BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM TP. HỒ CHÍ MINH

ĐỒ ÁN CUỐI KÌ

PROJECT 3

Thành phố Hồ Chí Minh – 2021

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM TP. HỒ CHÍ MINH

Trần Nguyễn Song Hiếu

ĐỒ ÁN CUỐI KÌ

Chuyên ngành: Khoa học máy tính
Mã số: KHMT831313

PROJECT 3

GIẢNG VIÊN:
TS. BÙI THANH HÙNG

Thành phố Hồ Chí Minh – 2021

LỜI CẢM ƠN

Tôi xin chân thành cảm ơn đến quý Thầy TS. Bùi Thanh Hùng– Giảng viên hướng dẫn đã truyền đạt những kiến thức nền tảng, những bài học lập trình, qua đó đã giúp đỡ tôi rất nhiều trong quá trình thực hiện Đồ án này.

ĐỒ ÁN ĐƯỢC HOÀN THÀNH
TẠI TRƯỜNG ĐẠI HỌC SƯ PHẠM TP. HỒ CHÍ MINH

Tôi xin cam đoan đây là sản phẩm đồ án của riêng tôi và được sự hướng dẫn của TS Bùi Thanh Hùng;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình. Trường Đại học Sư phạm TP. Hồ Chí Minh không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày tháng năm

Tác giả

(ký tên và ghi rõ họ tên)

Trần Nguyễn Song Hiếu

PHẦN ĐÁNH GIÁ CỦA GIẢNG VIÊN

_____

_____

_____

_____

_____

_____

_____

TP. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

TÓM TẮT

Bài toán dự đoán liên kết (Link Prediction) là bài toán dự đoán sự tồn tại liên kết giữa 2 thực thể trong một mạng. Đây là một bài toán quan trọng trong việc xử lý dữ liệu có cấu trúc mạng. Trong những nghiên cứu về bài toán này trước đây để đa số đều sử dụng đến các phương pháp heuristics với một hàm đánh giá từ đó tìm ra chỉ số tương đồng (similarity) giữa các thực thể từ đó dự đoán khả năng tồn tại liên kết giữa chúng. Tuy nhiên các phương pháp heuristics đều dựa trên những giả định về sự tồn tại liên kết giữa các thực thể, điều này dẫn đến khi những giả định này không chính xác thì kết quả của thuật toán bị giảm đi đáng kể. Thay vì sử dụng những phương pháp trên, phương pháp sử dụng mạng neural nhân tạo tỏ ra hứa hẹn hơn trong việc giải những bài toán về mạng này. Đã có một số công trình áp dụng cấu trúc đồ thị vào mạng neural nhân tạo như là mạng neural đệ quy RNN (Recursive Neural Networks) [1] và đặc biệt là sự ra đời của mô hình mạng neural đồ thị GNN (Graph Neural Networks) [2] đã thúc đẩy mạnh mẽ những nghiên cứu về việc ứng dụng của mô hình mạng neural đồ thị GNN vào việc giải bài toán dự đoán liên kết. Các kết quả thử nghiệm đã cho thấy, phương pháp sử dụng mạng neural đồ thị này là đặc biệt khả quan.

MỤC LỤC

DANH MỤC CÁC CHỮ VIẾT TẮT

GNN	Graph Neural Network
DGCNN	Dynamic Graph Convolution Neural Network
AUC	Area under the ROC Curve
RNN	Recursive Neural Network

DANH MỤC HÌNH VẼ

DANH MỤC CÁC BẢNG

DỰ ĐOÁN LIÊN KẾT - LINK PREDICTION

1.1 Giới thiệu về bài toán

Nhiều dữ liệu từ trong thế giới thực cho đến tự nhiên đều tồn tại dưới dạng liên kết, chẳng hạn như liên kết giữa các protein trong tế bào con người. Các liên kết này chứa rất nhiều thông tin như đặc tính của thực thể, cấu trúc mạng, hay sự phát triển của mạng.

Dự đoán liên kết là bài toán dự đoán sự tồn tại liên kết giữa các thực thể trong mạng có cấu trúc trong một khoảng thời gian. Các liên kết này gồm có 2 loại chính:

- Liên kết bị mất đi (missing) trong trường hợp dữ liệu có vấn đề cần phải sửa lỗi
- Liên kết mới trong tương lai (new) giữa 2 thực thể trong mạng.

Tương tác giữa các thực thể này chính là cơ sở của nhiều ứng dụng trong nhiều lĩnh vực như hóa học, sinh học, khoa học vật liệu, y khoa, hay mạng xã hội. Trong đó có thể kể đến như là gợi ý bạn bè trong mạng xã hội [3], đề xuất phim trong Netflix [4], hay là dự đoán chuỗi protein [5]. Việc xác định sự tồn tại liên kết giữa các thực thể này đòi hỏi một nỗ lực thực nghiệm đáng kể, và thậm chí có thể mất thời gian vô cùng lớn. Cho nên thay vì kiểm tra một cách mù quáng tất cả những liên kết này, việc sử dụng các phương pháp dự đoán liên kết có thể giúp các nhà khoa học, kĩ sư tập trung vào những liên kết có khả năng xảy ra nhất và do đó sẽ tiết kiệm đáng kể chi phí thử nghiệm.

Trong hơn một thập kỷ qua, đã có rất nhiều công trình được xuất bản về bài toán này. Bao gồm các công trình thuật toán, cải tiến, ứng dụng, thách thức và những hướng phát triển tương lai cho bài toán. Tuy nhiên, mặc dù đã có rất nhiều sự cố gắng, cũng như nhiều công trình nghiên cứu đã xuất bản nhưng vẫn chưa có một phương pháp nào có thể dự đoán liên kết tỏ ra nổi bật và mang lại hiệu quả như ý.

1.2 Phân tích yêu cầu bài toán

1.2.1 Yêu cầu của bài toán

Bài toán dự đoán liên kết có thể được mô tả như sau: xem xét một mạng có cấu trúc G = (V, E) trong đó V là tập hợp các đỉnh đồ thị và E là tập hợp cạnh của đồ thị. Nhiệm vụ của việc dự đoán liên kết là từ tập hợp V đỉnh, và một tập hợp con các liên kết đúng (các liên kết chắc chắn tồn tại), phải dự đoán được việc có tồn tại các liên kết giữa các thực thể chưa được quan sát hay không.

1.2.2 Các phương pháp giải quyết bài toán

1.2.2.1 Phương pháp Heuristic

Các hướng nghiên cứu ban đầu của bài toán này tập trung vào các phương pháp Heuristic. Các phương pháp này hoạt động dựa trên thuật toán đánh giá độ tương tự, trong đó mỗi cặp đỉnh x và y sẽ được gán một điểm s_{xy} được định nghĩa là độ giống nhau, hay độ gần gũi (similarity) giữa x và y. Tất cả các liên kết chưa được quan sát sẽ được đánh giá và xếp hạng dựa trên điểm s_{xy} của chúng, và những liên kết có kết nối với những đỉnh tương đồng sẽ có điểm số cao hơn.

Sự tương tự của các đỉnh có thể được đánh giá bằng các đặc trưng của bản thân các đỉnh. Hai đỉnh được coi là giống nhau nếu hai đỉnh đó có nhiều đặc trưng chung [6]. Tuy nhiên các đặc trưng của các đỉnh thường bị ẩn đi, và do đó các phương pháp này tập trung vào một chỉ số tương tự khác đó là tương tự về cấu trúc, chỉ dựa trên cấu trúc của mạng. Có rất nhiều chỉ số tương đồng và được phân loại theo nhiều cách khác nhau chẳng hạn như tương tự cục bộ chỉ xem xét các lân cận của các đỉnh đang dự đoán như một thước đo về độ tương tự tiêu biểu là chỉ số Common Neighbours (CN) [7]. Hay những chỉ số tương tự toàn mạng xem xét sự giống nhau về đường dẫn dựa trên cấu trúc của toàn mạng làm cơ sở tìm ra độ tương tự tiêu biểu là chỉ số Katz Index [8].

Hình 1 [9] là bảng so sánh độ chính xác của một số chỉ số tương tự cục bộ được đo lường bằng giá trị AUC. Mỗi con số thu được bằng cách lấy trung bình 10 lần tiến hành thực nghiệm. Thông tin chi tiết về các mạng được sử dụng có thể được tìm thấy tại [9].

Indices	PPI	NS	Grid	PB	INT	USAir
CN	0.889	**0.933**	**0.590**	0.925	**0.559**	0.937
Salton	0.869	0.911	0.585	0.874	0.552	0.898
Jaccard	0.888	**0.933**	**0.590**	0.882	**0.559**	0.901
Sørensen	0.888	**0.933**	**0.590**	0.881	**0.559**	0.902
HPI	0.868	0.911	0.585	0.852	0.552	0.857
HDI	0.888	**0.933**	**0.590**	0.877	**0.559**	0.895
LHN1	0.866	0.911	0.585	0.772	0.552	0.758
PA	0.828	0.623	0.446	0.907	0.464	0.886
AA	0.888	0.932	**0.590**	0.922	**0.559**	0.925
RA	**0.890**	**0.933**	**0.590**	**0.931**	**0.559**	**0.955**

Hình 1 Bảng so sánh độ chính xác của các chỉ số tương tự cục bộ

Hình 2 [9] là bảng so sánh độ chính xác của một số chỉ số tương tự toàn cục được đo lường bằng giá trị AUC. Mỗi con số thu được bằng cách lấy trung bình 10 lần tiến hành thực nghiệm. Thông tin chi tiết về các mạng được sử dụng có thể được tìm thấy tại [9].

AUC	PPI	NS	Grid	PB	INT	USAir
LP	0.970	**0.988**	0.697	**0.941**	0.943	**0.960**
LP*	0.970	**0.988**	0.697	0.939	0.941	0.959
Katz	**0.972**	**0.988**	**0.952**	0.936	**0.975**	0.956
LHN2	0.968	0.986	0.947	0.769	0.959	0.778

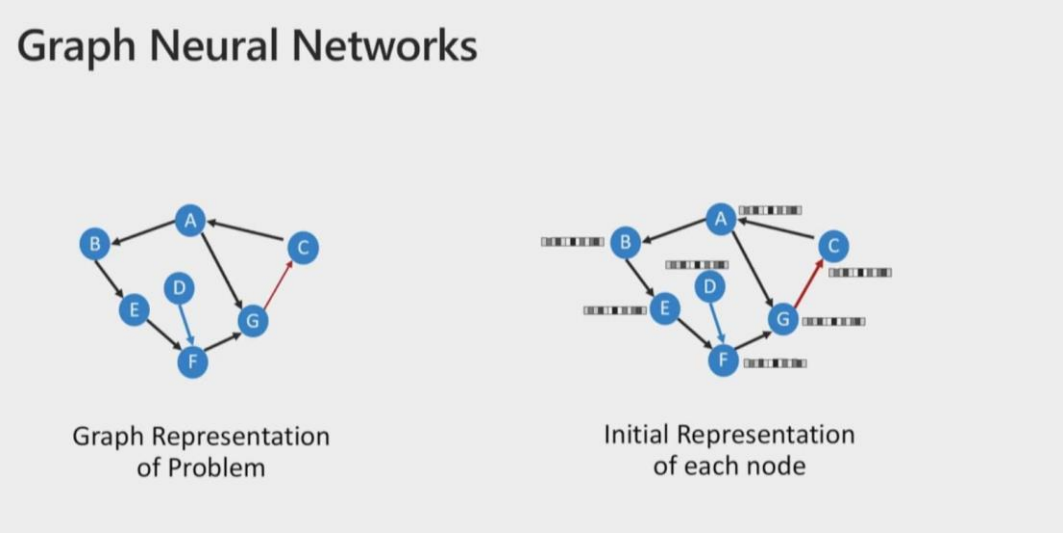
Hình 2 Bảng so sánh độ chính xác của các chỉ số tương tự toàn mạng

Như vậy ta thấy phương pháp này chủ yếu dựa vào các chỉ số đánh giá để xác định điểm số tương đồng giữa chúng. Tuy rằng đơn giản nhưng việc định nghĩa về độ giống nhau giữa các nút là một vấn đề không đơn giản. Trên thực tế, định nghĩa về độ giống nhau giữa các nút là một thách thức lớn. Các chỉ số tương đồng này có thể rất đơn giản nhưng cũng có thể rất phức tạp, có thể hoạt động tốt trên mạng này nhưng lại hoạt động kém trên mạng khác. Ví dụ chỉ số CN giả định rằng hai đỉnh có nhiều khả năng kết nối nếu nó có nhiều lân cận chung. Giả đỉnh này có vẻ đúng trong mạng gợi ý bạn bè, tuy nhiên lại không chính xác trong mạng Protein khi 2 Protein có lân cận giống nhau lại ít có khả năng liên kết hơn [10]. Vì vậy các phương pháp theo hướng này còn có rất nhiều hạn chế.

1.2.2.2 Phương pháp học máy

Trong những năm gần đây, mô hình mạng neural đồ thị (Graph Neural Network GNN) đã nổi lên như là một công cụ mạnh mẽ trong việc học dữ liệu có cấu trúc đồ thị [2] [11][12]. Nguyên nhân chính của lý do này là bởi vì học từ mạng neural đồ thị sẽ khiến cho việc học các đặc trưng của các nút và học cấu trúc của đồ thị thành một thể thống nhất. Bằng cách này, GNN đã thể hiện hiệu suất vượt trội trong các bài toán dự đoán liên kết [13][14][15].

Mạng Neural đồ thị (GNN) là một loại mạng chung cho các biểu diễn dưới dạng đồ thị. Bằng cách biểu diễn vấn đề dưới dạng đồ thị - mã hóa thông tin của các phần tử riêng lẻ dưới dạng các đỉnh và mối quan hệ giữa chúng dưới dạng các cạnh – GNN học được các mẫu từ trong đồ thị từ đó dự đoán ra mối quan hệ giữa các đỉnh chưa biết.



Hình 3 Biểu diễn đồ họa cho mạng GNN

Hình 3 là biểu diễn đồ họa cho một mạng GNN. Từ hình ta có thể thấy, mỗi đỉnh trong đồ thị đều chưa một vecto chứa đựng thông tin. Các vecto này không chỉ chứa thông tin của bản thân đỉnh đó, mà nó còn chứa thông tin của các đỉnh lân cận nó. Sau mỗi lần học các đỉnh sẽ càng ngày càng biết rõ hơn về các lân cận của nó thông qua việc cập nhật các thông tin của các lân cận và cuối cùng các đỉnh sẽ biết được vị trí của nó trên toàn bộ đồ thị. Như vậy có thể nói đầu ra của mạng GNN là 1 đồ thị mà trong đó mỗi đỉnh của đồ thị đều chứa đựng thông tin của nó và vị trí của nó trong đồ thị.

Tóm lại, từ những luận điểm nêu trên, Đồ án cho rằng phương pháp học máy sử dụng mạng GNN là một phương pháp vô cùng hứa hẹn. Khi xem xét tất cả các mạng đều được biểu diễn dưới dạng các đồ thị, kết hợp việc học đặc trưng của các phần tử riêng lẻ với vị trí của nó trong đồ thị sẽ cho ra kết quả chính xác hơn so với các phương pháp Heuristic.

1.2.3 Phương pháp đề xuất giải quyết bài toán

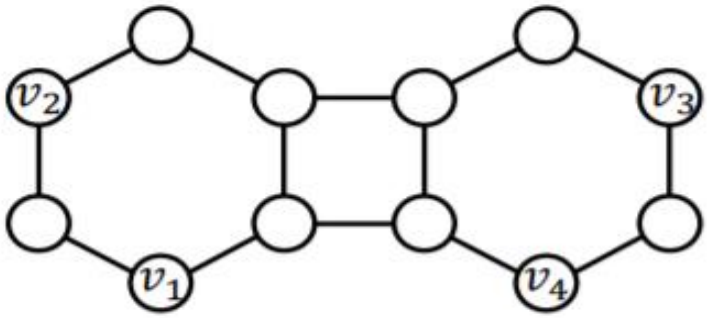
Như đã đề cập ở phần trên, Đồ án cho rằng phương pháp học máy sử dụng GNN là một phương pháp vô cùng hứa hẹn để giải quyết bài toán dự đoán liên kết. Hiện tại có 2 công trình dự đoán liên kết chính dựa trên GNN đó là:

- Graph Autoencoder (GAE) [13] : Trong phương pháp này, GNN lần đầu sẽ được áp dụng cho toàn bộ mạng. Sau đó, sẽ tổng hợp thông tin của 2 đỉnh nguồn và đích để đưa ra dự đoán về liên kết.
- SEAL [15]: Trong phương pháp này, một đồ thị con cục bộ sẽ được trích xuất xung quang liên kết cần dự đoán. Sau đó các đỉnh con trong đồ thị con này sẽ được gán các nhãn khác nhau tùy theo khoảng cách của chúng đến 2 đỉnh nguồn và đích. Cuối cùng mới áp dụng GNN lên đồ thị con này để tổng hợp thông tin và đưa ra dự đoán về liên kết.

Thoạt nhìn cả 2 công trình GAE và SEAL đều khá giống nhau khi đều sử dụng GNN để học các cấu trúc và đặc trưng của các đỉnh. Tuy nhiên, trong bài toán dự đoán liên kết phương pháp SEAL tỏ ra ưu việt hơn so với phương pháp GAE [16] vì 2 lý do cơ bản sau:

- Phương pháp SEAL trích xuất cục bộ 1 bộ phận đồ thị xung quanh liên kết cần dự đoán thay vì phải học cả cấu trúc của toàn bộ mang. Trong bài toán dự đoán liên kết, chúng ta có thể thấy những đỉnh nằm xa so với liên kết cần dự đoán thường ít ảnh hưởng hơn những đỉnh nằm gần nó. Vì vậy việc trích xuất đồ thị con sẽ tiết kiệm chi phí cho việc huấn luyện.
- Phương pháp SEAL kết hợp việc gán nhãn vào các đỉnh trong đồ thị con trước khi sử dụng GNN. Phương pháp này đặc biệt hữu hiệu bởi vì đôi khi GNN không phân biệt được một số liên kết có vai trò và có cấu trúc khác nhau. Xem Hình 4 ta thấy các đỉnh (v1,v4) và (v2,v3) là đẳng cấu vì vậy ta có thể dự đoán nếu liên kết (v1,v2) tồn tại thì (v3,v4) sẽ tồn tại vì

chúng là đẳng cấu với nhau. Tuy nhiên vì 2 đỉnh (v2,v3) là đẳng cấu nên GNN cũng sẽ dự đoán liên kết (v1,v2) tồn tại thì (v1,v3) sẽ tồn tại. Trên thực tế việc này là không đúng, ta có thể thấy được v1 gần với v2 hơn nhiều và chia sẻ nhiều hàng xóm chung hơn so với v1 và v3. Để giải quyết vấn đề này SEAL đã đề xuất phương pháp gán nhãn dựa vào khoảng cách giữa các đỉnh như là một đặc trưng thêm vào, như vậy GNN có thể học chính xác hơn.



Hình 4 Ví dụ về việc GNN không phân biệt được liên kết

Tổng hợp những luận điểm kể trên, Đồ án đề xuất phương pháp học máy sử dụng GNN và cụ thể là SEAL sẽ cho hiệu suất và độ chính xác cao hơn các phương pháp khác trong việc giải bài toán dự đoán liên kết. Do đó mục tiêu của Đồ án sẽ tập trung vào những mô hình học máy theo phương pháp SEAL để giải quyết bài toán dự đoán liên kết.

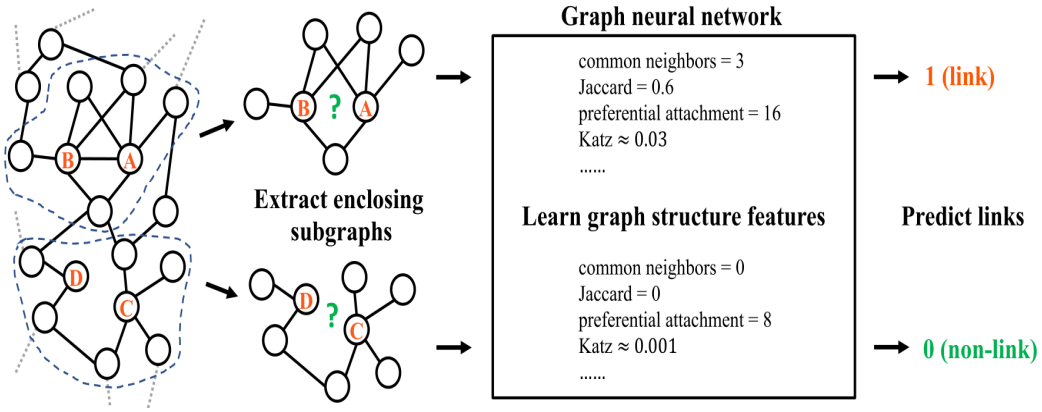
1.3 Phương pháp giải quyết bài toán

1.3.1 Mô hình tổng quát

Mô hình tổng quát của phương pháp SEAL để giải quyết bài toán được trình bày theo sơ đồ dưới đây. Trong mô hình này gồm có 3 phần chính:

- Phần 1: Biểu diễn dữ liệu dưới dạng đồ thị sau đó trích xuất các đồ thị con xung quanh liên kết cần dự đoán.

- Phần 2: Xây dựng ma trận đặc trưng của các đỉnh.
- Phần 3: Huấn luyện bằng các mô hình GNN (SAGE, DGCNN, GCN, …).



Hình 5 Mô hình tổng quát của phương pháp SEAL

1.3.2 Đặc trưng của mô hình đề xuất

1.3.2.1 Trích xuất các đồ thị con

Bước đầu tiên của phương pháp SEAL đó là trích xuất các đồ thị con xung quanh các đỉnh. Bao gồm các đỉnh chứa liên kết mà GNN cần phải huấn luyện (training) để xây dựng bộ dữ liệu huấn luyện và các đỉnh chứa liên kết mà GNN cần phải dự đoán để xây dựng bộ dữ liệu đào tạo. Mỗi cặp đỉnh có thể trích xuất 1 hoặc nhiều đồ thị con bao quanh cặp đỉnh đó để huấn luyện. Các đồ thị con này sẽ được biểu diễn dưới dạng ma trận kề (adjancency matrix) và coi như là input của GNN.

1.3.2.2 Xây dựng ma trận đặc trưng

Bước thứ 2 trong phương pháp SEAL chính là xây dựng ma trận thông tin đặc trưng của các đỉnh trong đồ thị con. Bước này rất quan trọng để có thể huấn luyện được mô hình GNN dự đoán liên kết hiệu quả. Ma trận thông tin đặc trưng trong phương pháp SEAL gồm 3 thành phần chính:

- Gán nhãn cho các đỉnh (structural node labels).
- Node Embedding.
- Đặc trưng của đỉnh (node attributes)

Gán nhãn cho các đỉnh:

Thành phần đầu tiên của ma trận thông tin đặc trưng là nhãn của các đỉnh. Mỗi nhãn trong đồ thị đều cần phải được gán nhãn trong đó hàm gán nhãn là 1 hàm f_l : $V \to N$ sẽ thực hiện gán một nhãn số nguyên f_l (i) cho đỉnh i. Mục đích của việc này là để đánh dấu các vai trò khác nhau của mỗi đỉnh trong đồ thị con:

- Các đỉnh x và y mà ta cần dự đoán liên kết hay còn gọi là các đỉnh trung tâm
- Các đỉnh có vị trí khác nhau so với vị trí trung tâm và có tầm quan trọng về mặt cấu trúc khác nhau so với liên kết cần dự đoán.

Việc gán nhãn cho các đỉnh là rất quan trọng trong việc huấn luyện GNN. Như đã đề cập ở phần trước, nếu chúng ta không gán nhãn thì GNN sẽ không thể phân biệt được tầm quan trọng của các nút trung tâm và có thể làm mất thông tin cấu trúc khi dự đoán sự tồn tại của liên kết.

Phương pháp SEAL đề xuất kĩ thuật gán nhãn Double-Radius Node Labeling (DRNL) [15] như sau:

- 2 đỉnh trung tâm x và y sẽ luôn được gán nhãn 1
- Các đỉnh i và j sẽ có cùng nhãn nếu d(i,x) = d(j,x) và d (i, y) = d (j, y). Tiêu chí này là vì vị trí của đỉnh bên trong đồ thị có thể biểu diễn bằng khoảng cách của nó đối với 2 nút trung tâm cụ thể là (d(i, x), d(i, y)). Do đó chúng ta để các nút nằm trong một quỹ đạo có cùng nhãn từ đó có thể phản ánh được vị trí tương đối và tầm quan trọng về cấu trúc của đỉnh đó các đỉnh trung tâm.

Node embedding and node attributes:

Bên cạnh nhãn cấu trúc của các đỉnh ma trận thông tin đặc trưng cũng chứa đựng thông tin về các đặc trưng tiềm ẩn hoặc rõ ràng của các đỉnh. Bằng việc kết hợp giữa Node embedding cùng với node attributes vào ma trận thông tin đặc trưng sẽ làm cho GNN học được chính xác hơn.

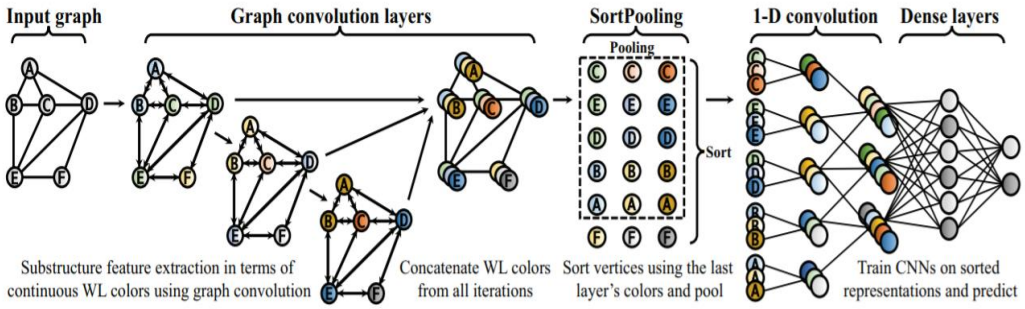
SEAL đề xuất phương pháp sinh ra Node embedding và node attributes như sau: giả sử chúng ta có một đồ thị G (V,E), và tập E_p là tập mẫu chứa các đỉnh có liên kết với $E_p \subseteq$ E và tập E_n là tập mẫu chứa các đỉnh không có liên kết trong đó $E \cap E_n = \emptyset$. Phương pháp SEAL đề xuất gọi là Negative Injection [15] theo đó thay vì các node embedding được tạo trực tiếp từ E thì chúng ta sẽ tạo ra tập E' = E $\cup$ E_n. Bằng cách này ma trận thông tin đặc trưng sẽ đồng thời có thông tin tồn tại liên kết giống nhau giữa tập mẫu chứa các đỉnh có liên kết và tập mẫu chứa các đỉnh không liên kết và sẽ học được chính xác hơn.

1.3.2.3 Phương pháp huấn luyện

Deep Graph Convolution Neural Network (DGCNN)

Mô hình DGCNN [17] bao gồm 3 giai đoạn:

- Lớp tích chập đồ thị (Graph Convolution Layers) trích xuất đặc trưng của các đồ thị con.
- SortPooling Layer sắp xếp các đặc điểm của các đỉnh trước đó và đồng bộ hóa kích thước của input
- Lớp tích chập truyền thống (Traditional Convolution Layers) và Dense Layers đọc các biểu diễn đồ thị kết quả sau đó chuyển về mô hình CNN truyền thống để tìm hiểu và đưa ra dự đoán liên kết.



Hình 6 Tổng quan mô hình DGCNN

Graph Convolution Layers

Cho 1 ma trận A với cấu trúc dữ liệu Adjacency Matrix và ma trận thông tin đặc trưng X, Graph Convolution Layer sẽ có cấu trúc như sau:

$$Z = f(\widetilde{D}^{-1}\tilde{A}X\widetilde{W}) \quad (1)$$

Trong đó:

- $\widetilde{D}$ là ma trận bậc
- $\tilde{A}$ = A + I (I là ma trận đơn vị)
- W là ma trận nhân chập
- f *là hàm lấy ngưỡng*

Sort Pooling Layer

DGCNN sử dụng một lớp Sort Pooling như là một cầu nối giữa lớp Graph Convolution Layers và Traditional Convolution Layers. Lớp này có tác dụng sắp xếp lại trạng thái của lớp Graph Convolution Layer cuối cùng thành một ma trận mô tả đặc trưng có thứ tự. Từ đó có thể nhanh chóng xác định được các đồ thị con đẳng cấu của đồ thị. Sau đó sử dụng phép toán max-k để đồng nhất nó về một kích thước trước khi đưa qua lớp Tranditinal Convolution Layers.

Traditional Convolution Layers

Đầu ra của lớp Sort Pooling là 1 tensor Z có kích thước k x $\sum_1^h c_t$ trong đó mỗi dòng đại diện cho số đỉnh và mỗi cột đại diện cho đặc trưng. Để thực hiện CNN, đầu

tiên cần phải chuẩn hóa kích thước tensor Z thành k x $\sum_1^h c_t$ x 1 vecto. Sau đó thực hiện nhân chập lớp 1-D convolution với size là $\sum_1^h c_t$ để áp dụng tuần tự bộ lọc lên tensor Z. Sau đó một số lớp Max-Pooling và 1-D convolution sẽ được thêm vào để tìm hiểu các đặc trưng cục bộ. Cuối cùng, là 1 lớp Fully-Connected để tổng hợp kết quả và đưa ra dự đoán.

1.4 Thực nghiệm

1.4.1 Dữ liệu

Đồ án tiến hành thực nghiệm trên bộ dữ liệu thực CORA[18]. Tập dữ liệu CORA bao gồm 2708 các bài báo khoa học về machine learning được phân loại thành 7 lớp. Trong đó mỗi đỉnh đại diện cho mỗi bài báo và các cạnh đại diện cho các trích dẫn giữa các bài báo với nhau. Mỗi bài báo trong tập dữ liệu CORA được mô tả bằng 1 vecto có giá trị 0-1 đại diện cho sự hiện diện của từ tương ứng trong từ điển. Từ điển bao gồm 1433 từ duy nhất. Tập dữ liệu CORA sẽ được xem như là tập dữ liệu có cấu trúc mạng có hướng.

1.4.2 Xử lý dữ liệu

Vì Đồ án sẽ sử dụng thư viện Pytorch để tiến hành thực nghiệm. Cho nên tập dữ liệu CORA phải được chuyển sang model Pytorch_Geometric. Ngoài ra Đồ án không thực hiện bất kì bước tiền xử lý dữ liệu nào khác.

1.4.3 Công nghệ sử dụng

Ngôn ngữ	Python 3.8
Thư viện	Pytorch
Môi trường	Google Colab

Bảng 1 Mô tả công nghệ sử dụng cho thực nghiệm bài toán dự đoán liên kết

1.4.4 Cách đánh giá

Area Under ROC Curve (AUC)

Đồ án sẽ sử dụng AUC làm độ đo đánh giá tính hiệu quả của mô hình đề xuất. AUC là một ước tính xác suất mà trong đó bộ phân loại sẽ xắp sếp trường hợp positive (có tồn tại liên kết) được chọn ngẫu nhiên cao hơn trường hợp negative (không tồn tại liên kết).

AUC có giá trị từ 0 cho đến 1. Một mô hình dự đoán sai hoàn toàn có giá trị AUC là 0 và mô hình dự đoán đúng hoàn toàn có giá trị AUC là 1.

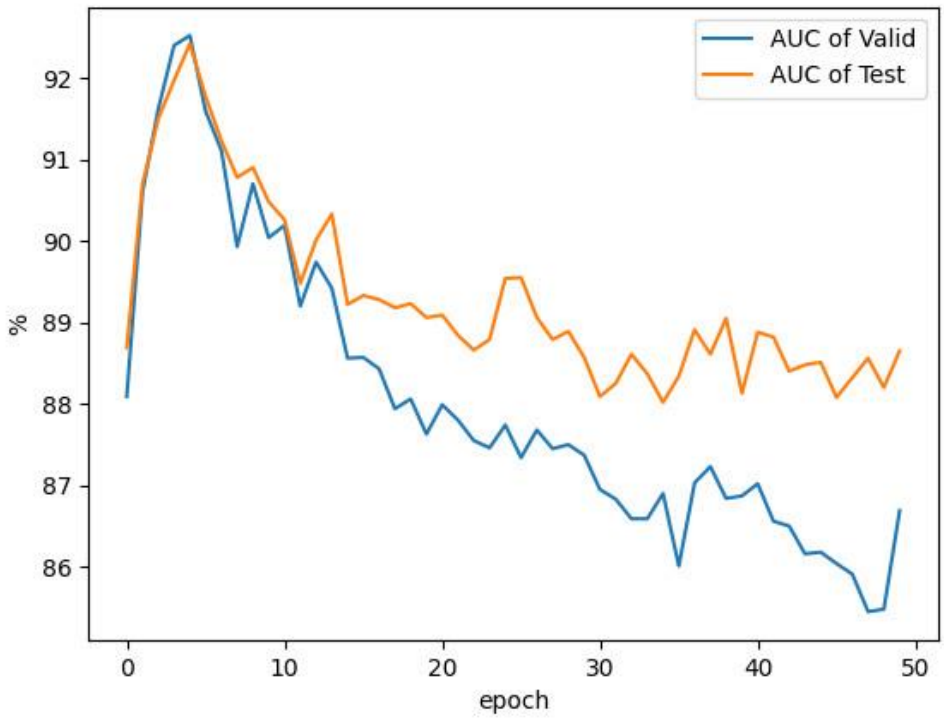
1.5 Kết quả đạt được

1.5.1 Tham số thực nghiệm

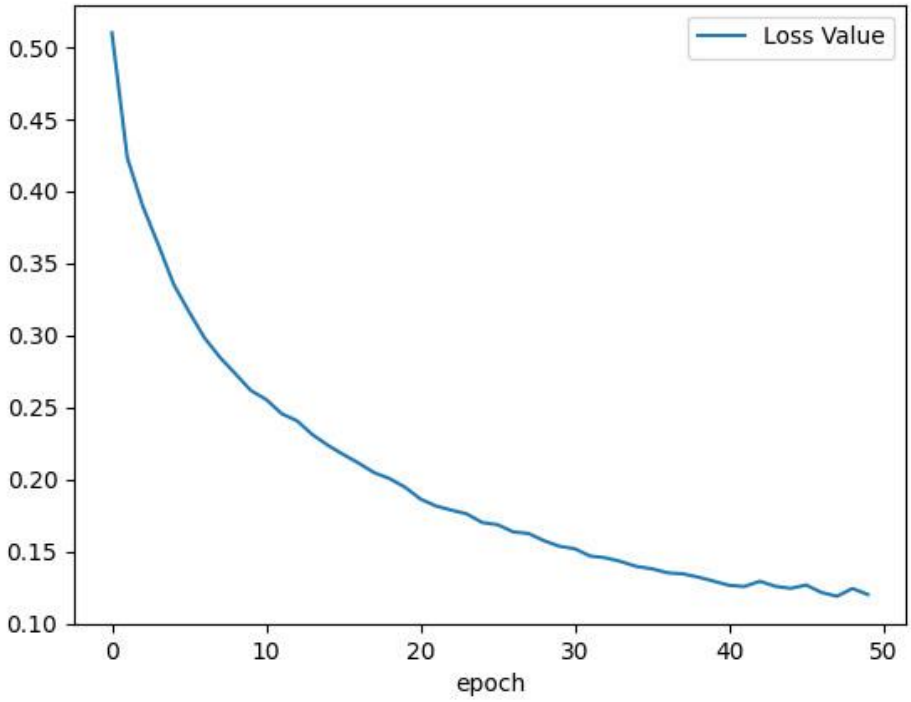
Đồ án sẽ tiến hành thực nghiệm trên tập dữ liệu Cora như đã trình bày ở phần trước. Tập dữ liệu sẽ được chia ra làm 8.5 phần cho huấn luyện mạng 0.5 phần để đánh giá trên tập huấn luyện và 1 phần để kiểm thử kết quả của mô hình.

Đồ án sẽ sử dụng mô hình DGCNN với 3 layers cho lớp Graph Convolutions. Với lớp Trainditional Convolutions, Đồ án sẽ sử dụng 2 lớp 1-D Convolution layers với 1 lớp MaxPooling ở giữa 2 lớp 1-D Convolution và cuối cùng là lớp FullyConnected sử dụng hàm Sigmoid. Hàm Loss Đồ án sẽ sử dụng hàm Binary Cross Entropy. Thuật toán tối ưu Đồ án sẽ sử dụng thuật toán Adam.

Đồ án sẽ sử dụng số Epochs là 50 sau đó chọn ra Epochs có chỉ số tốt nhất để tiến hành đánh giá kết quả.



Hình 7 Chỉ số AUC của tập Valid và tập Test trong quá trình huấn luyện



Hình 8 Chỉ số Loss trong qua trình huấn luyện

1.5.2 Kết quả đạt được

Bảng 3 so sánh kết quả thực nghiệm của mô hình Đồ án đề xuất với một số phương pháp Heuristics.

Phương pháp	**AUC**
SEAL với mô hình DGCNN	92.42
Common Neighbor (CN)	73.14
Adamic Adar (AA)	73.24

Bảng 2 So sánh kết quả thực nghiệm

Như đã thấy, mô hình mà Đồ án đề xuất có hiệu suất tốt hơn nhiều so với các phương pháp Heuristics sử dụng các chỉ số để đánh giá.

1.6 Kết luận

1.6.1 Kết quả đạt được

Về mặt lý thuyết, Đồ án đã tìm hiểu về các phương pháp giải quyết bài toán dự đoán liên kết, đồng thời Đồ án cũng đề xuất phương pháp học máy sử dụng phương pháp SEAL.

Về mặt thực nghiệm, Đồ án đã sử dụng tập dữ liệu CORA cho mô hình đề xuất cùng với 2 thuật toán Heuristics khác để so sánh. Kết quả thực nghiệm cho thấy mô hình đề xuất mang lại kết quả tốt hơn so với các phương pháp Heuristics.

1.6.2 Hạn chế

Các mô hình GNN nói chung đều nhạy cảm với dữ liệu bị nhiễu. Nếu dữ liệu của mô hình bị nhiễu (thêm/mất) cạnh có thể ảnh hưởng đến toàn bộ mô hình huấn luyện. Vì vậy nếu dữ liệu huấn luyện bị nhiễu do các cuộc tấn công hay người sử dụng có ý đồ xấu có thể ảnh hưởng nghiêm trọng đến tính chính xác của mô hình.

1.6.3 Hướng phát triển

Hầu hết các công trình gần đây đều hướng về việc phát triển sức mạnh của các mô hình GNN. Tuy nhiên lại không có nhiều công trình nghiên cứu phát triển khả năng biểu diễn các đỉnh, cạnh trong mô hình GNN. Phương pháp SEAL chỉ với việc thực hiện gán nhãn 1 cách đơn giản đã cải thiện đáng kể hiệu suất của mô hình GNN. Cho nên, trong tương lai, Đồ án dự kiến sẽ tiếp tục nghiên cứu hướng phát triển chính như sau

- Tìm hiểu các phương pháp biểu diễn các đặc trưng của các đỉnh và kết hợp nó vào mô hình GNN.
- Tìm hiểu các phương pháp khữ nhiễu để tiền xử lý dữ liệu trước khi đưa vào GNN

- Nghiên cứu thử nghiệm kết hợp các mô hình GNN để tìm ra mô hình tối ưu cho bài toán dự đoán liên kết

TÀI LIỆU THAM KHẢO

1. P. Frasconi, M. Gori, and A. Sperduti, (September 1998) “A General Framework for Adaptive Processing of Data Structures”, IEEE Transactions on Neural Networks, vol. 9, no. 5, pp. 768-786.
2. Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, Gabriele Monfardini, (January 2009) “The Graph Neural Network Model”, IEEE Transactions on Neural Networks, vol. 20, no. 1,.
3. Adamic, L. A. and Adar, E. Friends and neighbors on the web. Social networks, 25(3):211–230, 2003
4. Bennett, J., Lanning, S., et al. The netflix prize. In Proceedings of KDD cup and workshop, volume 2007, pp. 35. New York, 2007.
5. Qi, Y., Bar-Joseph, Z., and Klein-Seetharaman, J. Evaluation of different biological data and computational classification methods for use in protein interaction prediction. Proteins: Structure, Function, and Bioinformatics, 63(3): 490–500, 2006.
6. D. Lin, An information-theoretic definition of similarity, in Proceedings of the 15th International Conference on Machine Learning, Morgan Kaufman Publishers, San Francisco, 1998.
7. Barabasi, A.-L. and Albert, R. Emergence of scaling in ´ random networks. science, 286(5439):509–512, 1999.
8. L. Katz, A new status index derived from sociometric analysis, Psychmetrika 18 (1953) 39.
9. Lü, Linyuan, and Tao Zhou. "Link prediction in complex networks: A survey." *Physica A: statistical mechanics and its applications* 390.6 (2011): 1150-1170.
10. István A Kovács, Katja Luck, Kerstin Spirohn, Yang Wang, Carl Pollis, Sadie Schlabach, Wenting Bian, Dae-Kyum Kim, Nishka Kishore, Tong Hao, et al. Network-based prediction of protein interactions. bioRxiv, page 275529, 2018.

11. Bruna, J., Zaremba, W., Szlam, A., and LeCun, Y. Spectral networks and locally connected networks on graphs. arXiv preprint arXiv:1312.6203, 2013.
12. Li, Y., Tarlow, D., Brockschmidt, M., and Zemel, R. Gated graph sequence neural networks. arXiv preprint arXiv:1511.05493, 2015.
13. Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907, 2016a.
14. Chami, I., Ying, Z., Re, C., and Leskovec, J. Hyperbolic ´ graph convolutional neural networks. In Advances in neural information processing systems, pp. 4868–4879, 2019.
15. Zhang, M. and Chen, Y. Link prediction based on graph neural networks. In Advances in Neural Information Processing Systems, pp. 5165–5175, 2018.
16. Zhang, Muhan, et al. "Revisiting Graph Neural Networks for Link Prediction." *arXiv preprint arXiv:2010.16103* (2020).
17. Zhang, Muhan, et al. "An end-to-end deep learning architecture for graph classification." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. No. 1. 2018.
18. McCallum, A.; Nigam, K.; Rennie, J.; and Seymore, K. 2000. Automating the construction of internet portals with machine learning. Information Retrieval Journal.