

NHÚNG TRI THỨC DỰA TRÊN MẠNG TÍCH CHẬP ĐỒ THỊ

Knowledge Embedding Based Graph Convolution Network

Trần Đại Chí (18127070) Thái Hoàng Huy (18127109)

Trần Xuân Lộc (18127131) Võ Trần Quang Tuấn (18127248)

Lê Nhựt Nam (18120061)

Khoa Công nghệ Thông tin, Trường Đại học Khoa học Tự nhiên, Đại học Quốc gia TP.HCM

Ngày 12 tháng 8 năm 2022

Nội dung

- 1 Giới thiệu
- 2 Các công trình liên quan
- 3 Kiến thức nền tảng
- 4 Các phương pháp đề xuất
- 5 Kết quả thực nghiệm
- 6 Kết luận
- 7 Tài liệu tham khảo

Mạng tích chập đồ thị (GCN - Graph Convolution Network)

Mạng tích chập đồ thị, là một cách tiếp cận mới cho mô hình học bán giám sát dựa trên dữ liệu có cấu trúc đồ thị. Phương pháp chính dựa trên các loại mạng Nơ-ron tích chập thực thi trực tiếp trên đồ thị.

Đồ thị tri thức (KG - Knowledge Graph)

Đồ thị tri thức là phương pháp số hoá và tái cấu trúc các tri thức của nhân loại dưới dạng đồ thị bao gồm các nút và các cạnh tương ứng với các thực thể và các quan hệ giữa các thực thể.

Hạn chế của GCN và KG

Mặc dù GCN là một phương pháp mới giải quyết được nhiều bài toán sát thực tế, nhưng các phương pháp hiện tại vẫn có một số nhược điểm:

- Chỉ học những node dựa trên điều kiện của một số cạnh cố định
- Các cạnh của đồ thị phải là các quan hệ đồng nhất.

Bên cạnh đó, KG cũng có một số điểm yếu:

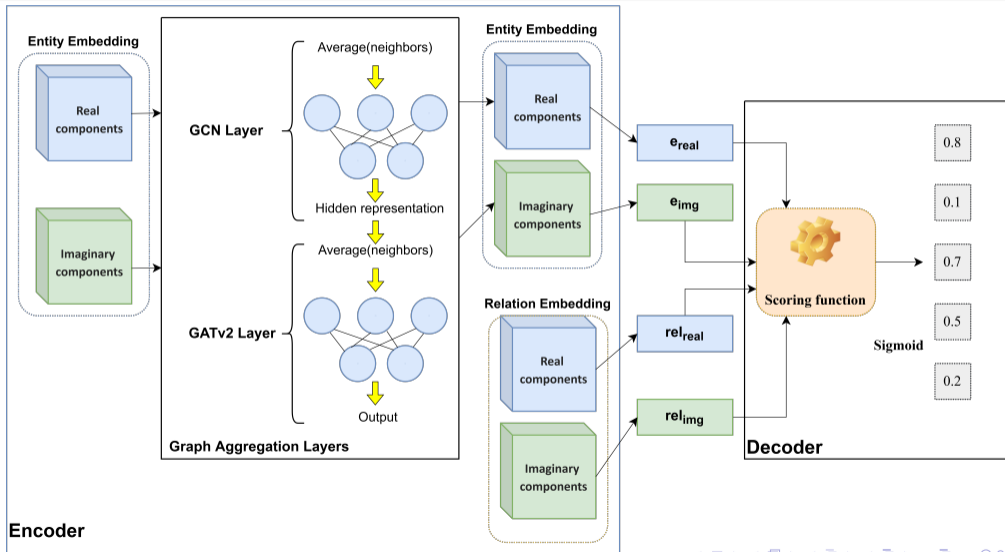
- Không tận dụng được việc lan truyền niềm tin của tích chập đồ thị trong quá trình học, mà chỉ học dựa trên hàm mục tiêu của bộ ba thực thể - quan hệ - thực thể.
- Không sử dụng cấu trúc đồ thị trong việc làm trơn không gian nhúng của thực thể và quan hệ.

Kết hợp KG và GCN, một phương pháp khả thi?

Kết hợp cả thể mạnh của GCN và KG để học cho cả thực thể và các mối quan hệ, liệu có hiệu quả hay không? Có nhưng chưa tối ưu ở việc nhúng các mối quan hệ cho quá trình huấn luyện. Trong đề tài này, chúng ta sẽ cải tiến:

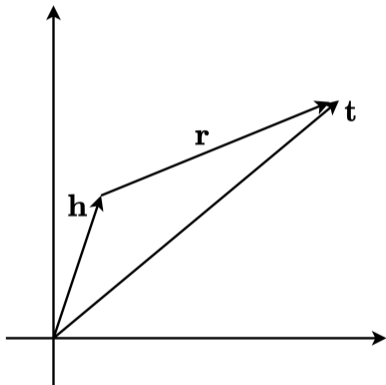
- Sử dụng một framework KE-GCN mới có thể cập nhật cả việc nhúng thực thể và mối quan hệ bằng phép toán tích chập đồ thị kết hợp các kỹ thuật nhúng tri thức khác nhau.
- Cung cấp một số khía cạnh đặc biệt, ưu và nhược điểm của phương pháp đề xuất.
- Thử nghiệm trên bộ dữ liệu chuẩn và quan sát sự vượt trội hơn các phương pháp cơ sở khác liên quan.

Công trình liên quan dựa trên mạng nơ ron

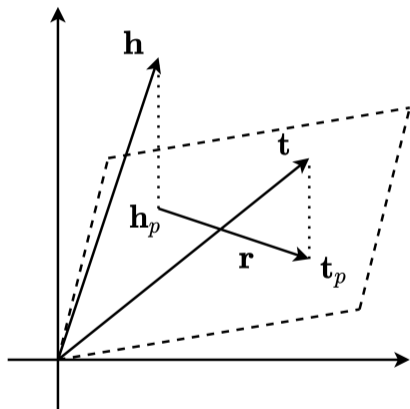


Công trình liên quan dựa trên nhúng tri thức

Phương pháp tính tiền khoảng cách



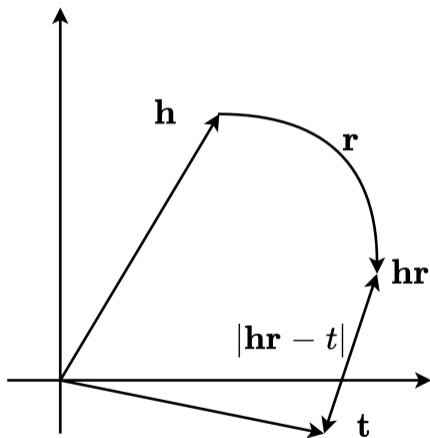
Hình 2: Translation-Distance: TransE



Hình 3: Translation-Distance: TransH

Công trình liên quan dựa trên nhúng tri thức

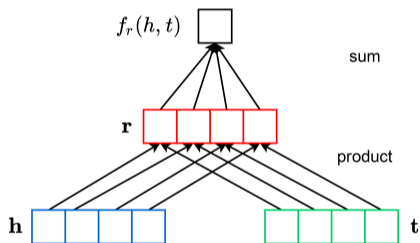
Phương pháp dựa trên phép xoay



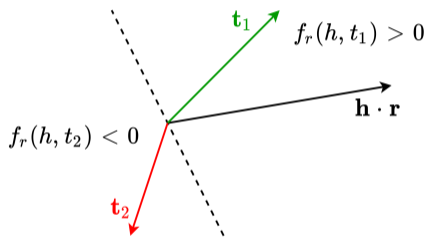
Hình 4: Rotational model: RotatE

Công trình liên quan dựa trên nhúng tri thức

Phương pháp dựa trên khớp ngữ nghĩa (tích trong)



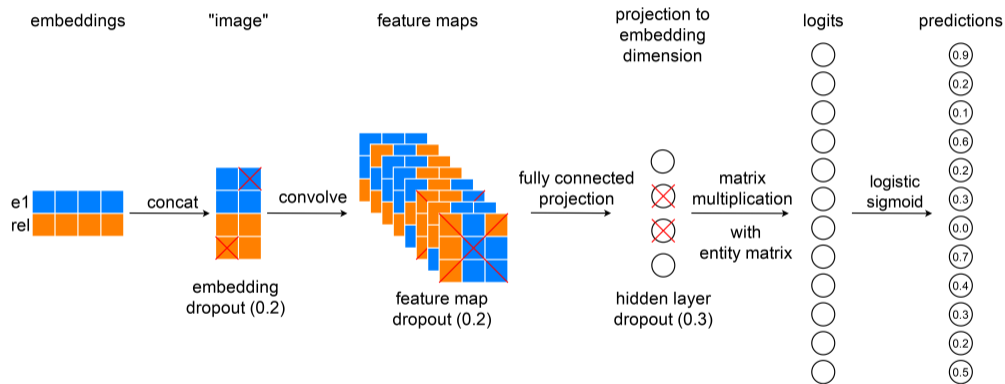
Hình 5: Semantic matching: DistMult



Hình 6: DistMult Geometric Perspective

Công trình liên quan dựa trên nhúng tri thức

Dựa trên mạng neural tích chập, capsule, ...



Hình 7: CNN-based: ConvE model

Nhắc lại: Node Embeddings

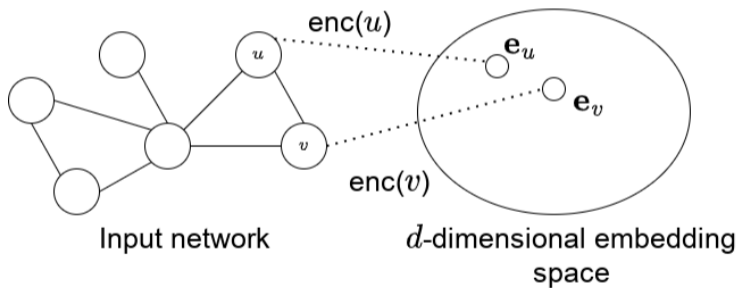
Ý tưởng cốt lõi: Nhúng những nút trong đồ thị vào không gian nhúng d -chiều sao cho những nút tương đồng trong đồ thị được nhúng gần nhau

$$f(\text{input graph}) = \text{2D node embeddings} \quad (1)$$

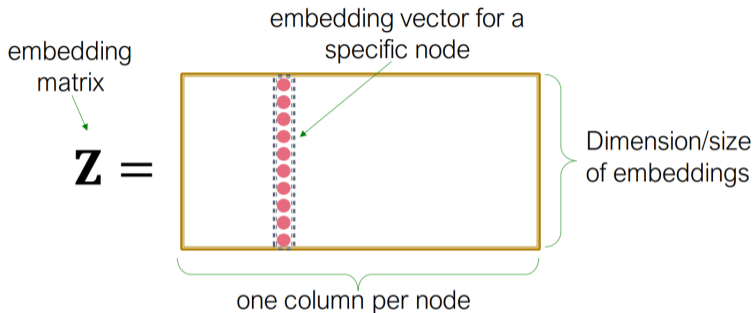
Vấn đề: Làm thế nào để học được hàm f

Nhắc lại: Node Embeddings

Mục tiêu: Định nghĩa được hàm similarity $(u, v) \approx \mathbf{e}_u^\top \mathbf{e}_v$



Nhắc lại: Shallow Encoding



$$\text{encode}(v) = \mathbf{e}_v = \mathbf{Z} \cdot v \quad (2)$$

trong đó:

- $\mathbf{Z} \in \mathbb{R}^{d \times |\mathcal{V}|}$: một ma trận mà mỗi cột là node embedding (ta cần học/ tối ưu)
- $v \in \mathbb{I}^{|\mathcal{V}|}$: indicator vector

Thể hiện đồ thị tri thức

Thể hiện đồ thị tri thức

- Các cạnh trong đồ thị tri thức được thể hiện như bộ ba (h, r, t) . Trong đồ thị tri thức động, các cạnh được thể hiện (h, r, t, τ)

Ý tưởng cốt lõi

- Mô hình hóa những thực thể và quan hệ trong không gian nhúng. Làm thế nào để mô hình hóa? Hàm mã hóa?
- Cho trước một bộ ba đúng (h, r, t) , nhúng của (h, r) phải nên "gần" với nhúng của t . Thế nào là "gần"? Hàm tương đồng?

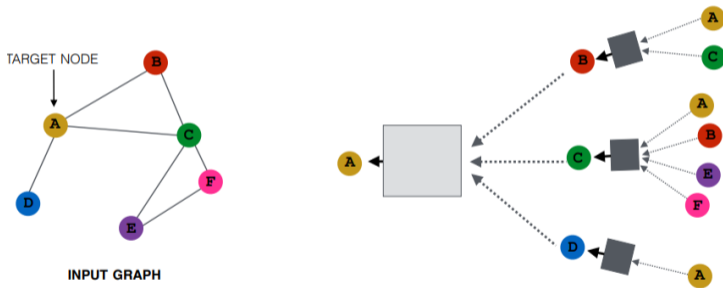
Học thể hiện tri thức

Những thành phần chính của học thể hiện

- Không gian thể hiện: point-wise; complex vector space; Gaussian distribution; manifolds; groups.
- Hàm tính điểm: distance-based; similarity-based.
- Chiến lược phát sinh mẫu âm: Uniform, Bernoulli; Adversarial, Self-Adversarial.
- Loss function: Margin-based loss; Logistic-based loss; Binary cross-entropy loss.

Graph Neural Networks

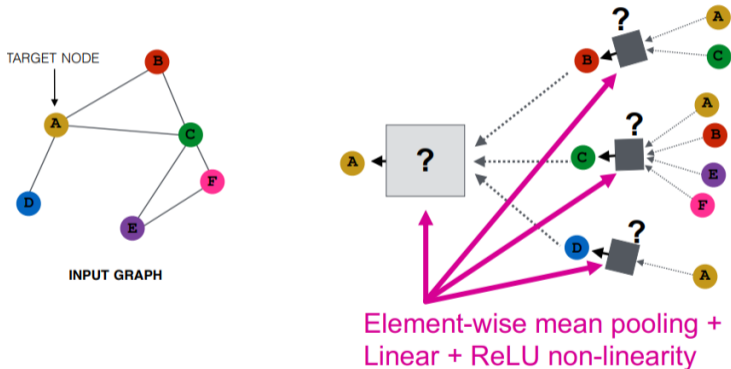
Lân cận của nút định nghĩa một đồ thị tính toán. Thông tin được lan truyền (propagate) và biến đổi (transform) trên toàn đồ thị. Mô hình học cách lan truyền thông tin trên toàn đồ thị để tính toán đặc trưng nút.



Hình 8: Mô hình tổng quát GNN

Graph Convolution Networks

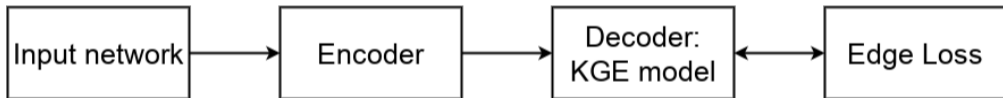
Ý tưởng chính: Phát sinh những node embedding dựa trên local network neighborhoods.



Hình 9: Mô hình tổng quát GCN

Framework for Knowledge Graphs

Để áp dụng các mô hình GNN cho dữ liệu đồ thị tri thức, các mô hình được xây dựng trên khung học encoder-decoder, trong đó các mô hình GNN đóng vai trò như encoder và các mô hình KGE đóng vai trò là decoder.



Hình 10: Khung học để áp dụng GNN cho dữ liệu đồ thị tri thức.

Phát biểu bài toán

Bài toán Entity Alignment

- Đầu vào (Input): Đồ thị tri thức $\mathcal{G}_1 = \{\mathcal{E}_1, \mathcal{R}_1, \mathcal{F}_1\}$, $\mathcal{F}_1 = (h, r, t)$ và $\mathcal{G}_2 = \{\mathcal{E}_2, \mathcal{R}_2, \mathcal{F}_2\}$, $\mathcal{F}_1 = (h, r, t)$
- Đầu ra (Output): $A = \{(e_1, e_2) \in \mathcal{E}_1 \times \mathcal{E}_2 \mid e_1 \equiv e_2\}$

Bài toán Knowledge Graph Entity Classification

- Đầu vào (Input): Đồ thị tri thức $\mathcal{G} = \{\mathcal{E}, \mathcal{R}, \mathcal{F}\}$, với những thực thể e
- Đầu ra (output): Luật phân lớp các thực thể trong đồ thị.

Tái định nghĩa Vanilla GCN

Dựa trên ý tưởng của việc nhúng đa lớp cho các nút trong đồ thị theo công thức Eq.3 và Eq.4.

$$\mathbf{m}_v^{l+1} = \sum_{u \in \mathcal{N}(v)} \mathbf{h}_u^l \quad (3)$$

$$\mathbf{h}_v^{l+1} = \sigma(\mathbf{W}^l(\mathbf{m}_v^{l+1} + \mathbf{h}_v^l)) \quad (4)$$

Trong đó:

- \mathbf{h}_v^l là vector nhúng của thực thể v tại lớp thứ l .
- \mathbf{m}_v^{l+1} là vector biểu diễn cho thông tin tổng hợp từ các láng giềng của thực thể v ở lớp thứ l .
- $\mathcal{N}(v)$ là tập hợp các láng giềng có liên kết trực tiếp với nút thứ v .
- \mathbf{W}^l là ma trận tham số của mô hình.

Tái định nghĩa Vanilla GCN (Tiếp)

Bằng cách áp dụng hàm tính điểm f để đo lường tính hợp lý của các quan hệ. Quan hệ chưa xuất hiện sẽ có điểm thấp hơn các quan hệ có sẵn.

Hàm tính điểm: $f(\mathbf{h}_u, \mathbf{h}_v) = \mathbf{h}_u^\top \mathbf{h}_v$. Ngoài ra, Cơ chế tính toán mới được định nghĩa như sau:

$$\mathbf{m}_v^{l+1} = \sum_{u \in \mathcal{N}(v)} \frac{\partial f(\mathbf{h}_u^l, \mathbf{h}_v^l)}{\partial \mathbf{h}_v^l} = \frac{\partial (\sum_{u \in \mathcal{N}(v)} f(\mathbf{h}_u^l, \mathbf{h}_v^l))}{\partial \mathbf{h}_v^l} \quad (5)$$

Kiến trúc của khung học đề xuất

Hàm tính điểm:

$$f(\mathbf{h}_u, \mathbf{h}_r, \mathbf{h}_v) \rightarrow \mathbb{R}$$

Cơ chế tính toán biểu diễn nút được phát triển lên từ Eq.5 như bên dưới:

$$\mathbf{m}_v^{l+1} = \sum_{(u,r) \in \mathcal{N}_{in}(v)} \mathbf{W}_r^l \frac{\partial f_{in}(\mathbf{h}_u^l, \mathbf{h}_r^l, \mathbf{h}_v^l)}{\partial \mathbf{h}_v^l} + \sum_{(u,r) \in \mathcal{N}_{out}(v)} \mathbf{W}_r^l \frac{\partial f_{out}(\mathbf{h}_v^l, \mathbf{h}_r^l, \mathbf{h}_u^l)}{\partial \mathbf{h}_v^l} \quad (6)$$

$$\mathbf{h}_v^{l+1} = \sigma_{ent}(\mathbf{m}_v^{l+1} + \mathbf{W}_0^l \mathbf{h}_v^l) \quad (7)$$

Trong đó:

- $\mathcal{N}_{in}(v) = \{(u, v) | u \xrightarrow{r} v\}$ là tập hợp các láng giềng có liên kết trong với thực thể v .
- $\mathcal{N}_{out}(v) = \{(u, v) | u \xleftarrow{r} v\}$ là tập hợp các láng giềng có liên kết ngoài với thực thể v .

Kiến trúc của khung học đề xuất (Tiếp)

Cập nhật nhúng quan hệ:

$$\mathbf{m}_r^{l+1} = \sum_{(u,v) \in \mathcal{N}(r)} \frac{\partial f_r(\mathbf{h}_u^l, \mathbf{h}_r^l, \mathbf{h}_v^l)}{\partial \mathbf{h}_r^l} \quad (8)$$

$$\mathbf{h}_r^{l+1} = \sigma_{rel}(\mathbf{W}_r^l(\mathbf{m}_r^{l+1} + \mathbf{h}_r^l)) \quad (9)$$

Trong đó:

- $\mathcal{N}_r = \{(u, v) | u \xrightarrow{r} v\}$ là danh sách các thực thể láng giềng có mối quan hệ với r .
- $\sigma_{rel}(\cdot)$ là hàm kích hoạt.
- $\partial f / \partial \mathbf{h}$ được tính thông qua **Auto Differentiation (AD)**.

Kiến trúc của khung học đề xuất (Tiếp)

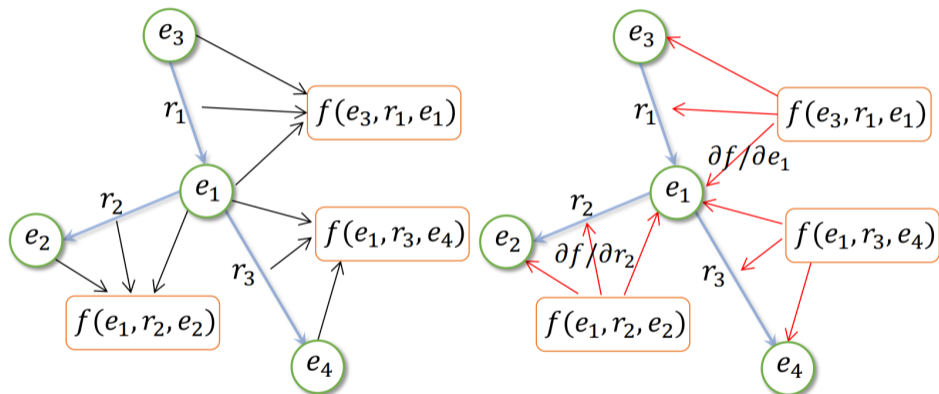
Cấu hình hàm lỗi:

- Phân loại thực thể: **cross-entropy**.
- KGs Alignment: sử dụng **khoảng cách giữa các vector nhúng** của 2 thực thể tới các đồ thị tri thức khác.

Các cấu hình khác:

- $f_{in} = f_{out} = f_r$ and $\mathbf{W}_r^l = \mathbf{W}^l$.
- Sử dụng hệ số α ở công thức Eq.7 and Eq.9 .Thay thế, \mathbf{m}_v^{l+1} bằng $\alpha \mathbf{m}_v^{l+1} / (|\mathcal{N}_{in}(v)| + |\mathcal{N}_{out}(v)|)$ và \mathbf{m}_r^{l+1} bằng $\alpha \mathbf{m}_r^{l+1} / |\mathcal{N}(r)|$.

Quá trình lan truyền thông tin của KE-GCN



Hình 11: Quá trình lan truyền thông tin của KE-GCN

Các tập dữ liệu thực nghiệm

	Tập dữ liệu	Thực thể	Quan hệ	Bộ ba
DBP _{ZH-EN}	Chinese	66,469	2,830	153,929
	English	98,125	2,317	237,674
DBP _{JA-EN}	Japanese	65,744	2,043	164,373
	English	95,680	2,096	233,319
DBP _{FR-EN}	French	66,858	1,379	192,191
	English	105,889	2,209	278,590

Bảng 1: Tóm tắt số liệu của tập dữ liệu cho bài toán liên kết đồ thị tri thức.

Các tập dữ liệu thực nghiệm (Tiếp)

Tập dữ liệu	AM	WN	FB15K
Thực thể	1,666,764	40,551	14,904
Quan hệ	133	18	1,341
Bộ ba	5,988,321	145,966	579,654
Đã gán nhãn	1,000	31,943	13,445
Số lớp	11	24	50

Bảng 2: Tóm tắt số liệu của các tập dữ liệu cho bài toán phân loại thực thể.

Cấu hình hàm lỗi

Bài toán liên kết đồ thị tri thức, hàm lỗi được định nghĩa như sau:

$$\mathcal{L} = \sum_{(u,v) \in S} \sum_{(u',v') \in S'_{(u,v)}} l(u, v, u', v') \quad (10)$$

$$l(u, v, u', v') = [||\mathbf{h}_u - \mathbf{h}_v||_1 + \gamma - ||\mathbf{h}_{u'} - \mathbf{h}_{v'}||_1]_+ \quad (11)$$

Trong đó:

- $[x]_+ = \max\{0, x\}$
- $S'_{(u,v)}$ là tập hợp các thực thể âm.
- $\gamma = 3$ là siêu tham số biên.
- Hệ số phân tách mỗi liên kết dương là 5.
- $S = \{(u, v)\}$ là tập liên kết thực thể huấn luyện, trong đó u là thực thể trong KG_1 và v là thực thể trong KG_2 .

Cấu hình hàm lỗi (Tiếp)

Trong bài toán phân loại thực thể, hàm lỗi được định nghĩa dựa trên 2 tác vụ khác nhau. Ở bài toán phân loại đa lớp (multi-class classification):

$$\mathcal{L} = - \sum_{u \in \mathcal{Y}_L} \sum_{c=1}^C Y_{uc} \ln \hat{Y}_{uc} \quad (12)$$

Ở bài toán phân loại đa nhãn (multi-label classification):

$$\mathcal{L} = - \sum_{u \in \mathcal{Y}_L} \sum_{c=1}^C Y_{uc} \ln \hat{Y}_{uc} + (1 - Y_{uc}) \ln(1 - \hat{Y}_{uc}) \quad (13)$$

Các độ đo I

Độ đo xếp hạng trung bình: là giá trị trung bình của thứ hạng các dự đoán giữa tất cả các ứng viên.

- $MR \in [1, \mathcal{E}]$.
- Giá trị MR càng nhỏ thì hiệu suất mô hình càng tốt.
- Công thức tính như Eq.14:

$$MR = \frac{1}{|Q|} \sum_{q \in Q} q \quad (14)$$

Trong đó: $|Q|$ là số lượng của các bộ ba trong tập đánh giá và q là xếp hạng của từng bộ ba.

Các độ đo II

Độ đo xếp hạng tương hỗ trung bình: là giá trị trung bình của các xếp hạng tương hỗ của kết quả cho một mẫu truy vấn Q .

- Nhằm khắc phục điểm yếu của độ đo MR.
- $MRR \in [0, 1]$.
- Giá trị càng lớn thì hiệu suất mô hình càng tốt.
- Công thức tính MRR như sau:

$$MRR = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{q} \quad (15)$$

Các độ đo III

Độ đo Hits@k: là xác suất xếp hạng của những bộ ba mà nhỏ hơn hoặc bằng k với k thường là 1, 3, 10.

- $\text{Hits}@k \in [0, 1]$.
- Giá trị càng lớn, hiệu suất mô hình càng tốt.
- Công thức tính Hits@k theo Eq.16:

$$\text{Hits}@k = \frac{|\{q \in Q : q \leq k\}|}{|Q|} \quad (16)$$

Các độ đo IV

Độ đo Precision@k: là sự tính toán số nhãn được dự đoán đúng chính xác trong tổng số các nhãn và k có nghĩa là chúng ta chỉ xét trong top-k các nhãn hàng đầu.

- $\text{Precision@k} \in [0, 1]$.
- Giá trị càng cao, hiệu suất mô hình càng tốt.
- Công thức tính Precision@k như bên dưới:

$$\text{NDCG@k} = \frac{\text{số nhãn liên quan trong top k các nhãn}}{k} \quad (17)$$

Các độ đo V

Độ đo NDCG@k: là một độ đo cải tiến của độ đo DCG mà đo lường chất lượng xếp hạng của các nhãn mô hình dự đoán ra được.

- $NDCG@k \in [0, 1]$.
- Giá trị càng cao, hiệu suất mô hình càng tốt.
- Tính NDCG@k theo công thức Eq.18:

$$NDCG@k = \frac{DCG}{iDCG} \quad (18)$$

Trong đó:

- $DCG = \sum_{i=1}^n \frac{relevance_i}{\log_2(i+1)}$ là thứ tự của nhãn ban đầu của một thực thể.
- $iDCG$ là thứ tự lí tưởng của nhãn thứ i của thực thể đó.

Cấu hình thực nghiệm I

Hai bài toán thực nghiệm chính bao gồm:

- Liên kết đồ thị (Liên kết thực thể và liên kết quan hệ).
- Phân loại thực thể.

Thực nghiệm trên GPU NVIDIA RTX 2080Ti và chạy 5 lần với các seed ngẫu nhiên.

Khởi tạo trọng số

Tối ưu hóa trọng số ngẫu nhiên của các đặc trưng đầu vào theo phân phối chuẩn cắt ngắn (truncated normal distribution).

Giả sử có một biến ngẫu nhiên $X \in (a, b)$ ($-\infty \leq a < b \leq \infty$) có một phân phối chuẩn với trung bình là μ và phương sai là σ^2 . Hàm mật độ xác suất của nó cho $a \leq x \leq b$ được tính như sau:

$$f(x; \mu, \sigma, a, b) = \frac{1}{\sigma} \frac{\phi\left(\frac{x-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)} \quad (19)$$

Trong đó:

- Hàm mật độ xác suất phân phối chuẩn: $\phi(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$
- Hàm phân phối tích lũy: $\Phi(x) = \frac{1}{2} \left(1 + \operatorname{erf}\left(\frac{x}{\sqrt{2}}\right)\right)$

Các siêu tham số của mô hình

Bảng 3 tóm tắt các siêu tham số tốt nhất của mô hình KE-GCN.

Bảng 3: Bảng tóm tắt các siêu tham số của mô hình KE-GCN.

task	dim	model	lr	alpha	beta	layer	epochs	seed	negs	gamma
Entity Alignment	200	QuatE	0.01	0.3	0.3	2	20.000	12.306	5	3.0
Relation Alignment	200	TransE	0.01	0.3	0.3	2	20.000	12.306	5	3.0
Entity Classification	32	TransE	0.01	0.3	0.3	2	1.000	12.306	5	3.0

Kết quả trên bài toán liên kết thực thể

Bảng 4: Kết quả của KE-GCN trên nhiệm vụ liên kết thực thể đồ thị tri thức trên tập dữ liệu DBP15K.

Models	DBP _{ZH-EN}			DBP _{JA-EN}			DBP _{FR-EN}		
	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
MTransE [2]	0.364	30.8	61.4	0.349	27.9	57.5	0.335	24.4	55.6
IPTransE [15]	0.516	40.6	73.5	0.474	36.7	69.3	0.451	33.3	68.5
JAPE [6]	0.490	41.2	74.5	0.476	36.3	68.5	0.430	32.4	66.7
AlignE [7]	0.581	47.2	79.2	0.563	44.8	78.9	0.599	48.1	82.4
GCN-Align [11]	0.549	41.3	74.4	0.546	39.9	74.5	0.532	37.3	74.5
MuGCN [1]	0.611	49.4	84.4	0.621	50.1	85.7	0.621	49.5	87.0
AliNet [8]	0.628	53.9	82.6	0.645	54.9	83.1	0.657	55.2	85.2
R-GCN [4]	0.564	46.3	73.4	0.571	47.1	75.4	0.570	46.9	75.8
W-GCN [5]	0.553	43.6	73.8	0.554	41.2	74.7	0.541	39.8	74.4
VR-GCN [12]	0.501	38.0	73.3	0.470	35.2	72.2	0.495	36.1	75.1
KBGAT [3]	0.582	48.0	77.3	0.582	47.6	77.7	0.593	47.4	80.9
CompGCN [10]	0.605	49.4	81.2	0.614	50.4	82.2	0.625	50.5	85.0
CompGCN [†]	0.628	52.8	81.1	0.629	52.8	81.5	0.641	52.6	85.4
KE-GCN	0.664	56.2	84.2	0.670	57.0	85.2	0.683	57.2	88.5

Trong đó CompGCN được đánh dấu với † chỉ ra sự kết hợp với các toán tử thành phần trong RotatE [9] và QuatE [14] trong khi đó CompGCN gốc chỉ có 3 toán tử là trừ, nhân và tương quan vòng tròn.

So sánh CompGCN với KE-GCN

Bảng 5: So sánh CompGCN với KE-GCN kết hợp với TransE và QuatE trên tập dữ liệu DBP_{ZH-EN}.

Models	MRR	Hits@1	Hits@10
CompGCN (TransE)	0.605 ± 0.003	49.4 ± 0.4	81.2 ± 0.3
KE-GCN (TransE)	0.648 ± 0.003	54.3 ± 0.3	83.4 ± 0.3
CompGCN (QuatE)	0.628 ± 0.003	52.8 ± 0.3	81.1 ± 0.4
KE-GCN (QuatE)	0.664 ± 0.004	56.2 ± 0.4	84.2 ± 0.4

Kết quả liên kết thực thể của KE-GCN trên DBP_{ZH-EN}

Bảng 6: Kết quả liên kết thực thể của KE-GCN trên tập dữ liệu DBP_{ZH-EN} mà kết hợp với các phương pháp nhúng đồ thị tri thức khác nhau.

KE-GCN (X)	MRR	Hits@1	Hits@10
X = TransE	0.648 ± 0.003	54.3 ± 0.3	83.4 ± 0.3
X = TransH	0.650 ± 0.003	54.3 ± 0.4	84.4 ± 0.3
X = DistMult	0.621 ± 0.003	52.0 ± 0.4	80.3 ± 0.4
X = TransD	0.635 ± 0.003	53.1 ± 0.3	82.7 ± 0.4
X = RotatE	0.653 ± 0.004	54.9 ± 0.4	83.8 ± 0.4
X = QuatE	0.664 ± 0.004	56.2 ± 0.4	84.2 ± 0.4

Kết quả liên kết thực thể của KE-GCN trên DBP_{JA-EN}

Bảng 7: Kết quả liên kết thực thể của KE-GCN trên tập dữ liệu DBP_{JA-EN} mà kết hợp với các phương pháp nhúng đồ thị tri thức khác nhau.

KE-GCN (X)	MRR	Hits@1	Hits@10
X = TransE	0.652 ± 0.003	54.8 ± 0.3	84.3 ± 0.3
X = TransH	0.654 ± 0.003	54.6 ± 0.4	85.5 ± 0.3
X = DistMult	0.622 ± 0.004	51.7 ± 0.5	81.2 ± 0.3
X = TransD	0.652 ± 0.001	54.5 ± 0.2	85.4 ± 0.2
X = RotatE	0.659 ± 0.002	55.7 ± 0.2	85.0 ± 0.3
X = QuatE	0.670 ± 0.001	57.0 ± 0.2	85.2 ± 0.3

Kết quả liên kết thực thể của KE-GCN trên DBP_{FR-EN}

Bảng 8: Kết quả liên kết thực thể của KE-GCN trên tập dữ liệu DBP_{FR-EN} mà kết hợp với các phương pháp nhúng đồ thị tri thức khác nhau.

KE-GCN (X)	MRR	Hits@1	Hits@10
X = TransE	0.669 ± 0.002	55.9 ± 0.2	87.5 ± 0.2
X = TransH	0.673 ± 0.002	56.1 ± 0.2	87.7 ± 0.2
X = DistMult	0.640 ± 0.002	52.4 ± 0.2	84.7 ± 0.2
X = TransD	0.660 ± 0.002	54.2 ± 0.2	87.6 ± 0.1
X = RotatE	0.673 ± 0.002	56.0 ± 0.3	88.2 ± 0.2
X = QuatE	0.683 ± 0.002	57.2 ± 0.3	88.5 ± 0.2

Kết quả liên kết quan hệ trên các tập dữ liệu DBP15K

Bảng 9: Kết quả liên kết quan hệ của KE-GCN và các mô hình khác trên các tập dữ liệu DBP15K mà trong đó các mô hình đều được kết hợp với cùng TransE.

Models	DBP_{ZH-EN}	DBP_{JA-EN}	DBP_{FR-EN}
VR-GCN	0.352 ± 0.006	0.335 ± 0.008	0.280 ± 0.017
KBGAT	0.341 ± 0.010	0.330 ± 0.013	0.274 ± 0.022
CompGCN	0.366 ± 0.007	0.347 ± 0.009	0.284 ± 0.015
KE-GCN	0.514 ± 0.006	0.466 ± 0.011	0.412 ± 0.021

Trung bình và độ lệch chuẩn của độ chính xác phân loại

Bảng 10: Trung bình và độ lệch chuẩn của độ chính xác phân loại của KE-GCN và các mô hình khác trên hai tập dữ liệu AM và WN cho nhiệm vụ phân loại đa lớp.

Models	AM	WN
GCN	86.2 \pm 1.4	53.4 \pm 0.2
R-GCN	89.3	55.1 \pm 0.6
W-GCN	90.2 \pm 0.9	54.2 \pm 0.5
KBGAT	85.7 \pm 1.7	53.7 \pm 1.1
CompGCN	90.6 \pm 0.2	55.9 \pm 0.4
KE-GCN	91.2 \pm 0.2	57.8 \pm 0.5

Trung bình và độ lệch chuẩn của P@1, P@5 và NDCG@5

Bảng 11: Trung bình và độ lệch chuẩn của Precision@1, Precision@5 và NDCG@5 trên tập dữ liệu FB15K cho nhiệm vụ phân loại đa nhãn.

Models	P@1	P@5	NDCG@5
GCN	86.1 \pm 0.3	69.0 \pm 0.3	82.7 \pm 0.2
R-GCN	91.7 \pm 0.6	73.0 \pm 0.4	89.5 \pm 0.6
W-GCN	91.2 \pm 0.6	72.8 \pm 0.3	88.6 \pm 0.5
KBGAT	90.5 \pm 0.7	72.4 \pm 0.8	87.5 \pm 0.8
CompGCN	92.5 \pm 0.1	74.0 \pm 0.3	90.1 \pm 0.2
KE-GCN	94.3 \pm 0.2	74.7 \pm 0.2	91.6 \pm 0.2

Độ chính xác phân loại thực thể của KE-GCN

Bảng 12: Độ chính xác phân loại thực thể của KE-GCN kết hợp với các phương pháp nhúng đồ thị tri thức khác nhau trên 2 tập dữ liệu AM và WN.

KE-GCN (X)	AM	WN
X = TransE	91.2 ± 0.2	57.8 ± 0.5
X = TransH	90.5 ± 0.3	57.4 ± 0.3
X = DistMult	89.5 ± 0.4	56.4 ± 0.1
X = TransD	90.1 ± 0.2	57.1 ± 0.2
X = RotatE	90.6 ± 0.4	56.6 ± 0.3
X = QuatE	91.0 ± 0.4	56.9 ± 0.3

Kết quả phân loại thực thể của KE-GCN




Bảng 13: Kết quả phân loại thực thể của KE-GCN kết hợp với các phương pháp nhúng đồ thị tri thức khác nhau trên tập dữ liệu FB15K.

KE-GCN (X)	P@1	P@5	NDCG@5
X = TransE	94.3 \pm 0.2	74.7 \pm 0.2	91.6 \pm 0.2
X = TransH	94.3 \pm 0.3	74.6 \pm 0.2	91.6 \pm 0.2
X = DistMult	94.2 \pm 0.1	72.4 \pm 0.4	89.9 \pm 0.3
X = TransD	93.9 \pm 0.4	74.5 \pm 0.2	91.3 \pm 0.2
X = RotatE	93.8 \pm 0.4	73.7 \pm 0.2	90.5 \pm 0.2
X = QuatE	94.9 \pm 0.4	74.2 \pm 0.3	91.6 \pm 0.2




Kết luận

- 1 Trong bài báo này, tác giả đề xuất ra KE-GCN mà tận dụng các phương pháp nhúng đồ thị tri thức trước đây kết hợp vào GCNs để mô hình hóa đồ thị đa quan hệ và sử dụng toán tử tích chập đồ thị để cập nhật các biểu diễn thực thể và quan hệ cùng lúc.
- 2 Thông qua các thí nghiệm cho 2 nhiệm vụ liên kết đồ thị tri thức và phân loại thực thể, KE-GCN đã cho thấy sự hiệu quả phương pháp đề xuất của mình.
- 3 Về công việc tương lai, tác giả đề xuất thay thế các mô hình đã kết hợp bằng các mô hình khác và nó cũng nên được cân nhắc để sử dụng KE-GCN trên các siêu đồ thị tri thức.




Tài liệu tham khảo I

-  Yixin Cao, Zhiyuan Liu, Chengjiang Li, Juanzi Li, and Tat-Seng Chua. Multi-channel graph neural network for entity alignment. [arXiv preprint arXiv:1908.09898, 2019.](#)
-  Muhao Chen, Yingtao Tian, Mohan Yang, and Carlo Zaniolo. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. [arXiv preprint arXiv:1611.03954, 2016.](#)
-  Deepak Nathani, Jatin Chauhan, Charu Sharma, and Manohar Kaul. Learning attention-based embeddings for relation prediction in knowledge graphs. [arXiv preprint arXiv:1906.01195, 2019.](#)




Tài liệu tham khảo II

-  Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling.
Modeling relational data with graph convolutional networks.
In [European semantic web conference](#), pages 593–607. Springer, 2018.
-  Chao Shang, Yun Tang, Jing Huang, Jinbo Bi, Xiaodong He, and Bowen Zhou.
End-to-end structure-aware convolutional networks for knowledge base completion.
In [Proceedings of the AAAI Conference on Artificial Intelligence](#), volume 33, pages 3060–3067, 2019.
-  Zequn Sun, Wei Hu, and Chengkai Li.
Cross-lingual entity alignment via joint attribute-preserving embedding.
In [International Semantic Web Conference](#), pages 628–644. Springer, 2017.




Tài liệu tham khảo III

-  Zequn Sun, Wei Hu, Qingheng Zhang, and Yuzhong Qu.
Bootstrapping entity alignment with knowledge graph embedding.
In [IJCAI](#), volume 18, pages 4396–4402, 2018.
-  Zequn Sun, Chengming Wang, Wei Hu, Muhao Chen, Jian Dai, Wei Zhang, and Yuzhong Qu.
Knowledge graph alignment network with gated multi-hop neighborhood aggregation.
In [Proceedings of the AAAI Conference on Artificial Intelligence](#), volume 34, pages 222–229, 2020.
-  Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang.
Rotate: Knowledge graph embedding by relational rotation in complex space.
[arXiv preprint arXiv:1902.10197](#), 2019.

Tài liệu tham khảo IV

-  Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. Composition-based multi-relational graph convolutional networks. [arXiv preprint arXiv:1911.03082](#), 2019.
-  Zhichun Wang, Qingsong Lv, Xiaohan Lan, and Yu Zhang. Cross-lingual knowledge graph alignment via graph convolutional networks. In [Proceedings of the 2018 conference on empirical methods in natural language processing](#), pages 349–357, 2018.
-  Rui Ye, Xin Li, Yujie Fang, Hongyu Zang, and Mingzhong Wang. A vectorized relational graph convolutional network for multi-relational network alignment. In [IJCAI](#), pages 4135–4141, 2019.

Tài liệu tham khảo V

-  Donghan Yu, Yiming Yang, Ruohong Zhang, and Yuexin Wu.
Knowledge embedding based graph convolutional network.
In [Proceedings of the Web Conference 2021, WWW '21](#), page 1619–1628,
New York, NY, USA, 2021. Association for Computing Machinery.
-  Shuai Zhang, Yi Tay, Lina Yao, and Qi Liu.
Quaternion knowledge graph embeddings.
[Advances in neural information processing systems](#), 32, 2019.
-  Hao Zhu, Ruobing Xie, Zhiyuan Liu, and Maosong Sun.
Iterative entity alignment via knowledge embeddings.
In [Proceedings of the International Joint Conference on Artificial Intelligence \(IJCAI\)](#), 2017.