurrent Neural Network	Mạng nơ ron hồi quy	RNN
Relation-guided Dual Adaptive Fusion	Kết hợp thích ứng kép hướng quan hệ (mô đun trong NativE)	ReDAF
Skip-gram with Negative Sampling	Skip-gram với lấy mẫu âm (phương pháp nhúng từ)	SGNS

Singular Value Decomposition	Phân tích giá trị đơn (phương pháp nhúng từ)	SVD
Translation-Based approach for Knowledge Graph Completion	Tiếp cận dựa trên dịch cho hoàn thiện đồ thị tri thức	TBKGC
Temporal Knowledge Graph Completion	Hoàn thiện đồ thị tri thức thời gian	TKGC
	Nhúng thực thể và quan hệ trong ĐTTT	TransE
VisualBERT-enhanced Knowledge Graph Completion	Hoàn thiện đồ thị tri thức đa phương thức được tăng cường BERT trực quan (mô hình)	VBKGC
Thuật ngữ không có viết tắt		
Relationship Instance/Fact	Thể hiện quan hệ/Dữ kiện	
Diachronic	Lịch đại	
Adversarial learning	Học đối nghịch	
Constrastive learning	Học tương phản	
Entity discovery	Phát hiện thực thể	
Knowledge acquisition	Thu thập tri thức	
Knowledge-aware applications	Các ứng dụng nhận thức tri thức	
Knowledge representation learning	Học biểu diễn tri thức	
Relation extraction	Trích xuất quan hệ	

Bảng ký hiệu Toán học

Ký hiệu	*Ý nghĩa*
$\|\|.\|\|_p$	Chuẩn L_p, $p = 1,2$ tùy thuộc vào từng mô hình
$\mathbb{R}^d$	Không gian thực d chiều
$\mathbb{C}^d$	Không gian phức d chiều
$*$	Toán tử tích chập.
$\circ$	Biểu thị phép nhân theo phần tử.
v^T	Véc-tơ chuyển vị của véc-tơ v
$\oplus$	Tích từng điểm (pointwise product)

Danh mục các bảng

Danh mục các hình vẽ

Mở đầu

Đồ thị tri thức (*ĐTTT*, Knowledge graph: *KG*) đóng vai trò quan trọng đặc biệt, cung cấp thông tin ngữ nghĩa phong phú và đã nổi lên như một động lực thúc đẩy các lĩnh vực tìm kiếm ngữ nghĩa và Trí tuệ nhân tạo (TTNT, Artificial Intelligence: *AI*) [1]. Như J. Barrasa và cộng sự nhận định, "*ĐTTT đang thay đổi* TTNT *bằng cách cung cấp ngữ cảnh. Điều đó dẫn đến khả năng giải thích, đa dạng hóa và cải thiện quá trình xử lý. Nếu* TTNT *đang thay đổi tương lai và* ĐTTT *đang thay đổi* TTNT*, thì theo tính chất bắc cầu,* ĐTTT *cũng đang thay đổi tương lai. Gần đây, sự quan tâm đến* ĐTTT *đã bùng nổ, với vô số các bài báo nghiên cứu giải pháp, báo cáo của nhà phân tích, nhóm và hội nghị"* và *"*ĐTTT *trở nên phổ biến một phần vì công nghệ đồ thị đã tăng tốc trong những năm gần đây nhưng cũng vì nhu cầu mạnh mẽ về việc hiểu dữ liệu*" [2]. Nói riêng, ĐTTT là một công cụ quan trọng cho học máy giải thích được (explainable machine learning) cũng như TTNT giải thích được (explainable artificial intelligence) [3], [4], [5]. Trong khu vực công nghiệp, ĐTTT rất quan trọng đối với nhiều doanh nghiệp ngày nay: Chúng cung cấp dữ liệu có cấu trúc và tri thức thực tế thúc đẩy nhiều sản phẩm và khiến các sản phẩm này trở nên thông minh và kỳ diệu hơn [6]. Một báo cáo của MarketsandMarkets vào tháng 01/2025 cho biết là quy mô thị trường ĐTTT đạt 1068 triệu đô la Mỹ vào năm 2024 và dự kiến đạt 6938 triệu đô la Mỹ vào năm 2030[1]. Thuật ngữ Cơ sở tri thức (Knowledge base) và thuật ngữ ĐTTT có thể được sử dụng thay thế cho nhau [7].

Các đơn vị tri thức cơ bản trong ĐTTT gồm các đỉnh biểu diễn các thực thể và các cạnh biểu diễn quan hệ giữa các thực thể đó. Cụ thể hơn, các quan hệ này được thể hiện dưới dạng bộ ba dữ kiện (thực thể đầu, quan hệ, thực thể đuôi). Đồ thị tri thức là sự kết hợp tổng hòa các thực thể, các quan hệ bộ ba, và cấu trúc đồ thị mang tính lan truyền, giàu tính dự đoán và suy luận. Sự kết hợp này chứa đựng rất nhiều thông tin, đặc trưng hữu ích cho các vấn đề đoán nhận, suy diễn. Nói một cách cụ thể hơn, ĐTTT là một biểu diễn có cấu trúc đồ thị các dữ kiện, bao hàm các thực thể, các quan hệ và các mô tả ngữ nghĩa [7]. Một cách hình thức, ĐTTT được biểu diễn dưới

[1] https://www.marketsandmarkets.com/Market-Reports/knowledge-graph-market-217920811.html

dạng một bộ ba $KG = (E, R, F)$, trong đó E là tập hữu hạn các thực thể (tập nút), R là tập hữu hạn các quan hệ (tập nhãn của các cung), còn F là một tập các thể hiện quan hệ (tập cung) $F \subseteq E \times R \times E$; một cung trong F được biểu diễn dưới dạng bộ ba (h, r, t) trong đó $h, t \in E$ và còn $r \in R$; trong cung (h, r, t), h được gọi là nút đầu, t được gọi là nút cuối và r được gọi là nhãn của cung. Trong nhiều trường hợp, "thể hiện quan hệ" (relationship instance) còn được gọi là "dữ kiện" (fact) trong ĐTTT.

Gần đây, nhu cầu *cung cấp ngữ cảnh có tính thời gian* đã trở nên hết sức cấp thiết, từ đó hình thành một kiểu ĐTTT mới - ĐTTT thời gian (Temporal Knowledge Graph: TKG) với sự bổ sung yếu tố thời gian vào tập các dữ kiện F trong ĐTTT thông thường. Một cung trong ĐTTT thời gian được biểu diễn dưới dạng một bộ bốn (h, r, t, τ) khi bổ sung vào cung (h, r, t) một giá trị thời gian τ, với τ là một giá trị theo một bộ đếm tem thời gian (timestampt) hoặc theo lịch đại (Diachronic).

Bốn nhóm chủ đề nghiên cứu chính (bài toán lớn) về ĐTTT là Học biểu diễn tri thức (Knowledge representation learning), Thu thập tri thức (Knowledge acquisition), ĐTTT thời gian (Temporal knowledge graph) và Các ứng dụng nhận thức tri thức (Knowledge-aware applications) [7]. Mỗi nhóm chủ đề nghiên cứu chính này lại được chia thành các chủ đề nghiên cứu cụ thể hơn, chẳng hạn, nhóm chủ đề Học biểu diễn tri thức bao gồm bốn chủ đề là Không gian biểu diễn, Hàm chấm điểm, Mô hình mã hóa và Nhúng với thông tin hỗ trợ.

Mặc dù ĐTTT có giá trị rất lớn trong nhiều ứng dụng, nhưng chúng thường được xây dựng thủ công hoặc bán tự động nên vẫn không đầy đủ vì một lượng lớn tri thức quý giá tồn tại ngầm hoặc bị bỏ sót dẫn tới ĐTTT hoạt động kém đi nhiều trong các ứng dụng thực tế [8]. Vì vậy, bài toán Thu thập tri thức với mục tiêu là giải quyết các vấn đề về tính không đầy đủ và thưa thớt trong ĐTTT có tầm quan trọng đặc biệt. Thu thập tri thức bao gồm ba bài toán con để giải quyết ba trường hợp chưa đầy đủ điển hình của ĐTTT là: (i) Phát hiện thực thể (entity discovery) để giải quyết tình huống tập thực thể E là chưa đầy đủ, (ii) Trích xuất quan hệ (relation extraction) để giải quyết tình huống tập quan hệ R là chưa đầy đủ, (iii) Hoàn thiện ĐTTT (Knowledge Graph Completion - KGC) để giải quyết tình huống tập dữ kiện F là chưa đầy đủ (trong trường hợp này, giả thiết rằng tập thực thể E và tập quan hệ R là đã đầy đủ) [7]. Một cách cụ thể hơn, bài toán Hoàn thiện ĐTTT cần xác định các bộ ba (h, r, t) có tồn tại thực sự hay không khi đã biết được hai thành phần trong các bộ

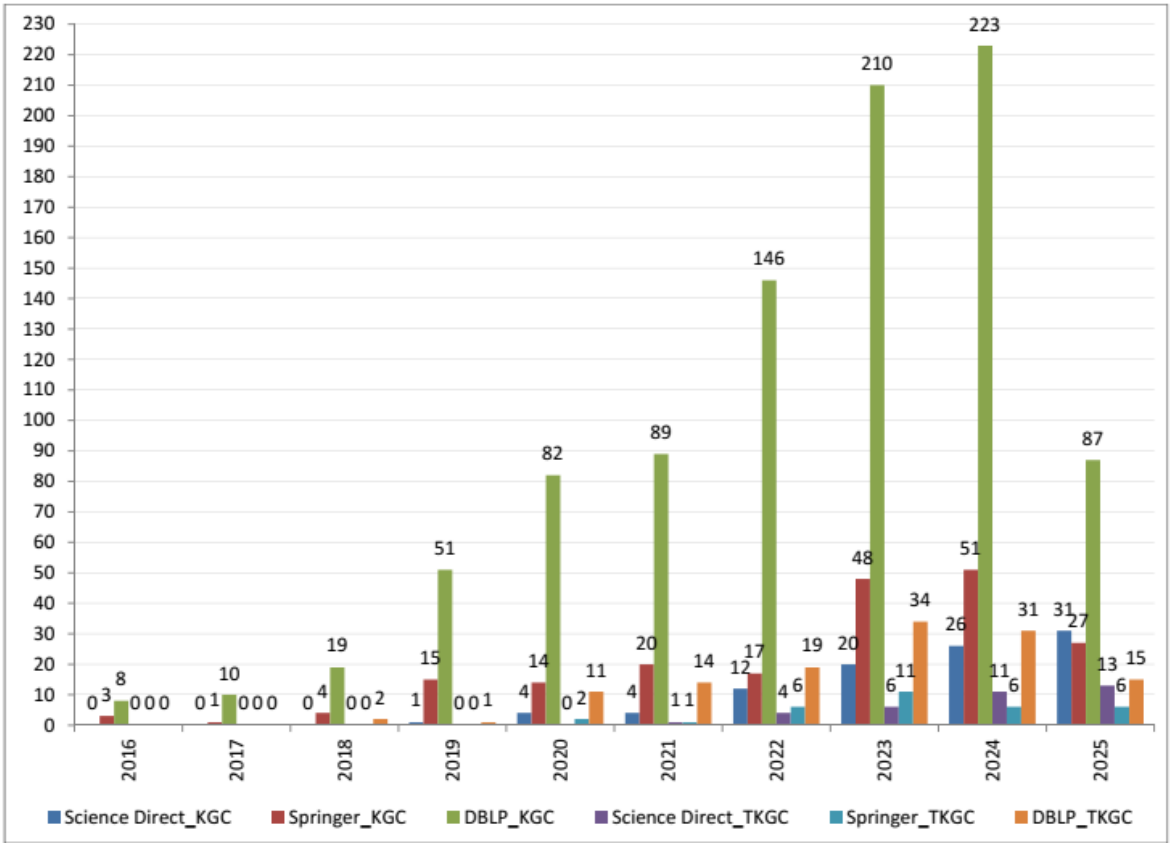
ba đó theo một trong ba dạng: $(?, r, t)$, $(h, r, ?)$ hoặc $(h, ?, t)$, trong đó, "?" chỉ thành phần còn bị thiếu. Một cách tương ứng, bài toán Hoàn thiện ĐTTT thời gian (Temporal Knowledge Graph Completion: TKGC) giải quyết vấn đề về sự tồn tại của các bộ bốn (h, r, t, τ) khi thiếu một thành phần trong bộ bốn này.

Theo T. Shen và cộng sự [9], quy trình hoàn thiện ĐTTT gồm ba khối chính là Huấn luyện mô hình, Xử lý ứng viên và Xác định dữ kiện. Đồng thời, hệ thống kỹ thuật hoàn thiện ĐTTT được phân loại thành ba nhóm chính là: (i) Kỹ thuật dựa trên thông tin cấu trúc/bộ ba theo các mô hình khớp ngôn ngữ và mô hình dịch, (ii) Kỹ thuật dựa trên thông tin bổ sung từ bên trong hoặc bên ngoài ĐTTT, và (iii) các kỹ thuật hoàn thiện ĐTTT khác. Theo xu thế chung tăng trưởng các nghiên cứu về ĐTTT, nghiên cứu về Hoàn thiện ĐTTT cũng có sự tăng trưởng vượt bậc trong khoảng mười năm trở lại đây.

Hình 0.1 cho một minh họa về số lượng các công trình khoa học có tiêu đề liên quan tới các cụm từ "Knowledge Graph Completion" và "Temporal Knowledge Graph Completion" được công bố trên Science Direct, Springer và DBLP. Hình 0.1 cho biết Hoàn thiện ĐTTT là một chủ đề nghiên cứu ngày càng được quan tâm trong thời gian gần đây, đồng thời Hoàn thiện ĐTTT thời gian là một chủ đề nghiên cứu chỉ mới xuất hiện từ năm 2018 trở lại đây. Thêm nữa, tìm kiếm trên Science Direct, Springer và DBLP, luận án nhận thấy rằng công bố khoa học có tiêu đề liên quan tới các cụm từ "Multi modal Knowledge Graph Completion" hoặc "Multimodal Knowledge Graph Completion" mới chỉ xuất hiện từ năm 2022 trở lại đây[2]. Điều đó có nghĩa là nghiên cứu phát triển các kỹ thuật học máy Hoàn thiện ĐTTT, Hoàn thiện ĐTTT đa phương thức, Hoàn thiện ĐTTT thời gian là rất cần thiết và có tính thời sự.

S. Ji và cộng sự [7] nhận định rằng hoàn thiện ĐTTT bao gồm các chủ đề nghiên cứu rất quan trọng về suy luận đường dẫn quan hệ, lập luận dựa trên luật, phân lớp bộ ba, đánh giá tính đúng đắn của bộ ba dữ kiện.

[2] https://dblp.dagstuhl.de/search?q=Multi+modal+Knowledge+Graph+Completion.



Hình 0.1. Một thống kê vào tháng 6/2025 về số công bố khoa học có liên quan tới Hoàn thiện ĐTTT (KGC) và Hoàn thiện ĐTTT thời gian (TKGC) trên Science Direct, Springer và DBLP

Qua phân tích về các khoảng trống nghiên cứu, T. Shen và cộng sự [9] đưa ra tám xu hướng nghiên cứu về hoàn thiện ĐTTT là: (1) Sử dụng thông tin bổ sung; (2) Dựa trên luật; (3) Sử dụng mô hình ngôn ngữ huấn luyện trước; (4) Miền ứng dụng cụ thể; (5) Nắm bắt tương tác giữa các ĐTTT cụ thể; (6) Lựa chọn không gian mô hình hóa; (7) Sử dụng học tăng cường và (8) Sử dụng học đa nhiệm. Giải quyết các thách thức từ các chủ đề nghiên cứu quan trọng cũng như tám triển vọng nghiên cứu về hoàn thiện ĐTTT trên đây là mục tiêu nghiên cứu của một số luận án Tiến sỹ trên thế giới, điển hình là [10], [11], [12], [13], [14].

Định hướng tới các chủ đề nghiên cứu quan trọng và xu hướng nghiên cứu về hoàn thiện ĐTTT như đã được đề cập, theo dòng nghiên cứu của các luận án Tiến sỹ trên thế giới, **luận án** đặt ra hai **câu hỏi nghiên cứu** với **mục tiêu** nghiên cứu tương ứng như sau:

- Câu hỏi nghiên cứu đầu tiên là về phương pháp và nguồn tài nguyên cho phép sử dụng thông tin bổ sung cho Hoàn thiện ĐTTT, tương ứng với xu hướng nghiên cứu (1). Luận án hướng mục tiêu đề xuất được các mô hình hoàn thiện ĐTTT bằng kỹ thuật học chuyển giao (transfer learning) để sử dụng thông tin bổ sung bên ngoài làm giàu thông tin bên trong cho hoàn thiện ĐTTT.
- Câu hỏi nghiên cứu thứ hai là về các kỹ thuật và mô hình Hoàn thiện ĐTTT trong các miền ứng dụng cụ thể có sử dụng mô hình ngôn ngữ huấn luyện trước, tương ứng với hai xu hướng nghiên cứu (3) và (4). Luận án hướng mục tiêu đề xuất được các kỹ thuật và mô hình hoàn thiện ĐTTT thời gian và hoàn thiện ĐTTT đa phương thức.

Đối tượng nghiên cứu của luận án là các mô hình hoàn thiện ĐTTT và các kỹ thuật được áp dụng trong các mô hình đó.

Phạm vi nghiên cứu của luận án tập trung tới các kỹ thuật và giải pháp cải tiến được áp dụng trong các mô hình hoàn thiện ĐTTT, cụ thể là Kỹ thuật học chuyển giao trên miền dữ liệu ĐTTT mở OlpBench, ReVerb45K, ReVerb20K, FB15K237, WN18RR; Kỹ thuật học đối nghịch (adversarial training) trên tập dữ liệu đa phương thức DB15K, MKG-W, MKG-Y; Kỹ thuật nhúng lịch đại (diachronic embedding) trên các miền dữ liệu về thời gian ICEWS14, ICEWS05-15, GDELT.

Luận án sử dụng **phương pháp nghiên cứu kết hợp** bao gồm:

- Phương pháp nghiên cứu định tính: Luận án tiến hành việc phân tích định tính (theo tiếp cận đọc – suy nghĩ – giải thích) các khái niệm và mô hình từ hệ thống tài liệu liên quan, tập trung vào các tài liệu nghiên cứu tổng quan (chẳng hạn như [15], [16], [7], [2], [9]) và các tài liệu nghiên cứu chuyên sâu phù hợp với các mục tiêu nghiên cứu của luận án, thông qua đó, đề xuất các kỹ thuật và mô hình hoàn thiện ĐTTT mới.
- Phương pháp nghiên cứu định lượng: Luận án tiến hành các nghiên cứu định lượng thông qua việc triển khai các hệ thống thực nghiệm tương ứng, tiến hành các kịch bản thực nghiệm để đánh giá hiệu năng của các kỹ thuật và mô hình đề xuất để kiểm chứng, đánh giá kết quả đối với các đề xuất của luận án.

Tham gia vào dòng nghiên cứu trên thế giới về hoàn thiện ĐTTT, luận án có ba đóng góp chính sau đây:

- Đề xuất mô hình hoàn thiện ĐTTT qua học chuyển giao BERT/FastText-GRU-KGC được cải tiến từ mô hình GloVe-GRU-KGC do V. Kocijan và T. Lukasiewicz đề xuất năm 2021 [17] theo ý tưởng khai thác tài nguyên nguồn từ mô hình ngôn ngữ BERT/FastText thay cho GloVe và cải tiến thành phần bộ mã hóa từ mô hình gốc GloVe-GRU-KGC. Triển khai thực nghiệm đánh giá hiệu năng của mô hình BERT/FastText-GRU-KGC cho thấy ý tưởng cải tiến là hợp lý khi đối sánh với mô hình GloVe-GRU-KGC gốc. Kết quả nghiên cứu này của luận án được công bố trong [AnhNTT1].
- Đề xuất hai mô hình hoàn thiện ĐTTT đa phương thức ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT được cải tiến từ mô hình AdaMF-MAT được Y. Zhang và cộng sự đề xuất năm 2024 [18], trong đó, ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT sử dụng mô hình nhúng ảnh Vision Transformer (ViT) và mô hình nhúng văn bản T5. Triển khai thực nghiệm đánh giá hiệu năng của mô hình ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT cho thấy ý tưởng cải tiến là hợp lý khi đối sánh với mô hình AdaMF-MAT gốc. Kết quả nghiên cứu này của luận án được công bố trong [AnhNTT4].
- Đề xuất hai mô hình hoàn thiện ĐTTT thời gian là DE-RotatE và DE-RotatE-sinc được cải tiến từ mô hình DE-SimplE do R. Goel và cộng sự đề xuất năm 2020 [19] theo ý tưởng sử dụng phép quay chiếu thay vì sử dụng phép tịnh tiến chiếu trong DE-SimplE. Triển khai thực nghiệm đánh giá hiệu năng của mô hình DE-RotatE và DE-RotatE-sinc cho thấy ý tưởng cải tiến là hợp lý khi đối sánh với mô hình DE-SimplE gốc. Kết quả nghiên cứu này của luận án được công bố trong [AnhNTT2, AnhNTT3].

Bố cục của luận án gồm phần Mở đầu và bốn chương nội dung, phần Kết luận và danh mục các tài liệu tham khảo. Nội dung chính của bốn chương của luận án được mô tả sơ bộ như sau:

- ***Chương 1*** trình bày một cách hệ thống về ĐTTT và hoàn thiện ĐTTT. Đối với ĐTTT bao gồm: khái niệm ĐTTT, phân loại ĐTTT (ĐTTT mở, ĐTTT đa phương thức, ĐTTT thời gian). Tiếp theo, luận án trình bày về hoàn thiện ĐTTT bao gồm: khái niệm hoàn thiện ĐTTT, quy trình chung cho hoàn thiện ĐTTT, một số mô hình hoàn thiện ĐTTT, hoàn thiện ĐTTT đa phương thức, hoàn thiện ĐTT thời gian. Tiếp theo, luận án trình bày về các độ đo đánh giá

hiệu năng của một mô hình hoàn thiện ĐTTT. Cuối cùng là mối liên hệ của các nghiên cứu trong luận án với một số chủ đề liên quan về hoàn thiện ĐTTT.

- ***Chương 2*** trình bày mô hình hoàn thiện ĐTTT qua học chuyển giao BERT/FastText-GRU-KGC của luận án, được cải tiến từ mô hình GloVe-GRU-KGC do V. Kocijan và T. Lukasiewicz đề xuất năm 2021 [17]. Chương này được bắt đầu bằng việc giới hiệu về Học chuyển giao, tiếp đó là việc phân tích mô hình GloVe-GRU-KGC. Phần quan trọng của chương là đề xuất mô hình BERT/FastText-GRU-KGC được xuất phát từ ý tưởng sử dụng BERT/FastText thay cho GloVe và cải tiến thành phần mã hóa trong GloVe-GRU-KGC. Các thành phần thuộc BERT/FastText-GRU-KGC được giới thiệu chi tiết, các thực nghiệm đánh giá hiệu năng của BERT/FastText-GRU-KGC được trình bày cụ thể và được đối sánh với GloVe-GRU-KGC để minh chứng sự hợp lý của đề xuất cải tiến.
- ***Chương 3*** trình bày hai mô hình hoàn thiện ĐTTT đa phương thức ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT do luận án đề xuất, dựa trên việc cải tiến mô hình AdaMF-MAT được Y. Zhang và cộng sự đề xuất năm 2024 [18]. Đầu tiên, chương này giới thiệu một vài mô hình hoàn thiện ĐTTT đa phương thức mới được công bố gần đây và đề cập tới hạn chế tính mất cân bằng của dữ liệu đa phương thức. Tiếp đó, mô hình AdaMF-MAT được trình bày. Phần quan trọng của chương là đề xuất hai mô hình ViT-AdaMF-MAT/T5-ViT-AdaMF-MAT dựa trên các ý tưởng cải tiến mô hình AdaMF-MAT. Các thành phần thuộc T5-ViT-AdaMF-MAT được giới thiệu chi tiết, các thực nghiệm đánh giá hiệu năng của ViT-AdaMF-MAT/T5-ViT-AdaMF-MAT được trình bày cụ thể và được đối sánh với AdaMF-MAT để minh chứng sự hợp lý của đề xuất cải tiến.
- ***Chương 4*** trình bày đề xuất của luận án về hai mô hình hoàn thiện ĐTTT thời gian là DE-RotatE và DE-RotatE-sinc được cải tiến từ mô hình DE-SimplE do R. Goel và cộng sự đề xuất năm 2020 [19]. Chương này bắt đầu bằng việc giới thiệu khái quát về nhúng lịch đại ĐTTT thời gian, tiếp đó, mô hình DE-SimplE được trình bày chi tiết. Phần quan trọng của chương là đề xuất hai mô hình DE-RotatE và DE-RotatE-sinc dựa trên ý tưởng sử dụng phép quay thay thế cho phép tịnh tiến trong DE-SimplE. Các thành phần trong mô hình DE-RotatE và DE-RotatE-sinc được giới thiệu chi tiết, các thực nghiệm đánh giá hiệu

năng của các mô hình này được trình bày cụ thể và được đối sánh với mô hình gốc DE-SimplE để minh chứng sự hợp lý của đề xuất cải tiến.

Chương 1. Đồ thị tri thức và hoàn thiện đồ thị tri thức

Chương đầu tiên của luận án trình bày một cách hệ thống về ĐTTT và hoàn thiện ĐTTT cùng với các nội dung liên quan. Mục đầu tiên cung cấp một cách khái quát về ĐTTT, bao gồm một khung phân loại các chủ đề nghiên cứu về ĐTTT. Hoàn thiện ĐTTT được trình bày khái quát trong mục thứ hai. Hai mục tiếp theo giới thiệu về hai kiểu ĐTTT đặc biệt (ĐTTT thời gian và ĐTTT đa phương thức) và hoàn thiện chúng. Các độ đo đánh giá hiệu năng của các mô hình hoàn thiện ĐTTT được đề cập trong mục tiếp đó. Mục thứ sáu giới thiệu về các xu hướng nghiên cứu điển hình về hoàn thiện ĐTTT. Mục cuối cùng của chương trình bày một cách sơ bộ về mối liên hệ của các nghiên cứu trong luận án với một số xu hướng nghiên cứu về hoàn thiện ĐTTT.

1.1 Đồ thị tri thức

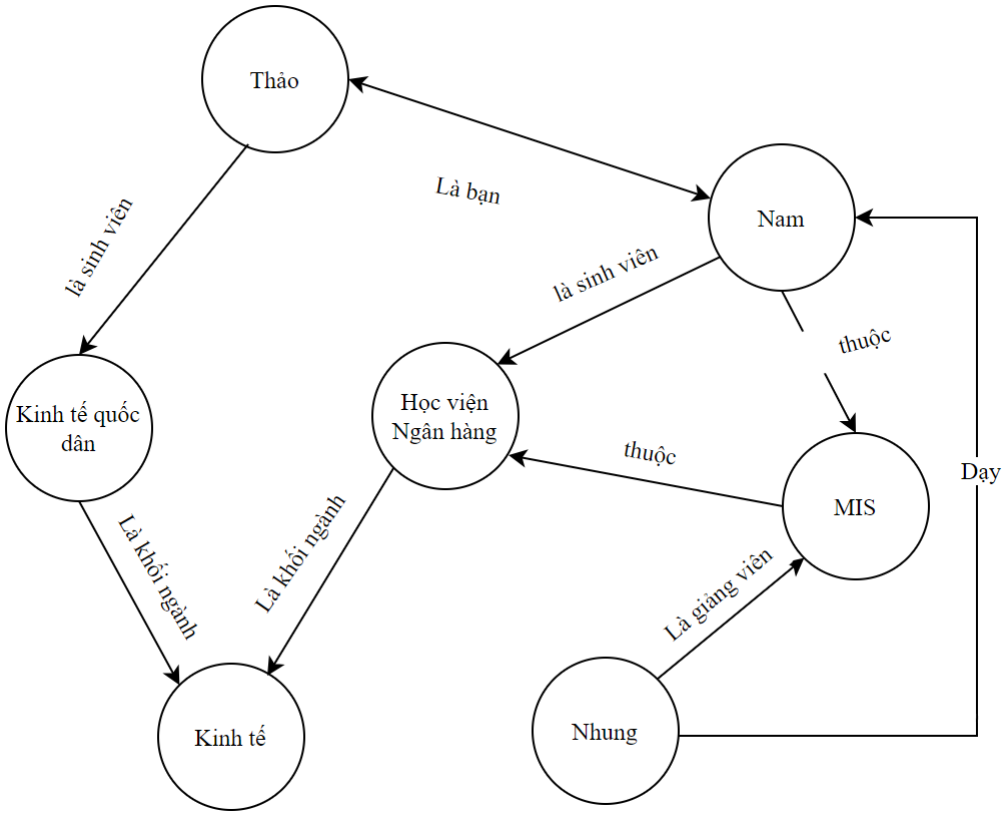
1.1.1 Khái niệm về đồ thị tri thức

Như đã được đề cập, ĐTTT là một biểu diễn có cấu trúc các dữ kiện, bao gồm các thực thể, các quan hệ và các mô tả ngữ nghĩa [7]. ĐTTT là một định dạng có cấu trúc về tri thức con người, đang thu hút rất nhiều sự quan tâm trong cả khoa học và công nghiệp [1], [20], [21], [22].

Định nghĩa 1.1 ĐTTT [7].

> *ĐTTT là một bộ ba $KG = (E, R, F)$, trong đó E là tập hữu hạn các thực thể (tập nút), R là tập hữu hạn các quan hệ (tập nhãn của các cung), còn F là một tập các thể hiện quan hệ (tập cung), $F \subseteq E \times R \times E$; một cung trong F được biểu diễn dưới dạng bộ ba (h, r, t) trong đó $h, t \in E$ và còn $r \in R$; trong cung (h, r, t), h được gọi là nút đầu, t được gọi là nút cuối và r được gọi là nhãn.*

Trong nhiều trường hợp, thể hiện quan hệ trong ĐTTT còn được gọi là "dữ kiện" (fact). Hình 1.1 cho một ví dụ minh họa về ĐTTT.



Hình 1.1 Ví dụ về ĐTTT.

1.1.2 Phân loại ĐTTT

Theo Liang và cộng sự [23], ĐTTT có thể được phân chia làm các loại khác nhau phụ thuộc vào đặc điểm, tính chất, đặc tính của các thực thể và quan hệ trong đó, trong đó, ba loại ĐTTT điển hình là:

- ĐTTT mở (Open Knowledge Graph, OKG).
- ĐTTT đa phương thức (Multi-Modal Knowledge Graph – MMKG).
- ĐTTT thời gian (Temporal Knowledge Graph – TKG).

Ngoài ra, theo Liang và cộng sự [23], hiện nay có một số kiểu ĐTTT mới dựa trên sự giao thoa của ba loại ĐTTT trên.

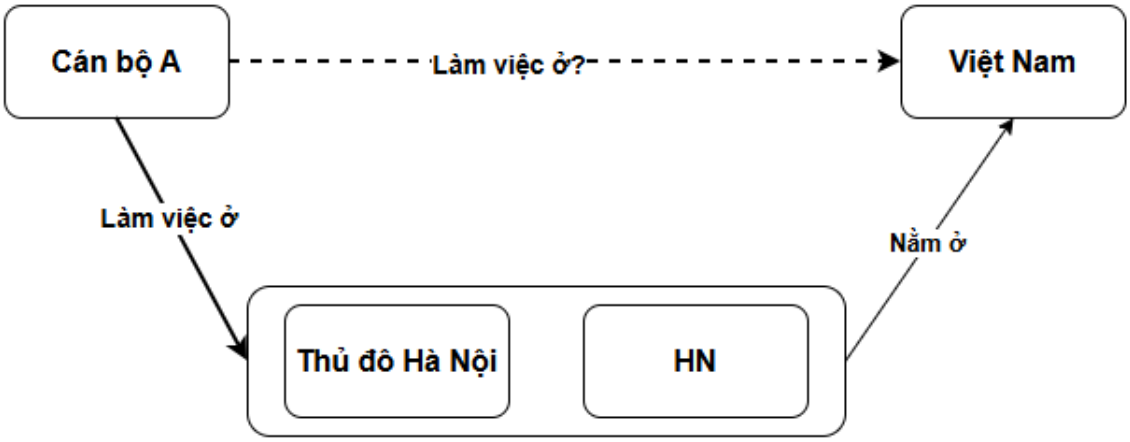
A. ĐTTT mở

Theo S. Gupta và cộng sự [24], khá nhiều ĐTTT hiện nay chưa được chuẩn hóa (uncanonicalized), nghĩa là các ĐTTT này chứa các thực thể chưa được chuẩn hóa (uncanonicalized) hoặc các quan hệ chưa được chuẩn hóa (uncanonicalied relations). Hơn nữa, trong thế giới thực, một thực thể hoặc một quan hệ có thể được

biểu diễn ở các dạng khác nhau. Do đó, quá trình trích xuất tự động từ các văn bản sẽ sinh ra các thực thể không được chuẩn hóa và quan hệ không được chuẩn hóa. Một ví dụ cụ thể như dưới đây:

- Ví dụ về thực thể không được chuẩn hóa: một thực thể như "Thủ đô Hà Nội" có thể xuất hiện dưới dạng "Hà Nội" hoặc "HN", thực thể "Việt Nam"" có thể xuất hiện dưới dạng "VN".
- Ví dụ về quan hệ không được chuẩn hóa: một quan hệ "nằm ở" có thể xuất hiện dưới dạng "thuộc vào".

Các lớp ĐTTT này được gọi là ĐTTT mở (OKC), Hình 1.2 cho một minh họa về ĐTTT. Trong Hình 1.2, các thực thể "Thủ đô Hà Nội" và "HN" có thể bị hiểu là hai thực thể riêng biệt. Chính vì vậy, bộ ba (Cán bộ A, làm việc ở, Việt Nam) có thể không xác định được là bộ ba đúng. Trên thực thế, đây là một bộ ba đúng vì hai thực thể "Thủ đô Hà Nội" và "HN" phải được coi là đồng nhất.



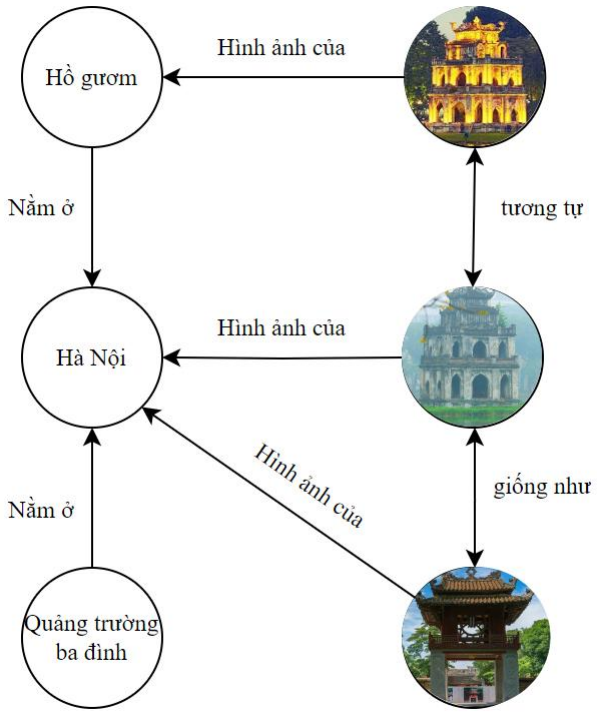
Hình 1.2. Ví dụ về một ĐTTT mở.

B. ĐTTT đa phương thức

ĐTTT đa phương thức (Multi-Modal Knowledge Graphs - MMKG) là một sự mở rộng của ĐTTT truyền thống thông qua việc tích hợp thêm các loại dữ liệu đa phương thức như ảnh, văn bản mô tả, hoặc các dạng dữ liệu khác ngoài văn bản thuần túy. Trong khi các ĐTTT thông thường biểu diễn tri thức bằng các bộ ba dưới dạng (thực thể đầu, quan hệ, thực thể đuôi) để mô tả các mối liên kết giữa các thực thể, ĐTTT đa phương thức mở rộng biểu diễn này bằng cách liên kết các thực thể với những nguồn dữ liệu mang tính vật lý và cảm quan hơn [25]. Cụ thể, ĐTTT đa phương

thức không chỉ lưu trữ các quan hệ định danh giữa các thực thể mà còn gắn kèm các thông tin bổ sung như ảnh, mô tả ngôn ngữ tự nhiên, hoặc các thuộc tính số nhằm tăng cường mức độ hiểu biết và biểu đạt tri thức. Nghĩa là, một ĐTTT đa phương thức có thể chứa các thông tin sốchi tiết và đồng thời cung cấp các liên kết trực tiếp đến hình ảnh minh họa cho mỗi thực thể. Việc kết hợp nhiều loại dữ liệu như vậy giúp các thực thể và quan hệ trong ĐTTT trở nên giàu ngữ nghĩa hơn, phản ánh tốt hơn đặc điểm của thế giới thực. Một số ví dụ:

- Thực thể "Ngày Giải phóng Miền Nam" có thể liên kết với hình ảnh của "Xe tăng Quân Giải phóng tiến vào Dinh Độc Lập", mô tả văn bản về "Ngày Giải phóng", mô tả video về ngày Giải phóng.
- Thực thể "Hà Nội" có thể liên kết với các ảnh đặc trưng như Tháp Rùa, ảnh Chùa Một Cột, các thực thể văn bản khác liên quan như Hồ gươm, Quảng trường Ba Đình.

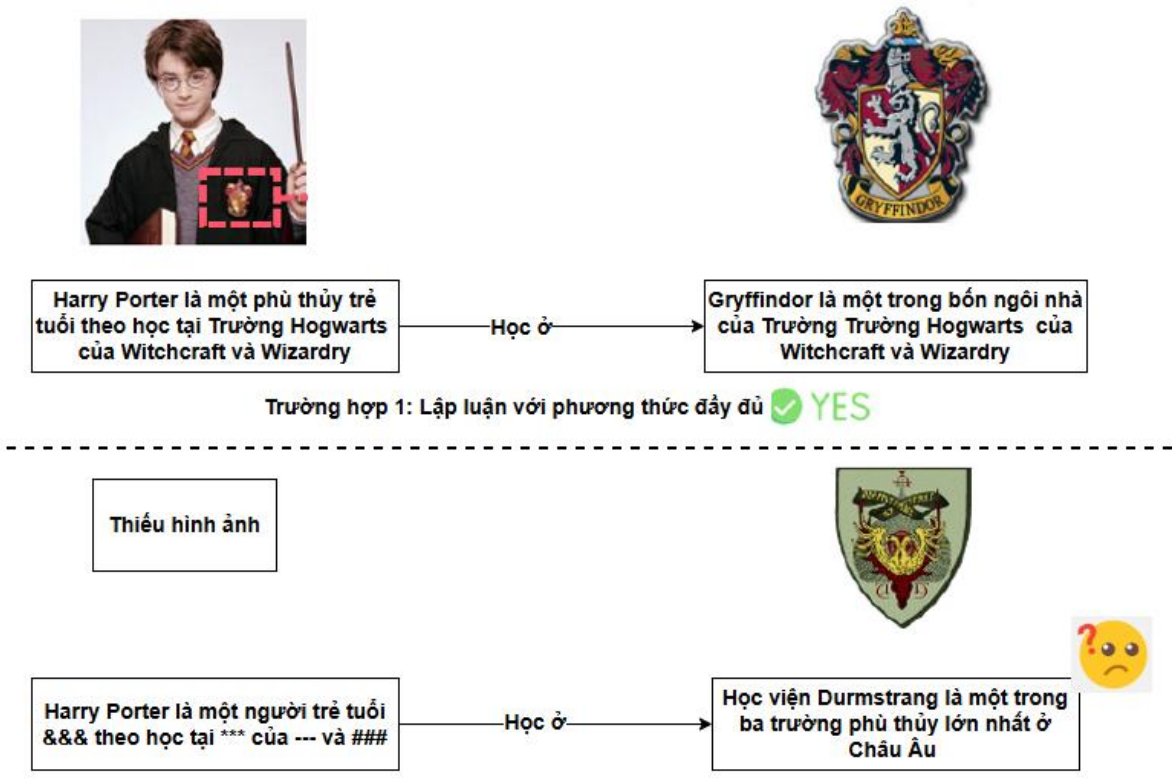


Hình 1.3. Ví dụ về ĐTTT đa phương thức N-MMKG (được mô phỏng theo ý tưởng từ [26]).

K. Liang và cộng sự [23] đưa ra định nghĩa về ĐTTT đa phương thức như sau.

Định nghĩa 1.2 ĐTTT đa phương thức [23].

ĐTTT đa phương thức $\mathcal{MKG}$ là một ĐTTT bao gồm các dữ kiện tri thức trong đó các thực thể tồn tại nhiều hơn một phương thức.

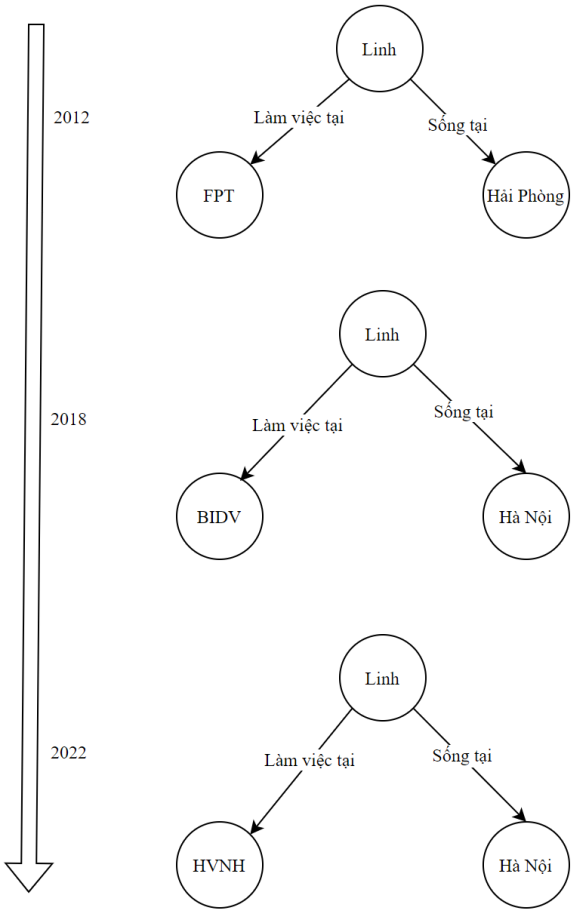


Hình 1.4. Nguyên nhân giới hạn thông tin phương thức [18].

Các thực thể ĐTTT đa phương thức bao gồm dữ liệu ở nhiều hơn một phương thức. Theo M. Wang và cộng sự [27], trên thực tế các ĐTTT được xây dựng và tập hợp từ nhiều nguồn dữ liệu không đồng nhất nhưng vẫn thiếu các yếu tố phương thức. Điều đó làm hạn chế việc sử dụng thông tin phương thức trong các mô hình hoàn thiện ĐTTT đa phương thức. Hình 1.4 là một ví dụ về việc giới hạn và thiếu thông tin đa phương thức đã ảnh hưởng đáng kể đến hiệu năng của quá trình hoàn thiện ĐTTT đa phương thức. Do đó, để đạt được hiệu năng tốt nhất, điều quan trọng là phải sử dụng hiệu quả các thông tin cần thiết cũng nhưng có được thông tin đa phương thức chất lượng cao. Điều này có nghĩa là các mô hình hoàn thiện ĐTTT đa phương thức cần quan tâm đến vấn đề mất cân bằng thông tin đa phương thức trong mô hình.

Theo Y. Zhang và cộng sự [18], các mô hình hoàn thiện ĐTTT đa phương thức trước đó chưa quan tâm đến vấn đề mất cân bằng thông tin đa phương thức trong mô hình. Chính vì điều này đã làm ảnh hưởng khá lớn đến hiệu năng của các mô hình.

C. ĐTTT thời gian



Hình 1.5. Ví dụ về sự thay đổi trong ĐTTT thời gian.

ĐTTT thời gian (Temporal knowledge graph, TKG) được hiểu là một cấu trúc ĐTTT động, ở đó các thông tin thực thể và quan hệ được thay đổi và cập nhật theo thời gian thực hoặc trong một khoảng thời gian nhất định. Điều này có ý nghĩa là thông tin các dữ kiện trong ĐTTT thời gian có thể thay đổi, được thêm mới, được cập nhật mới hoặc xóa bỏ theo thời gian. Ví dụ, trong một ĐTTT động liên quan đến các dữ liệu tài chính, thông tin về giá cổ phiếu, chỉ số tài chính hoặc thông tin về doanh nghiệp có thể được cập nhật hàng ngày, hàng tuần, hàng tháng. Hình 1.5 cho một ví dụ minh họa khác về ĐTTT thời gian.

K. Liang và cộng sự [23] đã đưa ra định nghĩa sau đây về ĐTTT thời gian.

Định nghĩa 1.3 ĐTTT thời gian [23].

ĐTTT thời gian được định nghĩa như là một dãy các ĐTTT tại các khung thời gian khác nhau $TKG = \{KG_1, \dots, KG_n\}$. *Một ĐTTT tại thời điểm* $\tau \in \mathcal{T}$ *được định nghĩa là* KG_τ. *Một dữ kiện có thời gian trong ĐTTT thời gian gian được ký hiệu là* (h, r, t, τ) *mô tả quan hệ* r *kết nối giữa thực thể đầu* h *và thực thể cuối* t *tại thời điểm* τ. *Bên cạnh đó, tính chất thuộc tính* (e, a, v, τ) *mô tả thực thể* e *có một thuộc tính* a *với giá trị thuộc tính* v *tại thời điểm* τ.

Như vậy, ĐTTT thời gian là một mở rộng của ĐTTT bằng cách thêm vào yếu tố thời gian. Nghĩa là, mỗi dữ kiện trong đó được xác định trong các (khoảng) thời điểm khác nhau. Do vậy, dữ kiện dược biểu diễn dưới dạng bộ bốn (h, r, t, τ), ở đó $h, t \in E$ là thực thể, $r \in R$ là quan hệ và $\tau \in \mathcal{T}$ là thời gian.

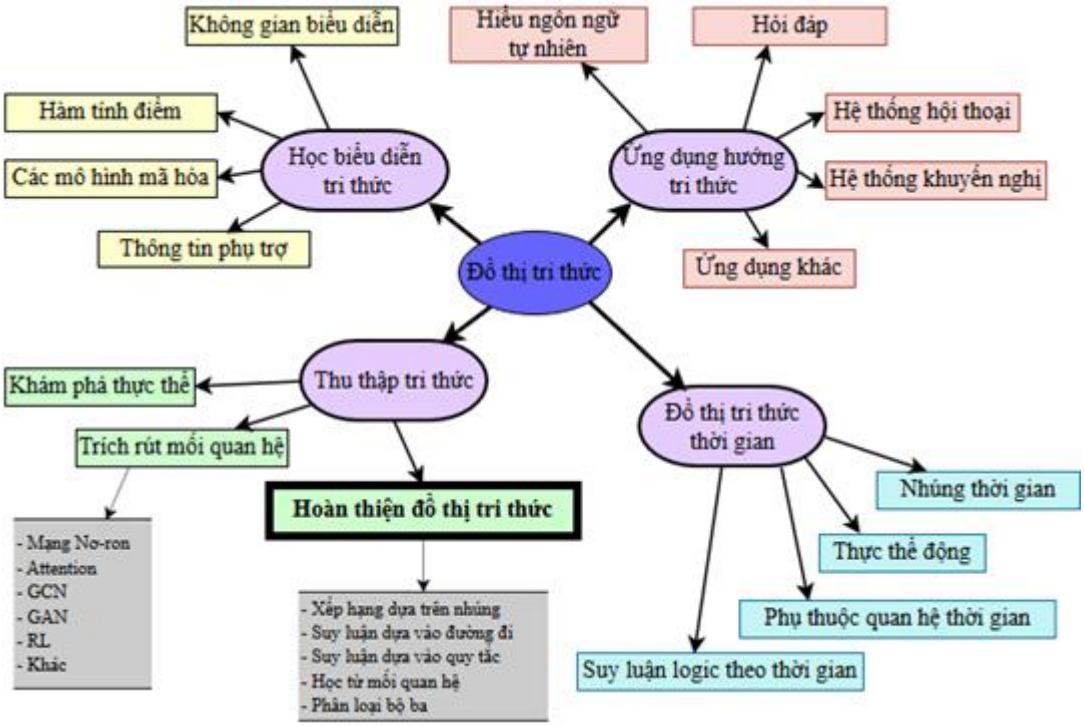
1.1.3 Khái quát về các bài toán nghiên cứu về ĐTTT

Hình 1.6 cung cấp một khung phân loại các bài toán nghiên cứu trong ĐTTT [7]. Bốn bài toán nghiên cứu về ĐTTT ở mức cao nhất là Học biểu diễn tri thức, Thu thập tri thức, ĐTTT thời gian và Các ứng dụng hướng tri thức dựa trên ĐTTT. Do các nội dung trình bày về Hoàn thiện ĐTTT thời gian (Mục 1.4) đã bao gói các bài toán nghiên cứu về ĐTTT thời gian cho nên luận án giới thiệu ba bài toán còn lại như dưới đây.

I. Học biểu diễn tri thức

Theo S. Ji và cộng sự [7], Học biểu diễn tri thức (Knowledge Representaion Learning) là một vấn đề nghiên cứu quan trọng của ĐTTT cho phép mở đường cho nhiều tác vụ thu nhận tri thức và các ứng dụng tiếp theo. Học biểu diễn tri thức được phân lớp theo bốn khía cạnh:

- Không gian biểu diễn (Representation Space).
- Hàm chấm điểm (Scoring Function).
- Mô hình mã hóa (Encoding Models).
- Thông tin phụ trợ (Auxiliary Information).



Hình 1.6. Khung phân loại nghiên cứu về ĐTTT (được thu gọn từ [7]).

Một quy trình làm việc rõ ràng để phát triển mô hình Học biểu diễn tri thức đã được cung cấp [7].

Chức năng của bốn thành phần cụ thể này là:

- Không gian biểu diễn là không gian mà các tập thực thể và quan hệ được biểu diễn.
- Hàm tính điểm sử dụng để đo lường tính hợp lý của bộ ba thực thể.
- Các mô hình mã hóa biểu diễn và học các tương tác quan hệ.
- Thông tin phụ trợ được tích hợp vào các phương pháp nhúng

Các biểu diễn số học được tạo ra từ việc huấn luyện các mô hình này có thể biểu thị các thực thể, quan hệ, tính chất và tương quan trong không gian tri thức. Các độ đo tính điểm thường được chia thành các hàm tính điểm dựa trên sự tương đồng và dựa trên khoảng cách.

II. Thu thập tri thức

Nhiệm vụ thu thập tri thức (Knowlegde Acquisition) chính là việc xây dựng ĐTTT có thể từ nhiều nguồn khác nhau như văn bản không cấu trúc, có cấu trúc hoặc

bán cấu trúc dựa vào quá trình trích xuất và biểu diễn thông tin dưới dạng bộ ba. Bên cạnh nhiệm vụ xây dựng ĐTTT thì mục đích của thu thập tri thức chính là mở rộng ĐTTT. Các bài toán mở rộng ĐTTT trong thu thập tri thức được chia làm ba loại chính:

- Hoàn thiện ĐTTT (Knowledge Graph Completion).
- Trích xuất quan hệ thực thể (Relation Extraction).
- Phát hiện thực thể (Entity Discovery).

Hoàn thiện ĐTTT nhằm mục đích mở rộng ĐTTT ban đầu, còn Trích xuất quan hệ thực thể và Phát hiện thực thể nhằm mục đích khám phá tri thức (các thực thể, các quan hệ và dữ liệu) mới từ văn bản nguồn. Hoàn thiện ĐTTT, Trích xuất quan hệ thực thể và Phát hiện thực thể lại được chia thành nhiều thể loại khác nhau dựa trên các phương pháp tiếp cận [5]. Do Hoàn thiện ĐTTT là chủ đề nghiên cứu chính của luận án này cho nên nó được giới thiệu chi tiết hơn ở các mục tiếp theo.

III. Ứng dụng hướng tri thức dựa trên ĐTTT

Ứng dụng nhận thức tri thức (Knowledge Aware Application) dựa trên ĐTTT được phân loại thành ứng dụng trong ĐTTT (dự đoán liên kết, đoán nhận thực thể có tên) và ứng dụng ngoài ĐTTT (trích xuất quan hệ, học biểu diễn ngôn ngữ, hệ thống hỏi-đáp, hệ thống tư vấn, v.v.). Ứng dụng trong ĐTTT và trích xuất quan hệ đã được đề cập ở đây. Dưới đây giới thiệu về ba ứng dụng hạ nguồn điển hình.

- Học biểu diễn ngôn ngữ (Language Representation Learning): Tích hợp ĐTTT làm tăng hiệu quả trong xây dựng mô hình ngôn ngữ huấn luyện trước, biểu diễn ngôn ngữ, v.v.
- Hỏi đáp (Question Answering): Các hệ thống hỏi đáp dựa trên ĐTTT để trả lời câu hỏi ngôn ngữ tự nhiên bằng các dữ kiện từ ĐTTT, đặc biệt tiếp cận sử dụng mạng nơ-ron biểu diễn câu hỏi và trả lời trong không gian ngữ nghĩa phân tán. Hai tình huống ứng dụng điển hình nhất là hệ thống hỏi đáp dữ kiện đơn và suy luận đa bước trong nhiệm vụ xây dựng câu trả lời.
- Hệ thống tư vấn (Recommender System): Sử dụng ĐTTT như một nguồn tài nguyên tri thức ngoài cung cấp thông tin phụ về thực thể, quan hệ và thuộc tính trong quá trình lọc thông tin để cung cấp tư vấn.

ĐTTT còn có nhiều ứng dụng nhận thức tri thức nội tại ĐTTT cũng trong nhiều tác vụ hạ nguồn bên ngoài ĐTTT.

Như đã được nhấn mạnh, ĐTTT đóng vai trò quan trọng đặc biệt, cung cấp thông tin ngữ nghĩa phong phú và đã nổi lên như một động lực thúc đẩy lĩnh vực TTNT [2], chúng cung cấp dữ liệu có cấu trúc và tri thức thực tế thúc đẩy nhiều sản phẩm và khiến các sản phẩm này trở nên thông minh và kỳ diệu hơn [6].

1.2 Hoàn thiện ĐTTT

Mặc dù ĐTTT có giá trị lớn trong các ứng dụng, nhưng chúng thường được xây dựng thủ công hoặc bán tự động nên vẫn chưa đầy đủ vì một lượng lớn tri thức quý giá tồn tại ngầm hoặc bỏ sót trong các ĐTTT, dẫn tới ĐTTT hoạt động kém hơn nhiều trong các ứng dụng thực tế. Trên thực tế các ĐTTT thường không đầy đủ, thiếu rất nhiều thể hiện quan hệ, ví dụ, theo Chen và cộng sự [28], trong Wikidata, chỉ có khoảng 30% số người có thuộc tính "ngày sinh" được xác định. Vì vậy, các bài toán về hoàn thiện ĐTTT (Knowledge Graph Completion – KGC) là rất cần thiết để giải quyết các vấn đề về thực trạng không đầy đủ đó.

1.2.1 Khái niệm về hoàn thiện đồ thị tri thức

Như đã được đề cập, ĐTTT là một tập hợp các dữ kiện được lưu trữ và mô tả theo một cách có cấu trúc hay một ĐTTT là một tập các bộ ba (h, r, t), ở đó h, t là các thực thể đầu và cuối, r là một quan hệ nhị phân giữa các thực thể. Để tăng tốc tiến trình xây dựng ĐTTT, các dữ kiện được trích xuất thông tin tự động từ các văn bản không có cấu trúc bằng các công cụ trích xuất thông tin mở (Open Information Extracted), ví dụ ReVerb [29], hoặc các công cụ thông tin dựa trên tiếp cận nơ-ron [30], [31].

Hoàn thiện ĐTTT, còn được gọi là Dự đoán liên kết (Link Prediction), nhằm mục đích tự động suy ra các dữ kiện còn thiếu cho một ĐTTT bằng cách học hỏi từ các dữ kiện hiện có trong ĐTTT đó. Hoàn thiện ĐTTT có thể được chia thành ba tác vụ nhỏ [9]:

- Phân lớp bộ ba: là một tác vụ hoàn thiện ĐTTT điển hình nhằm xác định xem có nên thêm một bộ ba (h, r, t) chưa có trong một ĐTTT vào ĐTTT đó hay không bằng cách ước tính xem bộ ba này có đúng hay không.

- Dự đoán liên kết: đề cập đến quá trình từ một bộ ba bị thiếu thực thể đầu (được ký hiệu là $(?, r, t)$) hoặc thực thể đuôi (được ký hiệu là $(h, r, ?)$) đi tìm thực thể bị thiếu "?" đó.
- Dự đoán quan hệ (được ký hiệu là $(h, ?, t)$): là đánh giá xác suất thiết lập các quan hệ cụ thể giữa hai thực thể [9].

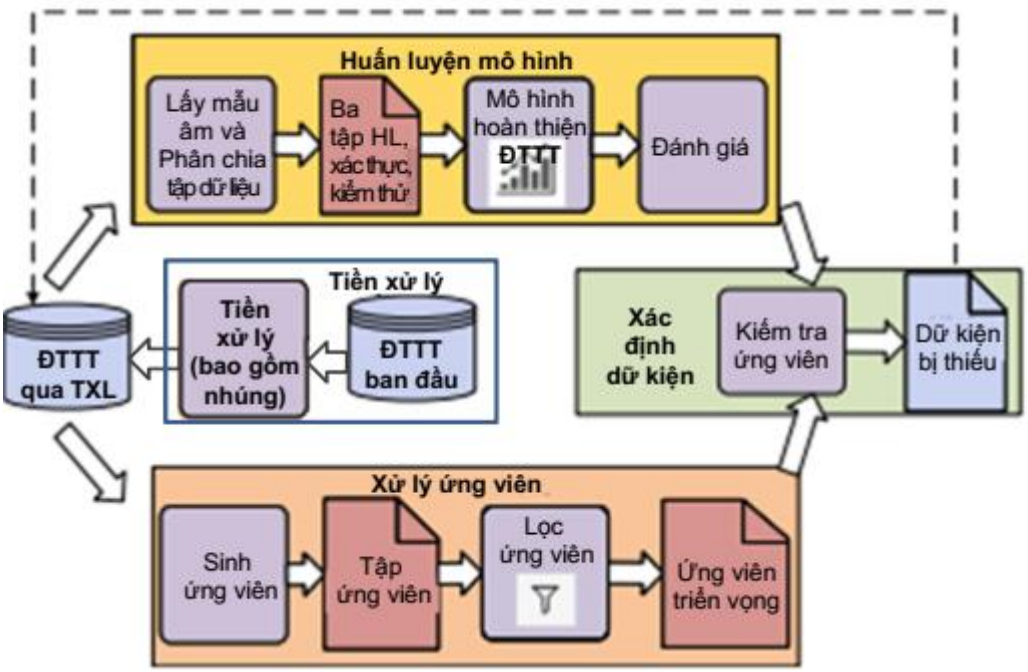
Từ đó, các thông tin thiếu có thể được suy ra bằng các thuật toán hoàn thiện ĐTTT (KGC), ví dụ như ConvE [32], TuckER [33], 5⋆E [34].

Tùy thuộc vào loại ĐTTT, đầu vào của mô hình hoàn thiện ĐTTT sẽ khác nhau, không chỉ đơn thuần là dữ liệu cấu trúc bộ ba mà còn có thể bao gồm nhiều loại dữ liệu bổ trợ khác nhau. Cụ thể đầu vào bao gồm:

- Tập các bộ ba đã biết (Tập dữ kiện gốc): Mỗi bộ ba được biểu diễn dưới dạng (h, r, t), ở đó h, t là các thực thể đầu và cuối, trong khi r là một quan hệ. Tập bộ ba này được sử dụng làm dữ liệu huấn luyện và là cơ sở để mô hình học biểu diễn ngữ nghĩa của thực thể và quan hệ.
- Tập thực thể và quan hệ: Là danh sách đầy đủ các thực thể $\mathcal{E}$ và tập quan hệ $\mathcal{R}$ xuất hiện trong tập dữ liệu. Thông tin này thường được dùng để khởi tạo không gian nhúng (embedding space) cho mô hình.
- Thông tin đa phương thức (đối với hoàn thiện ĐTTT đa phương thức): Trong một số mô hình hiện đại, đầu vào còn bao gồm các đặc trưng bổ sung như mô tả văn bản, đặc trưng hình ảnh, thông tin thời gian, nhằm tăng cường khả năng biểu diễn của mô hình. Những đặc trưng này đặc biệt quan trọng trong các mô hình hoàn thiện đồ thị tri thức đa phương thức.
- Tập tham số cấu hình: Bao gồm các siêu tham số như kích thước nhúng, số lượng bộ ba âm, tốc độ học, số epoch,... được sử dụng để điều chỉnh quá trình huấn luyện mô hình.

1.2.2 Quy trình chung hoàn thiện ĐTTT

Trước khi vào quy trình hoàn thiện ĐTTT, ĐTTT cần hoàn thiện cần được tiền xử lý để thực hiện các biểu diễn dữ liệu phù hợp với phương pháp hoàn thiện ĐTTT. Một tác vụ tiền xử lý phổ biến là tiến hành các phép nhúng thực thể, quan hệ và các đối tượng liên quan khác [35].



Hình 1.7. Quy trình chung hoàn thiện ĐTTT [9]; khối Tiền xử lý do luận án bổ sung.

Như minh họa ở Hình 1.7, quy trình chung hoàn thiện ĐTTT (sau khi ĐTTT đã được tiền xử lý) bao gồm ba khối chính: Huấn luyện mô hình, Xử lý ứng viên và Xác định dữ kiện [9]; hai khối sau thuộc về pha sử dụng mô hình trong khung học máy (khai phá dữ liệu) chung.

- **Huấn luyện mô hình.** Đầu tiên, trước khi xây dựng mô hình hoàn thiện ĐTTT, thường có một bước chuẩn bị dữ liệu, bao gồm lấy mẫu âm tính (đôi khi không phải là bước bắt buộc, cũng có thể thực hiện trực tuyến trong quá trình huấn luyện mô hình) và phân chia tập dữ liệu. Lấy mẫu âm tính nhằm mục đích thêm một lượng biến đổi các ví dụ âm tính vào ĐTTT gốc để giải quyết vấn đề ĐTTT chỉ chứa các ví dụ dương. Việc phân chia tập dữ liệu chịu trách nhiệm phân chia dữ liệu ĐTTT đang chờ xử lý thành một tập huấn luyện, một tập xác thực và một tập kiểm thử. Các tập dữ liệu phân chia tiếp theo sẽ được sử dụng để huấn luyện và đánh giá mô hình hoàn thiện ĐTTT. Sau đó, mô hình hoàn thiện ĐTTT thường là một mô hình phân lớp hoặc một mô hình xếp hạng, có mục tiêu là dự đoán xem một bộ ba ứng viên có đúng hay không đối với một ĐTTT. Một kết quả đánh giá thỏa mãn ở khâu cuối trong khối Huấn luyện mô hình có nghĩa là một mô hình hoàn thiện ĐTTT tốt đã được xây dựng xong.

Trong các mô hình xếp hạng, một hàm tính điểm được sử dụng để xếp hạng các bộ ba được đưa vào mô hình và quá trình huấn luyện mô hình có mục tiêu xác định ngưỡng phân tách bộ ba đúng và sai. Sau đây là một số hàm tính điểm điển hình:

$$\boldsymbol{f(h,r,t)} = -\| \mathbf{v_h} + \mathbf{v_r} - \mathbf{v_t} \|_{l_{1/2}} \quad \textit{(1-1)}$$

trong đó: $\mathbf{v}_r \in \mathbb{R}^k$. $l_{1/2}$ biểu thị là chuẩn L_1 hoặc chuẩn L_2 bình phương. Hàm này được dùng trong TransE.

$$\boldsymbol{f(h,r,t)} = \mathbf{v_h^T W_r v_t} \quad \textit{(1-2)}$$

trong đó: $\mathbf{W}_r \in \mathbb{R}^{k\times k}$ là một ma trận chéo. Hàm này được dùng trong DistMult.

$$\boldsymbol{f(h,r,t)} = \frac{1}{2}\left(\mathbf{v}_{h,1}^T \mathbf{W}_r \mathbf{v}_{t,2} + \mathbf{v}_{t,1}^T \mathbf{W}_{r^{-1}} \mathbf{v}_{h,2}\right) \quad \textit{(1-3)}$$

trong đó: $\mathbf{v}_{h,1}$, $\mathbf{v}_{h,2}$, $\mathbf{v}_{t,1}$, $\mathbf{v}_{t,1} \in \mathbb{R}^k$; $\mathbf{W}_r$ và $\mathbf{W}_{r^{-1}}$ là các ma trận chéo trong không gian $\mathbb{R}^{k\times k}$. Hàm này được dùng trong SimplE.

$$\boldsymbol{f(h,r,t)} = \mathbf{M} \times_1 \mathbf{v_h} \times_2 \mathbf{v_r} \times_3 \mathbf{v_t} \quad \textit{(1-4)}$$

trong đó, $\mathbf{v}_r \in \mathbb{R}^n$, $\mathbf{M} \in \mathbb{R}^{k\times n\times k}$, $\times_d$ là tích ten-xơ theo bậc thứ d. Hàm này được dùng trong TuckER.

$$f(h,r,t) = -\| P_r \mathbf{v}_h + b_r - \mathbf{v}_t \| \quad \textit{(1-5)}$$

trong đó, $P_r \in \mathbb{R}^{k\times k}$: là ma trận biến đổi (project matrix) tương tứng với quan hệ r; $b_r \in \mathbb{R}^k$ là véc-tơ dịch (bias) của quan hệ r. Hàm này được dùng trong 5★E.

$$\boldsymbol{f(h,r,t)} = \mathbf{v_t^T ReLU}\left(\mathbf{Wvec}\left(\mathbf{ReLU}(\mathbf{concat}(\bar{\mathbf{v}}_h, \bar{\mathbf{v}}_r) * \boldsymbol{\Omega})\right)\right) \quad \textit{(1-6)}$$

trong đó, $\bar{\mathbf{v}}_h, \bar{\mathbf{v}}_r$ là dạng chuyển đổi thành 2 chiều của véc-tơ $\mathbf{v}_h$, $\mathbf{v}_r$; $*$ ký hiệu một toán tử tích chập; Ω là tập các bộ lọc. Hàm này được dùng trong ConvE.

$$\boldsymbol{f(h,r,t)} = -\| \mathbf{v_h} \circ \mathbf{v_r} - \mathbf{v_t} \|_{l_{1/2}} \quad \textit{(1-7)}$$

trong đó, $\mathbf{v}_h$, $\mathbf{v}_r$, $\mathbf{v}_t \in \mathbb{C}^k$, $\circ$ biểu thị phép nhân theo phần tử. Hàm này được dùng trong RotatE.

Luận án kế thừa các hàm tính điểm từ các mô hình hoàn thiện ĐTTT được luận án cải tiến.

- **Xử lý ứng viên.** Quá trình xử lý ứng viên nhằm mục đích thu được các bộ ba có thể xác minh được. Các bộ ba đó sẽ được mô hình hoàn thiện ĐTTT đã học kiểm tra trong quá trình huấn luyện mô hình. Quá trình xử lý ứng viên bắt đầu bằng việc tạo tập ứng viên, tạo ra một tập ứng viên (tập các ứng viên là các bộ ba có thể đúng nhưng không có trong ĐTTT) dựa trên các thuật toán hoặc công

việc thủ công. Vì tập ứng viên được tạo ban đầu có xu hướng rất lớn bất kể các ứng viên có triển vọng hay không. Sau đó, nó phải trải qua bước lọc ứng viên để loại bỏ trước những ứng viên không có khả năng đó và đồng thời giữ lại càng nhiều ứng viên triển vọng càng tốt. Thông thường, công việc lọc được thực hiện bằng cách tạo ra một số luật lọc (còn được gọi là ‘‘chiến lược cắt tỉa’’) và áp dụng các luật này vào tập ứng viên để tạo ra tập cô đọng các ứng viên triển vọng nhất.

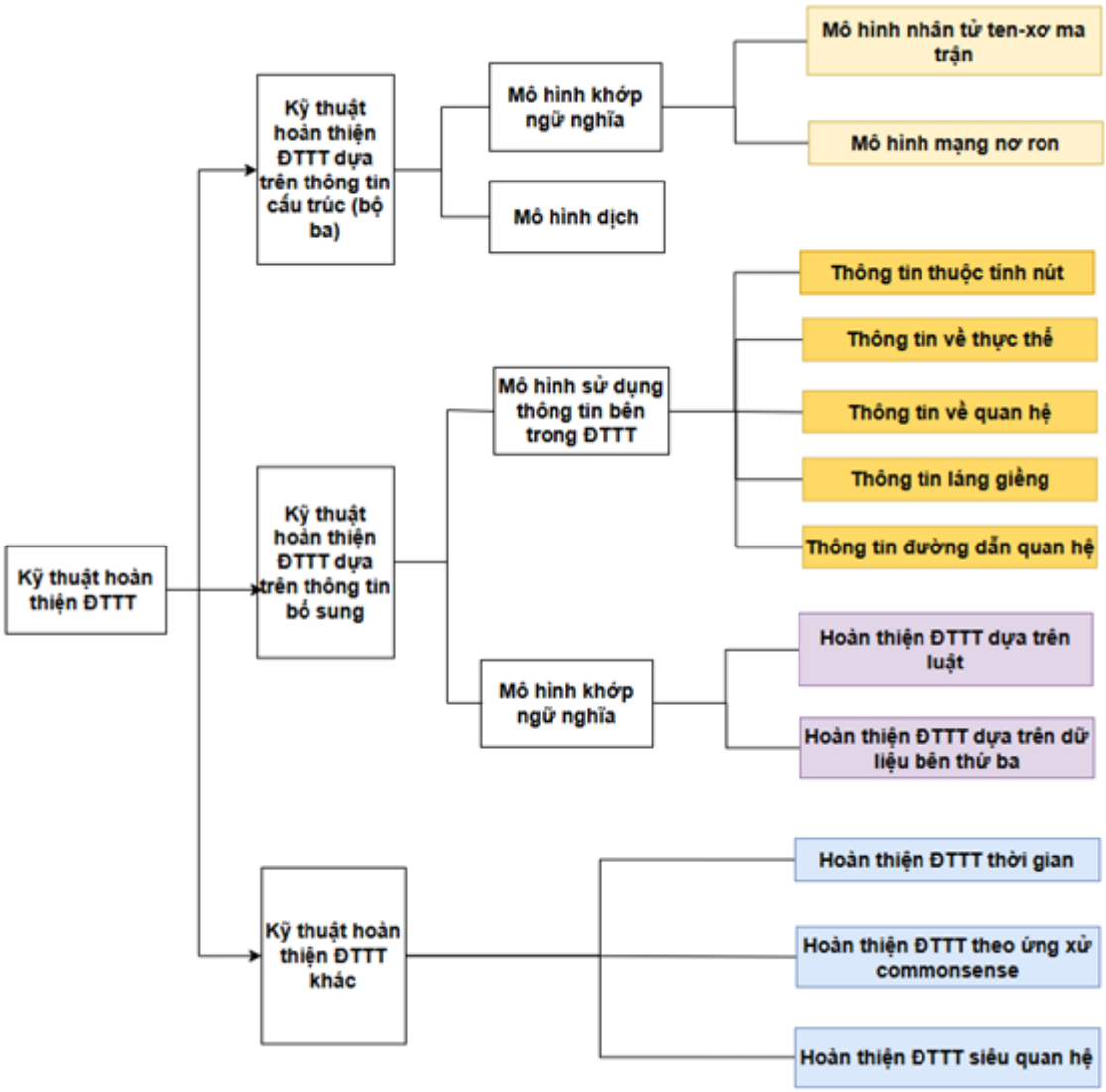
- **Xác định dữ kiện.** Cuối cùng, mô hình hoàn thiện ĐTTT đã học trong quá trình huấn luyện mô hình được áp dụng cho tập các ứng viên triển vọng ở trên được tạo ra bởi quá trình xử lý ứng viên, tạo ra tập các bộ ba bị thiếu được coi là đúng và có khả năng được thêm vào ĐTTT.

Các mô hình hoàn thiện ĐTTT được đề cập trong luận án này là tập trung vào khối Huấn luyện mô hình như được trình bày trên đây.

1.2.3 Hệ thống các kỹ thuật hoàn thiện ĐTTT

T. Shen và cộng sự [9] đã hệ thống hóa các kỹ thuật hoàn thiện ĐTTT như được minh họa ở Hình 1.8. Như vậy, có ba nhóm kỹ thuật chính hoàn thiện ĐTTT, cụ thể là:

- Kỹ thuật hoàn thiện ĐTTT dựa trên thông tin cấu trúc (Stuctural Information-based KGC Technologies).
- Kỹ thuật hoàn thiện ĐTTT dựa trên thông tin bổ sung (Additional Information-based KGC Technologies).
- Kỹ thuật hoàn thiện ĐTTT khác.



Hình 1.8. Một khung phân loại các kỹ thuật hoàn thiện ĐTTT [9].

Hai nhóm con kỹ thuật hoàn thiện ĐTTT dựa trên thông tin cấu trúc là nhóm kỹ thuật so khớp ngữ nghĩa (Semantic Matching Models) và nhóm kỹ thuật dịch (Translation Models) [9]. Các kỹ thuật so khớp ngữ nghĩa thực hiện việc tính toán các hàm chấm điểm dựa trên so khớp ngữ nghĩa bằng cách đo các điểm tương đồng ngữ nghĩa của các nhúng thực thể hoặc quan hệ trong không gian nhúng tiềm ẩn. Kỹ thuật so khớp ngữ nghĩa do A. Bordes và cộng sự đề xuất vào năm 2014 là một kỹ thuật tiêu biểu, sử dụng cấu trúc mạng nơ-ron để thu được sự khớp ngữ nghĩa, khai phá quan hệ ngữ nghĩa giữa các thực thể và các quan hệ. Hơn nữa, dòng nghiên cứu về kỹ thuật So khớp ngữ nghĩa được chia làm hai nhánh là: Kỹ thuật nhân tử ten-xơ ma trận (Tensor/Matrix Factorization Models) và kỹ thuật mạng nơ-ron nói chung (Neural Network Models). Kỹ thuật nhân tử ten-xơ ma trận được R. Socher và cộng

sự đề xuất năm 2013 theo ý tưởng cơ bản là thay thế lớp biến đổi tuyến tính trong mạng nơ-ron truyền thống bằng một ten-xơ song tuyến tính và kết nối với véc-tơ thực thể đầu và véc-tơ thực thể cuối trong các chiều khác. Một số nghiên cứu tiêu biểu về kỹ thuật mạng nơ-ron là Mô hình ConvE được T. Dettmers và cộng sự giới thiệu [32], Mô hình mạng tích chập đồ thị do M. Schlichtkrull và cộng sự giới thiệu vào năm 2018; Mô hình mạng tích chập nhận biết cấu trúc (Structure-Aware Convolutional Network – SACN) được C. Sang và cộng sự giới thiệu vào năm 2019. Các mô hình dựa trên kỹ thuật dịch thường mã hóa các thực thể dưới dạng nhúng chiều thấp và các quan hệ giữa các thực thể dưới dạng véc-tơ dịch, định nghĩa một hàm tính điểm dịch phụ thuộc vào quan hệ để đo xác suất của một bộ ba thông qua khoảng cách. Các kỹ thuật dịch vừa đơn giản và vừa có hiệu năng tốt đối với hoàn thiện ĐTTT. Các kỹ thuật dịch này thường định nghĩa một hàm tính điểm phụ thuộc vào quan hệ để đo xác suất của một bộ ba thông qua độ đo khoảng cách. Một mô hình hoàn thiện ĐTTT dựa trên kỹ thuật dịch rất phổ biến là mô hình TransE [36] và một họ các mô hình hoàn thiện ĐTTT được xây dựng và mở rộng từ mô hình TransE như TransH, TransR, TransD, lppTransD.

Hai nhóm con kỹ thuật hoàn thiện ĐTTT dựa trên thông tin bổ sung là dựa trên thông tin bổ sung bên trong ĐTTT (Internal Side Information Inside KGs) và dựa trên thông tin bổ sung bên ngoài ĐTTT (External Extra Information Outside KGs) [9]. Thông tin bổ sung bên trong ĐTTT gồm thông tin thuộc tính nút, thông tin liên quan đến thực thể, thông tin liên quan đến quan hệ, thông tin lân cận và thông tin đường dẫn quan hệ là những thông tin không thể bỏ qua, đóng vai trò quan trọng trong việc nắm bắt các đặc trưng hữu ích của nhúng tri thức cho hoàn thiện ĐTTT và các ứng dụng nhận thức tri thức. Thông tin bên ngoài ĐTTT thường được sử dụng vào hoàn thiện ĐTTT gồm luật suy diễn và thông tin phù trợ từ bên thứ ba. Một số mô hình hoàn thiện ĐTTT theo hướng này như: KBAT, DrWT, KBLRN.

Ba mô hình hoàn thiện ĐTTT điển hình không thuộc hai nhóm chính trên đây là các mô hình hoàn thiện ĐTTT thời gian, theo ứng xử và dựa trên siêu quan hệ [9].

1.2.4 Một số mô hình hoàn thiện ĐTTT điển hình

Dự đoán liên kết là một tác vụ cơ bản trong hoàn thiện ĐTTT bằng cách sử dụng các quan hệ có sẵn rồi suy luận ra các quan hệ mới để tạo ra một ĐTTT hoàn chỉnh. Tồn tại rất nhiều phương pháp được đề xuất để nghiên cứu về tác vụ dự đoán

liên kết dựa trên các cách tiếp cận và kỹ thuật biểu diễn khác nhau. Giữa các phương pháp này, các mô hình nhúng ĐTTT mang lại nhiều kết quả rõ nét. Nhúng ĐTTT được hình thành bằng cách tạo ra các nhúng thực thể và nhúng quan hệ, được trình bày theo hai định nghĩa dưới đây [19].

Định nghĩa 1.4.

Một nhúng thực thể, ký hiệu là $EEMB: \mathcal{E} \rightarrow \psi$, là một ánh xạ gán mỗi thực thể $e \in \mathcal{E}$ thành một biểu diễn ẩn trong ψ, trong đó ψ là một lớp không rỗng, chứa bộ các véc-tơ và/hoặc ma trận.

Định nghĩa 1.5

Một nhúng quan hệ, ký hiệu là $REMB: \mathcal{R} \rightarrow \psi$, là một ánh xạ gán mỗi quan hệ $r \in \mathcal{R}$ thành một biểu diễn ẩn trong ψ, trong đó ψ là một lớp không rỗng, chứa bộ các véc-tơ và/hoặc ma trận.

Lưu ý rằng lớp ẩn của một thực thể (quan hệ) là nhúng của thực thể (quan hệ) đó. Một mô hình nhúng ĐTTT được mô tả bằng hai thành phần: 1 – các ánh xạ $EEMB$, $REMB$; 2 – hàm tính điểm sẽ nhận đầu ra của các ánh xạ này làm dữ liệu đầu vào và cung cấp điểm số. Các tham số của biểu diễn ẩn được học từ dữ liệu. Trong mục này, luận án giới thiệu các mô hình hoàn thiện ĐTTT được coi là điển hình nhất, thường được sử dụng làm nền tảng phát triển các mô hình hoàn thiện ĐTTT khác, trong đó có các mô hình được luận án nghiên cứu và cải tiến.

1. Mô hình TransE

Mô hình hoàn thiện ĐTTT TransE được A. Bordes và cộng sự đề xuất vào năm 2013 [36]. Mô hình TransE được lấy cảm hứng từ các mô hình như mô hình Word2Vec Skip-gram. Ý tưởng chính của TransE là véc-tơ nhúng của mỗi quan hệ giữa hai thực thể được mô hình hóa bằng phép tịnh tiến trong không gian nhúng. Mỗi thực thể và quan hệ được biểu diễn dưới dạng một véc-tơ nhúng trong một không gian liên tục, qua hai ánh xạ nhúng $EEMB$ và $REMB$. Một cách chi tiết, mô hình TransE học các véc-tơ chiều thấp và dày cho mỗi thực thể và quan hệ, sao cho mỗi kiểu quan hệ tương ứng với một toán tử véc-tơ tịnh tiến tác động lên các véc-tơ biểu diễn cho thực thể. Nhiều mô hình hoàn thiện ĐTTT hiệu năng được đề xuất dựa trên việc cải tiến TransE, chẳng hạn như TransH (Translation on Hyperplanes), TransR,

DistMult [37], RotatE [38]. Hơn nữa, TransE còn được làm nền để xây dựng các mô hình hoàn thiện ĐTTT thời gian, chẳng hạn như TtransE [39], DE-TransE [19].

2. Mô hình DistMult

Mô hình DistMult được B. Yang và cộng sự đề xuất vào năm 2015 [37] cũng xây dựng ánh xạ nhúng thực thể $EEMB$ như TransE. Trong mô hình DistMult mỗi quan hệ được biểu diễn bởi một ma trận chéo dựa theo mô hình song tuyến tính. Các mô hình hoàn thiện ĐTTT thời gian TA-DistMult [40], DE-DistMult [19] sử dụng DistMult làm mô hình nền.

3. Mô hình SimplE

Trong mô hình hoàn thiện ĐTTT SimplE [41], S. M. Kazemi và D. Poole đề xuất giải pháp cải tiến đơn giản phân rã CP (Canonical Polyadic) nhằm cho phép hai nhúng của mỗi thực thể được học một cách độc lập, nghĩa là $\mathbf{v}_{e,1}, \mathbf{v}_{e,2} \in \mathbb{R}^k$ của mỗi thực thể được học một cách độc lập để khai thác lợi thế từ phép nghịch đảo quan hệ nhằm mục đích giải quyết vấn đề dòng chảy thông tin giữa chúng. Với mỗi thực thể $e \in E$, $\mathbf{v}_{e,1}$, được sử dụng khi e là thực thể đầu, $\mathbf{v}_{e,2}$ được sử dụng khi e là thực thể cuối. SimplE thực hiện nhúng thực thể $EEMB(e) = (\mathbf{v}_{e,1}, \mathbf{v}_{e,2})$. Mô hình hoàn thiện ĐTTT thời gian DE-SimplE [19] sử dụng SimplE làm mô hình nền.

4. Mô hình TuckER

Trong [33], I. B. Balažević và cộng sự đề xuất mô hình TuckER bằng cách sử dụng một mô hình tuyến tính dựa trên phân tích nhân tử ten-xơ (tensor factorization) nhằm mục đích dự đoán các liên kết bị thiếu trong ĐTTT bằng cách học hiệu quả các tương tác giữa các thực thể và quan hệ. Mô hình TuckER khá linh hoạt và tương thích với nhiều thiết lập khác nhau bằng cách điều chỉnh chiều của các ten-xơ lõi, cho phép nó khái quát tốt trên các ĐTTT với các kích thước và độ phức tạp khác nhau. Do đó, mô hình TuckER cung cấp một phương pháp mạnh mẽ và linh hoạt để hoàn thiện ĐTTT, tận dụng phân tích ten-xơ để mô hình hóa các tương tác phức tạp với ít tham số hơn và mang lại hiệu năng vượt trội trong các tác vụ dự đoán liên kết.

5. Mô hình 5⋆E

M. Nayyeri và cộng sự [34] giới thiệu mô hình 5⋆E tận dụng năm biến đổi phép chiếu (tịnh tiến, quay, đồng dạng, đảo ngược và phản xạ) để cải thiện chất lượng nhúng của các thực thể và các quan hệ trong ĐTTT. Mô hình được xây dựng nhằm

giải quyết một số hạn chế của các mô hình khác bằng cách cung cấp một quy trình nhúng sử dụng thông tin hình học. Ý tưởng chính của mô hình 5★E là áp dụng năm biến đổi phép chiếu – là các phép toán bảo toàn các đường thẳng vào không gian nhúng, cho phép biểu diễn chính xác và linh hoạt các mẫu quan hệ phức tạp trong hoàn thiện ĐTTT. Ngoài ra, phương pháp này cho phép mô hình nắm bắt tốt hơn các thuộc tính, quan hệ khác nhau như tính đối xứng, tính bắc cầu và tính nghịch đảo. Do vậy, mô hình giúp cho các nhúng có ý nghĩa hơn, dễ hiểu, dễ diễn giải hơn, phản ánh đầy đủ cấu trúc cơ bản của ĐTTT.

6. Mô hình ConvE

T. Dettmers và cộng sự [32] nhận thấy rằng các phương pháp hoàn thiện ĐTTT trước đó đều thực hiện trên các mô hình song tuyến tính đơn giản, thường không hiệu quả với các khung ĐTTT phức tạp. Mô hình ConvE được xây dựng bằng cách sử dụng mạng nơ-ron tích chập 2D để học các nhúng từ các thực thể và các quan hệ trong một ĐTTT. Mô hình ConvE áp dụng các phép toán tích chập 2D trên các phép nhúng để tạo thành các ma trận, từ đó cho phép mô hình nắm bắt được các tương tác phức tạp hơn giữa các thực thể và quan hệ.

7. Mô hình RotatE

Mô hình RotatE được Z. Sun và cộng sự đề xuất [38] cũng dựa trên ý tưởng từ mô hình TransE, nghĩa là dựa trên nhúng của các thực thể $EEMB$ và nhúng quan hệ $REMB$ để đưa ra dự đoán các quan hệ còn thiếu trong ĐTTT. Khác với TransE sử dụng phép tịnh tiến, RotatE sử dụng phép quay trong không gian phức.

1.3 Hoàn thiện ĐTTT đa phương thức

Như đã được đề cập, các thực thể trong ĐTTT đa phương thức tồn tại nhiều hơn một phương thức, nghĩa là mỗi thực thể $e \in \mathcal{E}$ có một tập các ảnh được ký hiệu là $\mathcal{V}_e$ và một mô tả văn bản tương ứng.

Trong thời gian gần đây, hoàn thiện ĐTTT đa phương thức đã được rất nhiều nhà khoa học quan tâm nghiên cứu. Theo Z. Chen và cộng sự [42], các mô hình hoàn thiện ĐTTT đa phương thức phần lớn được xây dựng và phát triển bằng cách mở rộng các phương pháp nhúng ĐTTT (Knowledge Graph Embedding – KGE) [36] sang môi trường đa phương thức. Hiện nay, hoàn thiện ĐTTT đa phương thức thường tập trung vào hai hướng phát triển sau đây:

- Hướng thứ nhất thiết kế các mô hình hoàn thiện ĐTTT đa phương thức là tăng cường biểu diễn thực thể bằng cách dựa trên thông tin đa phương thức (hình ảnh, văn bản) của chúng.
- Hướng thứ hai là cải thiện quá trình lấy mẫu âm của các thông tin đa phương thức.

1.3.1 Tiếp cận biểu diễn thực thể với thông tin đa phương thức

Bốn mô hình hoàn thiện ĐTTT đa phương thức điển hình theo tiếp cận biểu diễn thực thể với thông tin đa phương thức là IKRL, TBKGC, TransAE, VBKGC.

A. Mô hình IKRL

Theo hướng này, mô hình hoàn thiện ĐTTT đa phương thức đầu tiên được R. Xie và cộng sự [43] đề xuất và được gọi là mô hình IKRL (Image-embodied Knowledge Representation Learning). Trong mô hình IKRL, biểu diễn tri thức được học với cả các dữ kiện bộ ba và ảnh. Vì ảnh của thực thể có thể cung cấp thông tin trực quan cho nên ảnh đóng vai trò quan trọng trong việc học biểu diễn tri thức. Trong thực tế, mỗi thực thể có thể tồn tại nhiều ảnh đi kèm để cung cấp thông tin quan trọng giúp mô tả trực quan về hình dáng, hành vi của thực thể này. Quy trình xây dựng mô hình IKRL như sau:

- Đầu tiên, một bộ mã hóa ảnh bao gồm một mô-đun biểu diễn nơ-ron thần kinh và một mô-đun chiếu được thi hành để tạo ra biểu diễn dựa trên hình ảnh cho từng trường hợp.
- Tiếp theo, biểu diễn dựa trên ảnh tổng hợp cho từng thực thể được xây dựng bằng cách xem xét chung tất cả các trường hợp ảnh của nó bằng phương pháp dựa trên sự chú ý (attention-based method).
- Cuối cùng, học kết hợp các biểu diễn tri thức bằng các phương pháp dựa trên dịch chuyển (translation-based methods).

Hàm tính điểm trong IKRL cũng theo ý tưởng từ hàm tính điểm trong mô hình TransE [36] như Công thức *(1-8)* dưới đây.

$$f(h,r,t) = f_{SS} + f_{SI} + f_{IS} + f_{II} \qquad (1\text{-}8)$$

trong đó,

- Hàm f_{SS} được xác định bằng các biểu diễn dựa trên cấu trúc của các thực thể đầu và thực thể cuối.

- Hàm f_{II} được xác định bằng biểu diễn dựa trên ảnh của các thực thể đầu và thực thể cuối.
- Hàm f_{SI} và hàm f_{IS} xác định bằng cả biểu diễn dựa trên cấu trúc và biểu diễn dựa trên ảnh của các thực thể đầu và thực thể cuối tương ứng.

B. Mô hình TBKGC

Mô hình hoàn thiện ĐTTT đa phương thức TBKGC (Translation-Based for Knowledge Graph Completion) do H. M. Sergieh và cộng sự đề xuất [44]. Mô hình này được xây dựng bằng cách mở rộng thông tin ảnh và văn bản từ mô hình IKRL:

- Ngoài thông tin biểu diễn ảnh, một loại biểu diễn bên ngoài khác (nhúng ngôn ngữ cho các thực thể ĐTTT) được tích hợp để bổ sung thêm thông tin đa phương thức.
- Một mạng nơ-ron đơn giản được xây dựng và cho phép mở rộng kiến trúc một cách dễ dàng.
- Một hàm tính điểm được bổ sung để xem xét các biểu diễn đa phương thức của các thực thể trong ĐTTT.

C. Mô hình TransAE

Mô hình hoàn thiện ĐTTT đa phương thức TransAE được Z. Wang và cộng sự [48] đề xuất khi kết hợp giữa bộ mã hóa tự động đa phương thức (Multimodal Autoencoder) và mô hình TransE [35]. Trong mô hình TransAE, lớp ẩn của các bộ mã hóa tự động được sử dụng như là biểu diễn của các thực thể trong mô hình TransE. Do đó, nó không chỉ mã hóa tri thức có cấu trúc mà còn mã hóa cả tri thức đa phương thức, như tri thức ảnh và tri thức văn bản trong biểu diễn cuối cùng. Hàm tính điểm tổng trong mô hình TransAE được xây dựng như sau:

$$f(h,r,t) = f_S + f_{M1} + f_{M2} + f_{SM} + f_{MS} \qquad (1\text{-}9)$$

trong đó,

- f_S là hàm tính điểm của bộ ba dựa trên cấu trúc.
- f_{M1} và f_{M2} là các hàm tính điểm dựa trên tri thức đa phương thức. f_{M1} tính toán quan hệ giữa các biểu diễn đa phương thức của thực thể đầu và thực thể cuối khi được chiếu vào không gian có cấu trúc. f_{M2} áp dụng trên tổng của các nhúng đa phương thức và các nhúng có cấu trúc của các thực thể đầu và thực thể cuối.

- f_{SM} và f_{MS} để đảm bảo các biểu diễn có cấu trúc và các biểu diễn đa phương thức được học trong cùng một không gian. Các hàm tính điểm được xây dựng dựa theo hàm tính điểm trong [43].

D. Mô hình VBKGC

Mô hình hoàn thiện ĐTTT đa phương thức VBKGC (VisualBERT-enhanced Knowledge Graph Completion) được Y. Zhang và W. Zhang đề xuất [45] theo tiếp cận dựa trên tinh chỉnh cho các nhúng đa phương thức. Mô hình này cho phép nắm bắt thông tin đa phương thức hợp nhất sâu sắc cho các thực thể và tích hợp chúng vào trong quá trình hoàn thiện ĐTTT. Cụ thể là:

- Mô hình VBKGC là mô hình dựa trên nhúng và áp dụng VisualBERT như một bộ mã hóa đa phương thức để nắm bắt các đặc trưng đa phương thức hợp nhất của các thực thể.
- Một phương pháp lấy mẫu âm mới được xây dựng cho các kịch bản đa phương thức làm phù hợp với các kịch bản đa phương thức và có thể căn chỉnh các nhúng khác nhau cho các thực thể.

1.3.2 Tiếp cận lấy mẫu âm

Phương pháp lấy mẫu âm (Negative Sampling) trong biểu diễn ĐTTT [46], [47] là một kỹ thuật được sử dụng rộng rãi trong quá trình huấn luyện các mô hình nhúng ĐTTT, nhằm mục đích tạo ra các bộ ba âm bằng cách thay thế ngẫu nhiên các thực thể để tạo ra độ tương phản giữa mẫu dương và mẫu âm trong quá trình huấn luyện. Phương pháp này giúp cho mô hình nhúng ĐTTT đưa ra điểm số cao hơn cho các bộ ba dương.

Hai mô hình hoàn thiện ĐTTT đa phương thức điển hình theo tiếp cận lấy mẫu âm là MMRNS và MANS.

A. Mô hình MMRNS

Mô hình hoàn thiện ĐTTT đa phương thức MMRNS (MultiModal Relation-enhanced Negative Sampling) được D. Xu và cộng sự đề xuất [48]. Một cơ chế chú ý đa phương thức (Knowledge-guided Cross-modal Attention, KCA) được thiết kế, cho phép tích hợp đa quan hệ của cùng một thực thể để ước lượng các trọng số chú ý hai chiều của các đặc trưng đa ngữ nghĩa. Chi tiết, hai phần sau đây được xây dựng trong mô hình:

- Phần thứ nhất dùng để tóm tắt các tín hiệu đa phương thức thông qua sự chú ý tương hỗ đối với các đặc trưng không liên quan đến quan hệ.
- Phần thứ hai dùng để suy luận các yếu tố chú ý đa phương thức theo hai hướng với các quan hệ nhúng cho các đặc trưng có hướng dẫn quan hệ.

Dựa trên cơ chế chú ý đa phương thức KCA, sử dụng một hàm mất mát tương phản để xây dựng một bộ lấy mẫu ngữ nghĩa tương phản, nhằm mục đích học biểu diễn ngữ nghĩa tương đồng, ngữ nghĩa khác biệt giữa các mẫu âm và các mẫu dương để ước lượng phân phối mẫu. Mô hình hoàn thiện ĐTTT MMRNS xác định các mẫu âm bằng cách sử dụng kết hợp dữ liệu đa phương thức và các quan hệ ĐTTT phức tạp nhằm tăng cương biểu diễn ngữ nghĩa của các thực thể.

B. Mô hình MANS

Mô hình hoàn thiện ĐTTT đa phương thức MANS (Modality-Aware Negative Sampling) được Y. Zhang và cộng sự đề xuất [25]. Kiến trúc chính của mô hình bao gồm hai phần:

- Bộ mã hóa ảnh dùng để nắm bắt các đặc trưng ảnh của thực thể và chiếu chúng lên cùng không gian biểu diễn của các nhúng cấu trúc. Đối với các thực thể có nhiều hơn một ảnh, hàm trung bình được sử dụng để thu được một đặc trưng ảnh.
- Hàm tính điểm bao gồm bốn phần nhằm mục đích tính toán các nhúng trong cùng một không gian véc-tơ. Tất cả nhúng ảnh và nhúng cấu trúc đều được xem xét trong hàm tính điểm.

Phương pháp chính của mô hình MANS là dựa trên việc lấy mẫu âm của các ảnh, ký hiệu là MANS-V. Mô hình MANS-V nhằm mục đích lấy mẫu âm các nhúng ảnh từ những cái không thuộc về thực thể hiện tại để huấn luyện mô hình giúp xác định các đặc điểm hình tương ứng với từng thực thể. Mô hình cho phép đạt được sự căn chỉnh phương thức giữa nhúng có cấu trúc và nhúng ảnh. Ba phiên bản mô hình bổ sung được đề xuất dựa trên việc kết hợp mô hình MANS-V với tỷ lệ β các mẫu âm khác nhau: lấy mẫu âm hai giai đoạn (MANS-T); lấy mẫu âm lai (MANS-H) và lấy mẫu âm thích ứng (MANS-A).

1.3.3 Tiếp cận vấn đề mất cân bằng và thiếu dữ liệu

Phần này giới thiệu một số mô hình hoàn thiện ĐTTT đa phương thức theo các hướng tiếp cận khác nhau xử lý mất cân bằng và thiếu dữ liệu.

A. Mô hình MACO

Một số mô hình hoàn thiện ĐTTT đa phương thức thường bỏ qua hoặc chỉ sử dụng các giải pháp đơn giản (như khởi tạo ngẫu nhiên) để vấn đề thiếu dữ liệu đa phương thức (modality-missing problem) dẫn đến việc gây nhiễu và giảm hiểu năng của các mô hình hoàn thiện ĐTTT đa phương thức.

Nhằm khắc phục vấn đề này, Zhang và cộng sự [49] giới thiệu mô hình MACO (Modality Adversarial and Contrastive Framework) theo cách mục tiêu sau:

- Tạo ra các đặc trưng hình ảnh bị thiếu (missing visual features) dựa trên thông tin cấu trúc của thực thể. Điều này giúp làm giàu thông tin ĐTTT đa phương thức và cải tiến hiệu năng của mô hình hoàn thiện ĐTTT đa phương thức.
- Cải thiện chất lượng của đặc trưng được tạo ra thông qua việc kết hợp các cơ chế học đối nghịch (adversarial learning) và học tương phản (constrastive learning).

Mô hình MACO được cấu tạo bởi ba thành phần chính:

1. Bộ mã hóa đặc trưng:
    a. Bộ mã hóa cấu trúc (Structural Encoder) sử dụng mạng tích chập đồ thị quan hệ (relational graph convolution network) để thu nhận các đặc trưng cấu trúc của thực thể trong đồ thị.
    b. Bộ mã hóa ảnh (Visual Encoder) sử dụng Vision Transformer (ViT) [50] để thu nhận các đặc trưng ảnh của các thực thể.
2. Huấn luyện đối nghich đa phương thức (Modality – Adversarial Traning)
    a. Sử dụng một bộ phát G (Generator) và một bộ phân biệt D (Discriminator) trong khuôn khổ mạng đối kháng tạo sinh (Generative Adversarial Network, GAN).
    b. Bộ phát G: để tạo ra các đặc trưng ảnh phù hợp cho một thực thể nhất định dựa trên đặc trưng cấu trúc của nó.
    c. Bộ phân biệt D: dùng để phân biệt xem một cặp đặc trưng cấu trúc và đặc trưng ảnh có tương thích hay không.

d. Sau đó bộ phát G và bộ phân biệt D được huấn luyện theo kiểu đối kháng.

3. Hàm mất mát tương phản đa phương thức (Cross-modal Contrastive Loss):
    a. Thiết kế để tăng cường chất lượng của đặc trưng ảnh được tạo ra và của thiện sự ổn định của quá trình huấn luyện mạng GAN.
    b. Tối đa hóa thông tin hỗ trợ giữa đặc trưng cấu trúc và đặc trưng ảnh được tạo ra từ cùng một thực thể, coi chúng là một cặp đúng.

B. Mô hình NativE

Hiện nay, các ĐTTT đa phương thức trong thế giới thực đối mặt với hai thách thức:

- Vấn đề đa dạng: Trên thực tế, thông tin đa phương thức thường bao gồm nhiều loại khác nhau (ảnh, văn bản, số, âm thành, video), nhưng các phương pháp hoàn thiện ĐTTT chỉ tập trung vào các phương thức phổ biến như hình ảnh và văn bản. Điều này có thể hạn chế khả năng tổng quát hóa khi ĐTTT đa phương thức bao gồm nhiều phương thức hơn.
- Vấn đề mất cân bằng: Do sự phân phối thông tin đa phương thức không đồng đều giữa các thực thể.

Zhang và cộng sự [51] đề xuất mô hình NativE để giải quyết hai thách thức bên trên. Mô hình NativE bao gồm hai mô-đun chính:

1. Mô-đun kết hợp thích ứng kép hướng quan hệ (Relation-guided Dual Adaptive Fusion – ReDAF): Mô-đun này cho phép kết hợp thích ứng bất kỳ phương thức nào với sự hướng dẫn của quan hệ. ReDAF sử dụng một tập hợp các trọng thích ứng (adaptive weights) cho các phương thức khác nhau. Hơn nữa, mô-đun cho phép điều chỉnh linh hoạt các trọng số này khi thông tin phương thức bị mất cân bằng. "Nhiệt độ" theo quan hệ được sử dụng để điều chỉnh thêm sự phân bố trọng số, cung cấp ngữ cảnh quan hệ cho biểu diễn thực thể. ReDAF là một mô-đun kết hợp phương thức linh hoạt, có khả năng xử lý số lượng phương thức đầu vào không giới hạn.
2. Mô-đun huấn luyện đối kháng phương thức hợp tác (Collaborative Modality Adversarial Training – CoMAT) được thiết kế để tăng cường thông tin phương

thức bị mất cân bằng. CoMAT cải thiện các nhúng đa phương thức thông qua huấn luyện đối kháng, sử dụng dữ liệu phương thức hợp tác cụ thể cho từng thực thể để cân bằng phân phối thông tin đa phương thức. Hơn nữa, trong thiết kế của CoMAT, hàm tính điểm của mô hình hoàn thiện ĐTTT RotatE được sử dụng như một bộ phân biệt.

Một bộ tiêu chuẩn mới với tên gọi là WildKGC được xây dựng để đánh giá hiệu năng của mô hình NativE.

C. Mô hình MNFormer

Theo Wang và cộng sự [52], một số mô hình hoàn thiện ĐTTT đa phương thức chỉ tập trung vào việc tận dụng sự kết hợp đa phương thức để tăng cường khả năng biểu diễn của các thực thể và dự đoán liên kết mà chưa tính đến, một thách thức lớn là sự không đồng nhất giữa phương thức cấu trúc (structural modality) và phương thức ngữ nghĩa (semantics modality). Chính những điều này có thể làm giảm hiệu năng của các mô hình hoàn thiện ĐTTT. Để giải quyết các vấn đề này, mô hình MNFormer được đề xuất với một số đặc điểm:

1. MNFormer là một khung Transformer đa lớp mới được thiết kế để thu thập thông tin cấu trúc và ngữ nghĩa đồng thời tránh các vấn đề không đồng nhất.
2. MNFormer cho phép tích hợp đầy đủ cả đường dẫn láng giềng đa bước (multi-hop neighbor paths) và nhúng hình ảnh – văn bản.
3. Thay vì sử dụng các nhúng cấu trúc được tạo ngẫu nhiên, MNFormer biểu diễn thông tin cấu trúc của thực thể bằng cách sử dụng các phương thức hình ảnh và văn bản từ các nút láng giềng đa cấp của thực thể nguồn.

Mô hình MNFormer bao gồm ba bước chính:

1. Tiền huấn luyện ngữ nghĩa (Semantic Pre-training):
   - Sử dụng các mô hình đã được tiền huấn luyện như Vision Transformer (ViT) [50] và BERT [53] để trích xuất các đặc trưng từ ảnh và mô tả văn bản của các thực thể.
   - Các nhúng ảnh và văn bản được biểu diễn dưới dạng trung bình cộng của các đầu ra lớp cuối cùng từ ViT và BERT.

- Xây dựng phương thức kết hợp (fusion modality) được giới thiệu làm thuộc tính nhân tạo của thực thể. Phương thức này làm hàm mục tiêu cho thức thể cuối.
- Áp dụng chiến lược tìm kiếm đường dẫn để giới hạn số lượng và chọn các láng giềng cung cấp thông tin bổ sung có giá trị nhất (dựa trên độ tương đồng cosine).

2. Kết hợp láng giềng đa bước (Multi-hop Neigbor Fusion – MNF):
   - Là các mô-đun mã hóa được thiết kế để khám phá và bổ sung các đặc trưng ngữ nghĩa của phương thức ảnh, văn bản bằng cách kết hợp thông tin từ các láng giềng.
   - Mỗi mô-đun MNF bao gồm ba bước lặp lại: tự chú ý (self-attention), tương hỗ chú ý (mutual-attention) và chú ý quan hệ (relation-attention).
3. Giải mã đa phương thức:
   - Tích hợp đầu ra của tất cả các mô-đun MNF và sử dụng một lớp Transformer để dự đoán nhúng ngữ nghĩa chung của thực thể đích.
   - Thiết kế một hàm mất hướng (semantic direction loss) để tăng cường hiệu năng khớp bằng cách tăng độ tương đồng ngữ nghĩa giữa thực thể nguồn và thực thể đích chính xác trong phương thức tương ứng.

Một ưu điểm lớn nhất của MNFormer là giải quyết vấn đề kết hợp không đủ giữa các phương thức ngữ nghĩa và cấu trúc trong ĐTTT đa phương thức.

1.4 Hoàn thiện ĐTTT thời gian

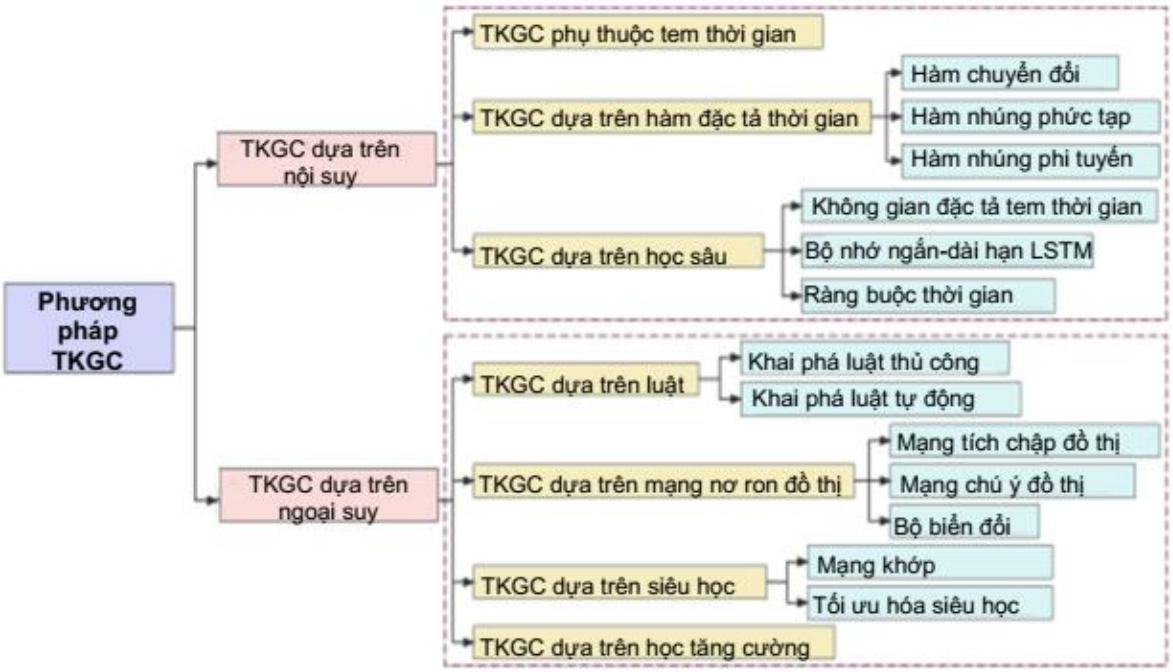
1.4.1 Giới thiệu về Hoàn thiện ĐTTT thời gian

Dựa trên vị trí bị thiếu trong bộ bốn (h, r, t, τ), bài toán hoàn thiện ĐTTT thời gian (Temporal Knowledge Graph Completion: TKGC) được chia ra thành các bài toán cụ thể hơn: dự đoán thực thể, dự đoán quan hệ và dự đoán thời gian. Dự đoán thực thể và quan hệ còn được gọi là dự đoán liên kết. Dự đoán thực thể là bài toán dự đoán thực thể bị thiếu tức là trả lời cho câu truy vấn có dạng $(?, r, t, \tau)$ hoặc $(h, r, ?, \tau)$, trong đó ký hiệu "?" biểu diễn thực thể còn thiếu cần dự đoán. Dự đoán quan hệ và dự đoán thời gian được định nghĩa tương tự với câu truy vấn tương ứng có dạng $(h, ?, t, \tau)$ và $(h, r, t, ?)$.

Kế thừa các mô hình hoàn thiện ĐTTT, về phổ biến, các mô hình hoàn thiện ĐTTT thời gian cũng xuất phát từ thành phần nhúng và chúng thường là phiên bản cải tiến một mô hình hoàn thiện ĐTTT hiện có để phù hợp với ĐTTT thời gian.

1.4.2 Khung phân loại các phương pháp Hoàn thiện ĐTTT thời gian

J. Wang và cộng sự [54] cung cấp một khung phân loại các phương pháp hoàn thiện ĐTTT thời gian như được minh họa ở Hình 1.9.



Hình 1.9. Khung phân loại các phương pháp hoàn thiện ĐTTT thời gian [54].

J. Wang và cộng sự [54] phân loại các phương pháp TKGC dựa trên việc chúng có dự báo các dữ kiện trong tương lai hay không thành hai nhóm phương pháp gồm dựa trên nội suy (Interpolation-based TKGCs) và dựa trên ngoại suy (Extrapolation-based TKGCs) như được minh họa ở Hình 1.9.

Các phương pháp hoàn thiện ĐTTT thời gian dựa trên nội suy thường hoàn thiện mục còn thiếu bằng cách phân tích tri thức đã biết trong ĐTTT thời gian và được phân loại chi tiết hơn thành ba loại dựa trên cách chúng xử lý thông tin thời gian như sau [54]:

1. Các phương pháp hoàn thiện ĐTTT thời gian phụ thuộc tem thời gian (Timestamps dependent-based TKGC) không áp đặt các phép toán lên tem thời gian.

2. Các phương pháp hoàn thiện ĐTTT thời gian dựa trên hàm đặc tả thời gian (Timestamps-specific functions-based TKGC) áp dụng các hàm đặc tả thời gian để có được nhúng của tem thời gian hoặc sự tiến hóa của các thực thể và quan hệ. Nhóm này được chia thành các nhóm con là Hàm chuyển đổi (Transformation Functions), Hàm nhúng phức tạp (Complex Embedding Functions), Hàm nhúng phi tuyến (Non-linear Embedding Functions).
3. Các phương pháp hoàn thiện ĐTTT thời gian dựa trên học sâu (Deep learning-based TKGC) sử dụng các thuật toán học sâu để mã hóa thông tin thời gian và điều tra. Nhóm này được chia thành các nhóm con là Không gian đặc tả tem thời gian (Timestamps-Specific Space), Bộ nhớ dài ngắn hạn (Long Short-Term Memory), Ràng buộc thời gian (Temporal Constraint).

Các phương pháp dựa trên ngoại suy tập trung vào các ĐTTT thời gian liên tục, cho phép dự đoán các sự kiện trong tương lai bằng cách học nhúng của các thực thể và quan hệ từ các ảnh chụp nhanh lịch sử và chúng được phân loại thành bốn nhóm con sau đây [54]:

1. Các phương pháp hoàn thiện ĐTTT thời gian dựa trên luật (Rule-based TKGC) áp dụng các luật logic để lý giải các sự kiện trong tương lai. Nhóm này được chia thành các nhóm con là Khai phá luật thủ công (Manual Rule Mining), Khai phá luật tự động (Automatic Rule Mining).
2. Các phương pháp hoàn thiện ĐTTT thời gian dựa trên mạng nơ-ron đồ thị (Graph neural network-based TKGC) thường sử dụng GNN (Graph neural network) và RNN (Recurrent neural network) để khám phá thông tin về cấu trúc và thời gian trong TKG. Nhóm này được chia thành các nhoims con là Mạng tích chập đồ thị (Graph convolutional network), Mạng chú ý đồ thị (Graph attention network), Bộ biến đổi (Transformer).
3. Các phương pháp hoàn thiện ĐTTT thời gian dựa trên siêu học (Meta learning-based TKGC) thiết kế metalearner để hướng dẫn quá trình học của mô hình. Nhóm này được chia thành các nhóm con là Mạng khớp (Matching network), Tối ưu hóa siêu học (Meta-optimization)
4. Các phương pháp hoàn thiện ĐTTT thời gian dựa trên học tăng cường (Reinforcement learning-based TKGC) giới thiệu chiến lược học tăng cường để đảm bảo rằng mô hình đạt được mục tiêu huấn luyện của mình tốt hơn.

J. Wang và cộng sự [54] trình bày chi tiết về các phương pháp hoàn thiện ĐTTT thời gian trong khung phân loại trên đây.

Hai mô hình hoàn thiện ĐTTT thời gian dựa trên nhúng lịch đại DE-RotatE, DE-RotatE-sinc do luận án đề xuất và các mô hình hoàn thiện ĐTTT thời gian DE-SimplE là thuộc nhóm con các phương pháp Hàm chuyển đổi và thuộc vào nhóm các phương pháp Nội suy đặc tả thời gian.

Ngoài ra, B. Cai và cộng sự cũng cung cấp một khung phân loại năm nhóm phương pháp TKGC [55].

1.5 Các độ đo đánh giá hiệu năng mô hình hoàn thiện ĐTTT

T. Shen và cộng sự [9] cũng hệ thống hóa các độ đo đánh giá hiệu năng các mô hình hoàn thiện ĐTTT. Ba độ đo khá đơn giản nhưng rất phổ biến để đánh giá hiệu năng của một mô hình, thuật toán hoàn thiện ĐTTT là $Hits@k$, Mean Rank (MR), Mean Reciprocal Rank (MRR); luận án cũng sử dụng ba độ đo này để đánh giá các mô hình hoàn thiện ĐTTT.

Trong các công thức dưới đây liên quan tới ba độ đo đánh giá, Q là tổng số tất cả các bộ ba trong tập dữ liệu kiểm thử hoặc tập dữ liệu xác thực.

1.5.1 Độ đo Hit@k

Độ đo $Hits@k$ (thường là $k = 1;\ 3;\ 10$) chỉ ra xác suất dự đoán đúng trong k bộ ứng viên hàng đầu bởi mô hình. Công thức tính toán được mô tả trong Công thức (1-10).

$$Hit@k = \frac{|\{q \in Q: q < k\}|}{|Q|} \qquad (1\text{-}10)$$

trong đó:

- q: số lượng mục dự đoán đúng bởi mô hình nằm trong k kết quả đầu tiên được xếp hạng theo điểm số dự đoán của mô hình.
- Giá trị của k thường nhỏ như $k \in \{1;\ 3;\ 5;\ 10\}$.
- Giá trị của chỉ số $Hit@k$ nằm từ 0 cho đến 1. Giá trị $Hit@k$ càng lớn, thì thuật toán hoạt động càng tốt.

Độ đo $Hit@k$ biểu diễn khả năng của thuật toán, mô hình dự đoán chính xác quan hệ giữa bộ ba, nghĩa là giá trị khả năng của mô hình, thuật toán hoàn thiện ĐTTT dự đoán các bộ ba chính xác. Với $Hits@1$ – mô hình chỉ được tính điểm nếu dự đoán đúng đứng ở vị trí đầu, $Hits@10$ – mô hình dự đoán đúng nằm trong top 10 vẫn được

chấp nhận. Đây là một chỉ số đánh giá không thể thiếu đối với các thuật toán, mô hình hoàn thiện ĐTTT.

1.5.2 Độ đo xếp hạng trung bình MR

Độ đo xếp hạng trung bình MR (Mean Rank) trong mô hình hoàn thiện ĐTTT là một độ đo hiệu năng phổ biến. Nó giúp đánh giá khả năng dự đoán chính xác các quan hệ bộ ba trong ĐTTT.

$$MR = \frac{1}{|Q|}\sum_{i=1} Rank_i \qquad (1\text{-}11)$$

trong đó:

- $Rank_i$: Hạng của bộ ba thứ i khi dự đoán. Giá trị được tính dựa trên vị trí thực tế của thực thể đúng sau khi mô hình đã xếp hạng tất cả các dự đoán.

Giá trị hạng $Rank_i$ là thứ hạng của bộ ba đúng, được xác định bằng cách thay thế thực thể hoặc quan hệ trong bộ ba và sắp xếp dự đoán theo điểm số từ cao xuống thấp. Chỉ số Mean Rank (MR) là giá trị trung bình của tất cả các hạng. Giá trị này càng nhỏ thì càng tốt, cho thấy mô hình dự đoán đúng với vị trí cao hơn.

1.5.3 Độ đo xếp hạng tương hỗ trung bình MRR

Độ đo xếp hạng tương hỗ trung bình MRR (Mean Reciprocal Rank) là một độ đo phổ biến dùng để đánh giá hiệu năng của các hệ thống dự đoán, đặc biệt trong bài toán hoàn thiện ĐTTT. Công thức tính toán được đưa ra trong Công thức (1-12).

$$MRR = \frac{1}{|Q|}\sum_{i=1} \frac{1}{Rank_i} \qquad (1\text{-}12)$$

trong đó:

- $Rank_i$: Thứ hạng của dự đoán đúng đầu tiên trong tập kết quả cho truy vấn thứ i. Nếu không có dự đoán đúng, giá trị Reciprocal Rank (RR) cho truy vấn đó là 0.

MRR là một độ đo thường được sử dụng để đánh giá hiệu quả của thuật toán tìm kiếm. Đây là giá trị trung bình của nghịch đảo thứ hạng của các dự đoán đúng đầu tiên. Chỉ số cho phép đánh giá các bộ ba dự đoán dựa trên việc xem xét nó đúng

hay không. Nếu bộ ba dự đoán đầu tiên mà đúng thì điểm số của nó là 1 và điểm số dự đoán đúng thứ hai là $\frac{1}{2}$, cứ tiếp tục như vậy. Với bộ ba thứ n thì điểm số của nó là $\frac{1}{n}$ và khi đó giá trị cuối cùng của MRR sẽ là tổng giá trị của tất cả điểm số. Giá trị MRR càng lớn thì hiệu quả dự đoán của mô hình càng tốt. Nếu chỉ trả về kết quả "Top 1" của ĐTTT, độ chính xác hoặc thu hồi sẽ kém, do vậy nhiều kết quả được trả về để tránh sai số lớn trong kết quả dự đoán.

1.5.4 Các độ đo khác

Một bộ phận các mô hình hoàn thiện ĐTTT dưới dạng bài toán phân lớp, vì vậy, một số độ đo dựa trên ma trận nhầm lẫn trong phân lớp như độ đo chính xác thống kê (accuracy), bộ độ đo hồi tưởng, chính xác (recall, precision) và F1 cũng được sử dụng. Ngoài ra, độ đo chính xác trung bình MAP (Mean Average Precision) cũng được sử dụng.

1.6 Xu hướng nghiên cứu hoàn thiện ĐTTT và một số luận án Tiến sỹ

1.6.1 Xu hướng nghiên cứu về hoàn thiện ĐTTT

T. Shen và cộng sự [9] nêu ra tám xu hướng (triển vọng) nghiên cứu chính về hoàn thiện ĐTTT như sau:

1. Sử dụng thông tin bổ sung, đặc biệt là thông tin bên ngoài ĐTTT (như luật và tài nguyên văn bản bên ngoài) cho hoàn thiện ĐTTT nhằm đạt được mục tiêu làm giàu tri thức bên trong ĐTTT.
2. Hoàn thiện ĐTTT dựa trên luật có triển vọng do các phương pháp dựa trên luật hoạt động rất tốt và là một giải pháp thay thế cạnh tranh cho các mô hình nhúng phổ biến.
3. Sử dụng các mô hình ngôn ngữ huấn luyện trước (Pre-trained Language Model) mới vào hoàn thiện ĐTTT là khả thi vì cho phép khai thác được khả năng (được xem là vô hạn) khi kết hợp hiệu quả các mô hình ngôn ngữ giàu ngữ nghĩa để đạt được các nhúng chất lượng cao cũng như nắm bắt thông tin ngữ nghĩa phong phú.
4. Hoàn thiện ĐTTT cho các ĐTTT miền cụ thể thực sự có ý nghĩa với giá trị ứng dụng thực tế tuyệt vời, sẽ được phát triển hơn nữa trong tương lai do chỉ

có một vài mô hình hoàn thiện ĐTTT hiện có cho các miền và nhu cầu cụ thể (ví dụ, các mô hình hoàn thiện ĐTTT CommonSense và ĐTTT siêu quan hệ).

5. Nắm bắt tương tác giữa các ĐTTT riêng biệt là rất hữu ích cho hoàn thiện ĐTTT. Một loạt các tác vụ đã xuất hiện với nhu cầu tương tác giữa các ĐTTT khác nhau, chẳng hạn như căn chỉnh thực thể, loại bỏ mơ hồ của thực thể, căn chỉnh thuộc tính, v.v., có thể tạo ra một số ý tưởng truyền cảm hứng bằng cách nghiên cứu nhúng thống nhất và hoàn thiện các loại và cấu trúc tri thức khác nhau. Nhân tiện, công trình của hoàn thiện ĐTTT liên quan đến ĐTTT đa ngôn ngữ là chưa đủ, nên việc khởi động hướng nghiên cứu này để bổ sung cho các ĐTTT đa ngôn ngữ được yêu cầu trong các ứng dụng thực tế là rất đáng giá.
6. Chọn không gian phù hợp hơn mô hình hóa cho hoàn thiện ĐTTT nhằm khắc phục hạn chế từ không gian mô hình hóa nhúng ĐTTT đang bị hạn hẹp (xem TransE và các phiên bản mở rộng). Cần có các nghiên cứu khám phá không gian véc-tơ nhúng hiệu quả và hợp lý hơn cho hoàn thiện ĐTTT để triển khai phép biến đổi tịnh tiến cơ bản hoặc phân tích ten-xơ các thực thể và quan hệ dễ dàng mô hình hóa các loại thực thể và quan hệ phức tạp, cùng với nhiều thông tin cấu trúc khác nhau.
7. Sử dụng học tăng cường trong hoàn thiện ĐTTT do học tăng cường (Reinforcement Learning) đã có nhiều ứng dụng hiệu quả trong dịch máy, tóm tắt văn bản, phân tích cú pháp ngữ nghĩa, v.v. Trong tương lai, khung học máy tăng cường có thể được phát triển để cùng suy luận với bộ ba ĐTTT và đề cập đến văn bản, điều này có thể giúp giải quyết tình huống có vấn đề khi ĐTTT không có đủ các đường dẫn suy luận.
8. Học đa nhiệm cho hoàn thiện ĐTTT để khai thác ưu thế vượt trội của học đa nhiệm so với học từng tác vụ riêng. Mô hình hoàn thiện ĐTTT có thể học và huấn luyện với các tác vụ dựa trên ĐTTT khác (hoặc các tác vụ phụ trợ được thiết kế phù hợp) theo một khuôn học đa nhiệm, có thể đạt được cả khả năng biểu diễn và khái quát hóa bằng cách chia sẻ thông tin chung giữa các tác vụ trong quá trình học, để đạt được hiệu năng tổng thể.

1.6.2 Một số luận án Tiến sỹ liên quan trên thế giới

Mục này giới thiệu sơ bộ về một số luận án Tiến sỹ điển hình tham gia vào dòng nghiên cứu trên thế giới theo tám triển vọng nghiên cứu về hoàn thiện ĐTTT.

1. Luận án của R. Biswas [10] có sử dụng thông tin bổ sung từ Danh mục Wikipedia (Xu hướng nghiên cứu 1) vào các miền dữ liệu cụ thể. Đầu tiên, một mô hình nhúng ĐTTT đa bước mới MADLINK được đề xuất cho Dự đoán liên kết; MADLINK xem xét thông tin theo ngữ cảnh của các thực thể bằng cách sử dụng các bước ngẫu nhiên cũng như các mô tả thực thể dạng văn bản của các thực thể. Tiếp đó, một kiến trúc mới khai thác thông tin có trong Mô hình ngôn ngữ nơ-ron theo ngữ cảnh được huấn luyện trước để đề xuất cho Phân lớp bộ ba. Thứ ba, những hạn chế của các mô hình dự đoán loại thực thể hiện đại (SoTA) đã được phân tích và một mô hình phân lớp thực thể mới CAT2Type được đề xuất để khai thác Danh mục Wikipedia; mô hình này cũng được dùng để dự đoán các loại thực thể chưa biết bị thiếu, tức là các thực thể mới được thêm vào trong ĐTTT. Cuối cùng, một kiến trúc mới GRAND được đề xuất để dự đoán các loại thực thể bị thiếu trong ĐTTT bằng cách sử dụng phân lớp đa nhãn, đa lớp và phân cấp bằng cách tận dụng các bước đồ thị chiến lược khác nhau trong ĐTTT. Các mô hình được đề xuất thiết lập rằng Mô hình ngôn ngữ nơ-ron và thông tin theo ngữ cảnh của các thực thể trong ĐTTT cùng với các kiến trúc mạng nơ-ron khác nhau có lợi cho Hoàn thiện ĐTTT. Các kết quả và quan sát đầy hứa hẹn mở ra phạm vi thú vị cho nghiên cứu trong tương lai liên quan đến việc khai thác các mô hình được đề xuất trong ĐTTT cụ thể theo miền như dữ liệu học thuật, dữ liệu y sinh, v.v. Hơn nữa, mô hình dự đoán liên kết có thể được khai thác như một mô hình cơ sở cho tác vụ căn chỉnh thực thể vì nó xem xét thông tin lân cận của các thực thể.
2. A. Boschin [11] cũng sử dụng thông tin bổ sung từ Danh mục Wikipedia (Xu hướng nghiên cứu 1), tập trung vào phương diện triển khai các mạng nơ-ron sâu vào hoàn thiện ĐTTT nhằm giải quyết sự phức tạp khi sử dụng các thư viện học sâu đa dạng. Luận án phát triển thư viện Python TorchKGE để cung cấp một thiết lập độc đáo cho việc triển khai các mô hình nhúng và một mô-đun đánh giá suy luận hiệu quả cao. Thư viện này dựa trên khả năng tăng tốc đồ họa của tính toán ten-xơ do PyTorch cung cấp, tương thích với các thư viện tối ưu hóa phổ biến và có sẵn dưới dạng mã nguồn mở. Sau đó, luận án xem xét việc làm giàu tự động của Wikidata bằng cách nhập các siêu liên kết các trang Wikipedia. Một nghiên cứu sơ bộ cho thấy biểu đồ các bài viết trên Wikipedia dày đặc hơn nhiều so với biểu đồ kiến thức tương ứng trong

Wikidata. Một phương pháp huấn luyện mới liên quan đến các quan hệ và phương pháp suy luận sử dụng các loại thực thể đã được đề xuất và các thử nghiệm cho thấy sự liên quan của phương pháp kết hợp, bao gồm cả trên một tập dữ liệu mới. Cuối cùng, luận án khám phá việc thu thập thực thể tự động như một nhiệm vụ phân lớp phân cấp. Điều đó dẫn đến việc thiết kế một mất mát theo thứ bậc mới được sử dụng để huấn luyện các mô hình dựa trên ten-xơ cùng với một loại bộ mã hóa mới. Các thử nghiệm trên hai tập dữ liệu đã cho phép hiểu rõ về tác động của tri thức trước đó về phân loại lớp có thể có đối với bộ phân lớp nhưng cũng củng cố trực giác rằng có thể học được thứ bậc từ các đặc trưng nếu tập dữ liệu đủ lớn.

3. Luận án của D. Yu [12] nhằm mục tiêu tích hợp thông tin văn bản vào mô hình hóa ĐTTT bằng cách sử dụng Mô hình ngôn ngữ được huấn luyện trước (Xu hướng nghiên cứu 3) trong hai tác vụ là: Thu thập tri thức để nâng cao chất lượng ĐTTT và Ứng dụng ĐTTT trong các hệ thống hỏi đáp. Luận án tiến hành một khung huấn luyện trước cùng học các véc-tơ biểu diễn của ĐTTT và văn bản được thi hành bằng một mô-đun huấn luyện kép ĐTTT-văn bản hỗ trợ lẫn nhau cho trích xuất quan hệ và phân lớp thực thể. Hơn nữa, một mô hình tạo văn bản tăng cường khả năng truy xuất được đề xuất, tận dụng các bộ ba có liên quan ngữ nghĩa từ các ĐTTT nhằm hướng dẫn việc tạo các thực thể bị thiếu trong các bộ ba để hoàn thiện ĐTTT. Sử dụng các ĐTTT để xây dựng các liên kết giữa các đoạn văn bản và thông tin cấu trúc này được sử dụng để xếp hạng lại và cắt tỉa các câu trả lời cho câu hỏi trong hệ thống hỏi đáp.

4. Luận án của F. Wang [13] có mục tiêu chính là giải quyết các vấn đề cơ bản về tính hoàn chỉnh (completeness) và tính nhất quán (consistency) trong các mô hình hoàn thiện ĐTTT qua khám phá sự kết hợp giữa kỹ thuật hoàn thiện ĐTTT và suy luận ngữ nghĩa lặp đi lặp lại (Xu hướng nghiên cứu 2). Luận án của F. Wang có ba đóng góp chính là:

    - Xây dựng mô hình thiện ĐTTT nhận thức lược đồ SICKLE dựa trên việc sử dụng nhiều chiến lược khác nhau ở nhiều cấp độ khác nhau.
    - Tối ưu hóa đường ống hoàn thiện ĐTTT SICKLE qua tính toán, sử dụng điểm tin cậy kết hợp để xếp hạng lại kết quả dự đoán và mở rộng mẫu âm tính có nhận thức lược đồ thành lấy mẫu âm tính thông tin.

- Khám phá tiếp cận thu thập và hoàn thiện tri thức theo kịch bản thực tế trong hệ thống nhật ký mạng để tiến hành lặp đi lặp lại.

5. Luận án của S. Liu [14] tập trung vào hai tác vụ hoàn thiện ĐTTT quy nạp và Tư vấn tăng cường tri thức quy nạp, đề xuất một phương pháp mới dựa trên mạng nơ-ron đồ thị (Graph Neural Netwworks: GNN) để giải quyết các thách thức đối với hai tác vụ trên đây (Xu hướng nghiên cứu 8). Thêm nữa, luận án xem xét lại các độ đo điểm chuẩn đánh giá mô hình hoàn thiện ĐTTT và đề xuất một phương pháp mới tạo thêm các điểm chuẩn cho phép hỗ trợ đánh giá theo kinh nghiệm năng lực nắm bắt các mô hình suy luận trong hoàn thiện ĐTTT.

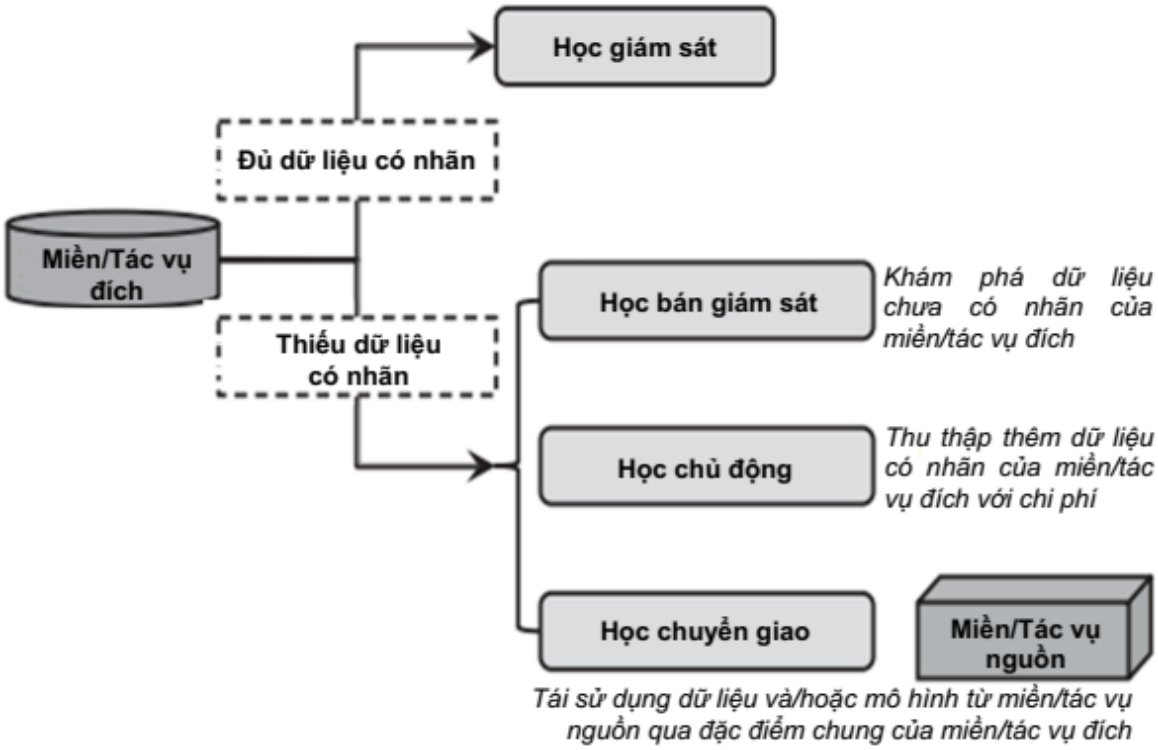
1.7 Liên hệ với nghiên cứu trong luận án

Như đã được đề cập, theo xu hướng nghiên cứu trên thế giới về hoàn thiện ĐTTT, luận án này tập trung vào các mô hình hoàn thiện ĐTTT dựa trên học chuyển giao (Xu hướng nghiên cứu 1), hoàn thiện ĐTTT đa phương thức (Xu hướng nghiên cứu 3, 4) và hoàn thiện ĐTTT thời gian (Xu hướng nghiên cứu 3, 4). Các mục con tiếp theo cung cấp các nội dung nền tảng cho các nghiên cứu của luận án.

1.7.1 Học chuyển giao cho hoàn thiện ĐTTT trong luận án

Định nghĩa 1.6 Học chuyển giao [56].

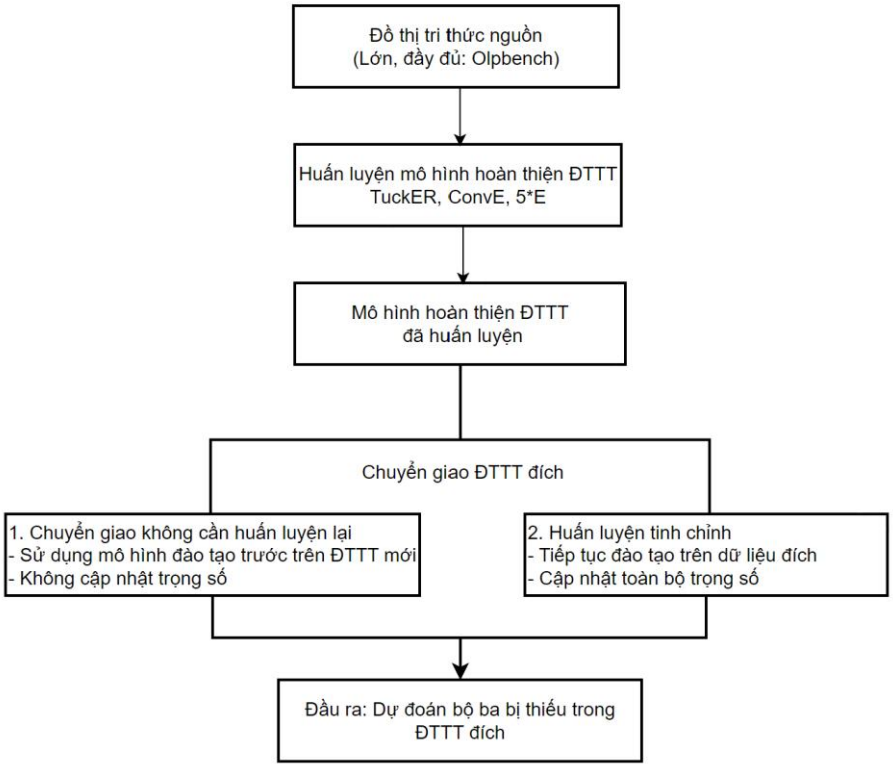
> *Cho một miền nguồn D_S và một tác vụ học tương ứng $\mathcal{T}_S$, một miền đích $\mathcal{D}_T$ và tác vụ học tương ứng $\mathcal{T}_T$. Học chuyển giao là phương pháp học nhằm mục đích cải thiện hàm dự đoán đích $f_T(.)$ trong $\mathcal{D}_T$ khi sử dụng tri thức trong $\mathcal{D}_S$ và $\mathcal{T}_S$, ở đây, $\mathcal{D}_S \neq \mathcal{D}_T$ hoặc $\mathcal{T}_S \neq \mathcal{T}_T$.*



Hình 1.10. Phân biệt học chuyển giao với một số kiểu học máy khác [57]

Lưu ý rằng, điều kiện cốt lõi cho học chuyển giao là miền/tác vụ đích phải có điểm chung với miền/tác vụ nguồn. Điều kiện "đặc điểm chung" để tái sử dụng dữ liệu/mô hình từ miền/tác vụ đích trong Hình 1.10 làm rõ thêm yêu cầu này. Ngoài ra, Hình 1.10 còn cho một đối sánh học chuyển giao với một số kiểu học máy giải quyết các tình huống thiếu dữ liệu mẫu.

Nghiên cứu việc ứng dụng học chuyển giao trong bài toán hoàn thiện ĐTTT đặc biệt trong tình huống các ĐTTT đích có dữ liệu hạn chế, như trong các miền chuyên ngành là rất cần thiết vì phương pháp này cho phép tận dụng tận dụng kiến thức từ đồ thị lớn hơn, đầy đủ hơn, khắc phục được hiện tượng dữ liệu mất cân bằng. Bên cạnh đó các quan hệ và thực thể giữa đồ thị đích và đồ thị nguồn có thể có sự tương đồng nên mô hình học được từ đồ thị nguồn giúp mô hình hiểu ngữ nghĩa tốt hơn, dễ thích nghi với đồ thị mới, tiết kiệm được chi phí, thời gian và tài nguyên khi không phải huấn luyện lại từ đầu.



Hình 1.11. Quy trình chuyển giao trong hoàn thiện ĐTTT (dựa theo Kocijian và cộng sự [17])

Các thành phần có thể chuyển giao trong mô hình học chuyển giao cho hoàn thiện ĐTTT đó là các vec-tơ nhúng thực thể học từ đồ thị nguồn, vec-tơ nhúng của quan hệ đã học để áp dụng cho quan hệ tương tự, các tham số mô hình như trọng số của hàm tính điểm, các mô hình thần kinh, kiến thức ngữ nghĩa từ văn bản như các nhúng từ giúp đại diện tốt cho thực thể, quan hệ, được minh họa trong quy trình học chuyển giao tại Hình 1.11.

Các chiến lược học chuyển giao trong hoàn thiện ĐTTT đó là:

- Chuyển giao không cần huấn luyện lại (zero-shot transfer): sử dụng trực tiếp mô hình học từ đồ thị nguồn cho đồ thị đích.
- Huấn luyện tinh chỉnh (Fine-tuning): dùng mô hình huấn luyện trước trên dữ liệu nguồn và tiếp tục huấn luyện trên dữ liệu đích.
- Học đa nhiệm (Multi-task learning): Học đồng thời trên nhiều đồ thị nguồn/đích.

- Học chuyển giao dựa trên module thích ứng (Adapter-based transfer): Là phương pháp chuyển giao học máy trong đó một mô hình lớn (đã tiền huấn luyện) được giữ nguyên, và ta chỉ thêm các module nhỏ gọi là adapter để tinh chỉnh mô hình cho một tác vụ mới hoặc miền dữ liệu mới.
- Học siêu cấp (Meta – learning): Mô hình học cách học để dễ dàng thích ứng nhanh với ĐTTT mới.

Luận án tập trung vào việc phân tích chi tiết mô hình học chuyển giao cho hoàn thiện ĐTTT GloVe-GRU-KGC do V. Kocijan và T. Lukasiewicz đề xuất năm 2021 [17]. Đề xuất mô hình hoàn thiện ĐTTT qua học chuyển giao BERT/FastText-GRU-KGC được cải tiến từ mô hình GloVe-GRU-KGC theo ý tưởng khai thác tài nguyên nguồn từ mô hình ngôn ngữ BERT/FastText để thay thế cho GloVe và cải tiến thành phần mã hóa trong GloVe-GRU-KGC. Chương 2 của luận án trình bày chi tiết đề xuất này của luận án.

1.7.2 Hoàn thiện ĐTTT đa phương thức trong luận án

Trong hoàn thiện ĐTTT đa phương thức (MMKGC), phân loại bộ ba không chỉ đánh giá độ đúng/ sai dựa trên quan hệ cấu trúc mà còn cần kết hợp dữ liệu đa phương thức như hình ảnh và văn bản mô tả. Điều này làm cho quá trình xây dựng và huấn luyện mô hình trở nên phức tạp hơn so với ĐTTT mở, vốn tập trung vào việc xử lý dữ liệu không chuẩn hóa từ văn bản mà không có sự hỗ trợ từ các phương thức khác. Bên cạnh đó, hoàn thiện ĐTTT đa phương thức yêu cầu mô hình có khả năng xử lý khi thiếu phương thức, trong khi ĐTTT mở (OKG) yêu cầu khả năng khái quát hóa mạnh mẽ khi gặp thực thể/quan hệ mới ngoài phân phối. Một trong những nhiệm vụ quan trọng trong bài toàn hoàn thiện ĐTTT đa phương thức là giải quyết vấn đề mất cân bằng thông tin giữa các phương thức để mô hình sử dụng hiệu quả các thông tin cần thiết cũng như có được thông tin đa phương thức chất lượng cao để tăng hiệu năng của quá trình dự đoán liên kết.

Luận án nghiên cứu, phân tích chi tiết mô hình AdaMF-MAT được Y. Zhang và cộng sự đề xuất năm 2024 [18], qua đó nhận ra được các ý tưởng cải tiến và đề xuất hai mô hình hoàn thiện ĐTTT đa phương thức ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT, trong đó, ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT tương ứng sử dụng mô hình nhúng ảnh Vision Transformer (ViT) và mô hình nhúng văn bản T5. Chương 3 của luận án trình bày chi tiết đề xuất này của luận án.

1.7.3 Hoàn thiện ĐTTT thời gian trong luận án

Hai mô hình DE-DistMult, DE-SimplE thuộc nhóm mô hình hoàn thiện ĐTTT thời gian dựa trên hàm đặc tả thời gian kiểu chuyển đổi do R. Goel và cộng sự đề xuất năm 2020 [19], trong đó, hàm nhúng lịch đại có thể thu được biểu diễn thực thế ở bất kỳ thời gian nào. DE-DistMult là sự kết hợp giữa mô hình nhúng lịch đại DE và mô hình hoàn thiện ĐTTT DistMult còn DE-SimplE là kết hợp giữa mô hình nhúng lịch DE và mô hình hoàn thiện ĐTTT SimplE. Bằng các thực nghiệm R. Goel và công sự đã chỉ ra rằng mô hình DE-SimplE có hiệu năng tốt nhất.

Luận án thực hiện việc phân tích chi tiết mô hình DE-SimplE, qua đó nhận ra được các ý tưởng cải tiến và đề xuất hai mô hình hoàn thiện ĐTTT thời gian là DE-RotatE và DE-RotatE-sinc được cải tiến từ mô hình DE-SimplE theo ý tưởng sử dụng phép quay chiếu thay vì sử dụng phép tịnh tiến chiếu trong DE-SimplE. Chương 4 của luận án trình bày chi tiết đề xuất này của luận án.

1.8 Kết luận Chương 1

Chương đầu tiên của luận án trình bày một cách hệ thống về ĐTTT và hoàn thiện ĐTTT và các nội dung liên quan. Đặc biệt, mối liên hệ giữa các nghiên cứu cùng các chương tương ứng của luận án với các xu hướng nghiên cứu về hoàn thiện ĐTTT đã được giới thiệu khái quát.

Chương 2 của luận án trình bày chi tiết đề xuất mô hình áp dụng học chuyển giao để bổ sung thông tin bên ngoài cho hoàn thiện ĐTTT của luận án. Chương 2 của luận án trình bày chi tiết đề xuất này của luận án

Chương 2. Hoàn thiện ĐTTT dựa trên học chuyển giao

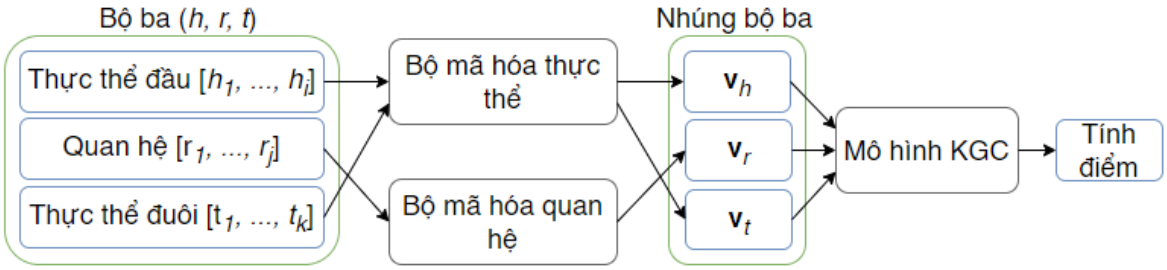
Chương này trình bày đề xuất của luận án về mô hình hoàn thiện ĐTTT qua học chuyển giao BERT/FastText-GRU-KGC được cải tiến từ mô hình GloVe-GRU-KGC do V. Kocijan và T. Lukasiewicz đề xuất [17]. Mục đầu tiên trình bày một cách khái quát mô hình GloVe-GRU-KGC và đưa ra các ý tưởng cải tiến. Mô hình hoàn thiện ĐTTT BERT/FastText-GRU-KGC được trình bày trong mục tiếp theo. Mục thứ ba trình bày quá trình thực nghiệm mô hình đề xuất BERT/FastText-GRU-KGC và đưa ra một số nhận xét hiệu năng của mô hình đề xuất qua kết quả thực nghiệm.

2.1 Mô hình hoàn thiện ĐTTT với học chuyển giao Glove-GRU-KGC

Như đã đề cập, V. Kocijan và T. Lukasiewicz [17] đề xuất một mô hình hoàn thiện ĐTTT qua học chuyển giao (luận án gọi là Glove-GRU-KGC; để thuận lợi, luận án gọi các mô hình kiểu này là KGC-TransLearning, với KGC là một mô hình hoàn thiện ĐTTT bất kỳ nào đó có thể được lựa chọn). Trong phần này, luận án trình bày về kiến trúc của mô hình KGC-TransLearning và cách thức các tác giả sử dụng một mô hình huấn luyện trước để khởi tạo mô hình cho tinh chỉnh. Mô hình KGC-TransLearning thực hiện huấn luyện trước bộ mã hóa dựa trên mạng nơ-ron hồi quy RNN để mã hóa các thực thể và các quan hệ từ các biểu diễn văn bản của chúng sang các véc-tơ nhúng của chúng kết hợp với một mô hình hoàn thiện ĐTTT trên một chuẩn hoàn thiện ĐTTT mở lớn. V. Kocijan và T. Lukasiewicz sử dụng mô hình hoàn thiện ĐTTT và các bộ mã hóa được huấn luyện trước để khởi tạo mô hình và sau đó tinh chỉnh trên tập dữ liệu nhỏ hơn. Cụ thể hơn nữa, các tham số trong mô hình hoàn thiện ĐTTT được chia sẻ giữa tất cả các bộ đầu vào và được sử dụng như một khởi tạo của các tham số giống nhau của mô hình tinh chỉnh. Khi khởi tạo các véc-tơ nhúng đầu vào cụ thể, V. Kocijan và T. Lukasiewicz giới thiệu và so sánh hai cách tiếp cận sau:

- Hướng tiếp cận đầu tiên: các bộ mã hóa của thực thể và quan hệ được huấn luyện trước cũng được sử dụng và huấn luyện trong suốt quá trình tinh chỉnh.
- Hướng tiếp cận thứ hai: các bộ mã hóa của thực thể và quan hệ được sử dụng ngay từ đầu để tính toán các giá trị khởi tạo của tất cả các nhúng của thực thể và quan hệ rồi sau đó loại bỏ chúng.

Mô hình KGC-TransLearning bao gồm hai bộ mã hóa, cụ thể: bộ mã hóa thứ nhất cho các thực thể và bộ mã hóa thứ hai cho các quan hệ. Sau đó, có thể kết hợp với một mô hình hoàn thiện ĐTTT bất kỳ nào đó. Cho bộ ba (h, r, t), bộ mã hóa thực thể được sử dụng để ánh xạ mỗi thực thể đầu h và thực thể đuôi t thành các véc-tơ nhúng $\mathbf{v}_h$, $\mathbf{v}_t$ tương ứng, trong khi bộ mã hóa quan hệ được sử dụng để ánh xạ mỗi quan hệ r thành véc-tơ nhúng $\mathbf{v}_r$ của nó. Sau đó, tất cả các nhúng này được sử dụng như là dữ liệu đầu vào cho bất kỳ một mô hình hoàn thiện ĐTTT nào đó được lựa chọn để đưa ra dự đoán kết quả (tương ứng với kết quả đúng đắn và phù hợp). Hơn nữa, cho phép sử dụng hàm mất mát (loss function) của bất kỳ một mô hình hoàn thiện ĐTTT nào đó. Sơ đồ của mô hình hoàn thiện ĐTTT dựa theo học chuyển giao KGC-TransLearning được mô tả ở Hình 2.1, trong đó, bộ mã hóa thực thể và bộ mã hóa quan hệ được sử dụng để ánh xạ một bộ ba (h, r, t) từ tên của chúng (biểu diễn văn bản) thành các véc-tơ nhúng $(\boldsymbol{v}_h, \boldsymbol{v}_r, \boldsymbol{v}_t)$; các véc-tơ nhúng này được sử dụng như là dữ liệu đầu vào cho một mô hình/thuật toán được lựa chọn để tính toán điểm số cho mỗi bộ ba..Tiếp theo, luận án trình bày một số đặc trưng của từng thành phần trong mô hình KGC-TransLearning.



Hình 2.1. Sơ đồ của mô hình hoàn thiện ĐTTT dựa theo học chuyển giao KGC-TransLearning [17].

2.1.1 Bộ mã hóa

V. Kocijan và T. Lukasiewicz [17] sử dụng hai cách tiếp cận để xây dựng ánh xạ từ một thực thể đến một không gian véc-tơ với số chiều thấp. Cụ thể hai cách tiếp cận này như sau:

- Hướng tiếp cận đầu tiên theo hướng truyền thống. Khi đó, ánh xạ mỗi thực thể và mỗi quan hệ thành nhúng riêng của chính nó, khởi tạo ngẫu nhiên và kết hợp huấn luyện với mô hình. Đây là cách tiếp cận mặc định được sử dụng bởi hầu hết các mô hình hoàn thiện ĐTTT. Hướng tiếp cận này khác biết so với

hướng tiếp cận dựa trên mô hình mạng nơ-ron hồi quy RNN. Đối với hướng này, để thuận tiện ký hiệu mô hình này là NoEncoder.

- Hướng tiếp cận thứ hai được V. Kocijan và T. Lukasiewicz thực nghiệm dựa trên mô hình mạng nơ-ron hồi quy RNN. Bằng cách sử dụng nền tảng mô hình mạng nơ-ron hồi quy RNN, các tác giả xây dựng ánh xạ để biến mỗi biểu diễn văn bản của một thực thể hoặc quan hệ (tên) thành nhúng tương ứng của nó. Hướng tiếp cận này được đặt tên là mô hình mã hóa GRU (Gated Recurrent Unit encoder). Chi tiết hướng tiếp cận này được mô tả qua hai giai đoạn sau:
  - Đầu tiên, mô hình nhúng GloVe (J. Pennington và cộng sự [58]) được sử dụng để gán mỗi từ thành một véc-tơ. GloVe là một mô hình để học các biểu diễn từ dựa trên cả thông tin thống kê toàn bộ và ngữ cảnh cục bộ từ một dữ liệu. Nó là mô hình biểu diễn dùng để xây dựng các nhúng dựa trên một ma trận lưu lại tần suất các từ xuất hiện cùng nhau trong một dữ liệu lớn. Ý tưởng chính được đề xuất là phân rã ma trận này để học biểu diễn các véc-tơ từ đó thu được nghĩa của từ.
  - Tiếp theo, những véc-tơ này được sử dụng làm đầu vào cho bộ mã hóa thực thể, được thực hiện như là một GRU. GRU là một cơ chế cổng trong mạng nơ-ron hồi quy được K. Cho và cộng sự [59] đề xuất vào năm 2014. GRU là một kiểu của mạng nơ-ron hồi quy RNN giúp khắc phục những hạn chế về vấn đề biến mất đạo hàm của mạng này. GRU phổ biến trong lĩnh vực học máy, đặc biệt là để xử lý dữ liệu tuần tự như văn bản vì chúng có thể xử lý hiệu quả các phụ thuộc theo thời gian. GRU tương tự như một LSTM (Long Short-Term Memory) với một cơ chế cổng để nhập hoặc quên một số đặc trưng nhất định nhưng không có véc-tơ ngữ cảnh hoặc cổng đầu ra. Điều đó dẫn đến GRU có ít tham số hơn LSTM.

V. Kocijan và T. Lukasiewicz [17] nhận định rằng, kích thước của mô hình NoEncoder tăng một cách tuyến tính theo số lượng thực thể. Trong khi đó, kích thước của mô hình mã hóa GRU tăng tuyến tính theo số lượng từ vựng. Trong thực tế, đối với các ĐTTT lớn, số lượng của từ vựng thường nhỏ hơn số lượng các thực thể và các quan hệ, do đó hướng tiếp cận thứ hai có thể giảm đáng kể việc sử dụng bộ nhớ.

2.1.2 Chuyển giao giữa các tập dữ liệu

V. Kocijan và T. Lukasiewicz [17] chỉ ra rằng quá trình huấn luyện trước luôn luôn được thực hiện bằng cách sử dụng các bộ mã hóa GRU. Bởi vì khi chuyển giao từ bộ mã hóa NoEncoder đến bất kỳ một bộ mã hóa nào khác đều yêu cầu phải có đối sánh thực thể. Trong quá trình tinh chỉnh được thực hiện bởi bộ mã hóa GRU, các

tham số của nó được khởi tạo từ bộ tham số GRU và được huấn luyện trước. Quá trình này cũng được áp dụng tương tự như cho từ vựng. Tuy nhiên, nếu từ vựng đích bao gồm bất kỳ một từ không được biết trước thì khi đó các nhúng từ của nó sẽ được khởi tạo một cách ngẫu nhiên.

So với khởi tạo của thiết lập NoEncoder, bộ mã hóa GRU huấn luyện trước được sử dụng để tạo ra các giá trị khởi tạo của tất cả các véc-tơ nhúng bằng cách mã hóa các biểu diễn từ của chúng. Khi đó, bất kỳ từ nào chưa biết đều được bỏ qua và thực thể với từ không biết được khởi tạo một cách ngẫu nhiên. Quy trình tương tự như vậy cũng được áp dụng cho các quan hệ.

2.1.3 Mô hình hoàn thiện ĐTTT GloVe-GRU-KGC

Chương 1 đã trình bày ba mô hình hoàn thiện ĐTTT là ConvE (T. Dettmers và cộng sự [32]), TuckER (I. B. Balažević và cộng sự [33]) và 5⋆E (M. Nayyeri và cộng sự [34]). Do ba mô hình hoàn thiện ĐTTT nói trên có hiệu năng tốt và tương đối phù hợp cho nên V. Kocijan và T. Lukasiewicz [17] đề xuất mô hình hoàn thiện ĐTTT dựa trên học chuyển giao GloVe-GRU-KGC với mô hình hoàn thiện ĐTTT nền là ba mô hình hoàn thiện ĐTTT. Khi đó, với mỗi bộ ba (h, r, t) qua bộ mã hóa sẽ thu được các nhúng tương ứng là $\mathbf{v}_h$, $\mathbf{v}_r$ và $\mathbf{v}_t$. Các nhúng này sẽ là dữ liệu đầu vào cho các mô hình hoàn thiện ĐTTT. Khi kết hợp với từng mô hình hoàn thiện ĐTTT nói trên thì quá trình hoạt động đối với từng mô hình hoàn thiện ĐTTT được V. Kocijan và T. Lukasiewicz [17] mô tả như sau:

- Mô hình TuckER gán một hàm tính điểm cho mỗi bộ ba bằng cách nhân các véc-tơ với một nhân ten-xơ $\mathcal{W} \in \mathbb{R}^{d_e \times d_e \times d_r}$, ở đó d_e là chiều của các thực thể và d_r là chiều của các quan hệ, giả định rằng $d_e = d_r$ khi thực nghiệm. Trong suốt quá trình chuyển giao, $\mathcal{W}$ từ mô hình huấn luyện trước được sử dụng để khởi tạo $\mathcal{W}$ trong mô hình tinh chỉnh.
- Mô hình ConvE gán một hàm tính điểm cho mỗi bộ ba bằng cách liên kết $\mathbf{v}_h$, $\mathbf{v}_r$, sau đó định hình lại chúng vào một ma trận $2D$ và chuyển chúng thông qua một mạng nơ-ron tích chập CNN. Kết quả đầu ra của mạng nơ-ron tích chập CNN này là một véc-tơ với số chiều d_e, véc-tơ này được nhân với $\mathbf{v}_t$. Sau đó, kết quả được tính tổng với một hệ số thiên lệch (bias) đặc trưng của thực thể cuối b_t cụ thể nào đó để thu được kết quả tính điểm cho bộ ba đó. Trong suốt quá trình chuyển giao, các siêu tham số của mô hình nơ-ron tích chập CNN huấn luyện trước được sử dụng như là khởi tạo của mô hình nơ-ron tích chập

CNN trong mô hình tinh chỉnh. Hệ số thiên lệch của mô hình tinh chỉnh được khởi tạo ngẫu nhiên vì chúng phụ thuộc vào từng thực thể cụ thể.

- Mô hình 5★E xem xét $\mathbf{v}_h$ và $\mathbf{v}_t$ là các đường chiếu phức hợp (complex projective lines), trong khi đó $\mathbf{v}_r$ là một ma trận phức với kích thước 2×2. Phương án này tương tự như một phép biến đổi Möbius đặc trưng cho quan hệ các các đường xạ ảnh. Không giống như hai mô hình hoàn thiện ĐTTT TuckER và ConvE, mô hình 5★E không có các tham số chia sẻ giữa các quan hệ và các thực thể khác nhau. Do đó quá trình huấn luyện trước chỉ đóng vai trò là sự khởi tạo cho các nhúng.

Trong suốt quá trình đánh giá, mô hình đưa ra một bộ ba bị thiếu thực thể đầu h hoặc thiếu thực thể cuối t. Sau đó, sử dụng để xếp hạng tất cả các thực thể có thể phù hợp dựa trên độ tương đồng khi nó xuất hiện như thế nào trong các vị trí bị thiếu thực thể. Bên cạnh đó, dựa trên ý tưởng từ [32], V. Kocijan và T. Lukasiewicz biến đổi các mẫu dự đoán thực thể đầu thành các mẫu dự đoán thực thể đuôi bằng cách sử dụng các quan hệ đối ngẫu r^{-1} cho mỗi quan hệ, như vậy bài toán được chuyển đổi từ $(?, r, t)$ sang $(t, r^{-1}, ?)$.

2.1.4 Ý tưởng cải tiến

Việc lựa chọn các mô hình để ánh xạ biểu diễn ĐTTT sang không gian véc-tơ đóng vai trò then chốt trong quá trình hoàn thiện đồ thị tri thức (KGC). Sự lựa chọn này quyết định chất lượng biểu diễn của thực thể và quan hệ, từ đó ảnh hưởng trực tiếp đến hiệu năng suy luận và dự đoán của mô hình. Những phép nhúng hiệu quả không chỉ giúp tạo ra các véc-tơ giàu thông tin ngữ nghĩa mà còn đặc biệt hữu ích đối với các thực thể hoặc quan hệ có kèm theo mô tả văn bản.

Trong các phương pháp nhúng từ truyền thống như Word2Vec, Glove... thì véc-tơ của từ là cố định, không phản ánh sự thay đổi ngữ cảnh và không xử lý được từ đa nghĩa. Do hạn chế này, các mô hình nhúng ngữ cảnh như BERT ra đời nhằm khắc phục bằng cách học biểu diễn động tùy thuộc vào bối cảnh, thích hợp với việc học các mẫu ngữ nghĩa phức tạp trong ngôn ngữ tự nhiên. BERT được xây dựng dựa trên kiến trúc Transformer encoder đa lớp hai chiều hỗ trợ đọc văn bản từ trái sang phải và từ phải sang trái cùng lúc để nắm bắt đầy đủ ngữ cảnh đầy đủ xung quanh một từ. BERT sử dụng cơ chế tự chú ý (self-attention) để tính trọng số của từng từ trong câu dựa trên toàn bộ câu, giúp mô hình biết từ nào cần chú trọng trong bối cảnh

cụ thể. Nhờ vậy, BERT có khả năng trích xuất đặc trưng giàu ngữ nghĩa phức tạp, hỗ trợ tốt hơn cho các nhiệm vụ suy luận và dự đoán trong ĐTTT.

Ngoài ra, trong ĐTTT có thể xuất hiện các từ mới, từ hiếm gặp hoặc tên riêng không có trong tập huấn luyện ban đầu. Để xử lý hiệu quả các trường hợp này, cần sử dụng các mô hình nhúng có khả năng khái quát tốt và tính toán nhanh. FastText là một lựa chọn phù hợp nhờ cơ chế nhúng dựa trên n-gram ký tự, cho phép mô hình học được biểu diễn cho cả những từ chưa từng xuất hiện.

Nhúng thực thể và quan hệ đóng vai trò là nền tảng cho việc học biểu diễn trong mô hình KGC. Chất lượng nhúng càng cao thì mô hình càng có khả năng học được các mối quan hệ phức tạp, phân biệt chính xác hơn giữa các bộ ba đúng và sai, từ đó nâng cao hiệu năng hoàn thiện ĐTTT. Đặc biệt, trong các đồ thị tri thức phức tạp, khi quan hệ giữa (thực thể đầu, quan hệ, thực thể cuối) mang tính phi tuyến mạnh, việc áp dụng các mô hình nhúng sâu sẽ giúp gia tăng tính linh hoạt của biểu diễn. Thêm nữa, theo nhận định của V. Kocijan và T. Lukasiewicz, mô hình KGC-TransLearning sẽ có kết quả tốt hơn khi mô hình sử dụng bộ mã hóa GRU thay vì sử dụng bộ mã hóa NoEncoder. GRU có khả năng nắm bắt các phụ thuộc theo chuỗi và duy trì thông tin ngữ cảnh hiệu quả, đặc biệt khi kết hợp với các tầng kết nối đầy đủ (Fully Connected – FC) để học nhúng bộ ba.

Từ những nhận định trên, luận án đưa ra ý tưởng đề xuất để cải tiến mô hình học chuyển giao trong hoàn thiện ĐTTT như sau:

- Sử dụng BERT hoặc FastText để ánh xạ các từ sang không gian véc-tơ, nhằm khai thác cả ngữ cảnh phức tạp (BERT) và xử lý hiệu quả từ mới/ từ hiếm (FastText).
- Sử dụng mô hình kết hợp GRU và kết nối đầy đủ (Fully Connected: FC) để học nhúng bộ ba, tăng khả năng mô hình hóa quan hệ phi tuyến và cải thiện độ chính xác trong phân biệt bộ ba hợp lệ.

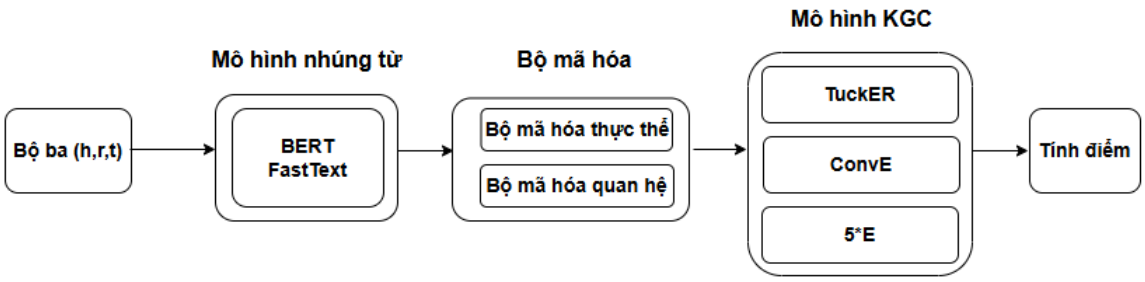
Đề xuất này vừa tận dụng sức mạnh của mô hình ngôn ngữ hiện đại để học đặc trưng ngữ nghĩa sâu, vừa khai thác khả năng mô hình hóa phụ thuộc tuần tự của GRU, từ đó tạo ra biểu diễn bộ ba giàu thông tin và nâng cao hiệu năng hoàn thiện ĐTTT.

2.2 Mô hình đề xuất BERT/FastText-GRU-KGC

Dựa vào mô hình hoàn thiện ĐTTT trong [17], luận án chia mô hình đề xuất thành ba thành phần chính như sau:

- Thành phần đầu tiên là một mô hình nhúng từ. Mô hình nhúng từ này được sử dụng để ánh xạ một bộ ba từ không gian từ sang không gian véc-tơ.
- Thành phần thứ hai là bộ mã hóa. Kết quả đầu ra của mô hình nhúng từ trước đó sẽ là dữ liệu đầu vào cho của bộ mã hóa. Bộ ba dữ liệu này qua bộ mã hóa sẽ tạo thành nhúng bộ ba tương ứng.
- Thành phần cuối cùng là một mô hình hoàn thiện ĐTTT nào đó. Mô hình hoàn thiện ĐTTT sẽ nhận các nhúng bộ ba làm đầu vào. Sau đó, sử dụng hàm tính điểm tương ứng của từng mô hình hoàn thiện ĐTTT để tính điểm cho từng nhúng bộ ba. Dựa vào điểm số để xác định một bộ ba là đúng hay là sai.

Ba thành phần trong mô hình đề xuất này độc lập và rời rạc với nhau, xem Hình 2.2.



Hình 2.2. Các thành phần của mô hình đề xuất.

Sự khác biệt giữa mô hình đề xuất với mô hình gốc Glove-GRU-KGC được thể hiện như sau:

- Thành phần đầu tiên – mô hình nhúng từ: Mô hình đề xuất cần sử dụng mô hình nhúng từ BERT/FastText thay vì sử dụng mô hình nhúng từ Glove.
- Thành phần thứ hai – bộ mã hóa: Mô hình đề xuất cần kết hợp giữa GRU và lớp kết nối đầy đủ (Fully Connected – FC) thay vì chỉ sử dụng GRU như trong mô hình Glove-GRU-KGC gốc.

Để phân biệt với mô hình gốc Glove-GRU-KGC do V. Kocijan và T. Lukasiewicz đề xuất, luận án gọi mô hình đề xuất là BERT/FastText-GRU-KGC. Từng thành phần trong mô hình đề xuất được trình bày chi tiết như sau đây.

2.2.1 Thành phần nhúng từ

Mô hình nhúng từ và bộ mã hóa đóng vai trò cầu nối giữa dữ liệu văn bản ban đầu và các mô hình hoàn thiện ĐTTT như TuckER, ConvE hay 5*E. Việc kết hợp linh hoạt giữa thông tin ngôn ngữ và cấu trúc biểu diễn góp phần cải thiện đáng kể hiệu quả của các mô hình hoàn thiện đồ thị tri thức, đặc biệt trong bối cảnh thiếu dữ liệu huấn luyện hoặc khi xuất hiện các thực thể/quan hệ mới.

Mô hình nhúng từ như BERT hoặc FastText có vai trò chính là ánh xạ thực thể và quan hệ được biểu diễn dưới dạng chuỗi văn bản sang các vector có mang thông tin ngữ nghĩa. Các mô hình này được huấn luyện trên tập dữ liệu văn bản quy mô lớn, cho phép chúng khai thác ngữ cảnh và cấu trúc ngôn ngữ để tạo ra các biểu diễn nhúng giàu thông tin. Một số đặc trưng của mô hình BERT và FastText được trình bày dưới đây.

I. Nhúng từ FastText

FastText được P. Bojanowki và cộng sự giới thiệu [60] nhằm khắc phục các nhược điểm về biểu diễn từ hiếm gặp hoặc không có trong tập dữ liệu. Thay vì huấn luyện cả từ, FastText chia mỗi từ thành các đoạn nhỏ hơn. Mỗi đoạn nhỏ hơn này được gọi là một túi ký tự n-gram. Việc phân chia được thực hiện bằng cách thêm vào các ký tự biên đặc biệt ‘<’ và ‘>’ tương ứng ở đầu và cuối của mỗi từ. Điều này cho phép phân biệt tiền tố và hậu tố với các chuỗi ký tự khác. Ví dụ: với 4-gram, từ “completion” sẽ được phân chia thành [‘<com’, ‘comp’, ‘ompl’, ‘mple’, ‘plet’, ‘leti’, ‘etio’, ‘tion’, ‘ion>’, ‘<completion>’], ở đó tiền tố ‘<’ cho biết bắt đầu của từ và hậu tố ‘>’ cho biết kết thúc của từ. Khi đó, véc-tơ của từ ‘completion’ sẽ bằng tổng tất cả các véc-tơ 4-gram này. Rõ ràng, bằng việc tách từ thành các túi ký tự n-gram như trên, nếu một từ không được nhìn thấy trong quá trình huấn luyện thì nó vẫn có thể tạo ra véc-tơ nhúng của từ bằng cách cộng các véc-tơ n-gram đã tồn tại. FastText rất phù với những ngôn ngữ giàu hình thái, nghĩa là ở đó ngôn ngữ được biểu diễn dưới nhiều hình thức khác nhau. Với những ngôn ngữ này, nhiều từ hiếm khi xuất hiện hoặc không xuất hiện trong quá trình huấn luyện, gây khó khăn cho quá trình biểu diễn từ. Đây là lợi thế lớn nhất của FastText.

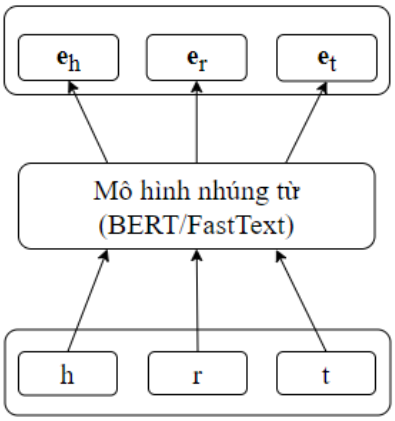
II. Nhúng từ BERT

BERT là viết tắt của cụm từ Bidirectional Encoder Representations from Transformer, là một mô hình biểu diễn ngôn ngữ được J. Devlin và cộng sự giới thiệu vào năm 2018 [53]. BERT được xây dựng để huấn luyện trước các biểu diễn hai chiều học sâu từ văn bản không được gán nhãn. Điểm nổi bật của BERT là nó cho phép điều hòa cân bằng cả phía trái và phía phải ngữ cảnh trong tất cả các lớp. BERT có thể được tinh chỉnh với chỉ một lớp đầu ra bổ sung để tạo ra mô hình tốt cho một lớp các tác vụ như hỏi đáp, suy luận ngôn ngữ …

BERT sử dụng cơ chế Attention của Transformer để mã hóa từ thành các véc-tơ có giá trị số. Một điểm khác biệt so với mô hình biểu diễn từ khác là Transformer đọc toàn tập dữ liệu đầu vào cùng một lúc. Chính điểm khác biệt này cho phép mô hình học được ngữ cảnh của một từ dựa vào các từ xung quanh nó. Ví dụ từ "con chuột" trong ngữ cảnh "con chuột máy tính" khác với trong ngữ cảnh "con mèo đang đuổi bắt con chuột". Để tăng khả năng hiểu và nhận biết ngữ cảnh, BERT sử dụng hai chiến lược học là ngôn ngữ mặt nạ (Masked Language Model - MLM) và dự đoán câu tiếp theo (Next Sentence Prediction - NSP). Mô hình MLM đánh dấu ngẫu nhiên một vài mã từ trong tập dữ liệu đầu vào, và đối tượng được sử dụng để dự đoán mã từ vựng gốc của các từ giấu chỉ dựa trên ngữ cảnh. Tác vụ NSP cho phép mô hình dự đoán câu tiếp theo dựa vào câu hiện tại. Đầu vào của mô hình là các cặp câu, sau đó mô hình học cách dự đoán xem câu thứ hai trong cặp câu vừa nhận vào này có phải là câu tiếp theo trong tài liệu gốc hay không. Nhờ áp dụng hai chiến lược này giúp BERT học một cách hiệu quả hơn các biểu diễn từ, tạo nên một mô hình huấn luyện trước vượt trội trong lĩnh vực xử lý ngôn ngữ tự nhiên.

Hiện nay có nhiều phiên bản khác nhau của mô hình BERT, các phiên bản đều dựa trên việc thay đổi kiến trúc của Transformer, trong đó tập trung ở ba tham số: L – số lượng các khối lớp con trong Transformer; H – kích thước của các véc-tơ nhúng (hay còn gọi là kích thước ẩn); A – số lượng head trong các lớp multi-head, mỗi một head sẽ thực hienj cơ chế self-attention. Hai phiên bản phổ biến là $\text{BERT}_{\text{base}}$ và $\text{BERT}_{\text{large}}$:

1) $\text{BERT}_{\text{base}}$: $L = 12, H = 768, A = 12$: tổng cộng có 110 triệu tham số.
2) $\text{BERT}_{\text{large}}$: $L = 24, H = 1024, A = 16$: tổng cộng có 340 triệu tham số.



Hình 2.3. Kiến trúc mô hình nhúng từ trong mô hình BERT/FastText-GRU-KGC.

Như đã đề cập bên trên, trong phần mô hình nhúng từ, luận án đề xuất thực nghiệm mô hình BERT hoặc FastText thay vì sử dụng mô hình Glove. Khi sử dụng mô hình BERT, luận án lựa chọn mô hình $\text{BERT}_{\text{large}}$. Dữ liệu đầu vào của mô hình nhúng từ là một bộ ba (h, r, t) dưới dạng văn bản tương ứng. Sau khi đi qua mô hình nhúng từ, bộ ba này sẽ được biến thành một véc-tơ nhúng từ tương ứng $(\mathbf{e}_h, \mathbf{e}_r, \mathbf{e}_t)$. Các véc-tơ nhúng từ này sẽ là dữ liệu đầu vào cho mô hình mã hóa tiếp theo. Chi tiết của kiến trúc mô hình nhúng từ xem tại Hình 2.3.

2.2.2 Thành phần mã hóa

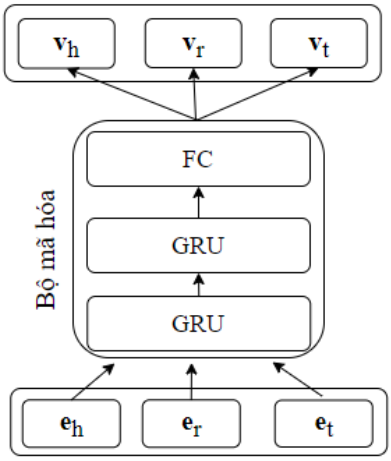
Sau khi thu được các véc-tơ nhúng từ mô hình ngôn ngữ, các bộ mã hóa được sử dụng để chuẩn hóa và điều chỉnh những véc-tơ này sao cho phù hợp với không gian biểu diễn của mô hình hoàn thiện ĐTTT.

Như đã được đề cập, quá trình học biểu diễn nhúng của các thực thể và quan hệ đóng một vai trò cực kỳ quan trọng trong việc hoàn thiện ĐTTT mở (ĐTTT chứa các thực thể chưa được chuẩn hóa). Kế thừa những ưu điểm nổi trội của bộ mã hóa GRU, do vậy mô hình đề xuất vẫn giữ nguyên bộ mã hóa GRU để học được các đặc trưng ngữ nghĩa tiềm ẩn trong chuỗi đầu vào, ví dụ như mối liên hệ theo ngữ cảnh giữa thực thể và quan hệ. Ngoài ra, mô hình đề xuất bổ sung thêm một lớp GRU nhằm mục đích tinh chỉnh tiếp biểu diễn từ lớp GRU đầu tiên và làm nổi bật các tương tác phi tuyến giữa các thực thể và quan hệ. Bên cạnh đó, để cải thiện hiệu năng của mô hình đề xuất, luận án sử dụng mô hình lớp kết nối đầy đủ FC với vai trò tổng hợp các đặc trưng trích xuất từ GRU để ánh xạ vào không gian nhúng cố định của mô hình hoàn thiện ĐTTT như TuckER, ConvE…

Lớp kết nối đầy đủ FC là một thành phần cơ bản trong mạng nơ-ron nhân tạo (Artificial Neural Networks – ANNs). Trong lớp kết nối đầy đủ, mỗi nơ-ron (hay còn gọi là nút) được kết nối với mọi nơ-ron trong lớp trước và với mọi nơ-ron trong lớp tiếp theo. Điều này có nghĩa là mỗi nơ-ron nhận đầu vào từ tất cả các nơ-ron trong lớp trước và đóng góp đầu ra của nó cho tất cả nơ-ron trong lớp tiếp theo. Mỗi kết nối giữa các nơ-ron sẽ có trọng số tương ứng, đây là một tham số có thể học mà mô hình điều chỉnh trong suốt quá trình huấn luyện.

Cấu trúc hai lớp GRU kết hợp với lớp kết nối đầy đủ FC mang lại lợi thế đáng kể trong việc học biểu diễn nhúng có chiều sâu ngữ nghĩa cao, đặc biệt hiệu quả trong các thiết lập dữ liệu ít nhãn hoặc kịch bản hoàn thiện tri thức trong bối cảnh chưa thấy (zero-shot). Cách tiếp cận này giúp mô hình hoàn thiện ĐTTT đạt được sự cân bằng giữa khả năng học đặc trưng mạnh mẽ và tính tổng quát hóa trên toàn đồ thị tri thức.

Dữ liệu đầu vào của mô hình mã hóa này là dữ liệu đầu ra của mô hình nhúng, cụ thể là các véc-tơ nhúng $(\mathbf{v}_h, \mathbf{v}_r, \mathbf{v}_t)$ tương ứng. Chi tiết mô hình mã hóa được minh họa tại Hình 2.4.

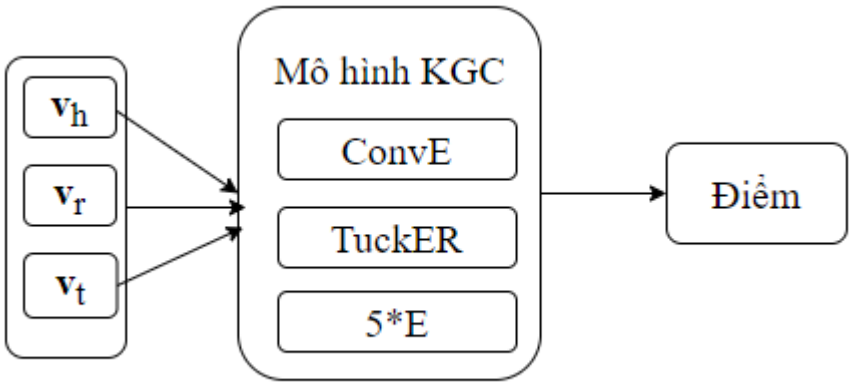


Hình 2.4. Kiến trúc mã hóa trong mô hình BERT/FastText-GRU-KGC.

Vì tính chất cũng như số lượng các quan hệ và thực thể là khác nhau, nên để mô hình có thể học và biểu diễn một cách tốt nhất các nhúng bộ ba thì ma trận trọng số của các thực thể và quan hệ sẽ được khởi tạo và cập nhật riêng biệt trong quá trình huấn luyện. Việc chia tách này giúp cho mô hình tối ưu hóa và đạt độ chính xác tốt hơn. Khi học chuyển giao được áp dụng, tham số của mô hình mã hóa sẽ được khởi tạo dựa vào tham số của mô hình được huấn luyện trước đó.

2.2.3 Thành phần hoàn thiện ĐTTT

Tương tự như mô hình Glove-GRU-KGC gốc của V. Kocijan và T. Lukawiewicz [17], mô hình đề xuất cũng sử dụng ba mô hình hoàn thiện ĐTTT có hiệu năng tốt nhất đó là TuckER, ConvE và 5⋆E. Đầu vào của thành phần hoàn thiện ĐTTT chính là bộ ba véc-tơ nhúng $(\mathbf{v}_h, \mathbf{v}_r, \mathbf{v}_t)$ (đây là kết quả thu được sau khi đi qua thành phần mã hóa). Khi bộ ba véc-tơ nhúng đi vào thành phần hoàn thiện ĐTTT, thì một trong các mô hình hoàn thiện ĐTTT được lựa chọn sẽ tìm cách học và tính điểm cho từng bộ ba này. Cuối cùng, đầu ra của mô hình đề xuất chính là kết quả điểm số của từng mô hình hoàn thiện ĐTTT trong thành phần này. Chi tiết về đầu vào và đầu ra của thành phần hoàn thiện ĐTTT xem tại Hình 2.5.



Hình 2.5. Đầu vào và đầu ra của thành phần hoàn thiện ĐTTT.

Để thuận tiện, luận án nhắc lại một số đặc trưng của từng mô hình hoàn thiện ĐTTT nguồn chuyển giao:

- TuckER: mô hình này sẽ tính điểm cho mỗi bộ ba bằng cách nhân các véc-tơ nhúng với một tensor $\mathcal{W} \in \mathbb{R}^{d_e \times d_r \times d_e}$, trong đó d_e là số chiều các thực thể và d_r là số chiều của các quan hệ. Để giảm số lượng siêu tham số, thông thường sẽ giả định rằng $d_e = d_r$. Đối với quá trình học chuyển giao, tensor $\mathcal{W}$ từ mô hình huấn luyện trước được sử dụng để khởi tạo tensor $\mathcal{W}$ trong mô hình tinh chỉnh.
- ConvE: Đối với quá trình học chuyển giao, các tham số của mạng tích chập CNN được huấn luyện trước. Sau đó, các tham số này được sử dụng là tham số khởi tạo của mạng tích chập CNN trong mô hình tinh chỉnh.

- 5★E: Khác với hai mô hình TuckER và ConvE, tham số không được chia sẻ giữa các quan hệ và thực thể khác nhau. Do vậy, trong quá trình học chuyển giao thì huấn luyện trước chỉ đóng vai trò là quá trình khởi tạo các véc-tơ nhúng.

2.3 Thực nghiệm mô hình đề xuất và đánh giá

2.3.1 Tập dữ liệu

Tương tự như các thực nghiệm với mô hình của V. Kocijan và T. Lukasiewicz [17] cũng như có những đánh giá công bằng so với hiệu năng của mô hình gốc, mô hình đề xuất cũng thực nghiệm với năm tập dữ liệu: OlpBench, ReVerb45K, ReVerb20K, FB15K237 và WN18RR. Đối với quá trình học chuyển giao, mô hình được huấn luyện trước trên tập dữ liệu OlpBench. Tập dữ liệu OlpBench là một trong những tập dữ liệu lớn về cơ sở tri thức mở. Sau đó, luận án tiến hành quá trình chuyển giao và tinh chỉnh trên bốn tập dữ liệu còn lại. Trong phần này, luận án mô tả một số đặc trưng chính về các tập dữ liệu này.

OlpBench [61]: OlpBench được tạo ra để đánh giá khả năng các mô hình nhúng ĐTTT có thể dự đoán được các dữ kiện mới bên trong một "thế giới mở" (kịch bản thế giới mở nghĩa là các thực thể mới và các quan hệ mới không ngừng được đưa vào). Khái niệm thế giới mở thì đối lập với thế giới đóng mà ở đó các tập thực thể và các tập quan hệ được cố định. Đây là tập dữ liệu hoàn thiện ĐTTT mở với kích thước lớn được lấy từ Wikipedia tiếng Anh và được sử dụng cho quá trình huấn luyện trước. Tập dữ liệu đi kèm với các bộ huấn luyện và bộ xác thực khác nhau. Tuy nhiên, trong thực nghiệm với mô hình đề xuất, luận án chỉ sử dụng bộ huấn luyện loại bỏ rò rỉ THOROUGH và bộ xác thực VALID-LINKED. Các tập dữ liệu này có chất lượng khá tốt.

ReVerb45K [62]: đây là tập dữ liệu hoàn thiện ĐTTT mở với kích thước nhỏ thu được thông qua công cụ trích rút thông tin mở ReVerb [29]. ReVerb45K là một phiên bản mở rộng của tập dữ liệu Ambiguous, bao gồm các cụm danh từ. Nó thu được bằng cách kết hợp thông tin từ ba nguồn sau: ReVerb Open KB, FreeBase và kho dữ liệu Clueweb09. Ở đó, ReVerb được thiết kế để trích rút thông tin từ bộ ba quan hệ "(subject, predicate, object)", với "subject" và "object" đề cập đến thực thể và "predicate" mô tả quan hệ được trích rút từ các văn bản. ReVerb thu được từ văn bản mà không cần dựa vào thuật ngữ học được xác định trước đó. Tập dữ liệu

ReVerb45K thu được bằng cách sử dụng tập dữ liệu ban đầu ReVerb, sau đó được chọn lọc với FreeBase và Clueweb09 để tạo ra khoảng 45.000 bộ ba chất lượng. Tập dữ liệu này ban đầu được sử dụng để đánh giá mô hình "các thực thể và các quan hệ chuẩn hóa" (Canonicalization of Entities and Relations), nhằm mục đích nhóm các thực thể và quan hệ đồng nghĩa trong các ĐTTT mở để tạo ra một bộ ba chuẩn hóa và mạch lạc hơn. Bên cạnh đó, còn giảm sự trùng lặp và mơ hồ trong dữ liệu được trích xuất bằng cách nhóm các thực thể và quan hệ tương tự lại với nhau. Tập dữ liệu ReVerb45K đóng vai trò chuẩn mực để đánh giá mức độ hiệu quả của một mô hình trong việc thực hiện các tác vụ chuẩn hóa. Tập dữ liệu ReVerb45K phản ánh những thách thức trong thế giới thực gặp phải khi làm việc với các hệ thống trích xuất thông tin mở. Nó giúp kiểm tra các mô hình nhằm mục đích cải thiện khả năng sử dụng kiến thức được trích xuất bằng cách giảm nhiễu và tránh trùng lặp. Đây là nguồn tài nguyên quan trọng trong lĩnh vực chuẩn hóa ĐTTT mở.

ReVerb20K [24]: tương tự như tập dữ liệu ReVerb45K. Tập dữ liệu ReVerb20K chứa khoảng 20.000 bộ ba, mỗi bộ ba có cấu trúc "(subject, predicate, object)". V. Kocijan và T. Lukasiewicz trong [17] đã chỉ ra rằng phần lớn mẫu kiểm thử trong ReVerb20K (99,9%) và ReVerb45K (99,3%) chứa ít nhất một thực thể hoặc quan hệ mà không xuất hiện trong tập dữ liệu OlpBench. Do vậy, yêu cầu khái quát hóa ngoài phân phối. Hai thực thể hoặc quan hệ được coi là khác nhau nếu biểu diễn văn bản của chúng khác nhau sau khi tiền xử lý (dùng chữ thường và loại bỏ khoảng trắng dư thừa). Nếu hai đầu vào vẫn khác nhau sau mỗi bước này, chuẩn hóa tiềm năng phải được thực hiện bởi mô hình.

FB15K237 [63]: Tập dữ liệu FB15K237 thường xuyên được sử dụng trong lĩnh vực nhúng ĐTTT và dự đoán liên kết. Nó là tập con của tập dữ liệu FB15K được xây dựng từ bộ Freebase (một cơ sở dữ liệu lớn chứa dữ liệu có cấu trúc bao gồm các thực thể và các quan hệ). Freebase chứa các thực thể và các dữ kiện mô tả tri thức trong thế giới thực. FB15K237 đã được loại bỏ các quan hệ thừa và các liên kết huấn luyện trực tiếp đối với bộ ba held-out nhằm mục đích làm cho tác vụ trở nên thực tế hơn.

WN18RR [32]: Tập dữ liệu WN18RR là tập con đã được tinh chỉnh của tập dữ liệu WN18. Tập dữ liệu WN18 được bắt nguồn từ cơ sở dữ liệu từ vựng WordNet. Tập dữ liệu WN18RR được tạo ra để giải quyết các vấn đề rò rỉ dữ liệu trong WN18 bằng cách loại bỏ các quan hệ nghịch đảo, từ đó biến nó thành các chuẩn mực cho

các tác vụ dự đoán liên kết trong hoàn thiện ĐTTT. Ngoài ra, V. Kocijan và T. Lukasiewicz [17] khẳng định rằng WN18RR khác với các tập dữ liệu khác trong nội dung của nó. Trong khi các tập dữ liệu khác miêu tả các thực thể thế giới thực và thường thu nhận từ Wikipedia hoặc một vài nguồn tài nguyên tương tự, WN18RR bao gồm thông tin trên các quan hệ từ và ngữ nghĩa giữa chúng, tạo ra một miền lớn nằm giữa huấn luyện trước và tinh chỉnh.

Bảng 2.1 cho số liệu thống kê về các số lượng quan hệ, thực thể và bộ ba của mỗi tập dữ liệu này.

Bảng 2.1. Số lượng thực thể, quan hệ và các bộ ba trong các tập dữ liệu.

	OlpBench	ReVerb20K	ReVerb45K	FB15K237	WN18RR
Thực thể	2,4M	11,1K	27K	14,5K	41,1K
Quan hệ	961K	11.1K	21,6K	237	11
Bộ ba huấn luyện	30M	15,5K	36K	272K	86,8K
Bộ ba xác thực	10K	1,6K	3,6K	17,5K	3K
Bộ ba kiểm thử	10K	2.4K	5.4K	20,5K	3K

2.3.2 Kịch bản thực nghiệm

Dựa theo các kịch bản đã được V. Kocijan và T. Lukasiewicz thực nghiệm [17], với mô hình đề xuất cũng được đánh giá và thực nghiệm với hai kịch bản khác nhau. Chi tiết về hai kịch bản này được mô tả như sau:

1) Kịch bản thứ nhất sử dụng một mô hình được huấn luyện trước để khởi tạo các tham số và các véc-tơ nhúng, sau đó thực hiện tinh chỉnh chúng. Đối với kịch bản này, giống như trong [17], tập dữ liệu OlpBench được sử dụng để huấn luyện trước. Vì tập dữ liệu OlpBench có kích thước tương đối lớn nên đối với kịch bản này luận án thực nghiệm sử dụng 1/10 kích thước của tập dữ liệu OlpBench. Ngoài ra, các dữ liệu này được lấy một cách ngẫu nhiên.
2) Kịch bản thứ hai là huấn luyện mô hình thực nghiệm từ đầu.

Với cả hai kịch bản trên, mô hình đề xuất trong luận án đều được đánh giá trên cả bốn tập dữ liệu tiêu chuẩn trong hoàn thiện ĐTTT là ReVerb20K, ReVerb45K, FB15K237 và WN18RR.

2.3.3 Cấu hình và thiết lập thực nghiệm mô hình đề xuất

I. Cấu hình thực nghiệm

Luận án thực nghiệm trên môi trường với thông số cho bởi Bảng 2.2.

Bảng 2.2. Thông số thực nghiệm mô hình

	Google Colab
Nguồn	https://colab.research.google.com
GPU	Nvidia K80/ T4
Tốc độ xử lý GPU	0,82GHz/ 1,59GHz
Dung lượng GPU	16,0 GB
Dung lượng RAM	12 GB

II. Độ đo đánh giá mô hình

Để có được đánh giá và so sánh cụ thể giữa mô hình đề xuất BERT/FastText-GRU-KGC và mô hình gốc Glove-GRU-KGC của V. Kocijan và T. Lukasiewicz trong [17] (lưu ý: luận án sẽ so sánh với kết quả tốt nhất của mô hình gốc), luận án sử dụng các độ đó là: MR, MRR, $Hit@n$ (luận án lựa chọn $n = 10$).

III. Thiết lập các mô hình

Phần này luận án sẽ trình bày về các thiết lập cho mô hình nhúng từ, mô hình mã hóa, mô hình huấn luyện trước, mô hình học chuyển giao.

- Mô hình nhúng từ:
  - Với kịch bản sử dụng mô hình nhúng từ BERT, mô hình đề xuất được gọi là BERT-GRU-KGC. Với mô hình này sẽ sử dụng $\text{BERT}_{\text{large}}$ cho mô hình huấn luyện trước để khởi tạo các nhúng bộ ba. Mô hình này được huấn luyện trước bằng cách sử dụng mô hình ngôn ngữ mặt nạ MLM và các véc-tơ nhúng được khởi tạo với số chiều là 100.

- Với kịch bản sử dụng mô hình nhúng từ FastText, mô hình đề xuất được gọi là FastText-GRU-KGC. Khi đó sử dụng mô hình được huấn luyện trên một triệu từ trên Wikipedia tiếng Anh 2017, kho dữ liệu webbase UMBC và tập dữ liệu tin tức statmt.org. Tổng cộng sẽ có 16 tỷ tokens. Đối với các quan hệ và thực thể được FastText huấn luyện trước sẽ được khởi tạo bằng nhúng từ tương ứng trong đó. Các từ còn lại không trong mô hình huấn luyện trước thì được khởi tạo một cách ngẫu nhiên. Trong trường hợp này, các thực thể và các quan hệ sẽ được ánh xạ vào không gian véc-tơ nhúng với kích thước là 300.

- Mô hình mã hóa
  - Với kịch bản sử dụng mô hình BERT-GRU-KGC, kích thước hidden_size của lớp GRU đối với thành phần mô hình hoàn thiện ĐTTT TuckER và $5\star$E sẽ là 100, trong khi đó đối với mô hình ConvE là 500. Trong trường hợp này, lớp kết nối đầy đủ FC sẽ trả về một véc-tơ nhúng đầu ra có kích thước bằng kích thước của hidden_size trong GRU.
  - Với kịch bản sử dụng mô hình FastText-GRU-KGC, kích thước của hidden_size của thành phần cả ba mô hình hoàn thiện ĐTTT đều là 300.

- Mô hình huấn luyện trước: với mô hình huấn luyện trước sẽ được huấn luyện trên tập dữ liệu OlpBench với kích thước lô huấn luyện là 1024. Tốc độ học được chọn từ $\{1.10^{-4}, 3.10^{-4}\}$, tỷ lệ dropout được chọn từ $\{0.3, 0.4\}$ đối với mô hình TuckER và ConvE. Tuy nhiên, đối với mô hình $5\star$E, tỷ lệ dropout sẽ không được sử dụng, thay vào đó sử dụng chính quy hóa N3 (N3 regularization) với các giá trị được lấy trong $\{0.1, 0.3\}$ theo [64]. Với mô hình TuckER và ConvE thì số chiều của véc-tơ nhúng được lựa chọn là 100 và 300. Trong khi đối với mô hình $5\star$E, số chiều véc-tơ nhúng được lựa chọn là 300 và 500. Kết quả tốt nhất của mỗi mô hình trong quá trình chạy sẽ được lưu lại và sử dụng cho việc học chuyển giao. Tất cả mô hình đều được huấn luyện với 100 epochs trên cụm DGX sử dụng Nvidia V100 GPU.

- Mô hình học chuyển giao: Tất cả quá trình học chuyển giao được thiết lập tương tự như huấn luyện trước. Tuy nhiên, các mô hình học chuyển giao được huấn luyện với số lượng epochs nhiều hơn (cỡ 200 epochs) và một số lượng lớn các siêu tham số. Ở đó, tốc độ học được lựa chọn từ

$\{3.10^{-5}, 3.10^{-4}, 1.10^{-4}, 1.10^{-2}\}$. Tỷ lệ dropout được lựa chọn từ $\{0.03, 0.1, 0.2, 0.3, 0.4\}$. Hệ số chính quy hóa N3 (N3 regularization) được lựa chọn từ $\{0.03, 0.1, 0.3\}$ và kích thước lô được lựa chọn từ $\{512, 1024, 2048, 4096\}$. Số chiều của các véc-tơ nhúng tương tự như số chiều của véc-tơ nhúng trong mô hình huấn luyện trước. Việc học chuyển giao được thực hiện trên GPU Tesla T4.

2.3.4 Kết quả và đánh giá

Trong phần này, luận án đưa ra một số kết quả và đánh giá khi thực hiện mô hình đề xuất với các cấu hình và tham số được đưa ra bên trên.

I. Kết quả khi huấn luyện trước.

Hiệu năng của các mô hình huấn luyện trước được đưa ra tại Bảng 2.3. Trong bảng này, kết quả tốt nhất được in đậm và kết quả tốt thứ hai được in nghiêng theo mỗi cột giá trị đánh giá.

Bảng 2.3. So sánh giữa các mô hình huấn luyện trước của mô hình đề xuất với kết quả tốt nhất của mô hình gốc trong [17] đối với tập dữ liệu OlpBench. Tại mỗi cột, kết quả in đậm là tốt nhất, kết quả gạch chân là tốt thứ hai.

Mô hình hoàn thiện ĐTTT	**Mô hình nhúng từ**	**MR ↓**	**MRR↑**	**H@10↑**
TuckER	BERT	**4.300**	**0,100**	**0,997**
	FastText	176.400	0,005	0,011
ConvE	BERT	229.400	0,004	0,009
	FastText	157.800	0,006	0,012
5★E	BERT	1.341.000	0,010	0,001
	FastText	167.400	0,004	0,011
GRU_5★E [17]	Glove	<u>60.100</u>	<u>0,055</u>	<u>0,101</u>

Từ bảng kết quả, luận án nhận thấy mô hình đề xuất sử dụng nhúng từ với BERT (BERT-GRU-KGC) cho kết quả tốt hơn khi sử dụng nhúng từ với FastText (FastText-GRU-KGC). Do đó, việc sử dụng phương pháp biểu diễn từ đối với các bài toán xử lý ngôn ngữ tự nhiên đóng một vai trò khá quan trọng. Hơn nữa, mô hình đề xuất, BERT-GRU-TuckER, là sự kết hợp giữa mô hình nhúng từ BERT và mô hình hoàn thiện ĐTTT TuckER cho kết quả vượt trội so với kết quả tốt nhất từ mô hình gốc GlovE-GRU-KGC của V. Kocijan và T. Lukasiewicz tại [17]. Một số nhận xét so sánh tỷ lệ cụ thể:

- Với đánh giá MR đạt 4300, giảm khoảng 14 lần (cụ thể, kết quả của mô hình trong GRU_5★E [17] là 60100, kết quả của mô hình BERT-GRU-TuckER là 4300).
- Với đánh giá MRR đạt $0{,}1$, tăng gần gấp 02 lần (cụ thể, kết quả của mô hình GRU_5★E [17] là 0,055, kết quả của mô hình BERT-GRU-TuckER là 0,997).

Theo đánh giá, kết quả thu được vẫn chưa là tốt nhất, do các thực nghiệm mới chỉ được tiến hành trên một phần của tập dữ liệu OlpBench, trong khi vẫn đánh giá trên toàn bộ tập kiểm thử gốc. Ngoài ra, việc huấn luyện trước tốn nhiều thời gian và tài nguyên phần cứng, do hạn chế về tài nguyên tính toán nên với mô hình đề xuất trong luận án chưa được thực nghiệm với các bộ siêu tham số khác nhau để có thể đưa ra kết quả tối ưu hơn.

II. Kết quả khi học chuyển giao

Kết quả với tập dữ liệu hoàn thiện ĐTTT mở: Kết quả thực nghiệm của mô hình đề xuất trên tập dữ liệu ReVerb20K và tập dữ liệu ReVerb45k được đưa ra tại Bảng 2.4. Qua kết quả thực nghiệm, luận án nhận thấy phần lớn các mô hình cho kết quả tốt hơn khi được tinh chỉnh với mô hình đã qua huấn luyện trước trên tập dữ liệu OlpBench.

Khi sử dụng mô hình nhúng từ FastText, mô hình đề xuất FastText-GRU-KGC, kết quả với kịch bản huấn luyện trước cải thiện tốt trên cả ba độ đo so với kết quả kịch bản huấn luyện từ đầu:

- Với tập dữ liệu ReVerb20K:
  - MR giảm nhiều nhất là $2{,}38$ lần.
  - MRR tăng nhiều nhất là $1{,}18$ lần.
  - $Hit@10$ tăng nhiều nhất là $1{,}20$ lần.
- Với tập dữ liệu ReVerb45K:

- MR giảm nhiều nhất là 2,30 lần.
- MRR tăng nhiều nhất là 1,24 lần.
- $Hit@10$ tăng nhiều nhất là 1,47 lần.

Bảng 2.4. So sánh kết quả giữa mô hình đề xuất và mô hình gốc trên các tập dữ liệu OKBC khác nhau. Tại mỗi cột, kết quả in đậm là tốt nhất, kết quả gạch chân là tốt thứ hai.

Mô hình hoàn thiện ĐTTT	Nhúng từ	Học chuyển giao	ReVerb20K			ReVerb45K		
			MR ↓	MRR↑	H@10↑	MR ↓	MRR↑	H@10↑
TuckER	BERT	Có	239	<u>0,977</u>	<u>0,977</u>	896	**0,955**	<u>0,964</u>
		Không	**46**	**0,996**	**0,996**	727	0,246	0,245
	FastText	Có	322	0,374	0,524	1196	0,265	0,387
		Không	765	0,317	0,436	2748	0,213	0,287
ConvE	BERT	Có	622	0,305	0,427	1432	0,159	0,276
		Không	1192	0,211	0,317	1390	0,286	0,452
	FastText	Có	390	0,347	0,489	1167	0,263	0,382
		Không	417	0,338	0,472	1472	0,275	0,399
5★E	BERT	Có	187	0,124	0,974	1094	0,112	0.761
		Không	<u>107</u>	0,493	0,988	**126**	<u>0,500</u>	**0,995**
	FastText	Có	293	0,392	0,548	842	0,289	0,551
		Không	563	0,331	0,466	1598	0,240	0,375
GRU_5★E [17]	Glove	Có	202	0,417	0,586	596	0,382	0,537
GRU_TuckER [17]	Glove	Có	245	0,397	0,558	706	0,331	0,477
GRU_ConvE [17]	Glove	Có	184	0,409	0,573	600	0,357	0,509
GRU_5★E [65] *	Glove	Có	201	0,415	0,589	<u>572</u>	0,376	0,534
GRU_TuckER [65] *	Glove	Có	256	0,394	0,556	808	0,322	0,458
GRU_ConvE [65] *	Glove	Có	184	0,408	0,570	582	0,353	0,507

Khi sử dụng mô hình nhúng từ BERT, mô hình đề xuất là BERT-GRU-KGC, kết quả kịch bản huấn luyện từ đầu tốt hơn kết quả kịch bản huấn luyện trước khi sử dụng mô hình TuckER và mô hình 5★E. Điều này xảy ra có thể được giải thích rằng do trong mô hình 5★E không chia sẻ các tham số trong quá trình học chuyển giao, mà việc này chỉ đóng vai trò là khởi tạo nhúng bộ ba.

Còn với mô hình TuckER có thể giải thích một trong những nguyên nhân chính dẫn đến hiệu năng suy giảm khi tinh chỉnh mô hình từ OLPBench sang ReVerb20K là sự khác biệt gần như tuyệt đối về tập thực thể và quan hệ giữa hai bộ dữ liệu khi có tới 99,9% mẫu kiểm thử trong ReVerb20K và 99,3% trong ReVerb45K chứa ít

nhất một thực thể hoặc quan hệ không xuất hiện trong OLPBench [17]. Sự khác biệt gần như tuyệt đối về tập thực thể/quan hệ giữa OLPBench và ReVerb20K dẫn đến mô hình BERT–KGC không thể tái sử dụng nhúng hiệu quả, gây suy giảm hiệu năng khi tinh chỉnh. Về cơ bản, hiệu năng của mô hình vẫn sẽ cải thiện nếu được huấn luyện trước trên tập dữ liệu OlpBench.

Bảng 2.5. So sánh kết quả giữa các mô hình KBC trên một số ĐTTT đã chuẩn hóa. Tại mỗi cột, kết quả in đậm là tốt nhất, kết quả gạch chân là tốt thứ hai.

Mô hình hoàn thiện ĐTTT	**Nhúng từ**	**Học chuyển giao**	**FB15K237**			**WN18RR**		
			MR↓	**MRR↑**	**H@10↑**	**MR↓**	**MRR↑**	**H@10↑**
TuckER	BERT	Có	1818	0,209	**0,626**	5186	0,166	0,271
		Không	1163	0,156	0,249	**615**	**0,985**	**0,985**
	FastText	Có	174	0,334	0,517	4013	0,432	0,471
		Không	187	0,328	0,526	4112	0,353	0,452
	GloVe [17]	Có	<u>151</u>	**0,363**	<u>0,550</u>	3456	0,467	0,529
ConvE	BERT	Có	704	0,181	0,288	<u>2540</u>	0,401	0,498
		Không	456	0,240	0,368	3897	0,389	0,442
	FastText	Có	188	0,317	0,490	4485	0,421	0,476
		Không	193	0,314	0,491	4782	0,398	0,450
	GloVe [17]	Có	200	0,325	0,510	5792	0,435	0,486
5⋆E	BERT	Có	996	0,166	0,256	2675	0,421	0,512
		Không	986	0,152	0,246	2986	0,388	0,493
	FastText	Có	168	0,336	0,521	3000	0,379	0,497
		Không	174	0,315	0,492	2880	0,386	0,502
	GloVe [17]	Có	**143**	<u>0,357</u>	0,544	2636	<u>0,492</u>	<u>0,582</u>

Ngoài ra, qua kết quả thực nghiệm, luận án nhận thấy kết quả của mô hình đề xuất BERT-GRU-KGC vẫn tốt hơn rất nhiều so với kết quả của mô hình đề xuất FastText-GRU-KGC. Cụ thể, khi so sánh kết quả tốt nhất của hai mô hình trên các tập dữ liệu (bao gồm cả kịch bản huấn luyện từ đầu và kịch bản huấn luyện trước):

- Với tập dữ liệu ReVerb20K:
  - Đối với MR: kết quả của mô hình BERT-GRU-KGC giảm 6,40 lần so với kết quả của mô hình FastText-GRU-KGC.
  - Đối với MRR: kết quả của mô hình BERT-GRU-KGC tăng 2,54 lần so với kết quả của mô hình FastText-GRU-KGC.
  - Đối với $Hit@10$: kết quả của mô hình BERT-GRU-KGC tăng 1,8 lần so với kết quả của FastText-GRU-KGC.
- Với tập dữ liệu ReVerb45K:
  - Đối với MR: kết quả của mô hình BERT-GRU-KGC giảm 6,68 lần so với kết quả của mô hình FastText-GRU-KGC.
  - Đối với MRR: kết quả của mô hình BERT-GRU-KGC tăng 3,47 lần so với kết quả của mô hình FastText-GRU-KGC.
  - Đối với $Hit@10$: Kết quả của mô hình BERT-GRU-KGC tăng 1,80 lần so với kết quả của mô hình FastText-GRU-KGC.

Ngoài ra, vào năm 2024, V. Kocijan, M. Jang và T. Lukasiewicz [65] công bố kết quả nghiên cứu mới tại tạp chí "Artificial Intelligence". Bài báo này bao gồm hai phần, trong đó phần đầu tiên giữ nguyên cách tiếp cận học chuyển giao trong [17] và đề xuất cải tiến thực nghiệm bằng cách lấy giá trị trung bình của 05 lần thực nghiệm thay vì đưa ra kết quả thực nghiệm tốt nhất như [17]. Các kết quả thực nghiệm [65] được đưa vào ba dòng cuối của Bảng 2.4. Qua so sánh, luận án nhận thấy các kết quả [65] khá tương đồng với các kết quả [17]. Tuy nhiên, các kết quả này cũng không tốt hơn so với kết quả của mô hình đề xuất trong luận án.

Tiếp theo luận án đưa ra so sánh về ảnh hưởng giữa các thành phần trong mô hình hoàn thiện ĐTTT khác nhau trong mô hình đề xuất và mô hình gốc của V. Kocijan và T. Lukasiewicz [17] (với mô hình gốc luận án sẽ lấy kết quả tốt nhất). Bên cạnh đó, để đánh giá ảnh hưởng của kịch bản huấn luyện trước so với kịch bản huấn luyện từ đầu đối với ĐTTT đã chuẩn hóa, luận án thực nghiệm trên các tập dữ liệu FB15K237 và WN18RR. Kết quả đưa ra tại Bảng 2.5. Từ bảng này, luận án nhận thấy rằng các kết quả tốt nhất thường rơi vào mô hình đã được huấn luyện trước (khi so sánh với các mô hình huấn luyện từ đầu). Bên cạnh đó, đối với tập dữ liệu đã được chuẩn hóa (FB15K237 và WN18RR) thì sự khác biệt này thấp hơn so với tập dữ liệu

mở chưa được chuẩn hóa (ReVerb20K và ReVerb45K). Nhìn chung, phần lớn kết quả của mô hình đề xuất đều tốt hơn kết quả tốt nhất của mô hình gốc của V. Kocijan và T. Lukasiewicz tại [17]. Dưới đây là một số nhận xét so sánh chi tiết:

- Đối với tập dữ liệu FB15K237: Khi so sánh độ đo MR và MRR thì luận án nhận thấy rằng kết quả của mô hình đề xuất BERT/FastText-GRU-KGC không tốt hơn so với kết quả của mô hình gốc Glove-GRU-KGC. Tuy nhiên, khi so sánh độ đo $Hit@10$ thì kết quả tốt nhất thu được bởi mô hình đề xuất BERT-GRU-KGC.
- Đối với tập dữ liệu WN18RR: Mô hình đề xuất BERT-GRU-TuckER tốt nhất ở cả ba độ đo MR, MRR, $Hit@10$.

Bằng thực nghiệm khi so sánh giữa việc sử dụng các thành phần mô hình nhúng từ khác nhau thì kết quả của mô hình nhúng từ BERT thường tốt nhất. Ngoài ra, với từng thành phần mô hình nhúng từ và từng thành phần mô hình hoàn thiện ĐTTT, thì kết quả khi sử dụng học chuyển giao phần lớn tốt hơn kết quả khi huấn luyện mô hình từ đầu. Điều này có thể dẫn đến tầm ảnh hưởng của việc học chuyển giao và quá trình sử dụng học chuyển giao hoàn toàn phù hợp với tập dữ liệu được chuẩn hóa trong hoàn thiện ĐTTT.

Hơn nữa, trong phần 2 của bài báo [65], V. Kocijan, M. Jang và T. Lukasiewicz đã xây dựng một bộ dữ liệu mới DOGE (Diagnostics of Open knowledge Graph Embeddings) với mục đích làm sáng tỏ hơn nữa và khẳng định tầm quan trọng của phương pháp học chuyển giao trong hoàn thiện ĐTTT. Các tác giả xây dựng bộ dữ liệu DOGE nhằm mục đích "chuẩn đoán" các mô hình đã được tiền huấn luyện, nhằm hiểu sâu hơn những gì chúng thực sự học được, thay vì chỉ đo lường hiệu suất tổng thể. Với bộ dữ liệu DOGE mới này, các tác giả tiến hành thực nghiệm theo ba mục tiêu khác nhau: đánh giá độ bao phủ và tính nhất quán của kiến thức tổng quát; đánh giá khả năng suy luận các sự kiện mới; phát hiện sự hiện diện và mức độ bền vững của định kiến giới tính. Luận án nhận thấy kết quả tốt nhất khi thực nghiệm trên bộ dữ liệu chuẩn hóa DOGE [65] như sau:

- $MR = 2{,}79$ (với mô hình hoàn thiện ĐTTT 5★E).
- $MRR = 0{,}702$ (với mô hình hoàn thiện ĐTTT 5★E).
- $Hits@10 = 0{,}57$ (với mô hình hoàn thiện ĐTTT 5★E).

Qua các kết quả trong Bảng 2.4, Bảng 2.5 thì kết quả tốt nhất của mô hình đề xuất trong luận án vẫn tốt hơn (mặc dù được thực nghiệm trên 05 bộ dữ liệu thông thường) các kết quả tốt nhất bên trên ở một số chỉ số MRR và $Hits@10$.

2.4 Kết luận Chương 2

Chương này tập trung trình bày kết quả nghiên cứu của luận án trong việc đề xuất mô hình hoàn thiện ĐTTT BERT/FastText-GRU-KGC được cải tiến từ GloVe-GRU-KGC [17]. Đánh giá thực nghiệm cho thấy các đề xuất cải tiến là hợp lý khi nhìn chung mô hình đề xuất có hiệu năng tốt hơn mô hình gốc. Kết quả nghiên cứu trong chương này được công bố trong [AnhNTT1] và đã nhận được tham chiếu từ một bài báo Scopus[3].

Chương 3 của luận án sẽ trình bày đề xuất của luận án về mô hình hoàn thiện ĐTTT đa phương thức.

[3] N. V. Hung, T. Q. Loi, N. T. Huong, T. T. T. Hang, and T. T. Huong, "AAFNDL - An Accurate Fake Information Recognition Model Using Deep Learning for the Vietnamese Language," *Informatics and Automation*, vol. 22, no. 4, pp. 795–825, 2023.

Chương 3. Hoàn thiện ĐTTT đa phương thức

Chương này trình bày về hai mô hình hoàn thiện ĐTTT đa phương thức ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT do luận án đề xuất dựa trên việc cải tiến mô hình hoàn thiện ĐTTT đa phương thức AdaMF-MAT (Adaptive Multi-modal Fusion and Modality Adversarial Training) được Y. Zhang và cộng sự đề xuất [18]. Mục đầu tiên giới thiệu về các thách thức từ mất cân bằng thông tin đa phương thức. Mục thứ hai phân tích khái quát về mô hình AdaMF-MAT do Y. Zhang và cộng sự đề xuất [18] và đưa ra một số nhận xét và ý tưởng cài tiến AdaMF-MAT. Hai mô hình đề xuất ViT-AdaMF-MAT, T5-ViT-AdaMF-MAT và thực nghiệm đánh giá hiệu năng của chúng được trình bày trong hai mục tiếp theo.

3.1 Mô hình Hoàn thiện ĐTTT đa phương thức AdaMF-MAT

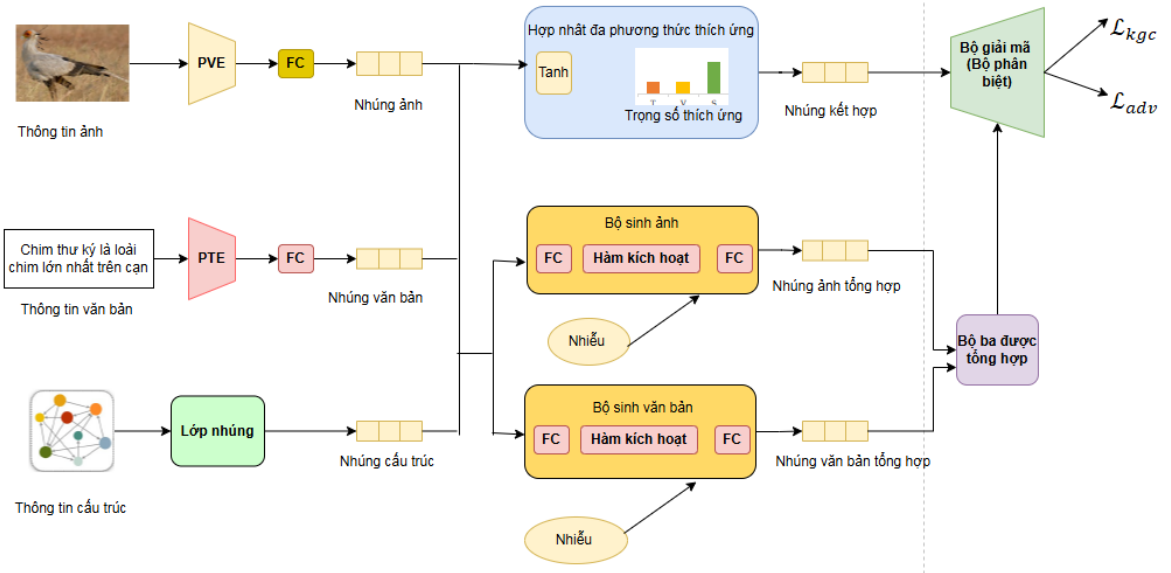
3.1.1 Giới thiệu chung

Phần này trình bày về mô hình hoàn thiện ĐTTT đa phương thức AdaMF-MAT, được Y. Zhang và cộng sự [18] đề xuất vào năm 2024.

Mô hình AdaMF-MAT được đề xuất để giải quyết các vấn đề không khớp thông tin giữa các thực thể. Đối với nhúng văn bản trong mô hình AdaMF-MAT, Y. Zhang và cộng sự lấy ra đặc trưng văn bản của mỗi thực thể cùng với mô tả của thực thể đó và mã hóa văn bản được được huấn luyện trước bằng cách sử dụng mô hình PTE (N. Reimers và I. Gurevych [66]). Sau đó, áp dụng mã duy nhất CLS (J. Devlin và cộng sự [53]) để trích rút đặc trưng văn bản cấp độ câu. Đối với thành phần nhúng ảnh, Y. Zhang và cộng sự đề xuất sử dụng bộ mã hóa ảnh huấn luyện trước (K. Simonyan và A. Ziserman [67]).

Để giải quyết vấn đề mất cân bằng về thông tin phương thức trong các mô hình hoàn thiện ĐTTT nói chung và một số mô hình hoàn thiện ĐTTT đa phương thức nói riêng, Y. Zhang và cộng sự [18] đề xuất một mô hình hoàn thiện ĐTTT đa phương thức mới bằng cách kết hợp giữa việc hợp nhất đa phương thức thích ứng AdaMF (Adaptive Multi-modal Fusion - AdaMF) và huấn luyện đối kháng theo phương thức MAT (Modality Adversarial Training – MAT) để tăng cường và sử dụng hiệu quả thông tin đa phương thức. Mô hình hoàn thiện ĐTTT đa phương thức AdaMF-MAT được tạo ra bằng cách kết hợp giữa hai mô-đun này, cụ thể:

- Mô-đun AdaMF cho phép trích rút có chọn lọn các đặc trưng đa phương thức thiết yếu từ các thực thể để tạo ra các nhúng biểu diễn kết hợp. Mô-đun này cho phép cải thiện việc mất cân bằng thông tin phương thức giữa nhúng cấu trúc, nhúng ảnh và nhúng văn bản để tạo ra nhúng biểu diễn tổng hợp.
- Mô-đun MAT để tạo ra các nhúng đa phương thức tổng hợp và xây dựng các ví dụ đối nghịch, nhằm mục đích cải tiến giới hạn thông tin đa phương thức trong suốt quá trình huấn luyện. Mô-đun MAT hướng đến việc tạo ra các mẫu đối nghịch với các nhúng đa phương thức tổng hợp để nâng cao quá trình huấn luyện trong mô hình hoàn thiện ĐTTT đa phương thức.



Hình 3.1. Mô hình AdaMF-MAT [18].

Hơn nữa, trong khi các mô hình hoàn thiện ĐTTT đa phương thức dựa theo phương pháp lấy mẫu âm thường thiết kế các chiến thuật phức tạp để lấy mẫu trong các ĐTTT đa phương thức, thì mô hình AdaMF-MAT trực tiếp tạo ra các mẫu tổng hợp với thông tin đa phương thức giàu ngữ nghĩa để cải tiến khả năng học nhúng đa phương thức.

Như được minh họa trên Hình 3.1, ba thành phần chính trong mô hình AdaMF-MAT [18] là: bộ mã hóa đặc trưng, sự kết hợp đa phương thức thích ứng và phương thức huấn luyện đối kháng.

Một mô hình hoàn thiện ĐTTT đa phương thức nhúng các thực thể và các quan hệ vào một không gian véc-tơ liên tục, trong đó, các nhúng để mô tả các đặc trưng phương thức khác nhau được kỹ hiệu như sau:

- Đối với mỗi thực thể $e \in \mathcal{E}$, nhúng cấu trúc ký hiệu là $\mathbf{e}_s$; nhúng ảnh ký hiệu là $\mathbf{e}_v$; nhúng văn bản ký hiệu là $\mathbf{e}_t$.
- Với mỗi quan hệ $r \in \mathcal{R}$, nhúng quan hệ r này là $\mathbf{e}_r$.

Tương tự như trong mô hình hoàn thiện ĐTTT nói chung, mô hình hoàn thiện ĐTTT đa phương thức cũng đo độ hợp lý của mỗi bộ ba $(h, r, t) \in \mathcal{T}$ với một hàm tính điểm $\mathcal{F}$, cho phép tính toán giá trị của các nhúng đã được xác định. Trong bước suy luận của mô hình hoàn thiện ĐTTT, với mỗi một truy vấn đã cho $(h, r, ?)$ và $(?, r, t)$, mô hình xếp hạng các điểm số của từng bộ ba tương ứng của mỗi thực thể ứng viên và đưa ra dự đoán.

3.1.2 Bộ mã hóa đặc trưng đa phương thức

Phần này trình bày về quá trình mã hóa trong AdaMF-MAT [18] để trích rút các đặc trưng đa phương thức từ các ảnh gốc và các mô tả văn bản của các thực thể. Đây là một bước rất quan trọng trong tất cả các mô hình hoàn thiện ĐTTT đa phương thức. Như được minh họa ở Hình 3.1, quá trình này bao gồm các bước sau:

1) Trước tiên đối với ảnh, bộ mã hóa ảnh được huấn luyện trước PVE (H. Bao và cộng sự [68]) được áp dụng để trích rút đặc trưng ảnh $\mathbf{f}_v$ đối với mỗi thực thể e bằng cách sử dụng hàm trung bình. Sau đó, thông tin toàn bộ hình ảnh được gom lại thành một véc-tơ duy nhất đại diện cho thực thể và chiếu đặc trưng ảnh lên không gian nhúng để thu được nhúng ảnh $\boldsymbol{e}_v$. Cụ thể công thức dưới đây.

$$\boldsymbol{f}_v = \frac{1}{|\mathcal{V}_e|} \sum_{img_i \in \mathcal{V}_e} PVE(img_i) \qquad (3\text{-}1)$$
$$\boldsymbol{e}_v = \boldsymbol{W}_v . \boldsymbol{f}_v + \boldsymbol{b}_v$$

trong đó, $\mathbf{W}_v$ và $\mathbf{b}_v$ là các tham số của lớp chiếu ảnh.

2) Tiếp theo, đối với nhúng văn bản, đặc trưng văn bản của mỗi thực thể được trích rút với mô tả của nó và bộ mã hóa văn bản huấn luyện trước PTE trong BeiT (N. Reimers và I. Gurevych [66]). Mã CLS (mô hình BERT [53]) được

áp dụng để thu được đặc trưng văn bản cấp độ câu $\mathbf{f}_t$. Sau đó, đặc trưng văn bản $\mathbf{f}_t$ được chiếu xuống không gian nhúng với các tham số lớp $\mathbf{W}_t$, $\mathbf{b}_t$ để thu được nhúng $\mathbf{e}_t$, được mô tả cụ thể bởi công thức dưới đây.

$$\boldsymbol{e}_t = \boldsymbol{W}_t.\boldsymbol{f}_t + \boldsymbol{b}_t \qquad (3\text{-}2)$$

3) Cuối cùng, lớp nhúng cấu trúc: Nhằm ánh xạ thực thể và quan hệ trong ĐTTT vào không gian nhúng thông qua các kỹ thuật nhúng truyền thống.

Sau đó, tập hợp các nhúng theo ba nguồn nêu trên được điều chỉnh kích thước và đưa qua các lớp đầy đủ FC để chuyển đổi kích thước nhúng và học biểu diễn chung trước khi hợp nhất.

3.1.3 Hợp nhất đa phương thức thích ứng

Tiếp theo, thành phần hợp nhất đa phương thức thích ứng trong mô hình hoàn thiện ĐTTT đa phương thức AdaMF-MAT được giới thiệu. Thành phần này tích hợp ba loại biểu diễn đa phương thức thông qua cơ chế học trọng số thích ứng. Hàm kích hoạt $Tanh$ được sử dụng để điều chỉnh đầu ra, trong khi đó trọng số học được xác định mức độ đóng góp của từng phương thức vào biểu diễn chung. Cách tiếp cận này cho phép mô hình thích nghi linh hoạt với trường hợp thiếu hụt một hoặc nhiều phương thức, bằng cách giảm trọng số tương ứng.

Trong mô hình hoàn thiện ĐTTT đa phương thức, thông tin của ba phương thức $m \in \mathcal{M} = \{s, v, t\}$ sẽ được xem xét để đo độ hợp lý của bộ ba. Cũng như một số mô hình hoàn thiện ĐTTT đa phương thức đã tồn tại trước đây, trong mô hình AdaMF-MAT, Y. Zhang và cộng sự cũng tính toán các phương thức độc lập. Sau đó, gộp kết quả lại. Một cơ chế hợp nhất đa phương thức thích ứng AdaMF được đề xuất để thu được kết quả tổng hợp. Đối với mỗi thực thể bất kỳ thì gọi nhúng của nó là $\mathbf{e}_m$, với $m \in \mathcal{M}$, mô hình AdaMF được tính toán theo công thức dưới đây

$$\alpha_m = \frac{exp(\boldsymbol{w}_m \oplus tanh(\boldsymbol{e}_m))}{\sum_{n \in \mathcal{M}} exp(\boldsymbol{w}_n \oplus tanh(\boldsymbol{e}_n))} \qquad (3\text{-}3)$$

trong đó

- α_m được gọi là trọng số của từng phương thức và thể hiện phương thức nào cung cấp thông tin quan trọng hơn trong từng ngữ cảnh cụ thể.

- Toán tử $\oplus$ gọi là tích từng điểm (pointwise product) và $\mathbf{w}_m$ véc-tơ có thể học của phương thức m dùng để tính mức độ quan trọng của phương thức m.

Tiếp theo, nhúng tổng hợp các phương thức ký hiệu là $\mathbf{e}_{joint}$ và được tính theo công thức dưới đây.

$$\boldsymbol{e}_{joint} = \sum_{m \in \mathcal{M}} \alpha_m \boldsymbol{e}_m \tag{3-4}$$

Với mỗi thực thể khác nhau, mô hình AdaMF cho phép học các trọng số phương thức khác nhau. Do vậy, quá trình này cho phép đạt được sự hợp nhất thông tin phương thức tích ứng để tạo ra các nhúng tổng hợp đại diện.

Sau đó, hàm tính điểm từ mô hình hoàn thiện ĐTTT RotatE (Z. Sun và cộng sự [38]) được sử dụng để đo độ hợp lý của các bộ ba. Công thức hàm tính điểm được mô tả dưới đây.

$$\mathcal{F}(h, r, t) = \left\| \boldsymbol{h}_{joint} \circ \boldsymbol{e}_r - \boldsymbol{t}_{joint} \right\| \tag{3-5}$$

trong đó, $\circ$ là toán tử quay trong không gian phức. Các nhúng tổng hợp của thực thể đầu và thực thể cuối được đưa vào trong hàm tính điểm.

Mục đích của mô hình là tăng điểm số đối với các bộ ba dương (bộ ba đúng) và giảm điểm số của bộ ba âm (bộ ba sai) để tối thiểu điểm số của $\mathcal{L}_{kgc}$. Do vậy các nhúng được huấn luyện bằng cách sử dụng hàm mất mát sigmoid như Z. Sun và cộng sự [38]. Hàm mất mát được cho theo Công thức (3-6) dưới đây.

$$\mathcal{L}_{kgc} = \frac{1}{|\mathcal{T}|} \sum_{(h,r,t) \in \mathcal{T}} \left(-log\, \sigma(\gamma - \mathcal{F}(h, r, t) - \sum_{i=1}^{K} p_i\, log\, \sigma(\mathcal{F}(h_i', r_i', t_i') - \gamma) \right) \tag{3-6}$$

trong đó

- σ là hàm sigmoid.
- γ là biên độ (margin).

- p_i là trọng số tự đối kháng cho mỗi bộ ba âm (bộ ba tiêu cực) (h_i', r_i', t_i') được tạo ra bởi phương pháp lấy mẫu âm bằng chiến lược thay thế ngẫu nhiên thực thể đầu và thực thể cuối của bộ ba đúng bằng một thực thể từ tập thực thể $\mathcal{E}$ (dựa theo nghiên cứu của A. Bordes và cộng sự [36]).

Trọng số p_i phản ánh mức độ khó phân biệt với bộ ba thật được cho bởi công thức dưới đây.

$$p_i = \frac{exp\left(\beta \mathcal{F}(h_i', r_i', t_i')\right)}{\sum_{j=1}^{K} exp\left(\beta \mathcal{F}\left(h_j', r_j', t_j'\right)\right)} \quad (3\text{-}7)$$

trong đó

- β là tham số.
- K là số lượng mẫu âm được tạo ra cho mỗi bộ ba đương.

Tác vụ chính của mô hình này là tối thiểu hàm số $\mathcal{L}_{kgc}$.

3.1.4 Huấn luyện đối kháng theo phương thức

Theo Y. Zhang và cộng sự [18], các nhúng đa phương thức của các thực thể chứa một số giới hạn thông tin và dẫn đến vấn đề mất cân bằng. Mặc dù mô hình AdaMF có thể chọn lọc và hợp nhất các nhúng đa phương thức vào một nhúng biểu diễn chung, nhưng mô hình này vẫn bị giới hạn bởi chất lượng của các nhúng đầu vào. Bên cạnh đó, nó không thể mở rộng các nhúng đa phương thức đã tồn tại để cung cấp thông tin phong phú hơn về ngữ nghĩa. Do đó, khi thông tin từ một hoặc nhiều phương thức là không đầy đủ, khả năng biểu diễn ngữ nghĩa của thực thể sẽ bị hạn chế, từ đó ảnh hưởng đến hiệu năng hoàn thiện ĐTTT.

Chính vì vậy, huấn luyện đối kháng AT (G. Mariani và cộng sự [69]) được sử dụng và một cơ chế huấn luyện đối kháng theo phương thức (Modaltily Adversarial Training – MAT) được thiết kế cho hoàn thiện ĐTTT đa phương thức để cải tiến việc mất cân bằng thông tin đa phương thức. Thay vì chỉ kết hợp các phương thức sẵn có (như AdaMF thông thường), cách tiếp cận này cho phép bổ sung thông tin từ các phương thức bị thiếu hoặc yếu thông qua việc sinh ra nhúng đối nghịch. Cơ chế MAT sử dụng một bộ tạo (generator) G để tạo ra các mẫu đối nghịch và một bộ phân biệt (discriminator) D để đo lường tính hợp lý của chúng và tuân thủ theo mô hình AT tổng quát. Đây được xem là đóng góp kỹ thuật quan trọng giúp nâng cao khả năng

khái quát hóa và cải thiện độ chính xác của hệ thống hoàn thiện đồ thị tri thức trong điều kiện dữ liệu đa phương thức không đồng nhất.

Tiếp theo, cơ chế thiết kế của bộ tạo G và bộ phân biệt D trong mô hình AdaMF-MAT được giới thiệu.

A. Bộ tạo G

Y. Zhang và cộng sự thiết kế một bộ tạo đối kháng phương thức để đảm nhận vai trò của G bằng cách tạo ra các nhúng đa phương thức với điều kiện dựa trên nhúng có cấu trúc $\mathbf{e}_s$, nhằm mục đích xây dựng mẫu đối kháng. Bộ tạo đối kháng phương thức là một mạng truyền thẳng hai tầng được cho bởi công thức dưới đây

$$G_m(\boldsymbol{e}_s, z) = \boldsymbol{W}_2 \cdot \delta(\boldsymbol{W}_1 \cdot [\boldsymbol{e}_s; z] + b_1) + b_2 \qquad (3\text{-}8)$$

trong đó

- $\mathbf{W}_1$, $\mathbf{W}_2$, b_1, b_2 là các tham số của các lớp truyền thẳng hai tầng.
- δ là hệ số hàm kích hoạt LeakyReLU (tham khảo từ [70]).
- $m \in \{v, t\}$ là ảnh hoặc phương thức văn bản.
- $z \sim \mathcal{N}(0,1)$ là nhiễu ngẫu nhiên được lấy từ một phân phối chuẩn (Gaussian hoặc uniform), đóng vai trò tạo tính đa dạng cho các nhúng sinh ra.

Từ Công thức (3-8), đối với mỗi bộ ba (h, r, t), nhúng đa phương thức nhân tạo cho mỗi thực thể đầu và thực thể cuối được tạo ra. Từ đó làm cơ sở để xây dựng các thực thể nhân tạo. Ví dụ với mỗi thực thể nhân tạo h^* bao gồm ba nhúng $\mathbf{h}_s^*$, $\mathbf{h}_v^*$, $\mathbf{h}_t^*$, ở đó $\mathbf{h}_v^*$, $\mathbf{h}_t^*$ được tạo ra bởi bộ huấn luyện đối kháng theo phương thức G_v và G_t.

Từ đó, bộ ba nhân tạo với hai thực thể nhân tạo đầu và cuối được xây dựng, cụ thể là $\mathcal{S}(h, r, t) = \{(h, r, t^*), (h^*, r, t), (h^*, r, t^*)\}$ đối với một bộ ba (h, r, t). Ở đây, $\mathcal{S}$ chứa một nhóm bộ ba nhân tạo cho (h, r, t), cái mà bao gồm ba mẫu đối kháng. Trong thực tế, các tác giả tạo ra các nhóm L của các bộ ba nhân tạo (h, r, t), và $\mathcal{S}$ sẽ chứa $3L$ bộ ba nhân tạo. Hơn nữa, bộ tạo chứa các nhiễu, nên các bộ ba nhân tạo này khác biệt với nhau.

B. Bộ phân biệt

Với các ví dụ đối kháng được tạo ra cùng với nhúng đa phương thức nhân tạo, bộ phân biệt D nhằm phân biệt bộ ba dương từ các bộ ba nhân tạo trong khi bộ tạo G

Thuật toán huấn luyện của mô hình AdaMF-MAT		
Đầu vào		Một lô của bộ ba huấn luyện $\mathcal{B}$ lấy mẫu từ tập $\mathcal{T}$, thông tin đa phương thức của các thực thể, bộ phân biệt D và bộ sinh G của mô hình AdaMF.
Đầu ra		Mô hình hoàn thiện ĐTTT đa phương thức với bộ phân biệt D được huấn luyện với MAT.
1	**for**	Bộ ba $(h, r, t) \in \mathcal{B}$ **do**
		// Huấn luyện bộ phân biệt D
2		Lấy các nhúng tổng hợp $\mathbf{h}_{joint}$, $\mathbf{t}_{joint}$
3		Tính toán điểm số của bộ ba $\mathcal{F}(h, r, t)$ và bộ ba âm $\mathcal{F}(h_i', r_i', t_i')$
4		Tính hàm mất mát kgc $\mathcal{L}_{kgc}$
5		Tạo tập ví dụ đối kháng $\mathcal{S}$ với bộ sinh G
6		Tính toán hàm mất mát đối kháng $\mathcal{L}_{adv}$
7		Tính toán hàm mất mát tổng $\mathcal{L}_{kgc} + \lambda\mathcal{L}_{adv}$
8		Lan truyền ngược và tối ưu bộ phân biệt D
		//Huấn luyện bộ sinh G
9		Lấy các nhúng tổng hợp $\mathbf{h}_{joint}$, $\mathbf{t}_{joint}$
10		Tạo tập ví dụ đối kháng $\mathcal{S}$ với bộ sinh G
11		Tính toán hàm mất mát đối kháng $\mathcal{L}_{adv}$
12		Lan truyền ngược và tối ưu bội sinh G
13	**End**	

Hình 3.2. Thuật toán huấn luyện mô hình AdaMF-MAT [18].

nhằm tạo các các bộ thực tế và đánh lừa bộ phân biệt D. Hàm tính điểm $\mathcal{F}$ trong AdaMF-MAT đánh giá mức độ đúng đắn của một bộ ba trong hoàn thiện ĐTTT, đóng

vai trò là bộ phân biệt D trong thiết lập huấn luyện đối kháng AT bởi vì $\mathcal{F}$ là bộ phân biệt tính khả thi của bộ ba và mục tiêu cần tăng cường đối kháng. Do đó, hàm mất mát của huấn luyện đối kháng được mô tả dưới đây.

$$\mathcal{L}_{adv} = \frac{1}{|\mathcal{T}|} \sum_{(h,r,t)\in\mathcal{T}} \left(-log\, \sigma(\gamma - \mathcal{F}(h,r,t) - \frac{1}{|\mathcal{S}|} \sum_{(h^*,r^*,t^*)\in\mathcal{S}(h,r,t)} log\, \sigma(\mathcal{F}(h^*,r^*,t^*) - \gamma) \right) \quad (3\text{-}9)$$

Hàm mất mát $\mathcal{L}_{adv}$ là thành phần cốt lõi trong thiết lập huấn luyện đối kháng của mô hình AdaMF-MAT, nhằm thúc đẩy mô hình phân biệt hiệu quả giữa các bộ ba thực và bộ ba nhân tạo được sinh từ nhúng đa phương thức. Bằng cách tối đa hóa khoảng cách điểm số giữa hai loại bộ ba, hàm này giúp mô hình cải thiện khả năng biểu diễn và khái quát hoá trong các tình huống thiếu hụt phương thức.

C. Hàm mục tiêu

Trong suốt quá trình huấn luyện, mô hình lặp lại quá trình huấn luyện bộ phân biệt D và bộ tạo G theo cách đối kháng, đồng thời vẫn tối ưu hàm mất mát $\mathcal{L}_{kgc}$ cho hoàn thiện ĐTTT đa phương thức. Do đó, mục tiêu huấn luyện có thể được mô tả theo biểu thức dưới đây.

$$\min_D \mathcal{L}_{kgc} + \min_D \max_G \lambda \mathcal{L}_{adv} \quad (3\text{-}10)$$

trong đó, λ là hệ số mất mát đối kháng. Thuật toán huấn luyện từng lô nhỏ trong mô hình hoàn thiện ĐTTT AdaMF-MAT được mô tả dưới đây (Hình 3.2).

3.1.5 Ý tưởng cải tiến

Đối với ĐTTT đa phương thức, việc lựa chọn mô hình học nhúng phù hợp cho mỗi phương thức sẽ ảnh hưởng trực tiếp đến hiệu năng của mô hình hoàn thiện ĐTTT vì mỗi phương thức mang một dạng thông tin riêng biệt, mô hình nhúng tốt phải bảo toàn được ngữ nghĩa của từng phương thức và đồng thời tạo điều kiện thuận lợi cho quá trình ánh xạ về không gian biểu diễn chung. Nếu mô hình nhúng không đủ mạnh, vector biểu diễn sẽ mất thông tin quan trọng, làm suy giảm khả năng học các đặc trưng sâu và dẫn tới sai lệch trong các tác vụ suy luận như dự đoán liên kết hoặc đánh giá tính hợp lệ của bộ ba. Hơn nữa, việc tích hợp các véc-tơ nhúng riêng lẻ vào không

gian chung đòi hỏi tính đồng nhất cao; khi đầu vào là các nhúng không tương thích, mô hình tổng hợp khó có thể học được một hàm kết hợp hiệu quả.

Trong các mô hình nhúng ảnh hiện nay, có thể phân loại theo ba hướng chính:

- **Mạng CNN truyền thống**: tiêu biểu như ResNet, Inception, EfficientNet – mạnh về nhận dạng đặc trưng cục bộ nhưng hạn chế khi xử lý ngữ cảnh dài và hiệu quả giảm khi dữ liệu lớn và phức tạp.
- **Mạng Transformer thị giác**: gồm Vision Transformer (ViT), BEiT, Swin Transformer – vượt trội trong khả năng học quan hệ toàn cục giữa các vùng ảnh nhờ cơ chế tự chú ý.
- **Mô hình học tự giám sát**: như MAE, DINO – tận dụng dữ liệu không gán nhãn để học biểu diễn giàu ngữ nghĩa.

Trong số đó, mô hình ViT (Vision Transformer) có những ưu thế vượt trội như với cơ chế tự chú ý là một thành phần cốt lõi trong kiến trúc Transformer giúp ViT rất hiệu quả trong việc học ngữ cảnh dài và quan hệ giữa các thành phần dữ liệu. Ngoài ra, ViT có ưu thế về khả năng kết nối với mô hình xử lý văn bản, nhờ chia sẻ cùng kiến trúc Transformer, từ đó hỗ trợ các bài toán đa phương thức như phân tích ngữ cảnh, học biểu diễn ngữ nghĩa và suy luận logic. ViT cũng có ưu điểm thực tiễn: nhẹ hơn các mô hình như BEiT, dễ dùng với các ảnh đã có nên tốc độ và hiệu quả của ViT nhanh hơn, đây cũng là một yếu tố quan trọng trong huấn luyện. ViT hoạt động tốt hơn khi mở rộng số layer, hidden size, quy mô dữ liệu.

Đối với nhúng văn bản, phần lớn các mô hình phổ biến hiện nay như BERT, RoBERTa, GPT đều dựa trên kiến trúc tổng thể bao gồm chỉ Encoder hoặc chỉ Decoder dẫn đến hạn chế về chức năng (BERT mạnh về biểu diễn ngữ cảnh nhưng không sinh văn bản, GPT mạnh về sinh văn bản nhưng yếu về biểu diễn). Để khắc phục điều này thì T5 (Text-to-Text Transfer Transformer) có kiến trúc tổng thể dạng Encoder – Decoder, điều này tạo ra nhúng mạnh ở Encoder, đồng thời sinh văn bản chất lượng ở Decoder mà các mô hình như BERT hoặc GPT không thể làm cùng lúc. Khi dùng Encoder, T5 có thể tạo ra nhúng mang thông tin ngữ nghĩa sâu, ngữ cảnh rộng và khả năng tổng quát hóa cao hơn so với nhiều mô hình embedding truyền thống.

Luận án nhận thấy rằng mô hình ViT và T5 có kiến trúc tương thích vì cùng dùng Transformer nên sự kết hợp giữa hai vector đầu ra dễ dàng hơn, giảm độ phức tạp và không cần adapter quá lớn. Quan trọng hơn, cả hai mô hình này đều học được biểu diễn sâu và ngữ nghĩa nên khi kết hợp tạo nhúng ảnh và văn bản sẽ tối ưu cho các tác vụ như hoàn thiện ĐTTT đa phương thức.

Từ những nhận định trên, luận án đưa ra ý tưởng đề xuất cải tiến nhúng của các thành phần thực thể đa phương thức để tăng hiệu năng của mô hình hoàn thiện ĐTTT đa phương thức. Cụ thể:

- Thay thế mô hình nhúng ảnh trong mô hình gốc bằng mô hình nhúng ảnh Vision Transformer (ViT) để khai thác tốt hơn ngữ cảnh toàn cục.
- Thay thế mô hình nhúng văn bản trong mô hình gốc bằng mô hình nhúng văn bản T5 để nâng cao hiệu quả trong việc học biểu diễn ngữ nghĩa và tổng quát hóa.

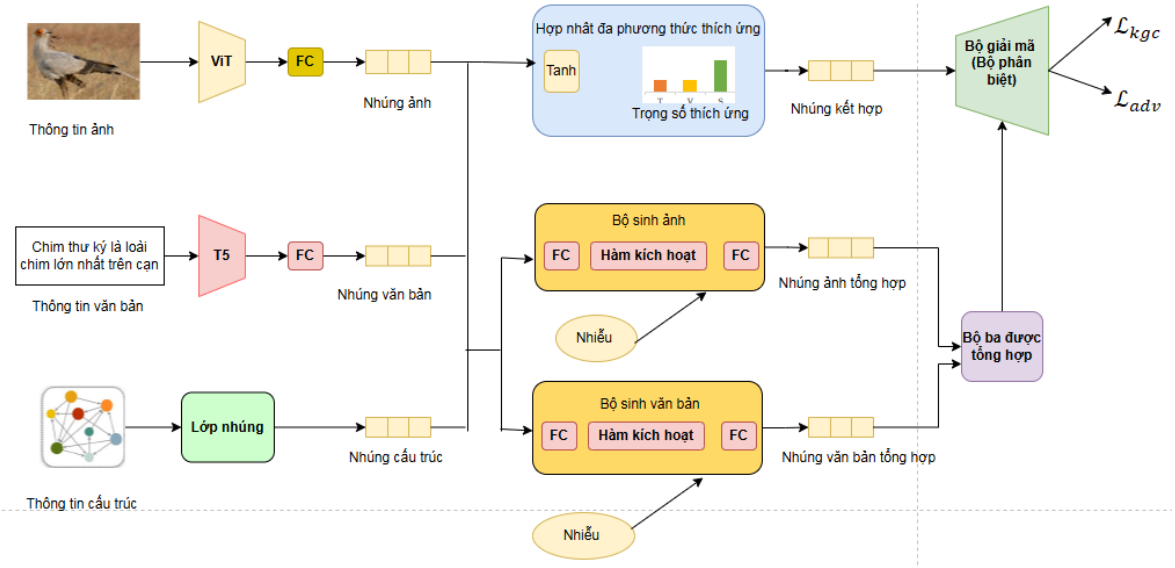
Luận án không chỉ dừng lại ở việc so sánh các mô hình nhúng đa phương thức mà còn xác định rõ sự tương thích kiến trúc giữa ViT và T5 nhằm tạo ra một không gian nhúng đa phương thức thống nhất vừa tận dụng khả năng học ngữ cảnh dài của ViT vừa khai thác sức mạnh tổng quát hóa của T5. Cách kết hợp này chưa được khai thác nhiều trong hoàn thiện ĐTTT, đặc biệt là trong nhiệm vụ hoàn thiện đồ thị tri thức đa phương thức. Ngoài ra, trong khi đa số công trình tập trung vào nhúng đơn phương thức, luận án hướng tới tích hợp thực thể đa phương thức trong không gian ĐTTT, giải quyết vấn đề thực tế khi nhiều tri thức trong đồ thị đến từ nguồn thông tin không đồng nhất (text, image).

3.2 Mô hình đề xuất ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT

Cấu trúc gồm hai thành phần chính AdaMF và MAT của mô hình AdaMF-MAT đã cho phép xử lý khá tốt tính mất cân bằng thông tin đa phương thức. Cho nên cấu trúc hai thành phần chính như vậy vẫn được giữ nguyên trong mô hình đề xuất của luận án. Luận án tập trung vào việc cải tiến quá trình mã hóa của mô hình AdaMF-MAT. Mô hình đề xuất sẽ cải tiến phần nhúng ảnh và nhúng thực thể từ mô hình gốc AdaMF-MAT. Cụ thể, hai thành phần nhúng trong mô hình đề xuất được thay thế như sau: thành phần nhúng ảnh sử dụng mô hình Vision Transformer do A. Dosovitskiy và cộng sự [50] giới thiệu; thành phần nhúng phương thức văn bản sử

dụng mô hình T5 "Text-to-Text-Transfer-Transformer" [71]. Luận án đề xuất hai mô hình cải tiến từ mô hình AdaMF-MAT như sau:

- Mô hình ViT-AdaMF-MAT chỉ thay thế mô-đun nhúng ảnh của AdaMF-MAT bằng ViT.
- Mô hình T5-ViT-AdaMF-MAT thay thế mô-đun nhúng văn bản bằng T5 và thay thế mô-đun ảnh bằng ViT.



Hình 3.3. Mô hình đề xuất T5-ViT-AdaMF-MAT

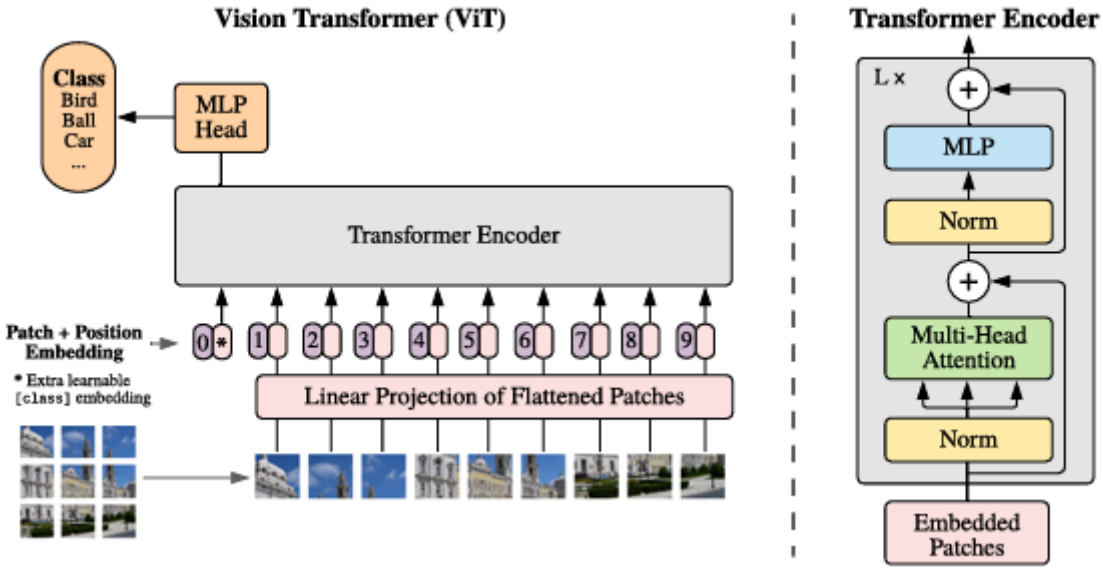
3.2.1 Kiến trúc mô hình T5-ViT-AdaMF-MAT đề xuất

Kiến trúc mô hình T5-ViT-AdaMF-MAT xem tại Hình 3.3; kiến trúc của ViT-AdaMF-MAT là hoàn toàn tương tự. Sau đây, luận án trình bày về các thành phần chính trong T5-ViT-AdaMF-MAT.

3.2.2 Nhúng ảnh bằng ViT

Dựa vào sự thành công của mô hình Transformer trong xử lý ngôn ngữ tự nhiên, A. Dosovitskiy và cộng sự [50] đã áp dụng trực tiếp một tiêu chuẩn Transformer lên các ảnh với một vài thay đổi nhỏ. Để làm như vậy, các tác giả phân chia một ảnh thành các phần nhỏ và cung cấp một dãy các nhúng tuyến tính tương ứng cho các phần này để tạo thành đầu vào cho mô hình Transformer. Các phần ảnh này được xử lý giống như là các tokens (các từ) trong mô hình xử lý ngôn ngữ tự nhiên. Các tác giả huấn luyện mô hình để phân loại ảnh theo phương pháp có giám

sát. Bằng các thực nghiệm, mô hình Vision Transformer (ViT) (cấu trúc mô hình ViT tại Hình 3.4) đạt kết quả rất tốt khi được tiền huấn luyện ở quy mô đủ lớn và chuyển giao sang tác vụ có ít dữ liệu hơn [50].



Hình 3.4. Mô hình Vision Transformer (ViT) [50]

Mô hình Transformer chuẩn nhận đầu vào là dãy các nhúng token 1 chiều. Khi đó, để xử lý ảnh hai chiều, các tác giả định hình lại ảnh 2 chiều $x \in \mathbb{R}^{H\times W\times C}$ thành một chuỗi các mảng hai chiều phẳng $x_p \in \mathbb{R}^{N\times(P^2C)}$, ở đó (H, W) là độ phân giải của ảnh gốc, C là số lượng các kênh, (P, P) là độ phân giải mỗi phân mảnh ảnh, và $N = HW/P^2$ là kết quả số lượng các phân mảnh (cũng đóng vai trò là chuỗi đầu vào hiệu quả cho Transformer). Transformer sử dụng véc-tơ phẳng với kích thước D chiều thông qua tất cả các lớp của nó. Do vậy, A. Dosovitskiy và cộng sự trải phẳng các chuỗi này ra và ánh xạ nó với phép chiếu tuyến tính D chiều được cho bởi công thức huấn luyện dưới đây. Đầu ra của phép chiếu này là những nhúng phân mảnh.

$$\boldsymbol{z}_0 = \left[\boldsymbol{x}_{class}; \boldsymbol{x}_p^1\boldsymbol{E}; \boldsymbol{x}_p^1\boldsymbol{E}; \ldots; \boldsymbol{x}_p^1\boldsymbol{E}\right] + \boldsymbol{E}_{pos} \qquad (3\text{-}11)$$
$$\boldsymbol{E} \in \mathbb{R}^{(P^2\cdot C)\times D}, \boldsymbol{E}_{pos} \in \mathbb{R}^{(N+1)\times D}$$

Tương tự như BERT, A. Dosovitskiy và cộng sự thêm một nhúng huấn luyện vào dãy các nhúng phân mảnh ($\mathbf{z}_0^0 = \mathbf{x}_{class}$) với trạng thái của nó tại đầu ra của bộ mã hóa Transformer ($\mathbf{z}_L^0$) đóng vai trò là biểu diễn ảnh $\mathbf{y}$ (Công thức (3-12)).

$$\mathbf{y} = LN(\mathbf{z}_L^0) \qquad (3\text{-}12)$$

Cả hai quá trình huấn luyện trước và tinh chỉnh với một đầu phân loại được gắn vào $\mathbf{z}_L^0$. Đầu phân loại này được thực hiện bởi một MLP với một lớp ẩn tại thời gian huấn luyện trước và một lớp tuyến tính duy nhất tại thời điểm tinh chỉnh.

Các nhúng vị trí được thêm vào các nhúng phân mảnh để giữ lại thông tin vị trí. Các tác giả sử dụng các nhúng vị trí 1D tiêu chuẩn. Dãy kết quả của các véc-tơ nhúng đóng vai trò là đầu vào cho bộ mã hóa.

Bộ mã hóa Transformer do A. Vaswani và cộng sự đề xuất [72] bao gồm các lớp xen kẽ với cơ chế tự chú ý đa đầu (Multiheaded Self-Attention - MSA) và các khối Perceptron đa tầng (Multilayer Perceptron – MLP) (Công thức ((3-13) và Công thức ((3-14)). Chuẩn hóa lớp (Layernorm) được áp dụng trước mỗi khối, và các kết nối dư sau mỗi khối. Khối Perceptron đa tầng chứa hai lớp với một hàm phi tuyến GELU (Gauss Error Linear Unit).

$$\mathbf{z}_l' = MSA\big(LN(\mathbf{z}_{l-1})\big) + \mathbf{z}_{l-1} \qquad (3\text{-}13)$$

$$l = 1 \dots L$$

$$\mathbf{z}_l = MLP\big(LN(\mathbf{z}_l')\big) + \mathbf{z}_l' \qquad (3\text{-}14)$$

$$l = 1 \dots L$$

A. Thiên vị quy nạp

Dựa trên việc bổ sung một số ràng buộc phù hợp, thiên vị quy nạp (inductive bias) cho phép một thuật toán ưu tiên một giải pháp (hoặc một cách diễn đạt) hơn một giải pháp khác, độc lập với dữ liệu quan sát, như vậy, thiên vị quy nạp vừa cải thiện việc tìm kiếm các giải pháp mà không làm giảm đáng kể hiệu năng, vừa giúp tìm ra các giải pháp tổng quát hóa theo cách mong muốn; tuy nhiên, thiên vị quy nạp không phù hợp cũng có thể dẫn đến hiệu năng dưới mức tối ưu bằng cách đưa ra các ràng buộc quá mạnh [73]. Vì vậy, thiên vị quy nạp thường được áp dụng vào các thuật toán học sâu để tăng cường tính khái quát hóa, tuy nhiên, trong một số trường hợp, giảm bớt thiên vị quy nạp sẽ giúp thuật toán học sâu tăng được độ hiệu năng.

Trong mô hình nhúng hình ảnh ViT, do chỉ có các lớp MLP là cục bộ và có tính chất bất biến, trong khi đó các lớp tự chú ý (self-attention) mang tính chất toàn

cục cho nên cấu trúc lân cận hai chiều được sử dụng rất hạn chế. Vì vậy, ViT ít thiên vị quy nạp ảnh hơn hơn so với các mô hình CNN trước đó.

B. Kiến trúc lai

Như một phương án thay thế cho các ô ảnh thô, chuỗi đầu vào có thể được tạo thành từ các bản đồ đặc trưng của một mạng nơ-ron tích chập CNN. Trong mô hình lai này, phép chiếu nhúng phân mảnh **E** (Công thức (3-11)) được áp dụng cho các phân mảnh mà trích rút từ một bản đồ đặc trưng của mạng CNN. Trong một số trường hợp đặc biệt cụ thể, các ô phân mảnh nhỏ có thể có kích thước 1×1, điều này có nghĩa là chuỗi đầu vào được tạo ra đơn giản bằng cách làm phẳng các chiều không gian của bản đồ đặc trưng và chiếu chúng sang không gian chiều Transformer. Khi đó, các nhúng đầu vào phân loại và các nhúng vị trí được cộng thêm vào.

Trong mô hình đề xuất, đối với nhúng ảnh luận án sử dụng mô hình Vision Transformer (ViT). Khi đó, đối với mỗi thực thể $e \in \mathcal{E}$ có ảnh v, luận án sẽ trích rút đặc trưng ảnh $\mathbf{f}_v$. Do đó, Công thức (3-1) được thay đổi như sau.

$$\mathbf{f}_v = \frac{1}{|\mathcal{V}_e|} \sum_{img_i \in \mathcal{V}_e} ViT(img_i) \qquad (3\text{-}15)$$

3.2.3 Nhúng văn bản bằng T5

Trong kỹ thuật học chuyển giao, một mô hình được tiền huấn luyện (pre-trained) trên tác vụ trước với nhiều dữ liệu. Sau đó, mô hình được tinh chỉnh cho một tác vụ tiếp theo. Đây là một kỹ thuật mạnh mẽ trong xử lý ngôn ngữ tự nhiên. Tiếp nối quá trình phát triển này, C. Raffel và cộng sự [71] đã xây dựng kỹ thuật học chuyển giao trong xử lý ngôn ngữ tự nhiên bằng cách giới thiệu một khung thống nhất, cho phép chuyển đổi tất cả các vấn đề ngôn ngữ dựa trên văn bản thành định dạng văn bản sang văn bản (text-to-text). Hơn nữa, kỹ thuật học chuyển giao cũng đã được áp dụng trong các mô hình hoàn thiện ĐTTT, được trình bày ở Chương 2 trong luận án này. Do vậy, trong phần nhúng phương thức văn bản cho mô hình đề xuất, luận án sử dụng mô hình T5 (Text-to-Text Transfer Transformer).

Theo C. Raffel và cộng sự [71], ý tưởng cơ bản trong mô hình T5 là giải quyết các vấn đề xử lý văn bản như một bài toán "text-to-text", tức là đầu vào nhận một văn bản và tạo ra một văn bản mới tại đầu ra. Phương pháp tiếp cận này được các tác giả lấy cảm hứng từ các khung thống nhất trước đây dành cho tác vụ xử lý văn bản như:

chuyển đổi tất cả các vấn đề văn bản thành bài toán trả lời câu hỏi, mô hình hóa ngôn ngữ, hoặc trích xuất văn bản. Một ưu điểm nổi bật của nền tảng văn bản sang văn bản được các tác giả chỉ ra đó là cho phép áp dụng trực tiếp cùng một mô hình, mục tiêu, quy trình huấn luyện và phương pháp giải mã cho tất cả các tác vụ.

Ban đầu, kỹ thuật học chuyển giao cho bài toán xử lý ngôn ngữ tự nhiên thường sử dụng mạng nơ-ron hồi quy. Tuy nhiên, thời gian gần đây các mô hình được xây dựng bằng cách sử dụng kiến trúc Transformer [72]. Khối xây dựng chính của kiến trúc Transformer là cơ chế tự chú ý (self-attention). Cơ chế tự chú ý là một biến thể của chú ý (attention), xử lý một chuỗi bằng cách thay thế từng phần tử bằng một trung bình có trọng số của các phần tử còn lại trong chuỗi. Mô hình gốc của Transformer bao gồm kiến trúc bộ mã hóa, bộ giải mã và hướng đến việc xử lý chuỗi sang chuỗi. C. Raffel và cộng sự [71] nhận xét rằng, gần đây các mô hình được xây dựng dựa trên một chồng lớp Transformer đơn lẻ đã trở nên phổ biến với các dạng tự chú ý khác nhau được sử dụng để tạo ra kiến trúc phù hợp với các tác vụ khác nhau như: mô hình ngôn ngữ, phân loại hoặc dự đoán phạm vi.

Theo đó, C. Raffel và cộng sự đã triển khai mô hình dựa trên kiến trúc Transformer gốc, bao gồm: bộ giải mã và bộ mã hóa. Đầu tiên, một chuỗi đầu vào gồm các token được ánh xạ thành một chuỗi các nhúng, sau đó truyền vào bộ mã hóa. Bộ mã hóa bao gồm một chồng các khối, mỗi khối gồm hai thành phần con: một lớp tự chú ý và một mạng truyền thẳng (feed forward network). Đối với mỗi thành phần con, các tác giả sử dụng chuẩn hóa tầng (layer normalization - được J. L. Ba và cộng sự giới thiệu [74]). Các tác giả [71] chỉ sử dụng một phiên bản đơn giản của chuẩn hóa tầng, trong đó các kích hoạt chỉ được tái chuẩn hóa mà không áp dụng thêm độ lệch cộng. Sau khi áp dụng chuẩn hóa tầng, một kết nối bỏ qua dư lượng (residual skip connection [75]) được thêm vào mỗi đầu vào và mỗi đầu ra của từng thành phần con. Dropout [76] được các tác giả sử dụng trong mạng truyền thẳng. Bộ giải mã có cấu trúc tương tự như bộ mã hóa, ở đó bao gồm một cơ chế chú ý tiêu chuẩn sau mỗi lớp tự chú ý. Cơ chế này tập trung vào đầu ra của bộ giải mã. Cơ chế tự chú ý trong bộ giải mã cũng sử dụng một dạng tự chú ý hồi quy (autoregressive) hoặc tự chú ý nhân quả (causal self-attention), trong đó mô hình chỉ được phép tập trung vào đầu ra trước đó. Đầu ra của khối giải mã cuối cùng được đưa vào một tầng dày đặc với đầu ra là hàm softmax, trong đó các trọng số được chia sẻ với ma trận nhúng đầu vào.

Tất cả cơ chế chú ý trong Transformer được chia thành các đầu độc lập, các đầu ra từ những đầu này được ghép nối trước khi chúng tiếp tục được xử lý thêm.

Do cơ chế tự chú ý không phụ thuộc vào thứ tự (nó là một toán tử trên các tập hợp), nên việc cung cấp tín hiệu vị trí rõ ràng cho Transformer là điều thường thấy. Trong khi các Transformer gốc sử dụng một tín hiệu vị trí dạng sóng sin hoặc các nhúng vị trí được học, thì các nghiên cứu gần đây sử dụng nhúng vị trí tương đối phổ biến hơn. Thay vì sử dụng một nhúng cố định cho mỗi vị trí, các nhúng vị trí tương đối sinh ra một nhúng được học khác nhau dựa vào khoảng cách giữa "khóa" và "truy vấn" trong cơ chế tự chú ý. C. Raffel và cộng sự [71] sử dụng một dạng đơn giản hóa của các nhúng vị trị, trong đó mỗi nhúng là một số vô hướng và được thêm vào logit tương ứng sử dụng cho tính toán các trọng số chú ý. Để tăng hiệu quả, các tác giả chia sẻ các tham số nhúng vị trí giữa tất cả các lớp trong mô hình, mặc dù trong một lớp cụ thể, mỗi đầu chú ý sử dụng một nhúng vị trí được học khác nhau. Thông thường, một số lượng cố định các nhúng được học, mỗi nhúng tương ứng với một khoảng cách "khóa-truy vấn" có thể xảy ra. Các tác giả cũng đã sử dụng 32 nhúng cho tất cả mô hình với các khoảng cách cho phép tăng kích thước theo hàm lôgarit cho đến 128, vượt quá mức này thì tất cả các vị trí tương đối đều được gán vào cùng một nhúng. Lưu ý rằng một lớp cụ thể không nhạy cảm với vị trí tương đối vượt quá 128 mã, nhưng các lớp sau đó có thể phát triển độ nhạy với các khoảng cách lớn hơn bằng cách kết hợp thông tin cục bộ từ các lớp trước đó. Các tác giả cũng nhấn mạnh rằng mô hình Transformer này tương đương với mô hình Transformer gốc [72] ngoại trừ việc loại bỏ độ lệch trong chuẩn hóa lớp, đặt chuẩn hóa lớp ra bên ngoài đường dẫn và sử dụng một lượt đồ nhúng vị trí khác.

Khi đó đối với mô hình T5-Vit-AdamMF-MAT, thành phần đặc trưng văn bản cấp độ câu $\mathbf{f}_t$ trong Công thức (3-2) thu được bằng công thức sau.

$$\mathbf{f}_t = T5(text) \qquad (3\text{-}16)$$

Sau đó, đặc trưng văn bản $\mathbf{f}_t$ được chiếu xuống không gian nhúng với các tham số lớp $\mathbf{W}_t$, $\mathbf{b}_t$ để thu được nhúng $\mathbf{e}_t$ (chi tiết xem tại Công thức (3-2)).

3.3 Thực nghiệm mô hình đề xuất và đánh giá

Trong phần này, luận án đánh giá hiệu năng của các mô hình đề xuất thông qua tác vụ dự đoán liên kết.

3.3.1 Cấu hình phần cứng thực nghiệm

Luận án thực nghiệm mô hình đề xuất với thông số được cho bởi Bảng 3.1.

Bảng 3.1. Thông số thực nghiệm mô hình

	Kaggle
Nguồn	https://www.kaggle.com
GPU	Nvidia P100
Tốc độ xử lý GPU	1,32GHz
Dung lượng GPU	16 GB
Dung lượng RAM	12 GB
Dung lượng bộ nhớ	5 GB

3.3.2 Tập dữ liệu

Giống các thực nghiệm được thực hiện như trong mô hình gốc của Y. Zhang và cộng sự [18], luận án cũng đánh giá hiệu năng của mô hình đề xuất thông qua các tập dữ liệu đa phương thức như DB15K, MKG-W và MKG-Y.

- Tập dữ liệu DB15K là một bộ tri thức đa phương thức được xây dựng từ tập dữ liệu DBpedia [81], với mục tiêu hỗ trợ các tác vụ như hoàn thiện ĐTTT đa phương thức. Mỗi thực thể trong DB15K được gắn với ít nhất một ảnh đại diện. Các ảnh được thu thập từ công cụ tìm kiếm ảnh, đảm bảo rằng mỗi thực thể có ít nhất một ảnh tương ứng. Các mối quan hệ được giữ nguyên theo cấu trúc RDF (Resource Description Framework) trong Dbpedia. Luôn đảm bảo tính liê kết giữa thông tin có cấu trúc (structural triples) và thông tin thị giác (visua embeddings).
- Hai tập dữ liệu MKG-W (Multimodal KG-Wikipedia) và MKG-Y (Multimodal KG-YAGO) [50], đây là các tập con của tập dữ liệu Wikidata [82] và tập dữ liệu YAGO [83], tương ứng. Tập dữ liệu MKG-W gắn thông tin thị giác (ảnh) và mô tả ngôn ngữ tự nhiên cho thực thể trong Wikipedia. Còn tập dữ liệu MKG-Y gắn hình ảnh và văn bản cho các thực thể thuộc YAGO. Mỗi thực thể trong cả hai tập dữ liệu có ít nhất một ảnh đại diện, được thu tập từ Wikimedia Commons hoặc qua tìm kiếm có kiểm duyệt.

- Tập dữ liệu MKG-W: dữ liệu ngôn ngữ (text description) thường mô tả ngắn về thực thể. Hình ảnh được ánh xạ với mã thực thể tương ứng. Đây là tập dữ liệu đa lĩnh vực. Được khai thác từ tập con của Wikidata nên phổ rộng hơn. Có độ phức tạp cấu trúc cao. Có ưu điểm là đa dạng thực thể và quan hệ, phản ảnh tri thức phong phúc, phù hợp với các mô hình phức tạp. Thường cần yêu cầu mô hình xử lý mạnh hơn để học hiệu quả.
- Tập dữ liệu MKG-Y: Các bộ ba trong đây được chọn lọc từ tập dữ liệu YAGO có tính ngữ nghĩa cao. Các quan hệ trong YAGO chủ yếu về con người, địa danh, sự kiện…. Trong đó, mỗi thực thể có một đoạn mô tả ngắn, thường trích từ Wikipedia hoặc YAGO. Đơn giản và hướng miền cụ thể hơn. Ít quan hệ, dễ huấn luyện, gọn nhẹ, thực thể dễ hiểu.

Các tập dữ liệu trên được chia ngẫu nhiên theo tỷ lệ 8: 1: 1 một cách tương ứng cho các tập huấn luyện, tập xác thực và tập kiểm thử. Thông tin chi tiết được cho tại Bảng 3.2.

Bảng 3.2. Thông tin các tập dữ liệu

Tập dữ liệu	*Số lượng thực thể*	*Số lượng ảnh*	*Số lượng văn bản*	*Số lượng quan hệ*	*Tập huấn luyện*	*Tập xác thực*	*Tập kiểm thử*
DB15K	12842	12818	9078	279	79222	9902	9904
MKG-W	15000	14463	14123	169	34196	4276	4274
MKG-Y	15000	14244	12305	28	21310	2665	2663

3.3.3 Phần mềm thực nghiệm và kịch bản thực nghiệm

Dựa trên phần mềm mô hình AdaMF-MAT [18], luận án xây dựng phần mềm thực nghiệm cho các mô hình đề xuất ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT trên nền tảng các phần mềm và công cụ sau: Kaggle (https://www.kaggle.com/), Pytorch, Thư viện Transformer (HuggingFace), Mô hình BERT, Mô hình T5 và Mô hình Vision Transformer (ViT).

Bảng 3.3. Thông số cài đặt cho từng tập dữ liệu.

Tập dữ liệu / Thông số	DB15K	MKG-W	MKG-Y
EMB_DIM	250	200	200
NUM_BATCH	1024	1024	1024
MARGIN	12	12	4
LR	$2e-5$	$2e-5$	$2e-5$
NEG_NUM	128	128	128
EPOCHS	1000	1000	1000

Luận án tiến hành thực nghiệm theo các kịch bản sau đây.

- Luận án tiến hành hai kịch bản cài đặt nhúng văn bản:
  - Nhúng văn bản bằng BERT (như mô hình gốc AdaMF-MAT) cho kết quả đầu ra là một ten-xơ có chiều là (số thực thể, 768) với dữ liệu đầu vào là một mảng các chuỗi văn bản có nội dung là tên các thực thể được ánh xạ tương ứng với chỉ số của chúng trong tập dữ liệu.
  - Nhúng văn bản bằng T5-small [71] với đầu vào là kết quả từ tokenizer. Cụ thể, luận án dùng attention mask để chỉ định các vị trí của token thực (tức là các token không phải padding). Tiếp theo, luận án lấy đầu ra từ các bộ mã hóa của mô hình T5, rồi sử dụng attention mask để thiết lập trọng số cho các nhúng, cụ thể: 0 đối với padding và 1 đối với các token còn lại. Cuối cùng, luận án tính giá trị trung bình có trọng số đối với các token không phải padding. Từ đó, thu được nhúng văn bản của các thực thể.
- Đối với mô hình nhúng ảnh, luận án sử dụng mô hình ViT (Vision Transformer [50]) với các thông số mặc định. Dữ liệu đầu vào của mô hình này là những tập ảnh khác nhau, trong đó mỗi tập ảnh sẽ tương ứng với một thực thể. Từ mỗi tập ảnh, mô hình nhúng ảnh sinh ra một ten-xơ trung bình của các ảnh trong tập đó (Lưu ý, nếu thực thể không có ảnh thì ngầm định một tensor gồm các giá trị là 1). Khi đó, đầu ra của mô hình nhúng ảnh là các tensor với số chiều là (số thực thể, 768).

Trong quá trình sinh nhúng, cần lưu ý sao cho các véc-tơ nhúng của thực thể có cùng thứ tự với nhau, nghĩa là véc-tơ nhúng văn bản của thực thể phải cùng vị trí với véc-tơ nhúng ảnh của thực thể đó.

Thông số cài đặt chi tiết theo từng tập dữ liệu được cho tại Bảng 3.3.

3.3.4 Độ đo hiệu năng

Tương tự như các mô hình thực nghiệm trên các tập dữ liệu khác nhau trong các chương trước và kết quả được đề xuất bởi Y. Zhang và cộng sự [18], luận án cũng đánh giá các mô hình đề xuất trong chương này trên các độ đo là: MRR và $Hits@n$ (với $n = 1; 3; 10$).

3.3.5 Kết quả thực nghiệm

Kết quả thực nghiệm được đưa ra tại Bảng 3.3, đối với các độ đo khi giá trị càng cao thì kết quả càng tốt.

Bảng 3.3. Kết quả thực nghiệm các mô hình. Giá trị in đậm là tốt nhất, giá trị gạch chân là tốt thứ hai theo từng cột.

Mô hình hoàn thiện ĐTTT	**Tập dữ liệu DB15K**				**Tập dữ liệu MKG-W**				**Tập dữ liệu MKG-Y**			
	MRR	*Hit@1*	*Hit@3*	*Hit@10*	*MRR*	*Hit@1*	*Hit@3*	*Hit@10*	*MRR*	*Hit@1*	*Hit@3*	*Hit@10*
AdaMF-MAT [18]	35,14	25,3	41,11	52,92	<u>35,85</u>	29,04	**39,01**	**48,42**	38,57	34,34	40,59	45,76
ViT-AdaMF-MAT	**36,04**	**26,39**	**41,88**	**53,53**	**36,01**	**29,75**	<u>38,69</u>	<u>47,59</u>	<u>38,63</u>	<u>34,83</u>	<u>40,63</u>	<u>45,29</u>
T5-ViT-AdaMF-MAT	<u>35,85</u>	<u>26,25</u>	<u>41,63</u>	<u>53,03</u>	35,33	28,88	38,08	47,00	**39,34**	**35,09**	**41,07**	**46,56**

3.3.6 Bàn luận

Nhìn vào bảng kết quả, luận án nhận thấy hầu hết kết quả của các mô hình đề xuất đều cho kết quả tốt hơn mô hình gốc AdaMF-MAT [18]. Cụ thể:

- Với tập dữ liệu DB15K, hiệu năng của hai mô hình đề xuất đều tốt hơn hiệu năng của mô hình gốc trên cả bốn chỉ số. Trong đó, mô hình ViT-AdaMF-MAT có kết quả tốt nhất trên cả bốn chỉ số so với mô hình đề xuất T5-ViT-AdaMF-MAT và mô hình gốc AdaMF-MAT.
  - Đối với chỉ số MRR, tỷ lệ tăng tối đa là $2{,}5\%$.
  - Đối với chỉ số $Hit@1$, tỷ lệ tăng tối đa là $4{,}3\%$.
  - Đối với chỉ số $Hit@3$, tỷ lệ tăng tối đa là $1{,}9\%$.
  - Đối với chỉ số $Hit@10$, tỷ lệ tăng tối đa là $1{,}2\%$.

Điều này cho thấy mô hình dự đoán chính xác hơn trên bộ dữ liệu DBK15, đặc biệt là với chỉ số nghiêm ngặt như Hit@1 (chỉ quan tâm kết quả đứng đầu), chứng tỏ mô hình tạo ra kết quả chính xác hơn trong việc dự đoán liên kết.

- Với tập dữ liệu MKG-W
  - Hiệu năng của mô hình đề xuất T5-ViT-AdaMF-MAT đều không tốt hơn hiệu năng của mô hình gốc AdaMF-MAT trên cả bốn chỉ số.
  - Hiệu năng của mô hình đề xuất ViT-AdaMF-MAT kém hơn hiệu năng của mô hình gốc trên các bộ chỉ số $Hit@3$ và $Hit@10$. Đối với các bộ chỉ số MRR và $Hit@1$ thì hiệu năng của mô hình đề xuất ViT-AdaMF-MAT có cải thiện so với hiệu năng của mô hình gốc. Cụ thể:
    - Đối với chỉ số MRR, tỷ lệ tăng là $0{,}4\%$;
    - Đối với chỉ số $Hit@1$, tỷ lệ tăng là $2{,}4\%$.
- Với tập dữ liệu MKG-Y, hiệu năng của hai mô hình đề xuất đều tốt hơn hiệu năng của mô gốc trên cả bốn chỉ số. Tuy nhiên, trong trường hợp này hiệu năng của mô hình T5-ViT-AdaMF-MAT tốt nhất khi so sánh trên cả bốn chỉ số. Cụ thể:
  - Đối với chỉ số MRR, tỷ lệ tăng tối đa là 2%.
  - Đối với chỉ số $Hit@1$, tỷ lệ tăng tối đa là $2{,}2\%$.

- Đối với chỉ số $Hit@3$, tỷ lệ tăng tối đa là 1,18%.
- Đối với chỉ số $Hit@10$, tỷ lệ tăng tối đa là 1,74%.

Bên cạnh đó, sau khi xem xét hai tập dữ liệu MKG-W và MKG-Y, luận án nhận thấy rằng mức độ chi tiết đối với phần mô tả của các thực thể trong tập dữ liệu MGK-W rất hạn chế. Do vậy, mô hình T5 không phát huy được hiệu quả và thế mạnh của nó trên các văn bản dài. Ngoài ra, tỷ lệ ảnh trên các thực thể trong tập dữ liệu MKG-W ít hơn là tỷ lệ ảnh trên các thực thể trong tập dữ liệu MKG-Y. Từ đó dẫn đến hiệu năng trên các độ đo của hai mô hình đề xuất trên tập dữ liệu MKG-W không được tốt bằng hiệu năng của trên các độ đo của mô hình gốc.

Tiếp theo, luận án so sánh hiệu năng "tốt nhất" (bộ dữ liệu DB15K) và hiệu năng "tồi nhất" (bộ dữ liệu MKG-W). Theo Bảng 3.4, số lượng thực thể của bộ DB15K (với số lượng là 12842) ít hơn số lượng thực thể của bộ MKG-W (với số lượng là 15000). Tuy nhiên, số lượng thực thể không có ảnh của bộ DB15K chỉ khoảng 800 thực thể. Trong khi đó, số lượng thực thể không có ảnh của bộ MKG-W lại khá cao, khoảng 6000 thực thể. Điều đó dẫn đến tỷ lệ không có ảnh/số lượng thực thể của bộ MKG-W cao hơn hẳn tỷ lệ không có ảnh/số lượng thực thể của bộ DB15K. Cụ thể tỷ lệ như trong Bảng 3.4 dưới đây.

Bảng 3.4. So sánh giữa bộ dữ liệu DB15K và MKG-W

	DB15K	**MKG-W**
Số lượng thực thể	12842	15000
Số lượng thực thể không có ảnh	800	6000
Số lượng thực thể không có ảnh/tổng số lượng thực thể	6,23%	40%

Một yếu tố khác, tỷ lệ cân bằng giữa các thực thể là người và các thực thể không phải con người trong bộ dữ liệu DB-15K tốt hơn trong bộ dữ liệu MKG-W. Tuy nhiên, khi nhìn vào các loại liên kết có trong các bộ dữ liệu, luận án nhận thấy những quan hệ này chủ yếu liên quan đến thực thể có tính chất con người.

3.4 Kết luận Chương 3

Chương này tập trung nghiên cứu về mô hình hoàn thiện ĐTTT đa phương thức: (i) Khảo sát một số mô hình hoàn thiện ĐTTT đa phương thức, (ii) Phân tích chi tiết mô hình AdaMF-MAT [18]. (iii Đề xuất hai mô hình hoàn thiện ĐTTT đa phương thức ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT được cải tiến từ mô hình AdaMF-MAT. Kết quả nghiên cứu trong chương này được công bố tại [AnhNTTT4].

Chương 4 của luận án sẽ trình bày đề xuất của luận án về mô hình hoàn thiện ĐTTT thời gian.

.

Chương 4. Hoàn thiện ĐTTT thời gian dựa trên nhúng lịch đại

Chương này trình bày đề xuất của luận án về hai mô hình hoàn thiện ĐTTT thời gian là DE-RotatE và DE-RotatE-sinc được cải tiến từ mô hình DE-SimplE do R. Goel và cộng sự đề xuất năm 2020 [19]. Mục đầu tiên giới thiệu mô hình nhúng lịch đại ĐTTT DE. Luận án tập trung trình bày quá trình học trong mô hình hoàn thiện ĐTTT thời gian DE-SimplE [19] trong mục thứ hai, đồng thời, đưa ra một số nhận xét và ý tưởng cải tiến DE-SimplE. Mục thứ ba trình bày đề xuất một số hướng tiếp cận để cải tiến hiệu năng của mô hình hoàn thiện ĐTTT thời gian DE-SimplE. Thực nghiệm và đánh giá hiệu năng của hai mô hình đề xuất trong mục thứ tư.

4.1 Nhúng lịch đại ĐTTT

Trong phần này, luận án trình bày về các khái niệm của nhúng lịch đại trong hoàn thiện ĐTTT thời gian. Gọi $\mathcal{E}$ là tập hữu hạn các thực thể, $\mathcal{R}$ là tập hữu hạn các kiểu quan hệ giữa các thực thể, và $\mathcal{T}$ là tập hữu hạn các dấu thời gian (timestamps). Đặt $\mathcal{W} \subset \mathcal{E} \times R \times \mathcal{E} \times \mathcal{T}$ là tập tất cả các dữ kiện trong thế giới thực, mô tả dưới dạng các bộ bốn $s = (h, r, t, \tau)$, trong đó h, t là các thực thể đầu và thực thể cuối, r là quan hệ giữa các thực thể, τ là thời gian. Đặt $\mathcal{W}^c$ là phần bù của $\mathcal{W}$, nghĩa là nếu $s \in \mathcal{W}$ thì $s \notin \mathcal{W}^c$. Một ĐTTT thời gian $\mathcal{G}$ là một tập con của $\mathcal{W}$. Yếu tố dấu thời gian trong ĐTTT thời gian có thể được xác định là một thời điểm (ví dụ 11:30 ngày 30/04/1975) hoặc một khoảng thời gian (ví dụ 7/1954 – 5/1975). Trong bài báo [19], R. Goel và cộng sự tập trung nghiên cứu các ĐTTT thời gian mà ở đó mỗi dữ kiện có một dấu thời gian duy nhất. Tuy nhiên, các tác giả cũng nhấn mạnh rằng cách tiếp cận này có thể mở rộng đối với những dữ kiện chứa các khoảng thời gian. Ví dụ một dữ kiện với dấu thời gian duy nhất có trong cơ sở dữ liệu là *("South Korea", "Sign formal agreement", "North Korea", "2014-04-09")*.

Bài toán hoàn thiện ĐTTT thời gian nhằm giải quyết vấn đề thiếu liên kết trong đồ thị thời gian, nghĩa là sử dụng đồ thị chưa đầy đủ $\mathcal{G}$ để suy ra bộ hoàn chỉnh $\mathcal{W}$.

4.1.1 Nhúng từ lịch đại

Các phương pháp nhúng từ truyền thống, như Word2Vec [77] hoặc GloVe [62], chỉ có thể tạo ra các biểu diễn tinh của từ dựa trên các thống kê cùng xuất hiện

bên trong một khung ngữ cảnh hoặc một đoạn văn bản cố định. Khi hiểu được cách các từ thay đổi như thế nào theo thời gian là chìa khóa cho các mô hình về ngôn ngữ hiểu về sự tiến hóa văn hóa các thời đại. Tuy nhiên, dữ liệu lịch sử về ý nghĩa thường khan hiếm, điều đó dẫn đến các lý thuyết khó phát triển và kiểm tra. Để giải quyết vấn đề này, nhúng từ lịch đại (Diachronic Word Embedding) được W. L. Hamilton và cộng sự giới thiệu lần đầu vào năm 2016 [78]. Nhúng từ lịch đại có thể được gọi là nhúng từ lịch sử hoặc nhúng từ thời gian. Đây là một kiểu của biểu diễn từ được sử dụng trong xử lý ngôn ngữ tự nhiên, dùng để nắm bắt những thay đổi ngữ nghĩa và ngữ cảnh của từ theo thời gian. Khác với nhúng từ truyền thống, nhúng từ lịch đại xem xét chiều và mô hình thời gian của từ để từ đó xem nó phát triển như thế nào trong các khoảng thời gian khác nhau. W. L. Hamilton và cộng sự đã xây dựng nhúng từ lịch đại bằng cách sau: đầu tiên xây dựng các nhúng trong mỗi khoảng thời gian và sắp xếp chúng theo chiều thời gian. Tiếp theo các độ đo được sử dụng để định lượng và đánh giá sự thay đổi về mặt ngữ nghĩa của từ qua các khoảng thời gian đó.

- Để xây dựng nhúng từ trong mỗi khoảng thời gian, các tác giả đã sử dụng ba phương pháp bao gồm: PPMI (positive point-wise mutual information), SVD (singular value decompostion) và SGNS (skip-gram with negative sampling). Các phương pháp này biểu diễn mỗi từ w_i bởi một véc-tơ $\mathbf{v}_i$ chứa đựng thông tin về số liệu thống kê đồng thời xuất hiện (co-occrurrence) của nó với các từ khác.
- Sắp xếp các kết quả nhúng theo thời gian: Để so sánh các véc-tơ từ $\mathbf{v}_i$ trong các khoảng thời gian khác nhau, các tác giả đã đưa các véc-tơ vào cùng hệ trục tọa độ. Đối với PPMI các véc-tơ tự căn chỉnh vì mỗi cột tương ứng với một từ ngữ cảnh. Tuy nhiên, các nhúng chiều thấp (low-dimensional) không tự căn chỉnh được do bản chất không duy nhất của phương pháp phân tích ma trận SVD và bản chất ngẫu nhiên của SGNS. Hơn nữa, cả hai phương pháp này đều có thể dẫn đến phép biến đổi trực giao tùy ý (orthogonal transformations). Mặc dù hai phương pháp này không ảnh hưởng đến độ tương đồng cosine từng cặp nhưng sẽ ngăn cản việc so sánh cùng từ theo thời gian. W. L. Hamilton và cộng sự [78] đề xuất sử dụng phương pháp trực giao Procrustes để căn chỉnh các nhúng chiều thấp đã được học. Các tác giả định nghĩa một ma trận các nhúng từ $\mathbf{W}^{(t)}$ được học tại thời điểm thứ t để sắp xếp theo các khoảng thời gian trong khi vẫn bảo toàn sự tương đồng cosine.

Mục tiêu đằng sau của nhúng từ lịch đại là nắm bắt ngữ nghĩa của ngôn ngữ tự nhiên một cách linh động, hàm ý rằng ý nghĩa của từ có thể thay đổi theo các khoảng thời gian trong khung cảnh văn hóa, xã hội, kỹ thuật, kinh tế, khoa học, kỹ thuật … Ví dụ, những từ như "tweet" và "cloud" có một ý nghĩa khác trong những năm gần đây do sự bùng nổ của mạng xã hội và điện toán đám mây. Nhúng từ lịch đại có rất nhiều ứng dụng trong các lĩnh vực khác nhau bao gồm ngôn ngữ học, lịch sử, nghiên cứu văn hóa, khoa học xã hội…

4.1.2 Nhúng lịch đại ĐTTT

Tương tự như trong hoàn thiện ĐTTT, mô hình nhúng đóng quan trọng và mang lại nhiều lợi thế trong hoàn thiện ĐTTT thời gian. Trước khi trình bày định nghĩa và chi tiết các thành phần trong nhúng lịch đại DE (Diachronic Embedding), luận án nhắc lại hai ký hiệu về thành phần nhúng đã được trình bày tại Chương 1.

- Nhúng thực thể ký hiệu là $EEMB$.
- Nhúng quan hệ ký hiệu là $REMB$.

Dựa trên ý tưởng của nhúng từ lịch đại (W. L. Hamilton và cộng sự [78]) và nhúng thực thể trong ĐTTT, R. Goel và cộng sự [19] giới thiệu mô hình nhúng lịch đại DE – một thành phần quan trọng trong mô hình hoàn thiện ĐTTT thời gian. Nhúng lịch đại DE là một ánh xạ với đầu vào là một cặp đôi bao gồm thực thể và dấu thời gian trong bộ bốn. Đầu ra của ánh xạ này là một biểu diễn ẩn lấy các đặc trưng của thực thể và dấu thời gian. Chi tiết về nhúng lịch đại DE được định nghĩa dưới đây.

Định nghĩa 4.1 Nhúng lịch đại [19].

> *Một nhúng lịch đại DE, ký hiệu là* $DEEMB : (\mathcal{E}, \mathcal{T}) \rightarrow \psi$*, là ánh xạ gán mỗi bộ đôi* (e, τ)*, trong đó* $e \in \mathcal{E}$ *và* $\tau \in \mathcal{T}$*, thành một biểu diễn ẩn trong* ψ *với* ψ *là một lớp không rỗng, chứ bộ các véc-tơ hoặc/và ma trận.*

Trước khi trình bày chi tiết về thành phần để xây dựng ánh xạ $DEEMB$, luận án đưa ra một số ký hiệu. Ký tự in thường được biểu diễn giá trị vô hướng, ký tự in đậm được biểu diễn cho véc-tơ, ký tự viết hoa in đậm biểu diễn cho ma trận. $\boldsymbol{z}[n]$ biểu diễn phần tử thứ n của véc-tơ $\boldsymbol{z}$. $\|\boldsymbol{z}\|$ biểu diễn chuẩn vủa véc-tơ $\boldsymbol{z}$ và $\boldsymbol{z}^{\mathrm{T}}$ biểu diễn véc-tơ chuyển vị. $\boldsymbol{z}_1 \otimes \boldsymbol{z}_2$ biểu diễn một véc-tơ $\boldsymbol{z} \in \mathbb{R}^{d_1 d_2}$ sao cho $\boldsymbol{z}[(n-1) * d_2 + m] = \boldsymbol{z}_1[n]\boldsymbol{z}_2[m]$ (véc-tơ phẳng của tích ten-xơ/ngoài của hai véc-tơ). Cho k véc-tơ $\boldsymbol{z}_1, \boldsymbol{z}_2, \dots, \boldsymbol{z}_k$ có cùng kích thước d, $\langle \boldsymbol{z}_1, \boldsymbol{z}_2, \dots, \boldsymbol{z}_k \rangle =$

$\sum_{n=1}^{d}(\boldsymbol{z}_1[n] * \boldsymbol{z}_2[n] *, \dots, * \boldsymbol{z}_k[n])$ biểu diễn tích vô hướng của k véc-tơ. Ký hiệu hàm tính điểm $\phi(h, r, t)$ đối với một bộ ba (h, r, t).

I. Ánh xạ nhúng lịch đại DE

Ánh xạ $DEEMB$ có thể xây dựng theo nhiều cách khác nhau phụ thuộc vào đặc trưng của các loại ĐTTT thời gian. Trong [19], R. Goel và cộng sự đề xuất một ánh xạ $DEEMB$ phù hợp đối với tập dữ liệu tri thức thời gian. Gọi $e \in \mathcal{E}$ là một thực thể (có thể là thực thể đầu hoặc thự thể cuối), $\boldsymbol{z}_e^{\tau} \in \mathbb{R}^{\boldsymbol{d}}$ là một véc-tơ thu được từ ánh xạ $DEEMB(e, \tau)$, nghĩa là $DEEMB(e, \tau) = (\dots, \boldsymbol{z}_e^{\tau}, \dots)$. Khi đó, $\boldsymbol{z}_e^{\tau}$ được xác định như công thức dưới đây.

$$\boldsymbol{z}_e^{\tau}[n] = \begin{cases} \boldsymbol{a}_e[n]\sigma(\boldsymbol{w}_e[n]\tau + \boldsymbol{b}_e[n]), & nếu\ 1 \le n \le \gamma d \\ \boldsymbol{a}_e[n], & nếu\ \gamma d < n \le d \end{cases} \qquad (4\text{-}1)$$

Trong Công thức (4-1), $\boldsymbol{a}_e \in \mathbb{R}^{\boldsymbol{d}}$ và $\boldsymbol{w}_e, \boldsymbol{b}_e \in \mathbb{R}^{\boldsymbol{d}}$ là các véc-tơ (cụ thể là thực thể) với các tham số được học và σ là hàm kích hoạt. Từ Công thức (4-1), nhận thấy rằng các thực thể có thể có vài đặc trưng thay đổi theo thời gian và một vài đặc trưng không thay đổi theo thời gian. Cụ thể, γd phần tử đầu tiên của véc-tơ là đặc trưng thời gian và $(1-\gamma)d$ phần tử còn lại là đặc trưng tĩnh. $0 \le \gamma \le 1$ là một siêu tham số điều khiển số lượng phần trăm đặc trưng thời gian, cụ thể phần nhúng động của thực thể chiếm tỷ lệ là γ và nhúng tĩnh của thực thể chiếm tỷ lệ là $1-\gamma$.

Công thức (4-1) được chia làm hai thành phần riêng biệt: thành phần 1 là nhúng tĩnh (không phụ thuộc thời gian) và thành phần 2 là nhúng động.

- Thành phần nhúng tĩnh ($\boldsymbol{a}_e[n]$): mỗi thực thể trong ĐTTT được ánh xạ với một véc-tơ nhúng. Véc-tơ này được học trong quá tình huấn luyện.
- Thành phần nhúng động $\boldsymbol{a}_e[n]\sigma(\boldsymbol{w}_e[n]\tau + \boldsymbol{b}_e[n])$: véc-tơ nhúng sẽ được biến đổi tỷ lệ với thời gian theo hàm kích hoạt σ.

Một quan sát đặc biệt cũng được R. Goel và cộng sự [19] nêu ra rằng, các đặc trưng tĩnh có thể thu được từ các đặc trưng thời gian khi bộ tối ưu hóa thiết lập một vài phần tử của $\boldsymbol{w}_e$ bằng 0. Khi mô hình hóa các đặc trưng tĩnh giúp giảm số lượng tham số học và ngăn ngừa hiện tượng quá khớp đối với tín hiệu thời gian.

Bằng cách học các tham số $\boldsymbol{w}_e$ và $\boldsymbol{b}_e$, mô hình sẽ học các đặc trưng thực thể bật và tắt tại các thời điểm khác nhau. Do vậy các dự đoán thời gian chính xác có thể được thực hiện tại bất kỳ thời điểm nào. Các tham số $\boldsymbol{a}_e$ điều khiển mức độ quan

trọng của các đặc trưng. Chính vì vậy, R. Goel và cộng sự đề xuất sử dụng hàm lượng giác sin là hàm kích hoạt cho Công thức (4-1). Qua nghiên cứu, luận án nhận thấy rằng, hàm tuần hoàn sin cho phép mô hình hình thành trạng thái tắt và bật. Khi đó, phần nhúng động của Công thức (4-1) là $f(\tau) = \boldsymbol{a}_e[n]\sin(\boldsymbol{w}_e[n]\tau + \boldsymbol{b}_e[n])$. Hàm số $f(\tau)$ là hàm đặc trưng phụ thuộc vào thời gian τ. Với $\boldsymbol{a}_e$ đưa ra giá trị tối đa của đặc trưng (giá trị càng cao thì đặc trưng càng quan trọng); $\boldsymbol{w}_e$ và $\boldsymbol{b}_e$ sẽ quyết định một đặc trưng sẽ thay đổi theo thời gian τ như thế nào. Do một số dữ kiện có chu kỳ và hàm kích hoạt sin có tính chất đặc biệt là tính lặp theo chu kỳ vô hạn của nó. Đó là lý do các tác giả đã đưa ra phương án lựa chọn này. Hơn nữa, bằng thực nghiệm các tác giả cũng chỉ ra rằng các hàm kích hoạt khác ($Tanh$, $Sigmoid$, $LeakyReLU$, $Squared\ Exponential$) có kết quả không tốt bằng hàm sin. Kết quả thực nghiệm so sánh được cho bởi Bảng 4.1.

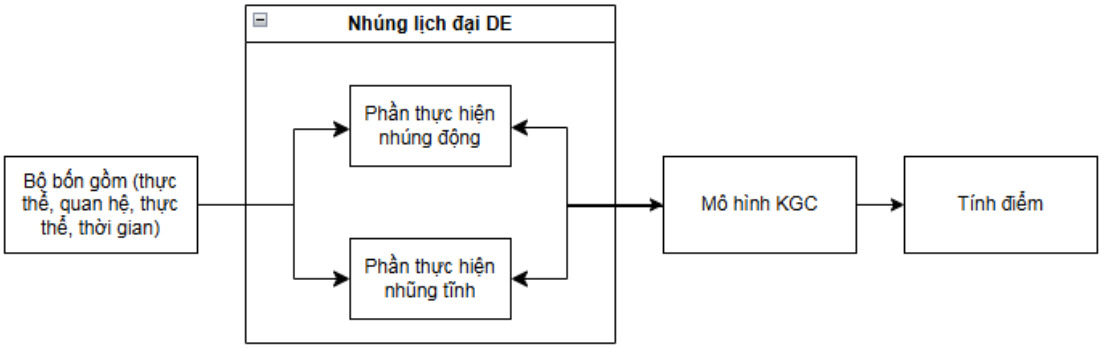
Bảng 4.1. Kết quả của mô hình hoàn thiện ĐTTT thời gian với nhúng lịch đại sử dụng các hàm kích hoạt khác nhau *[19].*

Hàm kích hoạt	$Hit@1$	$Hit@3$	$Hit@10$	MRR
Tanh	0,375	0,547	0,701	0,486
Sigmoid	0,370	0,546	0,706	0,484
Leaky ReLU	0,363	0,542	0,701	0,478
Squared Exponentail	0,390	0,568	**0,709**	**0,501**
Sin	**0,392**	**0,569**	0,708	**0,501**

4.2 Mô hình hoàn thiện ĐTTT thời gian nhúng lịch đại DE-KGC

Các mô hình nhúng (thực thể, quan hệ hoặc thời gian) trước đây thường chỉ cho phép mở rộng một hoặc một vài mô hình hoàn thiện ĐTTT sang mô hình hoàn thiện ĐTTT thời gian. Khác biệt so với các mô hình nhúng này, một trong những ưu điểm của nhúng lịch đại DE là cho phép kết hợp với rất nhiều mô hình hoàn thiện ĐTTT khác nhau như TuckER, TransE, DistMult, và SimplE bằng cách thay thế ánh xạ $EEMB$ của chúng bằng ánh xạ $DEEMB$. Nghĩa là dữ liệu đầu ra của ánh xạ $DEEMB$ có thể kết hợp đa dạng với bất kỳ một mô hình hoàn thiện ĐTTT tĩnh khác

để tạo thành một mô hình hoàn thiện ĐTTT thời gian. Để thuận tiện trong quá trình trình bày, luận án gọi mô hình hoàn thiện ĐTTT thời gian này là DE-KGC, một sự kết hợp giữa nhúng lịch đại DE và một mô hình hoàn thiện ĐTTT tĩnh bất kỳ. Cấu trúc của mô hình này được mô tả như Hình 4.1.



Hình 4.1. Mô hình DE-KGC [19].

R. Goel và cộng sự [19] chỉ ra rằng mô hình DE-SimplE là mô hình kết hợp giữa nhúng lịch đại DE và mô hình hoàn thiện ĐTTT SimplE, có kết quả tốt nhất về mặt lý thuyết cũng như thực nghiệm. Qua thực nghiệm và bằng các phép so sánh kết quả, các tác giả cũng đã chỉ ra rằng hiệu năng của mô hình DE-SimplE vượt trội so với các mô hình trước đó (TTransE [39], HyTE [79], TA-DistMult [40]). Hơn nữa, mô hình DE-SimplE cũng có kết quả vượt trội khi so sánh với các mô hình DE-DistMult (kết hợp DE và DistMult), DE-TransE (kết hợp DE và TransE).

Tính biểu cảm đầy đủ

Theo K. Xu và cộng sự [80], mặc dù tính biểu cảm đầy đủ là một tính chất rất quan trọng trong hoàn thiện các mô hình ĐTTT khác nhau. Tuy nhiên, tính chất biểu cảm đầy đủ này mới được chứng minh và xuất hiện trong một số mô hình hoàn thiện ĐTTT, như SimplE. Còn đối với ĐTTT thời gian, theo nghiên cứu thì hiện nay chưa có bất kỳ một mô hình nào được chứng minh có tính biểu cảm đầy đủ. Tuy nhiên, R. Goel và cộng sự chứng minh rằng mô hình DE-SimplE có tính biểu cảm đầy đủ.

Tính chất trên quan hệ miền

Ngoài việc, R. Goel và cộng sự chỉ ra tính biểu cảm đầy đủ trong mô hình DE-SimplE, các tác giả cũng quan tâm đến tri thức miền. Đối với các mô hình nhúng đồ thị tri thức tĩnh, một số nghiên cứu đã chỉ ra tính chất trên quan hệ của nó. Theo các

tác giả [19], trước đây chưa có một nghiên cứu đầy đủ nào trên các tính chất quan hệ đối với các mô hình hoàn thiện ĐTTT thời gian. Các tính chất trên quan hệ được tác giả chứng minh bằng kết quả dưới đây.

Mệnh đề 4.1 Các tính chất của DE-SimplE [19].

Mô hình DE-SimplE có tính chất đối xứng, bất đối xứng và nghịch đảo.

4.2.1 Quá trình học

Các dữ kiện trong ĐTTT $\mathcal{G}$ được phân chia vào bộ dữ liệu huấn luyện (training set), bộ dữ liệu đánh giá tinh chỉnh (validation set) và bộ dữ liệu kiểm thử (test set). Các tham số của mô hình được học bằng cách sử dụng kỹ thuật giảm độ dốc ngẫu nhiên SGD (Stochastic Gradient Descent) với các lô nhỏ (mini-batches). Gọi B là một tập con của tập huấn luyện (nghĩa là B có thể được hiểu là một lô nhỏ). Đối với một dữ kiện $f = (h, r, t, \tau) \in B$, R. Goel và cộng sự [19] tạo ra hai truy vấn: 1 - $(h, r, ?, \tau)$ và 2 - $(?, r, t, \tau)$. Đối với truy vấn đầu tiên, một tập câu trả lời tiềm năng được tạo ra là $C_{f,h}$ chứa h và n (được gọi là tỷ lệ âm) thực thể khác được lựa chọn ngẫu nhiên từ $\mathcal{E}$. Đối với truy vấn số hai, tương tự vậy một tập câu trả lời tiềm năng cũng được tạo ra là $C_{f,t}$. Sau đó, các tác giả tối thiểu hàm mất mát cross entropy bằng cách sử dụng kết quả từ [40], [81] để tính điểm $\mathcal{L}$ theo công thức dưới đây

$$\mathcal{L} = -\left(\sum_{f=(h,r,t,\tau)\in\mathcal{B}} \frac{\exp\big(\phi(h,r,t,\tau)\big)}{\sum_{t'\in C_{f,t}} \exp\left(\phi(h,r,t',\tau)\right)} + \frac{\exp\big(\phi(h,r,t,\tau)\big)}{\sum_{h'\in C_{f,h}} \exp\left(\phi(h',r,t,\tau)\right)} \right) \quad (4\text{-}2)$$

4.2.2 Ý tưởng cải tiến

Ưu điểm lớn nhất của phương pháp nhúng lịch đại DE từ [19] là việc đưa ra đề xuất cho phép kết hợp với bất kỳ mô hình hoàn thiện ĐTTT nào đó để tạo thành mô hình hoàn thiện ĐTTT thời gian. Trong quá trình tìm hiểu và nghiên cứu về phương pháp nhúng lịch đại DE, luận án nhận thấy hiệu năng của mô hình hoàn thiện ĐTTT thời gian với sự kết hợp sẽ bị ảnh hưởng khi:

- Thay thế mô hình hoàn thiện ĐTTT tĩnh phù hợp với tập dữ liệu vì việc lựa chọn mô hình phù hợp sẽ ảnh hưởng trực tiếp đến độ chính xác suy luận, khả năng tổng quát hóa với quan hệ và thực thể mới, tốc độ huấn luyện và sử dụng.
- Thay thế hàm kích hoạt vì hàm kích hoạt là thành phần cốt lõi trong mạng học sâu nên việc lựa chọn hàm kích hoạt phù hợp giúp mạng học được quan hệ phi tuyến tốt hơn, dự đoán chính xác hơn.

Trong [19], R. Goel và cộng sự kết hợp mô hình nhúng lịch đại với một số mô hình hoàn thiện ĐTTT khác nhau như: mô hình hoàn thiện ĐTTT TransE, DistMult, TuckER, RESCAL và SimplE.

Về mặt lý thuyết và thực nghiệm, R. Goel và cộng sự chỉ ra rằng phương pháp nhúng lịch đại DE khi kết hợp với mô hình hoàn thiên ĐTTT SimplE, để tạo thành mô hình hoàn thiện ĐTTT thời gian DE-SimplE có hiệu năng vượt trội. Mô hình SimplE do S. M. Kazemi và cộng sự [41] đề xuất đã được chứng minh một cách rõ ràng về mặt lý thuyết là có tính biểu cảm đầy đủ. Hơn nữa, bằng thực nghiệm S. M. Kazemi và cộng sự cũng chỉ ra rằng mô hình hoàn thiện ĐTTT SimplE (khi so sánh với các mô hình hoàn thiện ĐTTT TransE, DistMult, TuckER) cũng có hiệu năng vượt trội so với các mô hình hoàn thiện ĐTTT khác trên các tập dữ liệu tĩnh.

Trong các mô hình hoàn thiện ĐTTT, mô hình BoxE do R. Abboud và cộng sự giới thiệu [82] được đánh giá là mô hình có hiệu năng khá tốt. Đặc biệt, Abboud và cộng sự đã chỉ ra rằng mô hình hoàn thiện ĐTTT BoxE có hiệu năng vượt trội so với các mô hình hoàn thiện ĐTTT khác như TransE, DistMult, TuckER và SimplE. Giống với mô hình SimplE [41], mô hình BoxE đã được Abboud và cộng sự chứng minh là có tính biểu cảm đầy đủ. Hơn nữa, mô hình BoxE sử dụng các hộp căn chỉnh nên cung cấp nền tảng linh hoạt hơn để xử lý các mẫu quan hệ phức tạp. Khả năng biểu diễn các quan hệ phân cấp, "một-nhiều" và "nhiều-nhiều" của mô hình BoxE mang lại cho nó lợi thế về khả năng biểu đạt, đặc biệt là đối với các ĐTTT phức tạp. Hơn nữa, khi so sánh về khả năng diễn giải giữa hai mô hình BoxE và SimplE, luận án có các kết luận sau:

- Mô hình SimplE: cung cấp khả năng diễn giải hạn chế hơn vì nó dựa vào nhúng trên các véc-tơ truyền thống. Mặc dù mô hình cho phép biểu diễn các quan hệ theo cả hai hướng, nhưng việc hiểu lý do tại sao một số dữ kiện nhất định được dự đoán có thể khó hiểu hơn.

- Mô hình BoxE: mô hình này cung cấp khả năng diễn giải mạnh hơn nhiều do bản chất của mô hình là sử dụng hình học. Hơn nữa, với việc xác định vị trí của một thực thể bên trong hoặc bên ngoài hộp cung cấp một biểu diễn trực quan và rõ ràng về việc nó có thỏa mãn một quan hệ hay không. Từ đó, mô hình giúp diễn giải các quyết định một cách nhanh chóng.

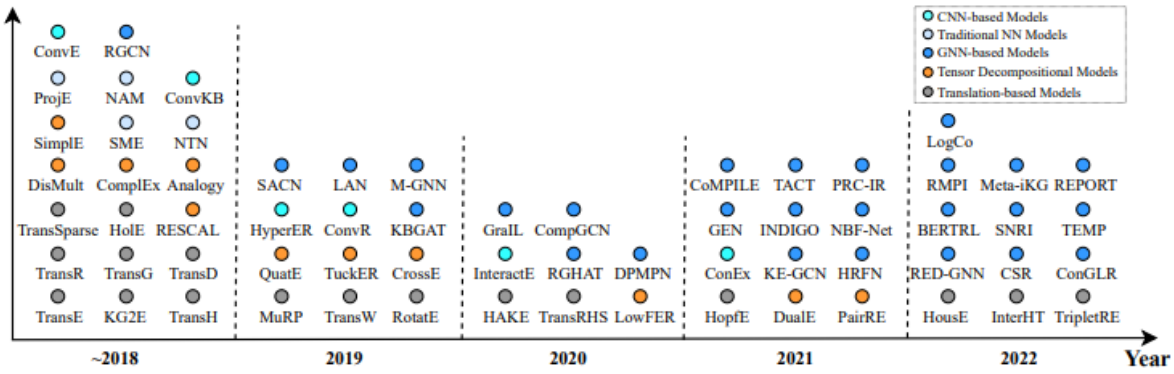
Mặc dù mô hình BoxE hoạt động tốt hơn mô hình SimplE đối với các ĐTTT tĩnh, trong nghiên cứu của J. Messner và cộng sự [83] thực nghiệm kết hợp mô hình hoàn thiện ĐTTT BoxE với nhúng lịch đại để tạo ra mô hình hoàn thiện ĐTTT thời gian DE-BoxE. Kết quả thực nghiệm với độ đo MRR trên cơ sở dữ liệu ICEWS14 [84] được cho bởi Bảng 4.2.

Bảng 4.2. So sánh giữa hai mô hình hoàn thiện ĐTTT thời gian DE-SimplE và mô hình hoàn thiện ĐTTT thời gian DE-BoxE.

Mô hình	$MRR \uparrow$
DE-SimplE	0.526
DE-BoxE	0.476

Thực nghiệm của J. Messner và cộng sự chỉ ra rằng không phải tất cả các mô hình hoàn thiện ĐTTT kết hợp với nhúng lịch đại để tạo ra mô hình hoàn thiện ĐTTT thời gian đều có kết quả tốt. Dựa theo nghiên cứu của J. Messner và cộng sự, luận án nhận thấy rằng việc kết quả của mô hình hoàn thiện ĐTTT thời gian DE-BoxE không tốt hơn kết quả của mô hình DE-SimplE có thể được giải thích bằng cách xem xét một cách chi tiết hệ thống của mô hình hoàn thiện ĐTTT BoxE và hàm khoảng cách. Trong mô hình hoàn thiện ĐTTT BoxE, một dữ kiện được coi là đúng khi và chỉ khi cả hai nhúng thực thể tương ứng bên trong các hộp đích của chúng. Trong khi đó, cách tiếp cận nhúng lịch đại DE là lựa chọn hàm kích hoạt dựa trên việc áp đặt quyền ưu tiên các đặc trưng thời gian được mong đợi hoạt động như thế nào theo dòng thời gian: Hàm kích hoạt sin giả định trước chu kỳ thay đổi của đặc trưng thời gian, trong khi hàm kích hoạt $sigmoid$ giả định tăng trưởng đơn điệu. Nếu các giả định này không phù hợp với dữ liệu, nó không chắc chắn rằng mỗi chiều thời gian có thể thiết lập để tất cả thực thể được đặt trong các hộp tương ứng của chúng. Hơn nữa, mô hình hoàn thiện ĐTTT BoxE coi quan hệ là các hộp chứa các thực thể, chính vì thế nếu như véc-tơ nhúng của các thực thể biến đổi theo chu kỳ thời gian (ví dụ biến đổi theo

hình sin) thì sẽ có rất nhiều khoảng thời gian nằm trong hộp. Ví dụ như dữ kiện buổi bảo vệ luận án Tiến sĩ của một nghiên cứu sinh thường chỉ đúng trong một khoảng thời gian ngắn (một buổi).



Hình 4.2. Tổng hợp sự xuất hiện của các mô hình hoàn thiện ĐTTT theo thời gian [23].

K. Liang và cộng sự (2024) [23] công bố một bảng tổng hợp về các loại mô hình hoàn thiện ĐTTT khác nhau (xem Hình 4.2). Có rất nhiều mô hình hoàn thiện ĐTTT khác nhau dựa trên nhúng được giới thiệu và xây dựng sau thời điểm xuất hiện của mô hình hoàn thiện ĐTTT SimplE. Các tác giả đã tổng hợp sự xuất hiện của các mô hình hoàn thiện ĐTTT theo dòng thời gian như chỉ ra ở Hình 4.2.

Thêm nữa, Sun và cộng sự [38] đã chứng minh rằng mô hình hoàn thiện ĐTTT RotatE xuất hiện sau này có thể mô hình hóa và suy luận ra bốn kiểu mẫu quan hệ. Cụ thể cho kết quả là: Quan hệ r trong mô hình RotatE có tính chất đối xứng, bất đối xứng và nghịch đảo, kết hợp. Bảng 4.3 tổng hợp tính chất quan hệ của một số mô hình.

Bảng 4.3. Tổng hợp tính chất quan hệ của một số mô hình hoàn thiện ĐTTT.

Mô hình	*Đối xứng*	*Phản xứng*	*Nghịch đảo*	*Kết hợp*
TransE	Không	Có	có	Có
DistMult	Có	Không	không	Không
SimplE	có	Có	Có	Không
RotatE	có	Có	Có	Có

Bên cạnh đó, qua thời gian thì mô hình hoàn thiện ĐTTT RotatE đã được chứng minh là mô hình có tốc độ tính toán nhanh vì sử dụng phép xoay chiều để biểu diễn quan hệ. Hơn nữa, mô hình hoàn thiện ĐTTT RotatE khá đơn giản và hiệu quả.

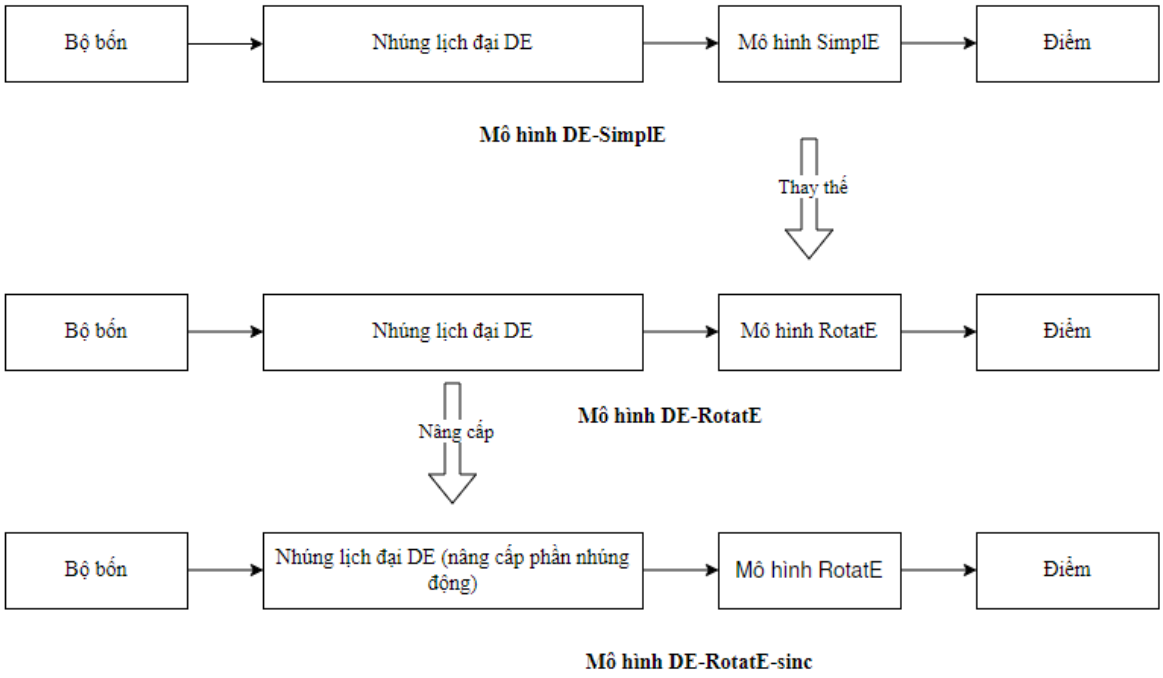
Với các đặc trưng tương đồng giữa mô hình hoàn thiện ĐTTT SimplE và mô hình hoàn thiện ĐTTT RotatE, và danh sách tổng kết về thông tin, hiệu năng của các mô hình hoàn thiện ĐTTT do K. Liang và cộng sự [23] tổng hợp, và tính chất về quan hệ của một số mô hình hoàn thiện ĐTTT tại Bảng 4.3 và các ưu điểm của mô hình hoàn thiện ĐTTT RotatE, luận án đề xuất phương án cải tiến :

- Đầu tiên là kết hợp mô hình nhúng lịch DE với mô hình hoàn thiện ĐTTT RotatE. Mô hình hoàn thiện ĐTTT thời gian mới được tạo ra có tên là DE-RotatE. Nghĩa là, thay đổi thành phần nhúng trong mô hình hoàn thiện ĐTTT RotatE bằng thành phần nhúng lịch đại DE.
- Thay đổi hàm kích hoạt sin bằng $sinc$ vì hàm $sinc$ ổn định hơn và giảm dao động dẫn tới học hiệu quả hơn.

Luận án đề xuất pipeline DE-RotatE + hàm sinc có thể áp dụng cho các ĐTTT phức tạp, nhiều loại quan hệ và có tính thời gian. Bên cạnh đó, cách phân tích của luận án cung cấp nguyên tắc chọn mô hình cơ sở (base KGC model) khi kết hợp với DE đó là không phải mọi mô hình đều phù hợp, mà cần xem xét sự tương thích giữa cơ chế nhúng (vector, hộp, phép xoay) và giả định thời gian (sin, sigmoid, sinc). Đề xuất có giá trị tham khảo cho các nghiên cứu tiếp theo về hoàn thiện ĐTTT thời gian.

4.3 Đề xuất hai mô hình DE-RotatE và DE-RotatE-sinc

Trong phần này, luận án trình bày về đề xuất cải tiến mô hình hoàn thiện ĐTTT thời gian từ mô hình gốc DE-SimplE [19] (luận án đề xuất cải tiến và so sánh với mô hình có hiệu năng tốt nhất). Phương án đề xuất cải tiến theo hai hướng tiếp cận nêu trên. Hai mô hình thực nghiệm đề xuất được gọi là mô hình hoàn thiện ĐTTT thời gian DE-RotatE (thay thế mô hình hoàn thiện ĐTTT SimplE bằng mô hình hoàn thiện ĐTTT RotatE [38]) và mô hình hoàn thiện ĐTTT thời gian DE-RotatE-sinc (thay thế mô hình hoàn thiện ĐTTT SimplE bằng mô hình hoàn thiện ĐTTT RotatE [38] và cải tiến thành phần trong Công thức (4-1)). Các mô hình đề xuất cải tiến được mô tả tổng thể trong Hình 4.3.



Hình 4.3. Hai phương pháp đề xuất cải tiến

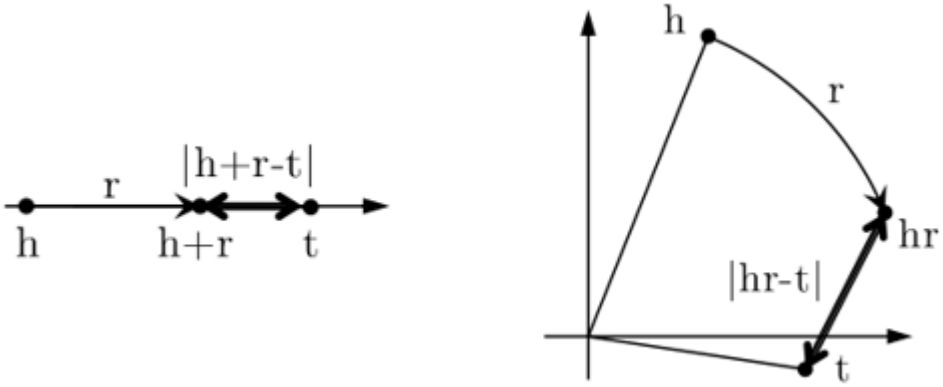
4.3.1 Mô hình đề xuất DE-RotatE

Mô hình hoàn thiện ĐTTT RotatE được Z. Sun và cộng sự đề xuất [38], là một mô hình được phát triển dựa trên TransE [36]. Cũng giống như các mô hình hoàn thiện ĐTTT trước đó, mô hình RotatE cũng dựa trên nhúng của các thực thể và quan hệ để đưa ra dự đoán các quan hệ còn thiếu trong ĐTTT. Trong khi mô hình hoàn thiện ĐTTT TransE xây dựng dựa trên phép tịnh tiến thì mô hình hoàn thiện ĐTTT RotatE được xây dựng dựa trên phép quay trong không gian phức. Trong mô hình hoàn thiện ĐTTT RotatE, các quan hệ được định nghĩa như các phép quay trong không gian véc-tơ phức. Hình 4.4 mô tả sự khác biệt giữa mô hình hoàn thiện ĐTTT TransE và mô hình hoàn thiện ĐTTT RotatE.

Xét một bộ ba (h, r, t), dựa trên đồng nhất thức Euler, Z. Sun và cộng sự đề xuất ánh xạ các thực thể đầu h và thực thể cuối t vào không gian nhúng phức, nghĩa là tạo thành các véc-tơ nhúng $\mathbf{v}_h, \mathbf{v}_t \in \mathbb{C}^k$, ở đó $\mathbb{C}^k$ là không gian nhúng phức k chiều. Sau đó, xây dựng một ánh xạ tạo ra bởi mỗi quan hệ r như một phép quay element-wise từ thực thể đầu h đến thực thể cuối t. Cụ thể như công thức dưới đây.

$$\boldsymbol{v}_t = \boldsymbol{v}_h \circ \boldsymbol{v}_r \qquad (4\text{-}3)$$

trong đó $|\mathbf{v}_{r_i}| = 1$ và $\circ$ là phép nhân Hadmard (tích element-wise).



Hình 4.4. Mô hình hoàn thiện ĐTTT TransE [36] (trái) và mô hình hoàn thiện ĐTTT RotatE [38] (phải).

Theo Công thức (4-3), mỗi phần tử trong không gian nhúng là $\mathbf{v}_{t_i} = \mathbf{v}_{h_i}\mathbf{v}_{r_i}$. Trong mô hình này, các tác giả hạn chế mô-đun của từng phần tử trong không gian nhúng phức, với $\mathbf{v}_r \in \mathbb{C}^k$, nghĩa là $\mathbf{v}_{r_i} \in \mathbb{C}$ sao cho $|\mathbf{v}_{r_i}| = 1$. Khi đó, $\mathbf{v}_{r_i}$ là $e^{i\theta_{r,i}}$, tương ứng với phép quay ngược chiều kim đồng hồ bởi $\theta_{r,i}$ radians từ gốc của mặt phẳng phức. Do vậy, mô hình có tên gọi là RotatE. Hàm số khoảng cách của RotatE được định nghĩa theo công thức sau.

$$d_r(\boldsymbol{v}_h, \boldsymbol{v}_t) = \|\boldsymbol{v}_h \circ \boldsymbol{v}_r - \boldsymbol{v}_t\| \qquad (4\text{-}4)$$

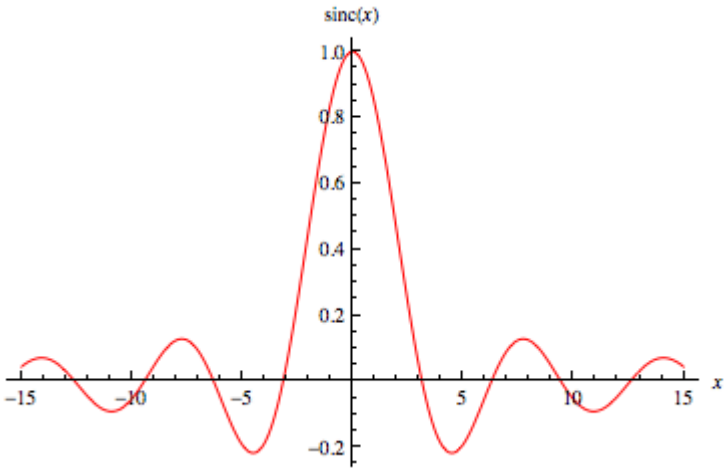
4.3.2 Mô hình đề xuất DE-RotatE-sinc

R. Goel và cộng sự [19] lựa chọn hàm $\sin(x)$ làm hàm kích hoạt vì tính chất tuần hoàn vô hạn của nó. Với tính chất tuần hoàn sẽ giúp mô hình nắm được tần suất của các dữ kiện có thực thể theo thời gian. Một số ví dụ thực thể như "kỳ thi", "kỳ nhập học" … Những thực thể dạng này liên quan đến dữ kiện có tần suất. Tuy nhiên, trên thực tế không phải dữ kiện nào cũng liên quan đến các thực thể có tần suất như vậy. Ví dụ dữ kiện "Steve Jobs là chủ tịch của Apple" chỉ đúng vào khoảng thời gian 1997- 2011.

Sau khi phân tích các thành phần trong Công thức (4-1), luận án nhận thấy hàm kích hoạt sẽ là yếu tố quan trọng làm ảnh hưởng đến hiệu năng và sự phù hợp

của nhúng lịch đại DE khi kết hợp với mô hình hoàn thiện ĐTTT để tạo thành mô hình hoàn thiện đồ thị tri thức thời gian. Ngoài ra, khi phân tích về ý nghĩa của các thực thể liên quan đến dữ kiện, trong các thực nghiệm tiếp theo, luận án đề xuất sử dụng hàm kích hoạt $\mathrm{sinc}(x)$ thay vì sử dụng hàm kích hoạt $\sin x$ như trong [19]. Hàm $\mathrm{sinc}(x)$ và đồ thị của nó được mô ta theo công thức và hình vẽ tương ứng dưới đây.

$$sinc(x) = \begin{cases} \frac{\sin x}{x}, x \neq 0 \\ 1, x = 0 \end{cases} \quad (4\text{-}5)$$



Hình 4.5. Đồ thị mô tả hàm $sinc(x)$

Hàm $\mathrm{sinc}(x)$ là một hàm số trong toán học, nó có nhiều ứng dụng trong xử lý tín hiệu và phân tích Fourier. Hàm số có hai phiên bản định nghĩa khác nhau phụ thuộc vào thời gian liên tục và thời gian rời rạc, nhưng khái niệm thì giống nhau trong cả hai phiên bản.

- Phiên bản không chuẩn hóa được định nghĩa như sau: $\mathrm{sinc}(x) = \frac{\sin x}{x}, x \neq 0$. Phiên bản không chuẩn hóa còn được gọi là hàm lấy mẫu.
- Phiên bản chuẩn hóa được định nghĩa như sau: $\mathrm{sinc}(x) = \frac{\sin \pi x}{\pi x}, x \neq 0$.

Về cơ bản thì hai phiên bản giống nhau, điểm khác biệt duy nhất là ở việc phiên bản chuẩn hóa chia tỷ lệ biến độc lập (theo trục x) với hệ số π. Ngoài ra, đối với phiên bản không chuẩn hóa thì tích phân xác định trên tập số thực bằng π, trong khi đó đối với phiên bản chuẩn hóa tích phân xác định trên tập số thực là 1. Trong cả hai phiên, tại điểm $x = 0$ đều có giá trị bằng 1.

Hàm $\mathrm{sinc}(x) = \frac{\sin x}{x}$ có tính chất đối xứng qua trục tung. Hơn nữa, nó là một biến thể của hàm $\sin x$ do vậy cũng có tính chất lặp vô hạn nhưng giá trị càng lớn khi gần về trục tung (trục đối xứng). Do đó, mô hình sẽ hoạt động hiệu quả hơn đối với những thực thể nằm trong dữ kiện với khoảng thời gian nhất định.

Trong phần này, luận án đề xuất sử dụng hàm kích hoạt $\mathrm{sinc}(x)$ (với phiên bản không chuẩn hóa) thay thế cho hàm $\sin(x)$ trong Công thức (4-1). Khi đó, công thức này trở thành:

$$f(\tau) = \boldsymbol{a}_e[n]\,\mathrm{sinc}(\boldsymbol{w}_e[n]\tau + \boldsymbol{b}_e[n])$$

Đi sâu hơn thì luận án nhận thấy do tính đối xứng của hàm số $\mathrm{sinc}(x)$ qua trục tung, nên tồn tại $\tau_1, \tau_2 \in \mathcal{T}$ và $\tau_1 \neq \tau_2$ sao cho $f(\tau_1) = f(\tau_2)$, nghĩa là tồn tại hai khoảng thời gian khác nhau cho cùng một giá trị. Về mặt thực tế thì điều này là bất hợp lý, có thể ảnh hưởng đến hiệu năng của mô hình đề xuất. Ví dụ ngày 01/08/2023 sinh viên đến trường X và gặp thầy hướng dẫn để trao đổi về đồ án. Từ đó ta có thể suy ra được thầy hướng dẫn ở trường X vào ngày 01/08/2023. Nhưng nếu 01/08/2023 sinh viên đến trường X và sinh viên gặp thầy hướng dẫn vào ngày 31/07/2023. Với dữ kiện này thì không thể kết luận được thầy hướng dẫn ở trường X vào ngày 31/07/2023. Nhưng do tính chất đối xứng của hàm $\mathrm{sinc}(x)$, có khả năng xảy ra dữ kiện "sinh viên gặp thầy hướng dẫn vào 01/08/2023" và "sinh viên gặp thầy hướng dẫn vào ngày 31/07/2023" có cùng một giá trị.

Để khắc phục hiện tượng này, luận án đề xuất bổ sung thêm một nhúng cho yếu tố thời gian vào sau kết quả của hàm $f(\tau)$ để cho hàm này mất tính đối xứng. Khi đó, các dữ kiện có cùng kết quả nhúng sẽ có giá trị khác nhau.

Hơn nữa, khi một dữ kiện xảy ra trong một khoảng thời gian dài thì số lượng véc-tơ nhúng cho mỗi ngày sẽ rất lớn, dẫn đến số lượng tham số mô hình cũng tăng theo. Do đó, luận án có thể chia thời gian thành nhiều khoảng bằng nhau và mỗi khoảng sẽ có một véc-tơ nhúng riêng, cụ thể là chia theo ngày. Ngoài ra, thời gian (cấu trúc gồm ngày, tháng, năm) sẽ được chuẩn hóa về khoảng $(-1, 1)$. Do vậy, công thức cho nhúng lịch đại trong mô hình DE-RotatE-sinc (cải tiến tiếp theo từ DE-RotatE) sẽ được mô tả dưới đây

$$\mathbf{v}_e^{\tau}[n] = \begin{cases} \boldsymbol{a}_e[n]\,\mathrm{sinc}(\boldsymbol{w}_e[n]\tau + \boldsymbol{b}_e[n]) + c_\tau, & nếu\ 1 \le n \le \gamma d \\ \boldsymbol{a}_e[n]\,, & nếu\ \gamma d < n \le d \end{cases} \qquad (4\text{-}6)$$

ở đó, c_τ là véc-tơ nhúng riêng cho từng khoảng thời gian.

Trong RotatE, không gian phức được biểu diễn bằng phần thực và ảo. Khi đó mỗi thực thể sẽ có 2 nhúng để biểu diễn cho phần thực và phần ảo. Sau đó cộng thêm nhúng động (chính là DE) cho cả 2 nhúng này để tạo ra một nhúng kết hợp giữa phần tĩnh và phần phụ thuộc thời gian. Sau đó các mô hình đề xuất áp dụng hàm tính điểm RotatE.

Độ phức tạp thời gian

Như lập luận của R. Goel và cộng sự [19] là khi mở rộng một mô hình hoàn thiện ĐTTT thành mô hình hoàn thiện ĐTTT thời gian bằng cách sử dụng nhúng lịch đại DE thì mô hình mới này sẽ không thay đổi độ phức tạp thời gian. Do đó, lập luận tương tự như đối với các mô hình DE-TransE, DE-DistMult và DE-SimplE [19], hai mô hình hoàn thiện ĐTTT thời gian do luận án đề xuất DE-RotatE, DE-RotatE-sinc đều có độ phức tạp thời gian là $O(d)$, ở đó d là kích thước của các nhúng.

4.4 Thực nghiệm mô hình đề xuất và đánh giá

4.4.1 Phần mềm và phần cứng

Dựa theo bộ mã nguồn được R. Goel và cộng sự [19] cung cấp, trong phần này của luận án cũng sử dụng ngôn ngữ Python và nền tảng Pytorch để thực nghiệm mô hình gốc DE-SimplE và hai mô hình đề xuất DE-RotatE và DE-RotatE-sinc.

Bảng 4.4. Thông tin phần cứng cho các thực nghiệm.

	Google Colab Pro+
Nguồn	https://colab.research.google.com
GPU	Nvidia K80/ T4
Tốc độ xử lý GPU	0,82GHz/ 1,59GHz
Dung lượng GPU	7,0 GB/ 40,0 GB
Dung lượng RAM	3,3 GB/ 83,5 GB
Dung lượng bộ nhớ	25,5 GB / 166,8 GB

Để thực nghiệm các mô hình, luận án sử dụng dung Google Colab Pro+, cụ thể được cho tại Bảng 4.4.

4.4.2 Tập dữ liệu thực nghiệm mô hình

Giống như các thực nghiệm trong [19], luận án cũng thực nghiệm với ba tập dữ liệu ICEWS14, ICEWS05-15 của A. García-Durán và cộng sự [40] và GDELT của K. Leetaru [85].

Bộ dữ liệu ICEWS kết hợp một cơ sở dữ liệu của các sự kiện chính trị và một hệ thống sử dụng những vấn đề này để đưa ra các cảnh báo sớm về xung đột. Nó được hỗ trợ bởi các cơ quan nghiên cứu quốc phòng của Mỹ. ICEWS14 và ICEWS05-15 là hai tập con của bộ dữ liệu ICEWS được trích xuất bởi các tác giả [39]: ICEWS14 chứa các sự kiện trong năm 2014 và ICEWS05-15 là các sự kiện từ 2005 đến 2015.

Bộ GDELT là bộ dữ liệu toàn cầu về các sự kiện, ngôn ngữ và giai điệu. Giống như các thực nghiệm trong [19], luận án dùng GDELT là một tập con dựa theo các sự kiện từ 01/04/2015 đến 31/03/2016.

Thông tin về các tập dữ liệu thực nghiệm được mô tả dưới đây.

Bảng 4.5. Thống kê về các tập dữ liệu ICEWS14, ICEWS05-15 và GDELT.

Datasets	$\|E\|$	$\|R\|$	$\|\mathcal{T}\|$	$\|train\|$	$\|validation\|$	$\|test\|$	$\|\mathcal{G}\|$
ICEWS14	7.128	230	365	72.826	8.941	8.963	90.730
ICEWS05-15	10.488	251	4017	386.962	46.275	46.092	479.329
GDELT	500	20	366	2.735.685	341.961	341.961	3.419.607

Luận án cũng đánh giá các kết quả thực nghiệm trên ba độ đo MR (Mearn Rank), MRR (Mearn Reciprocal Rank) và $Hit@10$ như trong [19].

4.4.3 Thông số cài đặt

Trong luận án này thực nghiệm hai mô hình là: mô hình hoàn thiện ĐTTT thời gian DE-RotatE và mô hình hoàn thiện ĐTTT thời gian DE-RotatE-sinc riêng biệt cùng với thông số cài đặt:

- Dữ liệu thời gian (ngày, tháng, năm) được chuẩn hóa về miền $(-1; 1)$.
- Tỷ lệ nhúng động là $0{,}32$ trên tổng số 100 chiều.
- Một số tham số khác được lựa chọn như trong [19], cụ thể là:
    - Tốc độ học là $0{,}001$.
    - dropout là $0{,}2$.
    - batch size là 512.
    - Số epochs được các tác giả lựa chọn là 500 cho cả ba tập dữ liệu. Do vấn đề hạn chế về phần cứng, nên luận án thay đổi số lượng epochs phụ thuộc vào từng tập dữ liệu:
        - ICEWS14: 500 epochs.
        - ICEWS05-15: 100 epochs.
        - GDELT: 80 – 90 epochs.

Tiếp theo, để diễn giải về cách thức lựa chọn tham số cho số lượng epochs trong thực nghiệm, luận án tiến hành các thực nghiệm trên nền tảng Kaggle với số lượng epochs khác nhau (tương ứng là 60, 80, 100, 120) trên bộ dữ liệu ICEWS05-15 và so sánh hiệu năng mô hình với mô hình gốc của tác giả.

4.4.4 Kết quả thực nghiệm

Mô hình đề xuất trong luận án được phát triển từ ý tưởng nhúng lịch đại DE trong [19] cho ĐTTT thời gian. Hơn nữa, mô hình hoàn thiện ĐTTT thời gian DE-SimplE được R. Goel và cộng sự chỉ ra là hiệu quả nhất. Do vậy, trong phần này để so sánh với kết quả của mô hình đề xuất, luận án sử dụng kết quả đã được công bố bởi R. Goel và cộng sự. Ngoài ra, luận án cũng huấn luyện thực nghiệm lại mô hình hoàn thiện ĐTTT thời gian DE-SimplE (để thuận tiện luận án ký hiệu là DE-SimplE-retrain) để đưa ra kết quả so sánh. Bảng 4.6 cho các kết quả thực nghiệm trên bộ ICEWS14.

Bảng 4.6. Kết quả thực nghiệm trên ĐTTT thời gian ICEWS14. Theo từng cột, kết quả tốt nhất in đậm, kết quả tốt thứ hai in nghiêng.

	ICEWS14				
Model	$MR \downarrow$	$MRR \uparrow$	$Hit@1 \uparrow$	$Hit@3 \uparrow$	$Hit@10 \uparrow$
DE-SimplE [19]	-	0,526	41,8	59,2	72,5
DE-SimplE-retrain	225	0,523	41,3	59,0	72,9
DE-RotatE	*184*	*0,538*	*42,7*	*60,5*	*74,3*
DE-RotatE-sinc	**181**	**0,555**	**44,8**	**62,5**	**75,5**

Kết quả thực nghiệm trên bộ ICEWS05 – 15 với số lượng epochs khác nhau được thể hiện ở Bảng 4.7.

Bảng 4.7. Kết quả thực nghiệm trên các ĐTTT thời gian ICEWS05-15 với số lượng epochs khác nhau

Độ đo	DE-RotatE-sinc (60)	DE-RotatE-sinc (80)	DE-RotatE-sinc (100)	DE-RotatE-sinc (120)	De-SimplE-sinc (100)
Hit@1 ↑	39,5	39,7	**40,1**	39,8	37,7
Hit@3 ↑	58,5	59,0	**59,3**	59,1	56,4
Hit@10 ↑	75,4	75,8	**76,0**	76,0	73,7
MR ↓	90	93	**96**	97	104
MMR ↑	0,518	0,520	**0,523**	0,522	0,499
Thời gian(s)	14708,7	19590,2	**24488,9**	29401,8	16794,4

Bảng 4.8. Kết quả thực nghiệm trên các ĐTTT thời gian ICEWS05-15 và GDELT

	ICEWS05-15				
Model	$MR\downarrow$	$MRR\uparrow$	$Hit@1\uparrow$	$Hit@3\uparrow$	$Hit@10\uparrow$
DE-SimplE [19]	-	*0,513*	*39,2*	*57,8*	*74,8*
DE-SimplE (retrain) với 100 epochs	104	0,499	37,9	56,4	73,6
DE-RotatE	104	0,495	37,4	55,8	73,0
DE-RotatE-sinc	**92**	**0,528**	**40,5**	**59,9**	**76,6**
	GDELT				
DE-SimplE [19]	-	**0,230**	14,1	24,8	**40.3**
DE-RotatE-sinc (80 epochs)	*66,93*	0,221	*14,12*	*23,75*	37,58
DE-RotaE-sinc (90 epochs)	**66,90**	*0,222*	**14,18**	**23,80**	*37,66*

Từ giá trị các độ đo Hit@k, MR, MRR thu được qua các epochs và thời gian cần để mô hình huấn luyện thì với số epoch 100 là lựa chọn tốt nhất về độ hiệu quả tính theo các độ đo lẫn thời gian tiêu tốn để huấn luyện mô hình. Vì vậy luận án lựa chọn tham số epochs là 100 để tiến hành các thực nghiệm với bộ ICEWS05 – 15 trên các nền tảng Colab tương tự như phần trình bày tiếp theo.

Kết quả thực nghiệm trên hai ĐTTT thời gian ICEWS05-15 và GDELT được cho ở

Bảng 4.8. Với ĐTTT thời gian GDELT, luận án không thực nghiệm huấn luyện lại mô hình DE-SimplE mà sử dụng trực tiếp kết quả từ bài báo gốc [19].

Từ các bảng kết quả thực nghiệm, luận án nhận thấy việc kết hợp nhúng lịch đại DE với mô hình hoàn thiện ĐTTT RotatE mang lại kết quả tương đối khả quan. Trong tất cả trường hợp thì kết quả của mô hình hoàn thiện ĐTTT thời gian DE-RotatE-sinc đều tốt hơn kết quả của mô hìn hoàn thiện ĐTTT thời gian DE-RotatE.

Điều đó, khẳng định rằng các thành phần trong Công thức (4-1) của nhúng lịch đại DE cũng ảnh hưởng nhất định đến kết quả của mô hình hoàn thiện ĐTTT thời gian DE-KBC. Kết quả thực nghiệm càng khẳng định rằng cần phải lựa chọn hàm kích hoạt phù hợp với mô hình hoàn thiện ĐTTT tĩnh và từng tập dữ liệu ĐTTT cụ thể. Chi tiết về kết quả có thể được so sánh dưới đây.

- Đối với ĐTTT thời gian ICEWS: kết quả tốt hơn rõ rệt trong tất cả các độ đo khi so sánh giữa mô hình hoàn thiện ĐTTT thời gian DE-RotatE-sinc và mô hình hoàn thiện ĐTTT thời gian DE-SimplE [19]:
  - Độ đo MRR: kết quả của mô hình DE-RotatE-sinc tăng $5{,}5\%$ so với mô hình DE-SimplE.
  - Độ đo $Hit@1$: kết quả của mô hình DE-RotatE-sinc tăng $7{,}2\%$ so với mô hình DE-SimplE.
  - Độ đo $Hit@3$: kết quả của mô hình DE-RotatE-sinc tăng $5{,}6\%$ so với mô hình DE-SimplE.
  - Độ đo $Hit@10$: kết quả của mô hình DE-RotatE-sinc tăng $4{,}1\%$ so với mô hình DE-SimplE.
- Đối với ĐTTT thời gian ICEWS05-15 (thời gian huấn luyện khoảng 172 giây/epoch). Luận án thực nghiệm với số lượng epochs nhỏ hơn so với mô hình gốc (bằng khoảng 20% so với mô hình gốc) thì kết quả của mô hình hoàn thiện ĐTTT thời gian DE-RotatE-sinc tốt hơn kết quả của mô hình hoàn thiện ĐTTT thời gian DE-SimplE trong tất cả độ đó. Nếu số lượng epochs tăng lên nữa thì luận án cho rằng kết quả sẽ còn cải thiện đáng kể.
  - Độ đo MRR: kết quả của mô hình DE-RotatE-sinc tăng 3% so với mô hình DE-SimplE.
  - Độ đo $Hit@1$: kết quả của mô hình DE-RotatE-sinc tăng $3{,}3\%$ so với mô hình DE-SimplE.
  - Độ đo $Hit@3$: kết quả của mô hình DE-RotatE-sinc tăng $3{,}6\%$ so với mô hình DE-SimplE.
  - Độ đo $Hit@10$: kết quả của mô hình DE-RotatE-sinc tăng $2{,}4\%$ so với mô hình DE-SimplE.
- Đối với ĐTTT thời gian GDELT, đây là cơ sơ tri thức khá lớn, thời gian huấn luyện diễn ra khá lâu (cụ thể huấn luyện khoảng 700 giây/epoch). Do vậy, luận án đưa ra kịch bản thực nghiệm và so sánh giữa mô hình hoàn thiện ĐTTT thời gian DE-RotatE-SimplE với số lượng epochs khác nhau (cụ thể là 80 và 90).

Kết quả thực nghiệm chỉ ra rằng khi số lượng epochs được huấn luyện tăng lên thì kết quả của năm độ đo đều có xu hướng tăng lên.

4.5 Kết luận Chương 4

Chương này tập trung nghiên cứu một số nội dung về hoàn thiện ĐTTT thời gian: (i) Nghiên cứu khái quát về hoàn thiện ĐTTT thời gian, (ii) Khảo sát một số mô hình hoàn thiện ĐTTT thời gian, (iii) Phân tích chi tiết mô hình DE-SimplE [19], (iv) Đề xuất hai mô hình hoàn thiện ĐTTT thời gian là DE-RotatE và DE-RotatE-sinc được cải tiến từ mô hình DE-SimplE, (v) Triển khai thực nghiệm đánh giá hiệu năng của mô hình DE-RotatE và DE-RotatE-sinc và đối sánh với mô hình DE-SimplE. Kết quả trong chương này của luận án đã được công bố tại [AnhNTT2, AnhNTT3]. Công trình [AnhNTT2] đã nhận được tham chiếu từ một bài báo Scopus[4].

[4] B. Liu, W. Yao, and H. Zhou, "Logic Rule Guided Multi-hop Temporal Knowledge Graph Reasoning," in *Proceedings of the 2024 5th International Conference on Computing, Networks and Internet of Things*, in CNIOT '24. New York, NY, USA: Association for Computing Machinery, 2024, pp. 49–55. doi: 10.1145/3670105.3670114.

Kết luận

Kết quả chính của luận án

Theo dòng nghiên cứu trên thế giới về hoàn thiện ĐTTT, luận án đã hoàn thiện các mục tiêu nghiên cứu với ba đóng góp chính sau đây:

- *Đầu tiên*, luận án đề xuất mô hình BERT/FastText-GRU-KBC được cải tiến từ mô hình GloVe-GRU-KBC [17] theo hai hướng: (i) lựa chọn các mô hình ngôn ngữ BERT/FastText có ưu thế hơn GloVe trong việc chuyển giao từ dữ liệu/mô hình nguồn; (ii) sử dụng lớp kết nối đầy đủ trong nhúng từ giúp mô hình hoàn thiện ĐTTT đạt được sự cân bằng giữa khả năng học đặc trưng mạnh mẽ và tính tổng quát hóa trên toàn đồ thị tri thức. Phát triển phần mềm và tiến hành các kịch bản thực nghiệm đánh giá hiệu năng mô hình đề xuất. Kết quả thực nghiệm cho thấy hiệu năng của mô hình đề xuất đã được cải thiện. Các kết quả nghiên cứu này đã được công bố [AnhNTTT1].
- *Thứ hai*, luận án luận án đề xuất hai mô hình hoàn thiện ĐTTT đa phương thức mới dựa vào mô hình gốc AdaMF-MAT [18], trong đó, mô hình ViT-AdaMF-MAT (thay thế nhúng ảnh của mô hình gốc bằng mô hình nhúng ảnh Vision Transformer ViT) và mô hình T5-ViT-AdaMF-MAT (thay thế nhúng ảnh bằng ViT và nhúng văn bản bằng mô hình T5). Phát triển phần mềm và tiến hành các kịch bản thực nghiệm đánh giá hiệu năng mô hình đề xuất ViT-AdaMF-MAT và T5-ViT-AdaMF-MAT. Qua kết quả thực nghiệm, luận án chỉ ra tầm ảnh hưởng của việc sử dụng các mô hình nhúng trong việc hoàn thiện ĐTTT đa phương thức. Các kết quả nghiên cứu này đã được công bố [AnhNTTT4].
- *Cuối cùng*, luận án đề xuất hai mô hình mới là DE-RotatE và DE-RotatE-sinc dựa trên cải tiên mô hình DE-SimplE [19], trong đó, mô hình DE-RotatE sử dụng mô hình thành phần RotatE thay cho mô hình thành phần SimplE, DE-RotatE-sinc không chỉ cải tiến như DE-RotatE mà còn cải tiến thành phần nhúng lịch đại DE thành nhúng động nhờ việc sử dụng hàm kích hoạt $\mathrm{sinc}(x)$ thay vì sử dụng hàm kích hoạt $\sin x$ như [19]. Phát triển phần mềm và tiến hành các kịch bản thực nghiệm đánh giá hiệu năng mô hình đề xuất DE-RotatE và DE-RotatE-sinc. Kết quả thực nghiệm cho thấy hiệu năng của các mô hình

đề xuất đã được cải thiện. Các kết quả nghiên cứu này đã được công bố [AnhNTTT2], [AnhNTTT3].

Hạn chế của luận án

Luận án đã đạt được một số kết quả nghiên cứu bước đầu trong hoàn thiện ĐTTT dựa trên học chuyển giao, hoàn thiện ĐTTT đa phương thức và hoàn thiện ĐTTT thời gian. Tuy nhiên, luận án còn một số hạn chế như sau:

- Luận án mới cải tiến được thành phần mô hình học chuyển giao từ mô hình hoàn thiện ĐTTT gốc dựa trên học chuyển giao (thay thế mô hình BERT, mô hình FastText cho mô hình Glove) và cải tiến phần mã hóa của mô hình hoàn thiện ĐTTT gốc bằng cách sử dụng lớp kết nối đầy đủ FC vào vị trí phù hợp để tăng hiệu năng hoạt động. Cần tiến hành thêm các nghiên cứu sâu hơn về các mô hình khai thác các thông tin bổ sung cho hoàn thiện ĐTTT.
- Luận án mới cải tiến thành phần mô hình nhúng ảnh và thành phần mô hình nhúng văn bản từ mô hình hoàn thiện ĐTTT đa phương thức gốc AdaMF-MAT. Chủ đề hoàn thiện hoàn thiện ĐTTT đa phương thức mới trong thời kỳ ban đầu, tạo miền rộng mở cho các nghiên cứu tiếp theo.
- Luận án mới cải tiến được thành phần mô hình hoàn thiện ĐTTT tĩnh cùng thành phần hàm kích hoạt của thành phần nhúng lịch đại DE trong mô hình hoàn thiện ĐTTT theo thời gian dựa trên nhúng lịch đại DE. Nghiên cứu phát triển nhiều hơn các kỹ thuật học máy theo hướng chú đề này cũng rất hấp dẫn.
- Một số kết quả thực nghiệm trong luận án còn chưa đạt được hiệu năng và mức độ cao như kỳ vọng. Trong đó, việc phân tích lỗi thử nghiệm chưa được tiến hành kỹ.

Nghiên cứu tiếp theo

Hiện thời, nghiên cứu sinh và nhóm nghiên cứu đang tiến hành các công việc sau đây:

- Bài báo [Kocijan 2024] là sự mở rộng của bài báo [17] và giới thiệu bộ dữ liệu DOGE (Diagnostics of Open knowledge Graph Embeddings) để làm sáng tỏ tác động của phương pháp học chuyển giao trong hoàn thiện ĐTTT và "chuẩn đoán" các mô hình đã được tiền huấn luyện, nhằm hiểu sâu hơn những gì chúng

thực sự học được, thay vì chỉ đo lường hiệu suất tổng thể. Bằng cách sử dụng bộ dữ liệu DOGE các tác giả hướng tới các mục tiêu sau:

- Độ bao phủ kiến thức: Kiểm tra xem kiến thức mà mô hình học được trong quá trình tiền huấn luyện bao phủ những lĩnh vực nào.
- Tính nhất quán với từ đồng nghĩa: khả năng nhận biết các thực thể hoặc quan hệ có tên gọi khác nhau nhưng cùng một ý nghĩa.
- Tính nhất quán với quan hệ đảo ngược: Kiểm tra xem mô hình có đưa ra dự đoán nhất quán cho một truy vấn và truy vấn đảo ngược của nó hay không.
- Khả năng suy luận diễn dịch: Đây là một trong những phần kiểm tra nhằm đánh giá liệu mô hình có thể kết hợp kiến thức nền (học từ tiền huấn luyện) với một sự thật mới (được cung cấp thêm) để suy ra một sự thật khác hay không.

Dựa trên nghiên cứu này, NCS sẽ tiếp tục tìm hiểu nghiên cứu kỹ thuật học chuyển giao đã đề xuất trong luận án trên các tập dữ liệu có tính chuyên biệt.

- Phân tích khung hoàn thiện ĐTTT đa phương thức CMR (Contrastive learning, Memorization, and Retrieval) gồm ba bước là Học đối chiếu tránh mâu thuẫn giữa các phương thức và đưa các hàng xóm ngữ nghĩa hữu ích đến gần nhau; Ghi nhớ lưu trữ rõ các hàng xóm ngữ nghĩa; Truy hồi tập các hàng xóm ngữ nghĩa trong giai đoạn kiểm thử để tăng cường suy luận [86]. Kết nối CRM với các tiếp cận làm giàu thông tin trong [87], [88], [89] để đề xuất ý tưởng cải tiến để xây dựng một mô hình hoàn thiện ĐTTT đa phương thức mới.
- Kết nối các kết quả nghiên cứu của luận án trong các mô hình hoàn thiện ĐTTT thời gian DE-RotatE và DE-RotatE-sinc với giải pháp tận dụng nói tích chập hai-ba chiều trong nhúng lịch đại trong [90] để nâng cấp, cải tiến các mô hình DE-RotatE và DE-RotatE-sinc.
- Khảo sát phân tích lỗi để làm cơ sở đối với những nghiên cứu tiếp theo.
- Tích hợp thành phần hoàn thiện ĐTTT vào các hệ thống chatbots.

Nghiên cứu sinh lập kế hoạch có kết quả nghiên cứu được chấp nhận công bố tại các hội nghị khoa học quốc tế vào Quý 4 năm 2025.

Danh mục công trình khoa học của tác giả liên quan tới luận án

1. [AnhNTT1] **Thuy-Anh Nguyen Thi**, Thi-Hong Vuong, Thi-Hanh Le, Xuan-Hieu Phan, Thi-Thao Le, Quang-Thuy Ha. *Knowledge Base Completion with Transfer Learning Using BERT and fastText*. KSE 2022: 1-6. **Scopus**, **DBLP**, 01 trích dẫn Scopus.
2. [AnhNTT2] **Thuy-Anh Nguyen Thi**, Viet-Phuong Ta, Xuan Hieu Phan, Quang-Thuy Ha. *An Improvement of Diachronic Embedding for Temporal Knowledge Graph Completion*. ACIIDS (2) 2023: 111-120. **Scopus**, **DBLP**, 01 trích dẫn Scopus.
3. [AnhNTT3] **Thuy-Anh Nguyen Thi**, Hieu Cong Nguyen, Xuan-Hieu Phan, Quang-Thuy Ha. *Time Factor in Diachronic Embedding for Temporal Knowledge Graph Completion*. Book of Abstracts, International Joint Conference on Rough Sets (IJCRS 2023), pp. 26-28.
4. [AnhNTT4] **Thuy-Anh Nguyen Thi**, Cong-Hoang Le, Viet-Phuong Ta, Xuan-Hieu Phan, Quang-Thuy Ha. *The Impact of Embeddings on the Performance of Multi-modal Knowledge Graph Completion*. KSE 2024 (chấp nhận đăng). **Scopus**.

Tài liệu tham khảo

[1]. Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, Axel-Cyrille Ngonga Ngomo, Axel Polleres, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan F. Sequeda, Steffen Staab, Antoine Zimmermann: *Knowledge Graphs*. ACM Comput. Surv. 54(4): 71:1-71:37 (2022).

[2]. Jesús Barrasa, Amy E. Hodler, and Jim Webber: *Knowledge Graphs: Data in Context for Responsive Businesses.* O'Reilly Media, Inc., 2021.

[3]. Ilaria Tiddi, Stefan Schlobach: *Knowledge graphs as tools for explainable machine learning: A survey*. Artif. Intell. 302: 103627 (2022).

[4]. Yushan Liu: *Machine learning with knowledge graphs for explainable Artificial Intelligence*. Ludwig Maximilian University of Munich, Germany, 2024.

[5]. Enayat Rajabi, Kobra Etminani: *Knowledge-graph-based explainable AI: A systematic review*. J. Inf. Sci. 50(4): 1019-1029 (2024).

[6]. Natalya Fridman Noy, Yuqing Gao, Anshu Jain, Anant Narayanan, Alan Patterson, Jamie Taylor: *Industry-scale knowledge graphs: lessons and challenges*. Commun. ACM 62(8): 36-43 (2019).

[7]. Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, Philip S. Yu: *A Survey on Knowledge Graphs: Representation, Acquisition, and Applications*. IEEE Trans. Neural Networks Learn. Syst. 33(2): 494-514 (2022).

[8]. Zhe Chen, Yuehan Wang, Bin Zhao, Jing Cheng, Xin Zhao, Zongtao Duan: *Knowledge Graph Completion: A Review*. IEEE Access 8: 192435-192456 (2020).

[9]. Tong Shen, Fu Zhang, Jingwei Cheng: *A comprehensive overview of knowledge graph completion*. Knowl. Based Syst. 255: 109597 (2022).

[10]. Russa Biswas: *Embedding Based Link Prediction for Knowledge Graph Completion*. Karlsruhe Institute of Technology, Germany, 2023.

[11]. Armand Boschin: *Machine learning techniques for automatic knowledge graph completion*. (Méthodes d'apprentissage automatique pour la complétion

de graphes de connaissances). Polytechnic Institute of Paris, Palaiseau, France, 2023

[12]. Donghan Yu: *Leveraging Textual Semantics for Knowledge Graph Acquisition and Application*. Carnegie Mellon University, 2023.

[13]. Fangong Wang: *Schema aware knowledge graph completions*. University of Edinburgh, 2024.

[14]. Shuwen Liu: *Deep Learning with Knowledge Graphs Using Graph Neural Networks*. University of Oxford, 2024.

[15]. Timothée Lesort, Vincenzo Lomonaco, Andrei Stoian, Davide Maltoni, David Filliat, Natalia Díaz Rodríguez: *Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges*. Inf. Fusion 58: 52-68 (2020).

[16]. Tom M. Mitchell, William W. Cohen, Estevam R. Hruschka Jr., Partha P. Talukdar, Bo Yang, Justin Betteridge, Andrew Carlson, Bhavana Dalvi Mishra, Matt Gardner, Bryan Kisiel, Jayant Krishnamurthy, Ni Lao, Kathryn Mazaitis, Thahir Mohamed, Ndapandula Nakashole, Emmanouil A. Platanios, Alan Ritter, Mehdi Samadi, Burr Settles, Richard C. Wang, Derry Wijaya, Abhinav Gupta, Xinlei Chen, Abulhair Saparov, Malcolm Greaves, Joel Welling: *Never-ending learning*. Commun. ACM 61(5): 103-115 (2018).

[17]. Vid Kocijan, Thomas Lukasiewicz: *Knowledge Base Completion Meets Transfer Learning*. EMNLP (1) 2021: 6521-6533.

[18]. Yichi Zhang, Zhuo Chen, Lei Liang, Huajun Chen, Wen Zhang: *Unleashing the Power of Imbalanced Modality Information for Multi-modal Knowledge Graph Completion*. LREC/COLING 2024: 17120-17130.

[19]. Rishab Goel, Seyed Mehran Kazemi, Marcus A. Brubaker, Pascal Poupart: *Diachronic Embedding for Temporal Knowledge Graph Completion*. AAAI 2020: 3988-3995.

[20]. Maximilian Nickel, Kevin Murphy, Volker Tresp, Evgeniy Gabrilovich: *A Review of Relational Machine Learning for Knowledge Graphs*. Proc. IEEE 104(1): 11-33 (2016).

[21]. Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, Wei Zhang: *Knowledge vault: a web-scale approach to probabilistic knowledge fusion*. KDD 2014: 601-610.

[22]. Quan Wang, Zhendong Mao, Bin Wang, Li Guo: *Knowledge Graph Embedding: A Survey of Approaches and Applications*. IEEE Trans. Knowl. Data Eng. 29(12): 2724-2743 (2017).

[23]. Ke Liang, Lingyuan Meng, Meng Liu, Yue Liu, Wenxuan Tu, Siwei Wang, Sihang Zhou, Xinwang Liu, Fuchun Sun, Kunlun He: *A Survey of Knowledge Graph Reasoning on Graph Types: Static, Dynamic, and Multi-Modal*. IEEE Trans. Pattern Anal. Mach. Intell. 46(12): 9456-9478 (2024).

[24]. Swapnil Gupta, Sreyash Kenkre, Partha P. Talukdar: *CaRe: Open Knowledge Graph Embeddings*. EMNLP/IJCNLP (1) 2019: 378-388.

[25]. Yichi Zhang, Mingyang Chen, Wen Zhang: *Modality-Aware Negative Sampling for Multi-modal Knowledge Graph Embedding*. IJCNN 2023: 1-8.

[26]. Xiangru Zhu, Zhixu Li, Xiaodan Wang, Xueyao Jiang, Penglei Sun, Xuwu Wang, Yanghua Xiao, Nicholas Jing Yuan: *Multi-Modal Knowledge Graph Construction and Application: A Survey*. IEEE Trans. Knowl. Data Eng. 36(2): 715-735 (2024).

[27]. Meng Wang, Sen Wang, Han Yang, Zheng Zhang, Xi Chen, Guilin Qi: *Is Visual Context Really Helpful for Knowledge Graph? A Representation Learning Perspective*. ACM Multimedia 2021: 2735-2743.

[28]. Mingyang Chen, Wen Zhang, Wei Zhang, Qiang Chen, Huajun Chen: *Meta Relational Learning for Few-Shot Link Prediction in Knowledge Graphs*. EMNLP/IJCNLP (1) 2019: 4216-4225.

[29]. Anthony Fader, Stephen Soderland, Oren Etzioni: *Identifying Relations for Open Information Extraction*. EMNLP 2011: 1535-1545.

[30]. Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, Ido Dagan: *Supervised Open Information Extraction*. NAACL-HLT 2018: 885-895.

[31]. Patrick Hohenecker, Frank Mtumbuka, Vid Kocijan, Thomas Lukasiewicz: *Systematic Comparison of Neural Architectures and Training Approaches for Open Information Extraction*. EMNLP (1) 2020: 8554-8565.

[32]. Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, Sebastian Riedel: *Convolutional 2D Knowledge Graph Embeddings*. AAAI 2018: 1811-1818.

[33]. Ivana Balazevic, Carl Allen, Timothy M. Hospedales: *TuckER: Tensor Factorization for Knowledge Graph Completion*. EMNLP/IJCNLP (1) 2019: 5184-5193.

[34]. Mojtaba Nayyeri, Sahar Vahdati, Can Aykul, Jens Lehmann: 5* *Knowledge Graph Embeddings with Projective Transformations*. AAAI 2021: 9064-9072.

[35]. Cheng Yang, Zhiyuan Liu, Cunchao Tu, Chuan Shi, Maosong Sun: *Network Embedding: Theories, Methods, and Applications*. Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool Publishers 2021, ISBN 978-3-031-00462-9, pp. 1-242.

[36]. Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, Oksana Yakhnenko: *Translating Embeddings for Modeling Multi-relational Data*. NIPS 2013: 2787-2795.

[37]. Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, Li Deng: *Embedding Entities and Relations for Learning and Inference in Knowledge Bases*. ICLR (Poster) 2015.

[38]. Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, Jian Tang: *RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space*. ICLR (Poster) 2019.

[39]. Tingsong Jiang, Tianyu Liu, Tao Ge, Lei Sha, Baobao Chang, Sujian Li, Zhifang Sui: *Towards Time-Aware Knowledge Graph Completion*. COLING 2016: 1715-1724.

[40]. Alberto García-Durán, Sebastijan Dumancic, Mathias Niepert: *Learning Sequence Encoders for Temporal Knowledge Graph Completion*. EMNLP 2018: 4816-4821.

[41]. Seyed Mehran Kazemi, David Poole: *SimplE Embedding for Link Prediction in Knowledge Graphs*. NeurIPS 2018: 4289-4300.

[42]. Zhuo Chen, Yichi Zhang, Yin Fang, Yuxia Geng, Lingbing Guo, Xiang Chen, Qian Li, Wen Zhang, Jiaoyan Chen, Yushan Zhu, Jiaqi Li, Xiaoze Liu, Jeff Z. Pan, Ningyu Zhang, Huajun Chen: *Knowledge Graphs Meet Multi-Modal Learning: A Comprehensive Survey*. CoRR abs/2402.05391 (2024).

[43]. Ruobing Xie, Zhiyuan Liu, Huanbo Luan, Maosong Sun: *Image-embodied Knowledge Representation Learning*. IJCAI 2017: 3140-3146.

[44]. Hatem Mousselly Sergieh, Teresa Botschen, Iryna Gurevych, Stefan Roth: *A Multimodal Translation-Based Approach for Knowledge Graph Representation Learning*. *SEM@NAACL-HLT 2018: 225-234.

[45]. Yichi Zhang, Wen Zhang: *Knowledge Graph Completion with Pre-trained Multimodal Transformer and Twins Negative Sampling*. CoRR abs/2209.07084 (2022).

[46]. Xiang Wang, Yaokun Xu, Xiangnan He, Yixin Cao, Meng Wang, Tat-Seng Chua: *Reinforced Negative Sampling over Knowledge Graph for Recommendation*. WWW 2020: 99-109.

[47]. Zhen Yang, Ming Ding, Chang Zhou, Hongxia Yang, Jingren Zhou, Jie Tang: *Understanding Negative Sampling in Graph Representation Learning*. KDD 2020: 1666-1676.

[48]. Derong Xu, Tong Xu, Shiwei Wu, Jingbo Zhou, Enhong Chen: *Relation-enhanced Negative Sampling for Multimodal Knowledge Graph Completion*. ACM Multimedia 2022: 3857-3866.

[49]. Yichi Zhang, Zhuo Chen, Wen Zhang: *MACO: A Modality Adversarial and Contrastive Framework for Modality-Missing Multi-modal Knowledge Graph Completion*. NLPCC (1) 2023: 123-134.

[50]. Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby: *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. ICLR 2021.

[51]. Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Binbin Hu, Ziqi Liu, Wen Zhang, Huajun Chen: *NativE: Multi-modal Knowledge Graph Completion in the Wild*. SIGIR 2024: 91-101.

[52]. Yunpeng Wang, Bo Ning, Xin Wang, Guanyu Li: *Multi-hop neighbor fusion enhanced hierarchical transformer for multi-modal knowledge graph completion*. World Wide Web (WWW) 27(1) (2024).

[53]. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova: *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. NAACL-HLT (1) 2019: 4171-4186.

[54]. Jiapu Wang, Boyue Wang, Meikang Qiu, Shirui Pan, Bo Xiong, Heng Liu, Linhao Luo, Tengfei Liu, Yongli Hu, Baocai Yin, Wen Gao: *A Survey on Temporal Knowledge Graph Completion: Taxonomy, Progress, and Prospects*. CoRR abs/2308.02457 (2023).

[55]. Borui Cai, Yong Xiang, Longxiang Gao, He Zhang, Yunfeng Li, Jianxin Li: *Temporal Knowledge Graph Completion: A Survey*. IJCAI 2023: 6545-6553.

[56]. Sinno Jialin Pan, Qiang Yang: *A Survey on Transfer Learning*. IEEE Trans. Knowl. Data Eng. 22(10): 1345-1359 (2010).

[57]. Qiang Yang, Yu Zhang, Wenyuan Dai, Sinno Jialin Pan: *Transfer Learning*. Cambridge University Press, 2020.

[58]. Jeffrey Pennington, Richard Socher, Christopher D. Manning: *Glove: Global Vectors for Word Representation*. EMNLP 2014: 1532-1543.

[59]. Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, Yoshua Bengio: *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. EMNLP 2014: 1724-1734.

[60]. Piotr Bojanowski, Edouard Grave, Armand Joulin, Tomás Mikolov: *Enriching Word Vectors with Subword Information*. Trans. Assoc. Comput. Linguistics 5: 135-146 (2017).

[61]. Samuel Broscheit, Kiril Gashteovski, Yanjie Wang, Rainer Gemulla: *Can We Predict New Facts with Open Knowledge Graph Embeddings? A Benchmark for Open Link Prediction*. ACL 2020: 2296-2308.

[62]. Shikhar Vashishth, Prince Jain, Partha P. Talukdar: *CESI: Canonicalizing Open Knowledge Bases using Embeddings and Side Information*. WWW 2018: 1317-1327.

[63]. Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, Michael Gamon: *Representing Text for Joint Embedding of Text and Knowledge Bases*. EMNLP 2015: 1499-1509.

[64]. Timothée Lacroix, Nicolas Usunier, Guillaume Obozinski: *Canonical Tensor Decomposition for Knowledge Base Completion*. ICML 2018: 2869-2878.

[65]. Vid Kocijan, Myeongjun Erik Jang, Thomas Lukasiewicz: *Pre-training and diagnosing knowledge base completion models*. Artif. Intell. 329: 104081 (2024).

[66]. Nils Reimers, Iryna Gurevych: *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. EMNLP/IJCNLP (1) 2019: 3980-3990.

[67]. Karen Simonyan, Andrew Zisserman: *Very Deep Convolutional Networks for Large-Scale Image Recognition*. ICLR 2015: 1-14.

[68]. Hangbo Bao, Li Dong, Songhao Piao, Furu Wei: *BEiT: BERT Pre-Training of Image Transformers*. ICLR 2022.

[69]. Giovanni Mariani, Florian Scheidegger, Roxana Istrate, Costas Bekas, A. Cristiano I. Malossi: *BAGAN: Data Augmentation with Balancing GAN*. CoRR abs/1803.09655 (2018).

[70]. Andrew L. Maas, Awni Y. Hannun, Andrew Y. Ng: *Rectifier Nonlinearities Improve Neural Network Acoustic Models*. ICML 2013.

[71]. Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J. Liu: *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer.* J. Mach. Learn. Res. 21: 140:1-140:67 (2020).

[72]. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin: *Attention is All you Need*. NIPS 2017: 5998-6008.

[73]. Peter W. Battaglia, Jessica B. Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinícius Flores Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, Çaglar Gülçehre, H. Francis Song, Andrew J. Ballard, Justin Gilmer, George E. Dahl, Ashish Vaswani, Kelsey R. Allen, Charles Nash, Victoria Langston, Chris Dyer, Nicolas Heess, Daan Wierstra, Pushmeet Kohli, Matthew M. Botvinick, Oriol Vinyals, Yujia Li, Razvan Pascanu: *Relational inductive biases, deep learning, and graph networks*. CoRR abs/1806.01261 (2018).

[74]. Lei Jimmy Ba, Jamie Ryan Kiros, Geoffrey E. Hinton: *Layer Normalization*. CoRR abs/1607.06450 (2016).

[75]. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun: *Deep Residual Learning for Image Recognition*. CVPR 2016: 770-778.

[76]. Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov: *Dropout: a simple way to prevent neural networks from overfitting*. J. Mach. Learn. Res. 15(1): 1929-1958 (2014).

[77]. Tomás Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean: *Efficient Estimation of Word Representations in Vector Space*. ICLR (Workshop Poster) 2013.

[78]. William L. Hamilton, Jure Leskovec, Dan Jurafsky: *Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change*. ACL (1) 2016: 1489-1501.

[79]. Shib Sankar Dasgupta, Swayambhu Nath Ray, Partha P. Talukdar: *HyTE: Hyperplane-based Temporally aware Knowledge Graph Embedding*. EMNLP 2018: 2001-2011.

[80]. Keyulu Xu, Weihua Hu, Jure Leskovec, Stefanie Jegelka: *How Powerful are Graph Neural Networks?* ICLR 2019.

[81]. Rudolf Kadlec, Ondrej Bajgar, Jan Kleindienst: *Knowledge Base Completion: Baselines Strike Back*. Rep4NLP@ACL 2017: 69-74.

[82]. Ralph Abboud, İsmail İlkan Ceylan, Thomas Lukasiewicz, Tommaso Salvatori: *BoxE: a box embedding model for knowledge base completion*. NeurIPS 2020.

[83]. Johannes Messner, Ralph Abboud, İsmail İlkan Ceylan: *Temporal Knowledge Graph Completion Using Box Embeddings*. AAAI 2022: 7779-7787.

[84]. Elizabeth Boschee, Jennifer Lautenschlager, Sean O'Brien, Steve Shellman, James Starz, Michael Ward: *ICEWS Coded Event Data*. Harvard Dataverse, 2015.

[85]. Kalev Leetaru, Philip A. Schrodt: *GDELT: Global Data on Events, Location and Tone, 1979–2012*. In: Proceedings of the International Studies Association Annual Convention, 2013.

[86]. Yu Zhao, Ying Zhang, Baohang Zhou, Xinying Qian, Kehui Song, Xiangrui Cai: *Contrast then Memorize: Semantic Neighbor Retrieval-Enhanced Inductive Multimodal Knowledge Graph Completion*. SIGIR 2024: 102-111.

[87]. Linyu Li, Zhi Jin, Yichi Zhang, Dongming Jin, Chengfeng Dou, Yuanpeng He, Xuan Zhang, Haiyan Zhao: *Towards Structure-aware Model for Multi-modal Knowledge Graph Completion*. 2025. [Online]. Available: https://arxiv.org/abs/2505.21973.

[88]. Yulou Shu, Wengen Li, Jiaqi Wang, Yichao Zhang, Jihong Guan, Shuigeng Zhou: C2RS: *Multimodal Knowledge Graph Completion with Cross-Modal Consistency and Relation Semantics*. IEEE Transactions on Artificial Intelligence, pp. 1–13, 2025.

[89]. Guoliang Zhu, Tao Ren, Dandan Wang, Jun Hu: *Let Modalities Teach Each Other: Modal-Collaborative Knowledge Extraction and Fusion for*

Multimodal Knowledge Graph Completion. NAACL (Findings) 2025: 6195-6207.

[90]. Mingsheng He, Lin Zhu, Luyi Bai: *Jointly leveraging 1D and 2D convolution on diachronic entity embedding for temporal knowledge graph completion*. Appl. Soft Comput. 176: 113144 (2025).