

**BỘ GIÁO DỤC  
VÀ ĐÀO TẠO**

**VIỆN HÀN LÂM KHOA HỌC  
VÀ CÔNG NGHỆ VIỆT NAM**

**HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ**

---



**NGUYỄN TUẤN KHANG**

**NGHIÊN CỨU PHÁT TRIỂN MỘT SỐ KỸ THUẬT  
GỢI Ý MUA HÀNG THEO PHIÊN  
DỰA TRÊN MÔ HÌNH HỌC SÂU**

**TÓM TẮT LUẬN ÁN TIẾN SĨ KHOA HỌC MÁY TÍNH**

**Mã số: 9 48 01 01**

**Hà Nội - 2023**

**Công trình được hoàn thành tại: Học viện Khoa học và Công nghệ -  
Viện Hàn lâm Khoa học và Công nghệ Việt Nam**

**Người hướng dẫn khoa học 1:** TS. Nguyễn Phú Bình

Đại học Victoria Wellington (New Zealand)

**Người hướng dẫn khoa học 2:** PGS. TS. Nguyễn Việt Anh

Viện Công nghệ thông tin

Viện Hàn lâm Khoa học và Công nghệ Việt Nam

**Phản biện 1:**

**Phản biện 2:**

**Phản biện 3:**

Luận án được bảo vệ trước Hội đồng chấm luận án tiến sĩ, họp tại Học viện Khoa học và Công nghệ - Viện Hàn lâm Khoa học và Công nghệ Việt Nam vào hồi ... giờ, ngày ... tháng ... năm 2023.

**Có thể tìm hiểu luận án tại:**

- Thư viện Học viện Khoa học và Công nghệ
- Thư viện Quốc gia Việt Nam

# Mở đầu

## 1 Tính cấp thiết của đề tài

Trong bối cảnh thương mại điện tử và dịch vụ trực tuyến đang phát triển nhanh chóng [1], hệ thống gợi ý đã trở thành một công cụ quan trọng để nâng cao trải nghiệm khách hàng và thúc đẩy sự phát triển kinh doanh. Các mô hình gợi ý truyền thống như *phương pháp đề xuất dựa trên nội dung* [2] và *phương pháp lọc dựa trên cộng tác* [3] chủ yếu tập trung vào sở thích cá nhân dài hạn và bỏ qua các tương tác ngắn hạn.

Với động cơ nghiên cứu như vậy, phương pháp *hệ gợi ý dựa trên phiên* (*Session-based recommendation*) đã được đề xuất, và nhiệm vụ của chúng là dự đoán hành vi tiếp theo của người dùng dựa trên hành vi của *phiên làm việc* hiện tại. Với góc nhìn này, tác giả nhấn mạnh tính cấp thiết của việc nghiên cứu các mô hình gợi ý hành vi mua sắm của khách hàng dựa trên phiên và khám phá những khả năng mới mà chúng mang lại cho việc đẩy mạnh lĩnh vực hệ thống gợi ý nhằm dự báo hành vi khách hàng [4].

## 2 Mục tiêu của luận án

### Đặt vấn đề

Phân tích phiên làm việc của khách hàng để dự báo khả năng họ sẽ mua sản phẩm nào hoặc lựa chọn sản phẩm nào tiếp theo là một bài toán dự báo khá phổ biến trong ngành thương mại điện tử. Việc dự báo này giúp cho doanh nghiệp đưa ra các ý tưởng bán hàng phù hợp trong quá trình người dùng tương tác với hệ thống bán hàng của mình.

### Đối tượng nghiên cứu

Đối tượng nghiên cứu của luận án này là chuỗi hành vi nhấp chuột trong quá trình lựa chọn sản phẩm của khách hàng. Chuỗi hành vi nhấp chuột được ghi nhận trong một phiên mua hàng trên một hệ thống thương mại điện tử hoặc nền tảng mạng xã hội nào đó.

### Mục tiêu nghiên cứu

Mục tiêu của luận án này là nghiên cứu và đề xuất mô hình dự báo hành vi lựa chọn sản phẩm trong phiên làm việc hiện tại của khách hàng với hệ thống bán hàng. Cụ thể hơn, luận án này có một số mục tiêu nghiên cứu chính như sau:

- Nghiên cứu và đề xuất cách thức biểu diễn dữ liệu phiên làm việc.
- Nghiên cứu và đề xuất một số mô hình mạng nơ-ron học sâu và mạng nơ-ron đồ thị nhằm xây dựng mô hình dự báo hành vi mua hàng.
- Thực nghiệm một số phương án khác nhau và so sánh với một số mô hình cơ sở nhằm đánh giá tính hiệu quả của mô hình đề xuất.

### Phạm vi nghiên cứu

Phạm vi nghiên cứu tiếp cận với hai bài toán cụ thể sau:

- Bài toán 1 trả lời câu hỏi "*Với danh sách sản phẩm đang lựa chọn trong phiên tương tác hiện tại thì khả năng khách hàng có mua hàng không, và nếu mua thì khả năng họ chọn mặt hàng nào?*".
- Bài toán 2 mang tính tổng quát hơn khi trả lời câu hỏi "*Với danh sách sản phẩm đang lựa chọn trong phiên tương tác hiện tại thì khả năng khách hàng sẽ chọn những sản phẩm nào tiếp theo?*".

### 3 Phương pháp nghiên cứu

Bài toán 1 là bài toán nhị phân mua hàng đơn giản, luận án đề xuất hai mô hình mạng nơ-ron là mạng học rộng và sâu và mạng học máy biến đổi để phân tích phiên làm việc dưới dạng bảng (*tabular data*) gồm các thuộc tính có dữ liệu chuỗi số và danh mục (các đối tượng dữ liệu rời rạc) nhằm dự báo hành vi có mua hàng hay không của khách hàng. Hai mô hình mạng nơ-ron này khá đơn giản và phù hợp với các phiên dữ liệu dạng bảng, tuy nhiên điểm hạn chế là chỉ đánh giá dữ liệu theo từng phiên cụ thể (*intra-session*), mà không đánh giá được mối quan hệ giữa các phiên dữ liệu trong cả bộ dữ liệu lớn.

Với Bài toán 2 nhằm xây dựng hệ gợi ý *top-k*, phương pháp nghiên cứu cần cải tiến bằng cách tìm hiểu và đề xuất phương án biểu diễn dữ liệu phiên làm việc và đặc biệt hơn là khả năng thể hiện rõ mối quan hệ giữa hàng triệu phiên làm việc trong bộ dữ liệu thực tế, khái niệm này gọi là *inter-session* [5]. Đồ thị là hướng tiếp cận rất phù hợp nhằm biểu diễn dữ liệu phiên làm việc của hàng triệu khách hàng trong quá trình lựa chọn trên một tập các sản phẩm của một hệ thống nào đó [6]. Với góc độ mô hình kiến trúc, luận án nghiên cứu và đề xuất sử dụng mô hình nơ-ron đồ thị để xây dựng mô hình gợi ý cho Bài toán 2.

### 4 Bố cục luận án

Bố cục của luận án gồm phần Mở đầu và bốn chương nội dung, và phần Kết luận được mô tả ngắn gọn như sau:

- *"Mở đầu"*: Phần mở đầu trình bày tổng quan về bài toán nghiên cứu, tính cấp thiết và ý nghĩa khoa học thực tiễn của đề tài.
- *Chương 1 "Tổng quan về hệ gợi ý"*: Chương 1 trình bày về bài toán gợi ý mà nhiều hệ thống bán hàng thương mại điện tử hay các nền tảng mạng xã hội đang triển khai. Chương này nêu định nghĩa và phát biểu hai bài toán ứng với hai mục tiêu cụ thể của luận án được nêu ở phần Mở đầu, gồm Bài toán 1 là mô hình dự báo nhị phân có mua hàng hay không và Bài toán 2 là hệ gợi ý *top-k* dựa theo phiên làm việc hiện tại của khách hàng khi nhấp chuột lựa chọn sản phẩm trên hệ thống bán hàng.
- *Chương 2 "Đề xuất mô hình mạng nơ-ron học sâu giải bài toán mua hàng"*: Chương 2 giải quyết Bài toán 1 của luận án trả lời câu hỏi *"khách hàng có mua hàng trong phiên làm việc hiện tại không?"*. Chương này đề xuất hai mô hình mạng nơ-ron cụ thể gồm mạng nơ-ron rộng & sâu và mạng nơ-ron biến đổi để xây dựng mô hình dự báo mua hàng.
- *Chương 3 "Đề xuất mô hình mạng nơ-ron đồ thị giải bài toán top-k"*: Chương 3 giải quyết Bài toán 2 mang tính tổng quát của luận án là bài toán *top-k*. Chương này trình bày một số phương án thiết kế đồ thị để mô hình hóa thông tin đầu vào là phiên làm việc của khách hàng, gồm hai đồ thị đơn  $\mathcal{G}$ ,  $\mathcal{H}$  và một đồ thị đa quan hệ  $\mathcal{K}$ .
- *Chương 4 "Đề xuất phương pháp nhúng cho mô hình mạng nơ-ron đồ thị"*: Nhằm tiếp tục cải tiến mô hình GNN đề xuất ở chương 3, chương 4 đề xuất phép biến đổi trên đồ thị để nâng cao hiệu quả của mô hình. Tác giả đề xuất tối ưu hóa mô hình mạng nơ-ron đồ thị GNN bằng cách đề xuất mới một lớp nhúng đồ thị đặc biệt nhằm cải tiến mô hình dự báo *top-k*. Chương này thiết kế lớp nhúng phiên sử dụng phép biến đổi nhúng kết hợp bao gồm nhúng đỉnh, nhúng đồ thị và nhúng nhãn.
- *"Kết luận"*: Phần cuối cùng đưa ra các kết luận chung và nhận xét kết quả đạt được của luận án để giải thích rõ động cơ nghiên cứu và các bước cải tiến các mô hình.

# Chương 1 | Tổng quan về hệ gợi ý và một số mô hình mạng nơ-ron học sâu

## 1.1 Bài toán hệ gợi ý

### 1.1.1 Tổng quan về hệ gợi ý

Có khá nhiều hệ thống gợi ý khác nhau tùy theo ngữ cảnh bài toán [7]. Đơn giản nhất, hệ thống gợi ý dựa vào thông tin lịch sử hoặc sở thích của người dùng đã được lưu lại để tìm ra sản phẩm phù hợp nhất [8]. Hệ thống hoạt động kiểu này khá dễ hiểu nhưng lại gặp nhiều thách thức khi cần đưa ra gợi ý cho người dùng mới, trong khi hệ thống chưa ghi nhận được thông tin lịch sử gì từ họ. Một hình thức mới về hệ thống gợi ý chỉ dựa vào quá trình tương tác hiện tại của người dùng, gọi là phiên làm việc. Dựa vào thông tin phiên làm việc, hệ thống có thể đưa ra gợi ý cho người dùng chỉ sau vài ba chuỗi sự kiện tương tác của họ với hệ thống, mô hình này được gọi là hệ thống gợi ý dựa vào phiên làm việc [9].

### 1.1.2 Phân loại bài toán hệ gợi ý

Mỗi loại hệ thống gợi ý sử dụng các thuật toán và kỹ thuật khác nhau để tìm hiểu và phân tích dữ liệu, từ đó đưa ra các gợi ý phù hợp với sở thích và nhu cầu của người dùng.

- Hệ gợi ý dựa trên nội dung (*Content-Based Filtering*).
- Hệ gợi ý dựa trên sự cộng tác (*Collaborative Filtering*).
- Hệ gợi ý kết hợp (*Hybrid Recommendation Systems*).
- Hệ gợi ý dựa trên tri thức (*Knowledge-Based Recommendation Systems*).
- Hệ gợi ý dựa trên bối cảnh (*Context-Aware Recommendation Systems*).
- Hệ gợi ý dựa trên học tăng cường (*Reinforcement Learning-Based Recommendation Systems*).
- Hệ gợi ý dựa trên phiên làm việc (*Session-Based Recommendation Systems*).

## 1.2 Hai bài toán cơ sở

### 1.2.1 Định nghĩa phiên làm việc

**Định nghĩa 1.** *Phiên làm việc của khách hàng là một chuỗi các sự kiện nhấp chuột khi lựa chọn sản phẩm và được hệ thống ghi nhận dưới dạng véc-tơ  $s = \{id_1, id_2, \dots, id_c\}$  trong đó  $id_i$  là mã định danh sản phẩm,  $c$  là số lượt sản phẩm được nhấp chọn trong phiên làm việc  $s$  và cũng chính là độ dài của phiên làm việc đó.*

### 1.2.2 Bài toán 1 - Dự báo hành vi mua hàng

**Bài toán 1.** *Cho một chuỗi nhấp chuột có tính thứ tự theo thời gian được sinh ra từ một phiên làm việc của khách hàng khi lựa chọn sản phẩm, cần xây dựng mô hình dự báo xem liệu khách hàng có mua hàng trong phiên làm việc hiện tại không?*

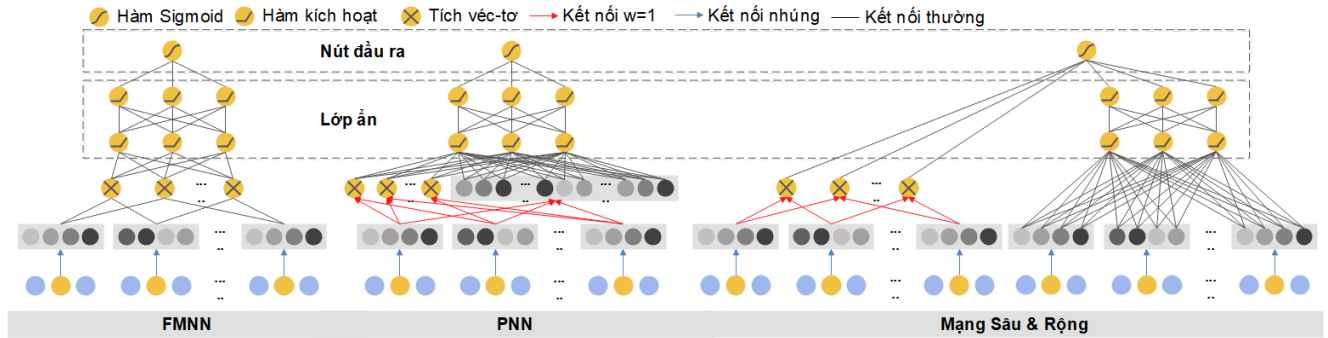
### 1.2.3 Bài toán 2 - Hệ gợi ý top - k

**Bài toán 2.** *Cho một chuỗi nhấp chuột có tính thứ tự theo thời gian được sinh ra từ một phiên làm việc của khách hàng khi lựa chọn sản phẩm, cần xây dựng mô hình gợi ý xem liệu khách hàng lựa chọn mặt hàng nào tiếp theo trong phiên làm việc hiện tại?*

## 1.3 Lý thuyết mạng nơ-ron học sâu

### 1.3.1 Mô hình mạng nơ-ron học sâu truyền thẳng

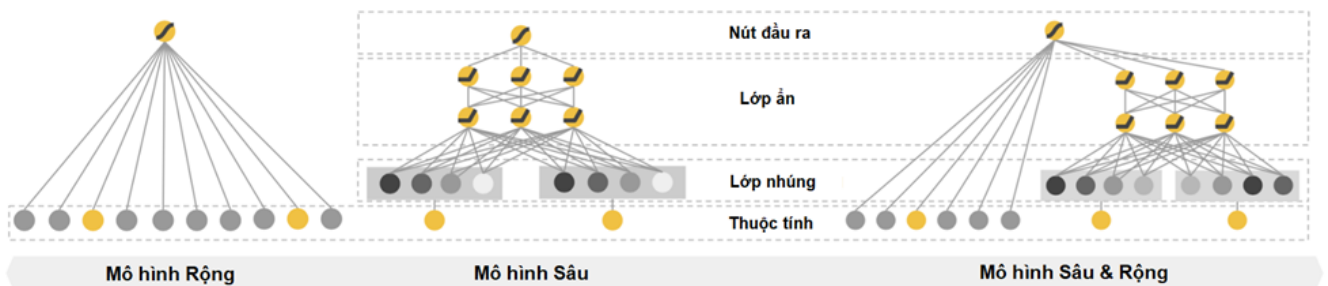
Phần này nghiên cứu một số mô hình cải tiến cụ thể của mạng nơ-ron truyền thẳng FNN nhằm cung cấp cái nhìn tổng quan hơn về kỹ thuật học sâu trong việc giải quyết Bài toán 1. Ba mô hình có tính chất tương tự như FNN nhưng khác nhau ở phương pháp tiền xử lý lớp nhúng trước khi vào lớp học sâu truyền thẳng. Các biến thể của mô hình FNN được minh họa ở Hình 1.1.



Hình 1.1: Một số mô hình nơ-ron sử dụng trong dự báo chuỗi nhấp chuột

### 1.3.2 Mô hình mạng nơ-ron rộng và sâu

Với hướng nghiên cứu ứng dụng mạng nơ-ron học sâu cho Bài toán 1, tác giả sử dụng mạng nơ-ron học rộng và sâu để phục vụ mục tiêu đề ra. Mô hình này được đề xuất năm 2016 bởi một nhóm làm việc trong Google [10].



Hình 1.2: Sơ đồ cấu trúc mạng nơ-ron rộng và sâu

Mô hình rộng và sâu là một mạng nơ-ron hỗn hợp với cấu trúc bao gồm hai nhánh được mô tả như sau:

#### Phần Rộng

Phần rộng là mô hình tuyến tính có dạng:

$$y = W^T x + b \quad (1.1)$$

Trường thuộc tính đầu vào bao gồm các thuộc tính thô và một số thuộc tính đặc biệt được tạo ra bằng phép biến đổi tích chéo như công thức 1.2:

$$\varphi_k(x) = \prod_{i=1}^d x_i^{c_{ki}}, c_{ki} \in \{0, 1\} \quad (1.2)$$

trong đó  $c_{ki}$  nhận giá trị 1 nếu thuộc tính thứ  $i$  nằm trong biến đổi thứ  $k$  của  $\varphi_k$ , và nhận giá trị 0 nếu ngược lại.

## Phần Sâu

Phần sâu là mạng nơ-ron học sâu truyền thẳng kết hợp kỹ thuật nhúng, lớp đầu tiên của mạng truyền thẳng là lớp nhúng thuộc tính. Đầu ra của lớp nhúng có dạng  $a^{(0)} = [e_1, e_2, \dots, e_m]$  với  $m$  là số trường thuộc tính, trong đó  $e_i$  là véc-tơ nhúng của trường thuộc tính thứ  $i$ . Các véc-tơ này kết hợp với các thuộc tính dạng số được truyền vào các lớp ẩn tiếp theo của mạng nơ-ron học sâu:

$$a^{l+1} = \sigma(W^{(l)}a^{(l)} + b^{(l)}) \quad (1.3)$$

trong đó  $\sigma$  là hàm kích hoạt, thường là hàm *ReLU* có dạng  $f(x) = x^+ = \max(0, x)$ ;  $W^{(l)}$ ,  $a^{(l)}$ , và  $b^{(l)}$  đầu ra và độ lệch của lớp nơ-ron thứ  $l$ .

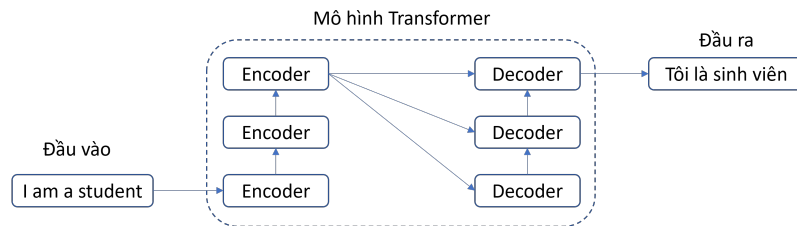
Quá trình học của mạng diễn ra đồng thời đối với cả hai phần để tạo ra kết quả cuối cùng của mô hình dự báo tổng hợp theo công thức 1.4

$$\hat{y} = Sigmoid(y_R + y_S) = \frac{1}{1 + e^{-(y_R + y_S)}} \quad (1.4)$$

trong đó  $\hat{y} \in (0, 1)$  là giá trị dự báo khả năng mua hàng,  $y_R$  là đầu ra của phần rộng và  $y_S$  là đầu ra của phần sâu.

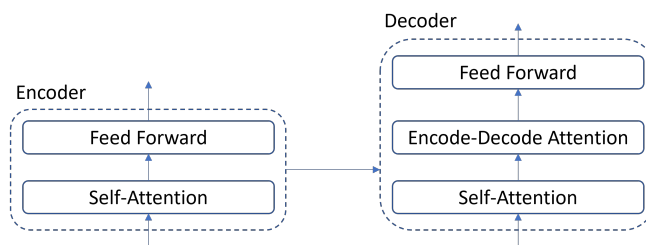
### 1.3.3 Mô hình mạng nơ-ron biến đổi

Mô hình biến đổi *Transformer* bao gồm hai mô-dun chính là khối mã hóa (*encoder*) và khối giải mã (*decoder*) được mô tả như Hình 1.3:



Hình 1.3: Mô hình minh họa kiến trúc *Transformer*

Kiến trúc *Transformer* tiếp cận khá giống với các mạng nơ-ron học sâu cơ bản như trình bày ở phần trên gồm W&DNN, FNN, PNN... vì nó cũng sử dụng kết hợp lớp nhúng và mạng nơ-ron truyền thẳng FNN. Tuy nhiên có 2 điểm khác là (1) kiến trúc *Transformer* sử dụng lớp nhúng theo cơ chế tự chú ý để biến đổi dữ liệu đầu vào theo dạng chuỗi tuần tự, (2) các khối này được xếp lớp với nhau để xử lý song song được nhiều thuộc tính khác nhau từ chuỗi dữ liệu đầu vào.



Hình 1.4: Các lớp chi tiết của kiến trúc *Transformer*



## 1.4 Lý thuyết mạng nơ-ron đồ thị

### 1.4.1 Định nghĩa về đồ thị

Theo định nghĩa cơ bản thì đồ thị là một tập các đối tượng gọi là đỉnh nối với nhau bởi các cạnh, mà ở đây cạnh thể hiện một quan hệ cụ thể nào đó giữa hai đỉnh. Tùy từng bài toán cụ thể mà cạnh có thể có hướng hoặc vô hướng, và tương ứng đồ thị khi đó cũng được gọi là có hướng hoặc vô hướng như một số phát biểu sau.

**Định nghĩa 2.** Một đồ thị đơn  $G$  gồm một tập không rỗng  $V$  mà các phần tử của nó gọi là các đỉnh và một tập  $E$  mà các phần tử của nó gọi là các cạnh, đó là các cặp không sắp xếp thứ tự các đỉnh phân biệt. Đồ thị này còn gọi là đồ thị vô hướng (undirected graph).

Biểu thức toán học biểu diễn đồ thị mô tả theo Công thức 1.5.

$$G = (V, E) \tag{1.5}$$

trong đó

- $V = \{v_1, v_2, \dots, v_n\}$  là tập các đỉnh của đồ thị, và số đỉnh  $n = |V|$ .
- $E = \{e_1, e_2, \dots, e_m\}$  là tập các cạnh của đồ thị, và số cạnh  $m = |E|$ .

**Định nghĩa 3.** Một đồ thị có hướng (directed graph)  $G = (V, E)$  gồm tập các đỉnh  $V$  và tập các cạnh  $E$  là các cặp có thứ tự của các phần tử thuộc  $V$ .

Với các dạng đồ thị phức tạp hơn, chúng có thể có nhiều loại cạnh khác nhau nối giữa các đỉnh. Đồ thị này được gọi là đồ thị đa quan hệ (multi-relational graphs) vì nó chứa nhiều tầng quan hệ khác nhau [11]. Với dạng đồ thị đa quan hệ, chúng ta cần thêm tham số để chỉ ra loại quan hệ (loại cạnh) giữa 2 đỉnh  $(u, v)$  thông qua một hàm  $f$  nào đó sao cho  $f(e) = (u, v)$ .

**Định nghĩa 4.** Một đồ thị đa quan hệ vô hướng  $G = (V, E)$  gồm tập các đỉnh  $V$ , một tập các cạnh  $E$  và một hàm  $f$  từ  $E$  tới  $\{\{u, v\} | u, v \in V, u \neq v\}$ . Các cạnh  $e_1$  và  $e_2$  được gọi là cạnh song song hay cạnh bội nếu  $f(e_1) = f(e_2)$ .

**Định nghĩa 5.** Một đồ thị đa quan hệ có hướng  $G = (V, E)$  gồm tập các đỉnh  $V$ , một tập các cạnh  $E$  và một hàm  $f$  từ  $E$  tới  $\{\{u, v\} | u, v \in V\}$ . Các cạnh  $e_1$  và  $e_2$  được gọi là cạnh song song hay cạnh bội nếu  $f(e_1) = f(e_2)$ .

**Định nghĩa 6. (đỉnh kề)** Hai đỉnh  $u$  và  $v$  trong một đồ thị vô hướng  $G$  được gọi là liền kề nếu  $\{u, v\}$  là một cạnh của đồ thị  $G$ . Nếu  $e = \{u, v\}$  thì  $e$  gọi là cạnh liền thuộc với các đỉnh  $u$  và  $v$ . Cạnh  $e$  còn được gọi là cạnh nối các đỉnh  $u$  và  $v$ , và các đỉnh  $u$  và  $v$  gọi là các điểm đầu mút của cạnh  $\{u, v\}$ .

**Định nghĩa 7.** Khi  $e = \{u, v\}$  là cạnh của đồ thị có hướng  $G$  thì  $u$  được gọi là đỉnh nối tới  $v$  và  $v$  được gọi là đỉnh được nối từ  $u$ . Đỉnh  $u$  gọi là đỉnh đầu, đỉnh  $v$  gọi là đỉnh cuối của cạnh  $\{u, v\}$ .

**Định nghĩa 8. (bậc của đỉnh)** Bậc của một đỉnh trong đồ thị vô hướng là số các cạnh liền thuộc với nó. Ký hiệu bậc của đỉnh  $v$  là  $\text{deg}(v)$ .

**Định nghĩa 9.** Với đồ thị có hướng, bậc vào (incoming degree) của đỉnh  $v$  ký hiệu là  $\text{deg}^-(v)$  là số các cạnh có đỉnh cuối là  $v$ . Bậc ra (outgoing degree) của đỉnh  $v$  ký hiệu là  $\text{deg}^+(v)$  là số các cạnh có đỉnh đầu là  $v$ .

**Định nghĩa 10. (đường đi)** Một đường  $P$  đi từ đỉnh  $v_1$  tới đỉnh  $v_k$  là tập các đỉnh  $\{v_1, v_2, \dots, v_k\}$  sao cho tồn tại  $(v_i, v_{i+1}) \in E, \forall i : 1 \leq i < k$ . Đường đi  $P$  có độ dài là  $P(v_1, v_k) = k - 1$  do không tính đỉnh khởi đầu  $v_1$ , độ dài này cũng chính là số lượng cạnh chứa trong đường đi đó.



## 1.4.2 Biểu diễn đồ thị

### a. Danh sách kề

Danh sách kề (*adjacency list*) là danh sách biểu diễn tất cả các cạnh của một đồ thị. Nếu đồ thị vô hướng, mỗi phần tử của danh sách là một cặp hai đỉnh là hai đầu của cạnh tương ứng. Nếu đồ thị có hướng, mỗi phần tử là một cặp có thứ tự gồm hai đỉnh là đỉnh đầu và đỉnh cuối của cung tương ứng.

Hình 1.5 minh họa cách biểu diễn đồ thị bằng danh sách kề



Hình 1.5: Biểu diễn đồ thị bằng danh sách kề

### b. Ma trận kề

Khi biểu diễn đồ thị sử dụng danh sách kề thì việc xây dựng thuật toán có thể sẽ rất cồng kềnh nếu đồ thị có nhiều cạnh, để đơn giản hóa việc tính toán ta có thể biểu diễn đồ thị bằng ma trận kề (*adjacency matrix*).

Giả sử  $G = (V, E)$  là một đồ thị đơn có  $n$  đỉnh, ta có thể biểu diễn đồ thị bằng một ma trận  $A_G = [a_{ij}] \in \mathbb{R}^{n \times n}$ , ma trận này còn được gọi là ma trận kề:

- $a_{ij} = 1$  nếu  $\{v_i, v_j\} \in E$ .
- $a_{ij} = 0$  nếu không có cạnh nối đỉnh  $v_i$  với đỉnh  $v_j$ .
- Quy ước  $a_{ii} = 0$  với  $\forall_i$ .

Với trường hợp biểu diễn đồ thị có trọng số, thì giá trị  $a_{ij} = w(i, j)$  là trọng số của cạnh của hai đỉnh liền kề  $v_i$  nối tới  $v_j$ .

## 1.4.3 Mô hình mạng nơ-ron đồ thị

Mô hình mạng nơ-ron đồ thị được giới thiệu đầu tiên vào năm 2005 [12], GNN là một loại mạng nơ-ron hoạt động trực tiếp trên cấu trúc đồ thị. Với việc sử dụng nơ-ron như là các nút trong cấu trúc mạng, từng nút sẽ chứa thông tin của riêng nó và thu thập thêm các thông tin từ các nút lân cận thể hiện mối tương quan giữa chúng trong đồ thị. Các nút này sẽ được bố cục và kết hợp với nhau theo một kiến trúc mô hình cụ thể nào đó để từ đó đưa ra dự đoán hoặc phân loại kết quả. Thông thường cái bài toán GNN sẽ tập trung giải quyết một số vấn đề như sau [13]:

- Phân loại nút (*Node classification*).
- Dự đoán kết nối (*Link prediction*).
- Phát hiện cụm (*Clustering detection*).
- Phân loại đồ thị (*Graph classification*).

## 1.5 Phép biến đổi nhúng

### 1.5.1 Khái niệm phép biến đổi nhúng

Trong lĩnh vực học máy, phép biến đổi nhúng (*embedding*) là một kỹ thuật được sử dụng để biến đổi các dữ liệu thuộc tính rời rạc, chẳng hạn như từ hay danh mục, thành dạng các véc-tơ liên tục trong một không gian chiều thấp hơn [14]. Như vậy, phép biến đổi nhúng ánh xạ mỗi biến rời rạc thành một véc-tơ số thực, có thể được sử dụng làm đầu vào cho một mạng nơ-ron.

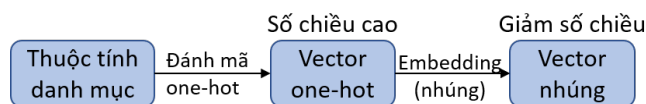
Các phép biến đổi nhúng có thể sử dụng với nhiều loại dữ liệu khác nhau ví dụ như dữ liệu rời rạc, văn bản, dữ liệu chuỗi thời gian (*time series*), hình ảnh hay đồ thị. Phần tiếp theo của luận án sẽ trình bày một số kỹ thuật nhúng được sử dụng trong các chương tiếp theo của luận án, bao gồm:

- Kỹ thuật nhúng dữ liệu có dạng rời rạc sử dụng cho mạng nơ-ron học sâu truyền thẳng được đề xuất trong chương 2 và chương 3.
- Kỹ thuật nhúng dữ liệu có dạng chuỗi tuần tự (ví dụ như câu văn bản) sử dụng cho mạng nơ-ron biến đổi được đề xuất trong chương 2, hoặc dữ liệu chuỗi thời gian sử dụng cho mạng nơ-ron hồi quy.
- Kỹ thuật nhúng dữ liệu có dạng đồ thị sử dụng cho mạng nơ-ron đồ thị được đề xuất trong chương 4.

### 1.5.2 Phép biến đổi nhúng với dữ liệu rời rạc

Hai loại dữ liệu phổ biến nhất là dữ liệu *liên tục* và *rời rạc*, được xếp vào dạng dữ liệu dạng bảng (*tabular*) [15]. Dữ liệu liên tục được biểu diễn bởi các số thực, trong khi đó giá trị rời rạc như trường *danh mục sản phẩm* được biểu diễn bởi các nhãn chữ hoặc nhãn số. Thực tế việc đánh nhãn chỉ là cách biểu diễn thuận tiện cho bộ từ điển giá trị của một thuộc tính rời rạc nào đó, các nhãn này thực sự không mang giá trị có ích nào như các thuộc tính liên tục. Loại dữ liệu này được gọi là thuộc tính danh mục, chúng có thể có thứ tự hoặc không.

Điểm lưu ý là mô hình nơ-ron không phù hợp khi xử lý loại dữ liệu danh mục vì tính rời rạc của chúng [16], do đó các thuộc tính rời rạc cần phải được biến đổi sang dạng véc-tơ để thể hiện được tính liên tục trong miền giá trị của chúng. Các đối véc-tơ sau khi biến đổi sẽ giúp cải thiện khả năng học của các mô hình nơ-ron trong việc ghi nhớ sự tương quan giữa các giá trị rời rạc của từng thuộc tính cũng như mối tương tác giữa các thuộc tính. Phép biến đổi gồm hai bước như Hình 1.6.



Hình 1.6: Biến đổi thuộc tính danh mục thành véc-tơ nhúng

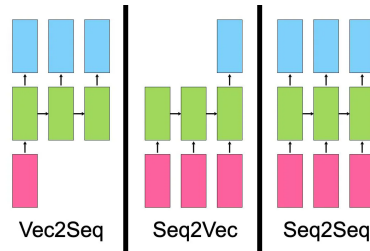
Phép biến đổi nhúng thuộc tính (*feature embedding*) là kỹ thuật xây dựng véc-tơ đặc trưng cho một thuộc tính danh mục trong không gian đa chiều thuộc miền giá trị của nó [17]. Kỹ thuật này tìm cách biểu diễn và sắp xếp lại các phần tử có mức ảnh hưởng giống nhau ở gần nhau để (1) tìm ra tính liên tục của dữ liệu trong không gian nhúng, và (2) nắm bắt được mối quan hệ giữa các danh mục rời rạc của thuộc tính từ đó giúp mạng nơ-ron học sâu có thể học hiệu quả hơn. Với kỹ thuật này, véc-tơ nhúng sau khi biến đổi có số chiều thấp hơn và các thành phần của véc-tơ là số thực thay vì chỉ là giá trị 0 và 1 như véc-tơ one-hot.

### 1.5.3 Phép biến đổi nhúng với dữ liệu theo chuỗi tuần tự

Các mô hình mạng nơ-ron học sâu cơ bản (ví dụ như mạng nơ-ron truyền thẳng) có thể xử lý tốt dữ liệu dạng số và danh mục tuy nhiên nó lại không xử lý được các dạng dữ liệu chuỗi tuần tự (*sequential data*) ví dụ như dữ liệu chuỗi từ trong câu hoặc chuỗi thời gian. Như vậy, mô hình mạng nơ-ron khi xử lý văn bản không chỉ tính toán từng từ trong câu mà còn phải xem xét cách các từ đó xuất hiện theo thứ tự và liên quan đến nhau như thế nào. Ý nghĩa của các từ có thể thay đổi tùy thuộc vào các từ khác xuất hiện trước và sau chúng trong câu.

#### a. Chuỗi dữ liệu tuần tự dạng văn bản

Có ba kỹ thuật biến đổi kết hợp với phép nhúng như Hình 1.7 sử dụng mạng nơ-ron để xử lý dữ liệu chuỗi tuần tự:



Hình 1.7: Các kỹ thuật xử lý dữ liệu chuỗi dữ liệu tuần tự cho mạng nơ-ron

#### b. Chuỗi dữ liệu tuần tự dạng thời gian

Dữ liệu chuỗi thời gian cũng khá phổ biến ví dụ như bảng giá chứng khoán, tín hiệu điện tâm đồ hay phức tạp hơn khi thu thập tín hiệu đa biến (*multivariate time series*) từ các thiết bị IoT hay điện thoại thông minh. Với dạng dữ liệu chuỗi thời gian này thì cần các mô hình mạng nơ-ron phù hợp hơn ví dụ như mạng nơ-ron tích chập CNN [18] hoặc mạng nơ-ron hồi quy RNN. Đặc biệt khi cần phải làm việc với dữ liệu chuỗi thời gian đa biến, kỹ thuật phân tích thành phần chính PCA (*Principal Component Analysis*), dù không hoàn toàn được tính là một kỹ thuật nhúng, là một phương pháp rất phổ biến [19] để thực hiện việc phân tích và giảm chiều dữ liệu đa biến này.

### 1.5.4 Phép biến đổi nhúng với dữ liệu đồ thị

Kỹ thuật nhúng với dữ liệu đồ thị, gọi là phép nhúng đồ thị, là một kỹ thuật cho phép biểu diễn một đồ thị dưới dạng các véc-tơ có số chiều cao. Điều này cho phép sử dụng các thuật toán học máy hoặc mạng nơ-ron phù hợp để xử lý và phân tích các thông tin trong đồ thị, chẳng hạn như phân loại nút, dự đoán liên kết và phân cụm đồ thị.

Có nhiều cách thực hiện phép nhúng đồ thị, ví dụ như phương pháp *random walk* [20], *deep walk* [21], phân tích ma trận (*matrix factorization*) [22] và một số phương pháp khác dựa trên mạng nơ-ron học sâu. Kết quả của phép nhúng đồ thị có rất nhiều ứng dụng trong thực tế, ví dụ như phân tích mạng xã hội hay xây dựng hệ thống gợi ý. Ví dụ, nó có thể được sử dụng để phân cụm các người dùng tương tự trong mạng xã hội hoặc để đề xuất các sản phẩm tương tự cho khách hàng trong quá trình mua hàng.

# Chương 2 | Đề xuất mô hình mạng nơ-ron học sâu cho bài toán mua hàng

Chương 2 trình bày phương pháp tiếp cận giải Bài toán 1, đây là bài toán nhị phân dự báo khách hàng có mua hàng trong phiên làm việc hiện tại hay không. Chương này đề xuất sử dụng hai mạng nơ-ron học sâu, gồm mạng nơ-ron rộng & sâu và mạng nơ-ron biến đổi, để học dữ liệu dạng chuỗi biểu diễn thông tin phiên làm việc của khách hàng.

## 2.1 Phát biểu bài toán

Giả sử tập dữ liệu huấn luyện bao gồm  $n$  mẫu  $(\mathcal{X}, y)$ , trong đó  $\mathcal{X}$  là chuỗi dữ liệu được ghi nhận với  $m$  trường thuộc tính liên quan tới khách hàng và sản phẩm, và  $y \in (0, 1)$  là nhãn tương ứng với hành vi mua của khách hàng ( $y = 1$  nếu khách hàng mua sản phẩm, và  $y = 0$  trong trường hợp ngược lại). Như vậy, Bài toán 1 là xây dựng mô hình dự báo  $y \approx \hat{y} = f(x)$  nhằm ước tính xác suất của người dùng có mua hàng dựa vào chuỗi dữ liệu đầu vào hay không.

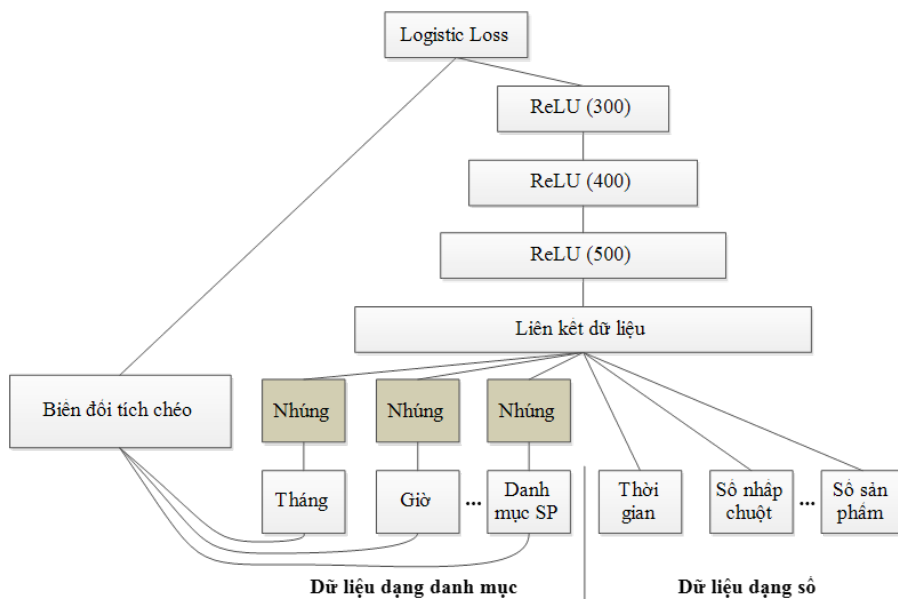
## 2.2 Các mô hình đề xuất

### 2.2.1 Mạng nơ-ron học rộng và sâu

Mô hình mạng học rộng và sâu được đề xuất với thiết kế kiến trúc như sau:

- **Phần Rộng:** gồm 2 lớp truyền thẳng, với lớp đầu ra có một nơ-ron và lớp đầu vào có số nơ-ron xác định bằng:  $N = N_{cat} + N_{num}$ , trong đó,  $N$  là số nơ-ron của lớp đầu vào,  $N_{cat}$  là số trường thuộc tính dạng danh mục và  $N_{num}$  là số cặp tương tác chéo của các trường thuộc tính dạng danh mục.
- **Phần Sâu:** gồm 6 lớp truyền thẳng, trong đó có 1 lớp đầu vào với số nơ-ron bằng số trường thuộc tính, 1 lớp nhúng, 3 lớp ẩn với số nơ-ron lần lượt được lấy bằng  $400 - 400 - 400$  và 1 lớp đầu ra với 1 nơ-ron. Các nơ-ron ẩn sử dụng hàm kích hoạt *ReLU*, nơ-ron đầu ra sử dụng hàm kích hoạt *sigmoid*.

Cấu trúc mô hình được sử dụng thể hiện ở Hình 2.1.



Hình 2.1: Cấu trúc mô hình rộng và sâu sử dụng trong dự báo chuỗi nhấp chuột

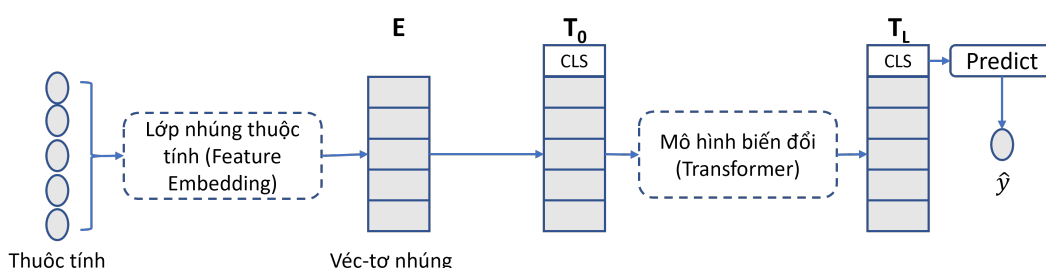
Mô hình mạng đề xuất này có các điểm cải tiến như sau:

- Đề xuất sử dụng phép nhúng với các thuộc tính dạng danh mục và liên kết dữ liệu với các thuộc tính còn lại nhằm tạo ra một véc-tơ nhúng đặc trưng cho phiên làm việc.
- Xây dựng kiến trúc mạng với một số lớp nơ-ron ở nhánh học sâu (nhánh FNN).
- Thực hiện phép biến đổi tích chéo giữa một số cặp thuộc tính nhằm tìm ra các tương tác ẩn của các trường thuộc tính.

Việc kết hợp đồng thời hai kỹ thuật học sâu và rộng giúp cho mô hình dự báo được chính xác hơn so với các mô hình chỉ sử dụng một kỹ thuật.

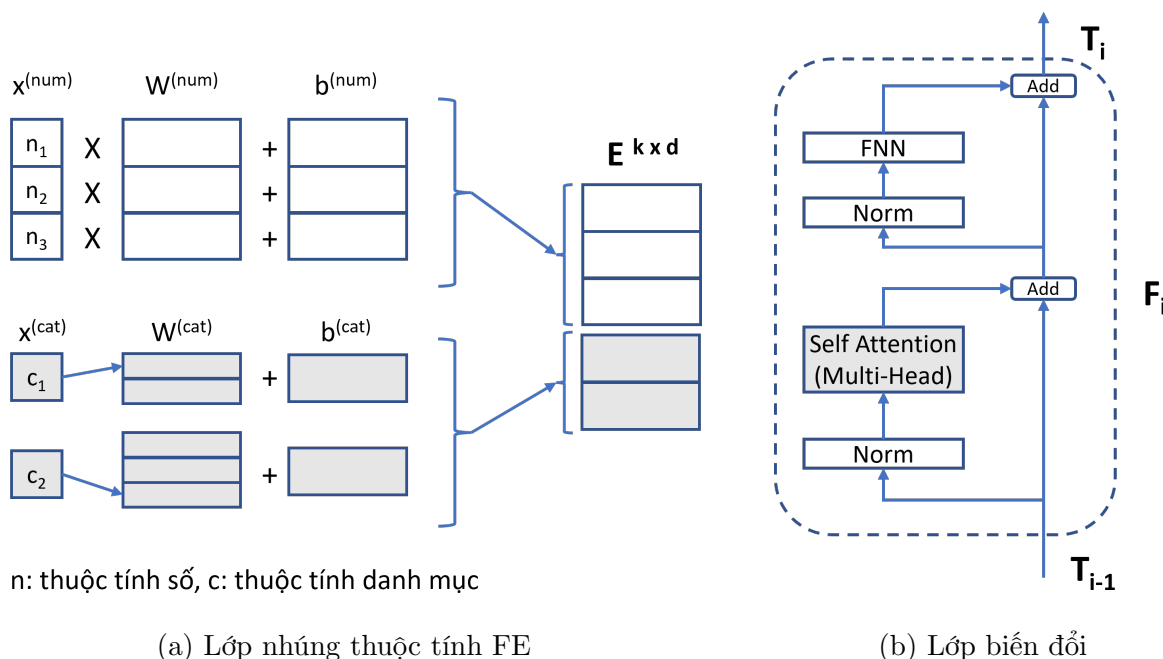
### 2.2.2 Mạng nơ-ron biến đổi

Tác giả nghiên cứu đề xuất một kiến trúc *Transformer* cải tiến bằng cách bổ sung lớp nhúng thuộc tính giúp mô hình huấn luyện làm việc tối ưu hơn trên dữ liệu dạng bảng như được mô tả ở Hình 2.2, gọi là mô hình *FE-Transformer*. Mô hình này đề xuất thêm lớp nhúng nhằm biến đổi tất cả các thuộc tính gồm cả dạng số và danh mục rời rạc thành các véc-tơ nhúng, các bước sau sẽ áp dụng một chuỗi các lớp *Transformer* cho các véc-tơ nhúng đó. Do đó, mỗi lớp *Transformer* có khả năng học được các đặc trưng riêng biệt trong bộ dữ liệu.



Hình 2.2: Kiến trúc *FE-Transformer*

Thiết kế chi tiết của hai thành phần của kiến trúc *FE-Transformer* được biểu diễn như ở Hình 2.3:



(a) Lớp nhúng thuộc tính FE

(b) Lớp biến đổi

Hình 2.3: Thiết kế lớp cho mô hình *FE-Transformer*

## 2.3 Kỹ thuật thực nghiệm

### 2.3.1 Bộ dữ liệu thực nghiệm

Phần thực nghiệm này sử dụng bộ dữ liệu cung cấp bởi Yoochoose GmbH.

### 2.3.2 Xử lý và trích chọn đặc trưng

Bảng 2.1 liệt kê các thuộc tính cơ sở đã được trích chọn.

Bảng 2.1: Danh sách các thuộc tính trích chọn

I	Thuộc tính sản phẩm (2 thuộc tính)		
1	Product ID	Danh mục	Mã sản phẩm
2	Cat ID	Danh mục	Mã danh mục của sản phẩm
II	Thuộc tính phiên (11 thuộc tính)		
3	The First Product	Danh mục	Sản phẩm đầu tiên trong phiên
4	The Pre Product	Danh mục	Sản phẩm trước đó trong phiên
5	Session Duration	Số	Độ dài của phiên
6	Current Duration	Số	Thời gian tính từ đầu phiên
7	#Clicks/Session	Số	Số lượng nhấp trong phiên
8	#Products/Session	Số	Số lượng sản phẩm trong phiên
9	#Clicks So Far	Số	Số lượng nhấp tới hiện tại trong phiên
10	#Products So Far	Số	Số lượng sản phẩm được nhấp tới hiện tại
11	#Views of Product	Số	Số lượng views sản phẩm này trong phiên
12	#Products of the same Cat	Số	Số lượng sản phẩm trong cùng danh mục
13	#Cats	Số	Số lượng danh mục chứa cùng sản phẩm
III	Thuộc tính thời gian chi tiết theo giờ, phút, giây (9 thuộc tính)		
14-16	Session Start	Danh mục	Thời điểm phiên bắt đầu
17-19	The first time that product is clicked	Danh mục	Thời điểm đầu tiên lựa chọn sản phẩm
20-22	Current Time	Danh mục	Thời điểm hiện tại
IV	Thuộc tính boolean (4 thuộc tính)		
23	The most clicked product	Boolean	Sản phẩm được click nhiều nhất trong phiên
24	The most viewed product	Boolean	Sản phẩm được xem nhiều nhất trong phiên
25	The first clicked product	Boolean	Sản phẩm được click đầu tiên trong phiên
26	The most viewed category	Boolean	Danh mục được xem nhiều nhất trong phiên

### 2.3.3 Cách thức chia dữ liệu

Toàn bộ tập dữ liệu được chia ngẫu nhiên theo tỷ lệ 60% để huấn luyện, 20% để đánh giá mức độ hiệu quả trong quá trình tối ưu cấu trúc mạng, 20% để kiểm tra và so sánh giữa các mô hình mạng dự kiến trong quá trình xây dựng cấu trúc mạng.

Bảng 2.2: Bảng thống kê số lượng nhãn của các tập dữ liệu sau khi chia

Dữ liệu	Nhãn mua	Nhãn không mua	Tổng
Tập huấn luyện	325.966	5.593.860	5.919.826
Tập kiểm thử	81.808	1.398.149	1.479.957
Tập thực nghiệm	101.922	1.748.024	1.849.946

### 2.3.4 Độ đo đánh giá mô hình

Nhằm tìm kiếm mô hình dự báo tốt nhất, phần thực nghiệm sử dụng các chỉ số cơ bản sau để tiến hành phân tích đánh giá các cấu trúc mạng khác nhau:

- AUC (*Area Under the Curve*).

- Logloss (*Logarithmic Loss*).
- Độ chính xác (*Accuracy*).

## 2.4 Kết quả thực nghiệm

### 2.4.1 Kết quả thực nghiệm

Bảng 2.3: So sánh hiệu quả giữa các mô hình trong dự báo chuỗi nhấp chuột

Mô hình	AUC	Logloss	Accuracy
<b>LR</b>	0,7604	0,5842	0,6967
<b>FNN</b>	0,8521	0,6145	0,7789
<b>FMNN</b>	0,8620	0,5061	0,7814
<b>PNN</b>	0,8596	0,5332	0,7808
<b>W&amp;DNN</b>	0,8670	0,4519	0,7826
<b>FE-Transformer</b>	0,7868	0,1844	0,9449

### 2.4.2 So sánh với các nghiên cứu liên quan

Nghiên cứu cũng tiến hành so sánh kết quả với nhóm Yandex Data Factory về nhất trong cuộc thi RecSys Challenge 2015, cùng sử dụng bộ dữ liệu Yoochoose [23]. Theo nghiên cứu này, họ sử dụng phương pháp kết hợp bao gồm: Cây phân rã (*Gradient Boosted Decision Tree*) + Mạng phân tích nhân tử FM + Phân tích *Singular Value Decomposition* (SVD) với kết quả AUC = 0,85 và độ chính xác Accuracy = 0,77. Như vậy có thể thấy nghiên cứu hiện tại cho kết quả tốt hơn với tài nguyên tính toán ít hơn.

Các đóng góp của việc đề xuất và thiết kế hai mạng nơ-ron học sâu như sau:

- Cả hai mô hình sử dụng kiến trúc mạng nơ-ron học sâu truyền thẳng cải tiến. Mô hình W&DNN sử dụng mạng FNN có kết hợp với mô hình tuyến tính ở nhánh học rộng. Mô hình FE-Transformer sử dụng lớp tự chú ý để học được các đặc trưng từ các thành phần quan trọng trong phiên làm việc.
- Mô hình W&DNN sử dụng lớp nhúng ở nhánh sâu và phép biến đổi tích chéo ở nhánh rộng, giúp cho mô hình có thể nắm bắt được các trường thuộc tính bậc thấp và bậc cao. Mô hình FE-Transformer được cải tiến với lớp nhúng thuộc tính FE.

## 2.5 Kết luận chương

Chương này nghiên cứu và đề xuất sử dụng hai mô hình mạng nơ-ron cụ thể gồm mạng rộng & sâu và mạng biến đổi để giải quyết Bài toán 1 nhằm dự báo khả năng mua sắm của khách hàng trên cơ sở dữ liệu nhấp chuột. Kết quả cho thấy mô hình rộng và sâu có những khả năng vượt trội hơn: (1) không cần tiền huấn luyện, (2) có thể học được tương tác bậc thấp lẫn bậc cao của các trường thuộc tính, (3) tận dụng được khả năng ghi nhớ của mô hình tuyến tính và khả năng tổng quát hóa của mạng nơ-ron học sâu vào trong cùng một mô hình. Mô hình biến đổi có khả năng xử lý tốt dữ liệu tuần tự sau khi áp dụng một lớp nhúng thuộc tính. Kết quả nghiên cứu của mô hình học sâu và rộng được công bố ở công trình [A-1], và mô hình biến đổi được gửi đi công bố ở công trình [A-8] (để đảm bảo tính đa dạng trong thực nghiệm, công trình [A-8] sử dụng bộ dữ liệu khác so với Luận án này).

Một kết luận quan trọng cho Bài toán 1 là từ kết quả thu được cho thấy việc dự báo hành vi mua của khách hàng với độ chính xác cao có thể được thực hiện bằng cách chỉ dựa trên phân tích chuỗi nhấp chuột trong phiên làm việc hiện tại, mà không cần xét đến thông tin quá khứ của người sử dụng.



# Chương 3 | Đề xuất mô hình mạng nơ-ron đồ thị cho bài toán top-k

Chương 3 trình bày cách thức tiếp cận giải quyết Bài toán 2 trong việc xây dựng mô hình gợi ý top - k. Cụ thể chương này đề xuất biểu diễn dữ liệu phiên làm việc dưới dạng đồ thị, từ đó nghiên cứu đề xuất sử dụng mạng nơ-ron đồ thị để xây dựng bài toán SR gợi ý top - k.

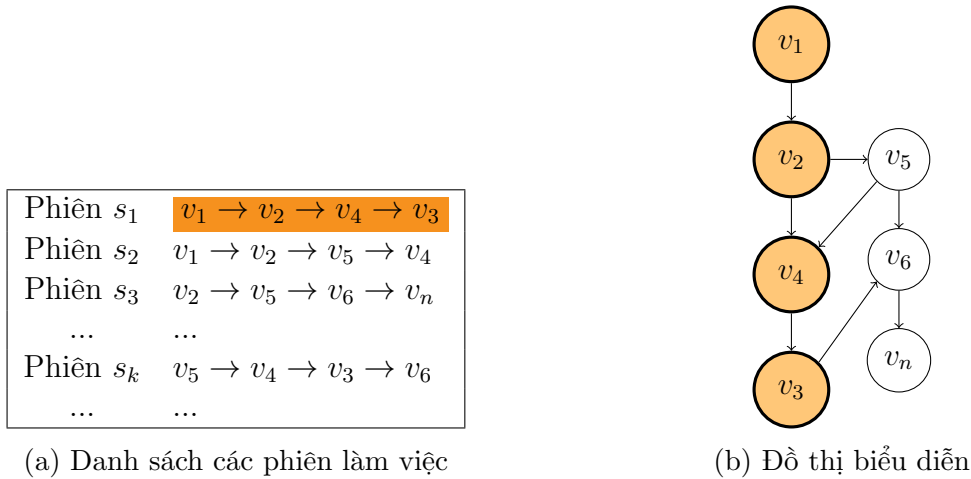
## 3.1 Phát biểu bài toán

Bài toán top - k là một hệ thống gợi ý sản phẩm (ví dụ như bộ phim, bản nhạc hay sản phẩm khi mua hàng...) cho người dùng dựa trên tương tác của họ và cả của người khác với hệ thống. Hệ thống gợi ý sẽ xếp hạng tất cả các sản phẩm đề xuất theo thứ tự giảm dần của xác suất khả năng được người dùng lựa chọn, và sẽ giới hạn trả về top - k sản phẩm được đề xuất.

## 3.2 Đề xuất thiết kế đồ thị

### 3.2.1 Biểu diễn phiên làm việc bằng đồ thị

Một phiên làm việc  $s$  có thể được biểu diễn bằng một đồ thị có hướng  $G_s = (V_s, E_s)$ . Trong đó, mỗi đỉnh thể hiện là sản phẩm  $v_{s,i} \in V$  ( $V$  là tập đỉnh tổng thể của toàn bộ hệ thống). Minh họa biểu diễn đồ thị từ các phiên làm việc  $s_k$  được thể hiện như Hình 3.1.



Hình 3.1: Minh họa biểu diễn phiên làm việc bằng đồ thị

Tương tự như đồ thị, khi biểu diễn phiên làm việc dưới dạng đồ thị, ta có một số định nghĩa:

**Định nghĩa 11. (độ dài đường đi cục bộ)** Giả sử  $v_i$  và  $v_j$  là 2 sản phẩm bất kỳ được nhấp trong phiên  $s$  với thứ tự nhấp lần lượt là  $x$  và  $y$  với  $x < y$ . Độ dài đường đi từ nhấp  $v_i$  tới nhấp  $v_j$  trong phiên làm việc  $s$  ký hiệu là  $p_s(v_i, v_j)$  thỏa mãn công thức:  $p_s(v_i, v_j) = y - x$

**Định nghĩa 12. (p-nhấp)** Hai nhấp vào sản phẩm  $v_i$  và  $v_j$  trong một phiên làm việc  $s$  được gọi là p-nhấp nếu thành phần  $v_j$  được nhấp sau  $v_i$  đúng  $p$  lần nhấp trong phiên làm việc  $s$ . Nói cách khác, hai nhấp  $v_i$  và  $v_j$  trong một phiên làm việc  $s$  là p-nhấp nếu và chỉ nếu  $p_s(v_i, v_j) = p$ .

**Định nghĩa 13. (nhấp kề)** Hai nhấp vào sản phẩm  $v_i$  và  $v_j$  trong một phiên làm việc  $s$  được gọi là nhấp kề nếu thành phần  $v_j$  được nhấp ngay sau  $v_i$  trong phiên làm việc  $s$ . Nói cách khác, hai nhấp  $v_i$  và  $v_j$  trong một phiên làm việc  $s$  là nhấp kề nếu và chỉ nếu  $p_s(v_i, v_j) = 1$ .

**Định nghĩa 14. (trọng số nhấp kề)** Hai nhấp vào sản phẩm  $v_i$  và  $v_j$  trong một phiên làm việc  $s$  có trọng số là số lượng nhấp kề tạo bởi 2 sản phẩm  $v_i$  và  $v_j$  trong phiên làm việc  $s$ , được ký hiệu là  $w_s^{v_i, v_j}$ . Trọng số này được gọi là trọng số nhấp kề.

**Định nghĩa 15. (trọng số p-nhập)** Hai nhập vào sản phẩm  $v_i$  và  $v_j$  trong một phiên làm việc  $s$  có trọng số là số lượng p-nhập tạo bởi 2 sản phẩm  $v_i$  và  $v_j$  trong phiên làm việc  $s$ , được ký hiệu là  $w_{s,p}^{v_i,v_j}$ . Trọng số này được gọi là trọng số p-nhập.

**Định nghĩa 16. (đường đi toàn cục)** Một đường đi  $P$  từ nhập  $v_1$  tới nhập  $v_k$  mà ở đó các nhập  $v_1$  tới  $v_k$  có thể nằm ở nhiều phiên khác nhau, thì đường đi toàn cục giữa 2 nhập đó chính là đường đi giữa 2 đỉnh ở đồ thị tổng thể  $G$  biểu diễn toàn bộ tập phiên làm việc, ký hiệu là  $P(v_1, v_k)$ .

Câu hỏi đặt ra là: "Với tập đỉnh  $V = \{v_1, v_2, \dots, v_n\}$  có số lượng  $n$  sản phẩm cố định thì khi biểu diễn đồ thị tổng thể  $G$  cần xây dựng tập cạnh  $E$  và trọng số cạnh như thế nào cho hiệu quả?".

### 3.2.2 Đề xuất thiết kế đồ thị

Phần này đề xuất một số phương án xây dựng đồ thị  $G$  từ tập danh sách phiên làm việc của các khách hàng. Cụ thể tác giả đề xuất 3 dạng đồ thị sau:

#### a. Đồ thị $\mathcal{G}$

Gọi  $\mathcal{G}$  là một đồ thị thoả mãn ma trận kề  $M_G \in \mathbb{R}^{n \times n}$  với  $M_G^{v_i,v_j}$  là số lần sản phẩm  $v_j$  được nhập kể ngay sau khi nhập sản phẩm  $v_i$  trong một phiên. Ta có:

$$M_G^{v_i,v_j} = \sum_s w_s^{v_i,v_j}, \forall s \quad (3.1)$$

trong đó  $w_s^{v_i,v_j}$  là "trọng số nhập kề" của 2 đỉnh  $v_i, v_j$  trong phiên làm việc  $s$ .

#### b. Đồ thị $\mathcal{H}$

Gọi  $\mathcal{H}$  là một đồ thị thoả mãn ma trận kề  $M_H \in \mathbb{R}^{n \times n}$  với  $M_H^{v_i,v_j}$  là số lần sản phẩm  $v_j$  được nhập sau khi nhập sản phẩm  $v_i$  trong một phiên. Ta có:

$$M_H^{v_i,v_j} = \sum_s \sum_{p=0}^{|s|} w_{s,p}^{v_i,v_j}, \forall s \quad (3.2)$$

trong đó  $w_{s,p}^{v_i,v_j}$  là "trọng số p-nhập" của 2 đỉnh  $v_i, v_j$  trong phiên làm việc  $s$ .

#### c. Đồ thị $\mathcal{K}$

Giả sử  $c$  là số lượng nhập nhiều nhất của một phiên trong tập dữ liệu. Gọi  $\mathcal{K}$  là một đồ thị thoả mãn khối ma trận kề  $M_K \in \mathbb{R}^{n \times n \times c}$  với  $M_K^{v_i,v_j}[p]$  là tổng số lần sản phẩm  $v_j$  được nhập sau khi nhập sản phẩm  $v_i$  đúng  $p$  lần nhập trong một phiên. Ta có:

$$M_K^{v_i,v_j}[p] = \sum_s w_{s,p}^{v_i,v_j} \quad (3.3)$$

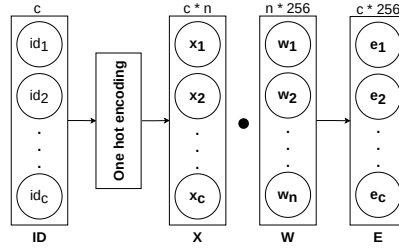
## 3.3 Các mô hình đề xuất

### 3.3.1 Mạng nơ-ron truyền thẳng (FNN)

Phần này đề xuất sử dụng mạng nơ-ron truyền thẳng FNN như ở chương 2 nhưng giải quyết Bài toán 2 là xây dựng mô hình gợi ý top-k thay vì Bài toán 1.

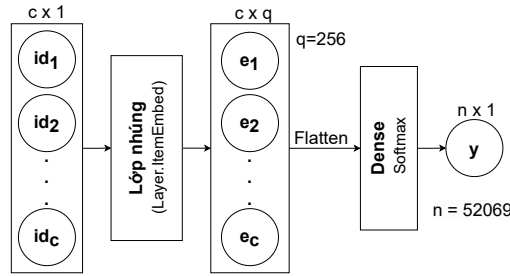
#### a. Lớp nhúng sản phẩm

Luận án đề xuất xây dựng lớp nhúng sản phẩm như Hình 3.2. Lớp nhúng này sẽ được dùng làm lớp cơ sở để xây dựng một số mô hình khác nhau trong luận án này.



Hình 3.2: Lớp nhúng sản phẩm (*Layer.ItemEmbed*)

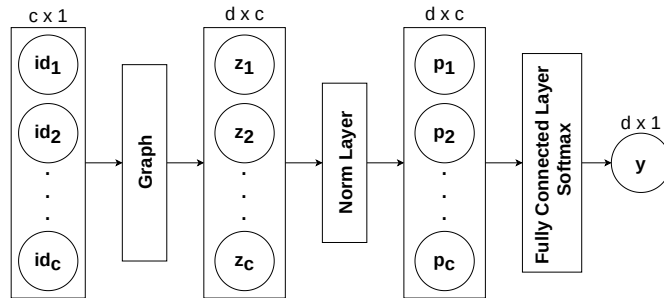
b. Mô hình mạng nơ-ron truyền thẳng



Hình 3.3: Mô hình FNN cơ sở

3.3.2 Mạng nơ-ron đồ thị (*GNN*)

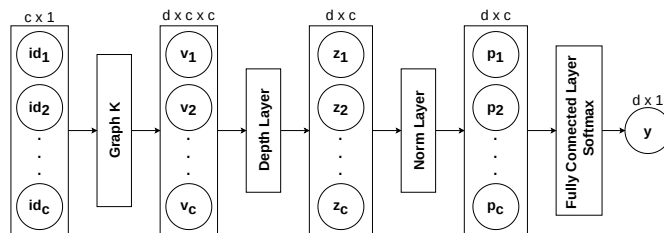
a. Mô hình mạng nơ-ron cho đồ thị  $\mathcal{G}$  và  $\mathcal{H}$



Hình 3.4: Mô hình mạng nơ-ron cho đồ thị  $\mathcal{G}$  và  $\mathcal{H}$

b. Mô hình mạng nơ-ron cho đồ thị  $\mathcal{K}$

Để cải tiến mô hình mạng nơ-ron đồ thị khi phải làm việc với đồ thị đa quan hệ  $\mathcal{K}$  với trọng số cạnh là véc-tơ  $c$  chiều, luận án đề xuất sử dụng thêm một lớp học sâu như Hình 3.5.



Hình 3.5: Mô hình mạng nơ-ron cho đồ thị  $\mathcal{K}$

## 3.4 Kỹ thuật thực nghiệm

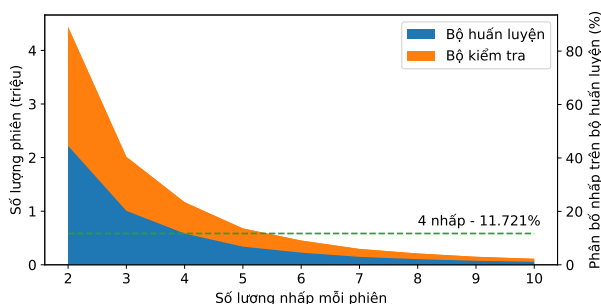
### 3.4.1 Tiền xử lý dữ liệu

Bộ dữ liệu sau bước tiền xử lý được mô tả ở Bảng 3.1.

Bảng 3.1: Thống kê về bộ dữ liệu nhập Yoochoose sau khi tiền xử lý

	Bộ huấn luyện	Bộ kiểm tra	Tổng
Số lượng phiên	7.990.018	1.996.408	9.986.426
Số lượng sản phẩm	52.069	38.733	52.069
Số lượng nhập	31.744.233	7.926.322	39.670.555
Số nhập lớn nhất	200	200	200
Số nhập nhỏ nhất	2	2	2
Số nhập trung bình	3,97	3,97	3,97

Biểu đồ phân bố số lượng phiên được nhập từ 1 tới 10 lần ở Hình 3.6, do số lượng phiên có nhập lớn hơn 10 rất nhỏ nên không cần thể hiện trong biểu đồ này:



Hình 3.6: Biểu đồ phân bố số lượng nhập chuột (sau khi tiền xử lý)

### 3.4.2 Chuẩn hóa dữ liệu huấn luyện

Các phiên dữ liệu trong bộ dữ liệu gốc có số lượng nhập khác nhau nên không thể dùng ngay cho các mô hình phân loại. Để có được dữ liệu đào tạo phù hợp cho các mô hình, tác giả đề xuất một số thuật toán chuẩn hóa dữ liệu huấn luyện theo đúng tiêu chuẩn đầu vào đã được thiết kế cho các mô hình đề xuất.

#### a. Chuẩn hóa dữ liệu huấn luyện cho mô hình FNN

Mô hình FNN là mô hình cơ sở không sử dụng đồ thị, vì vậy thuật toán chuẩn hóa dữ liệu khá đơn giản và được thể hiện như mô hình 3.7:

Giải mã của các bước chuẩn hóa dữ liệu trên được mô tả tại Thuật toán 3.1:

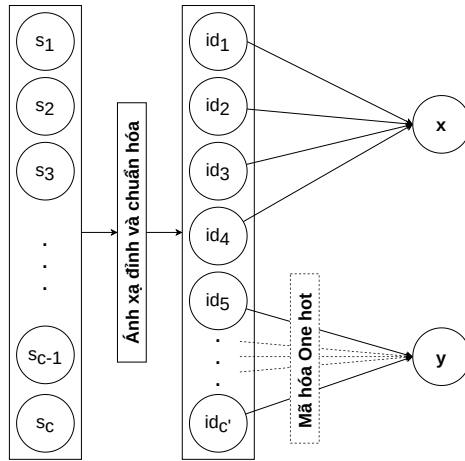
#### b. Chuẩn hóa dữ liệu huấn luyện cho mô hình GNN

Để có những véc-tơ chuẩn đầu vào cho các mô hình sử dụng đồ thị, các bước chuẩn hóa được mô tả như Hình 3.8 với mỗi phiên của từng đồ thị.

Giải mã của các bước chuẩn hóa dữ liệu trên được mô tả tại Thuật toán 3.2:

### 3.4.3 Độ đo đánh giá mô hình

Đề xuất các độ đo  $Recall@k$ ,  $MRR@k$  và  $ACCs@k$  để đánh giá hệ gợi ý top - k.



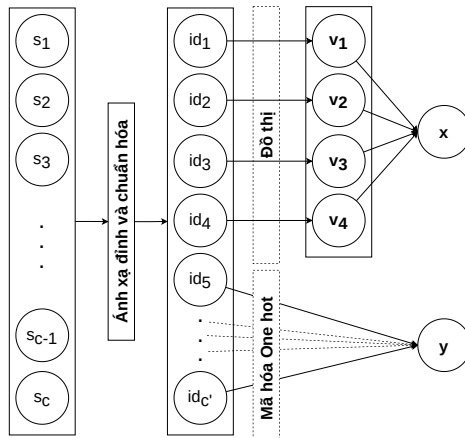
Hình 3.7: Mô hình chuẩn hóa dữ liệu huấn luyện cho mô hình FNN

**Algorithm 3.1:** Thuật toán NORM.FNN:

Chuẩn hóa dữ liệu huấn luyện cho mô hình FNN

**Input:**  $s = \{id_1, id_2, \dots, id_c\}$   
**Output:** Dữ liệu đầu vào huấn luyện là  $x$  và đầu ra huấn luyện  $y$

- 1  $c' \leftarrow c$ ;
- 2 **while**  $c' < 5$  **do**
- 3     Thêm vào cuối  $s$  một nhập  $None$ ;
- 4      $c' \leftarrow c' + 1$ ;
- 5  $\mathbf{x} \leftarrow \{id_1, id_2, id_3, id_4\}$ ;
- 6  $Z \leftarrow \{id_5, id_6, \dots, id_{c'}\}$ ;
- 7  $\mathbf{y} \leftarrow OneHotEncoding(Z)$
- 8 **return**  $\mathbf{x} \in \mathbb{R}^4, \mathbf{y} \in \mathbb{R}^{n \times 2}$ ;



Hình 3.8: Mô hình chuẩn hóa dữ liệu huấn luyện cho các mô hình GNN

$$Recall@k = \frac{1}{n} \sum_{i=0}^{n-1} \frac{|S_{pred}^i \cap S_{labels}^i|}{|S_{labels}^i|} \quad (3.4)$$

$$MRR@k = \frac{1}{n} \sum_{i=0}^{n-1} RR(id_*^i, S_{pred}^i) \quad (3.5)$$

**Algorithm 3.2:** Thuật toán NORM.GNN:

Chuẩn hóa dữ liệu dữ liệu huấn luyện cho các mô hình GNN

---

**Input:**  $s = \{id_1, id_2, id_3, \dots, id_{c-1}, id_c\}$   
**Output:** Dữ liệu đầu vào huấn luyện là  $x$  và đầu ra huấn luyện  $y$

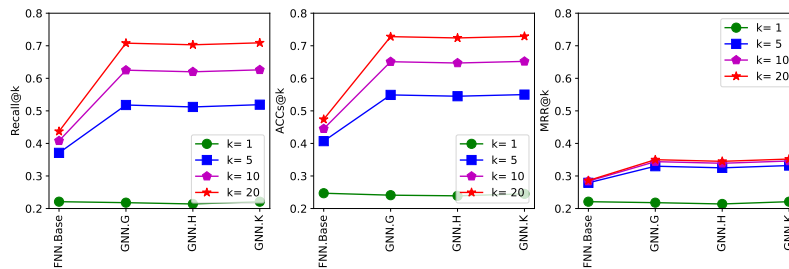
- 1  $c' \leftarrow c$ ;
- 2 **while**  $c' < 5$  **do**
- 3     Thêm vào cuối  $s$  một nhập  $None$ ;
- 4      $c' \leftarrow c' + 1$ ;
- 5  $\mathbf{x} \leftarrow \{\}$ ;
- 6 **for**  $i \leftarrow 1$  **to**  $4$  **by**  $1$  **do**
- 7     **if**  $id_i == None$  **then**
- 8          $v_i \leftarrow$  vec-tơ toàn 0;
- 9     **else**
- 10          $v_i \leftarrow$  vec-tơ trọng số của đỉnh  $id_i$  trong đồ thị;
- 11     Thêm  $v_i$  vào  $\mathbf{x}$
- 12  $Z \leftarrow \{id_5, id_6, \dots, id_{c'}\}$ ;
- 13  $\mathbf{y} \leftarrow OneHotEncoding(Z)$
- 14 **return**  $\mathbf{x} \in \mathbb{R}^4$ ,  $\mathbf{y} \in \mathbb{R}^{n \times 2}$ ;

---

$$ACC_{s@k} = \frac{1}{n} \sum_{i=0}^{n-1} \min(1, |S_{pred}^i \cap S_{labels}^i|) \quad (3.6)$$

### 3.5 Kết quả và nhận xét

Hình 3.9 biểu diễn kết quả của các mô hình sử dụng trong quá trình thực nghiệm.



Hình 3.9: Biểu đồ kết quả so sánh các mô hình GNN với FNN

### 3.6 Kết luận chương

Chương này tác giả đề xuất thiết kế 3 đồ thị khác nhau gồm đồ thị đơn  $\mathcal{G}$ , đồ thị đơn  $\mathcal{H}$  và đồ thị đa quan hệ  $\mathcal{K}$ . Các đồ thị này khác nhau về cách thức thiết kế tập cạnh và trọng số cạnh trong việc biểu diễn mối quan hệ giữa các nhập, bao gồm cả quan hệ trong phiên làm việc cục bộ và giữa các phiên làm việc toàn cục trong tập dữ liệu. Kết quả thực nghiệm cho thấy mô hình GNN kết hợp với đồ thị biểu diễn phiên làm việc cho kết quả rất khả quan so với mô hình mạng nơ-ron truyền thẳng FNN không dùng đồ thị. Kết luận của chương khẳng định mạng nơ-ron đồ thị GNN hoàn toàn có thể được sử dụng để xây dựng hệ thống gợi ý  $top-k$ .

# Chương 4 | Đề xuất cải tiến mô hình GNN với phép nhúng

Với kết quả đạt được ở Chương 3 cho Bài toán 2 bằng cách biểu diễn phiên làm việc dưới dạng đồ thị, tuy nhiên vẫn có một thách thức đặt ra là mô hình đề xuất phải xử lý bài toán đa nhãn với số lượng nhãn tương đương với số lượng đỉnh của đồ thị là rất lớn.

## 4.1 Thách thức của bài toán phân loại đa nhãn

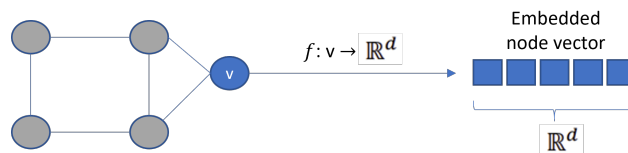
Phân loại đa nhãn là một vấn đề khó khăn trong máy học do nhiều lý do như sự phụ thuộc giữa nhãn, không gian nhãn lớn, dữ liệu mất cân bằng và trích xuất đặc trưng.

## 4.2 Phương pháp nhúng đồ thị

**Định nghĩa 17. Phép nhúng đồ thị.** Phép nhúng đồ thị là một kỹ thuật để biểu diễn một đồ thị dưới dạng các véc-tơ có số chiều cao với mục đích hỗ trợ các thuật toán học máy xử lý và phân tích thông tin của đồ thị, ví dụ như phân loại nút, dự đoán liên kết và phân cụm đồ thị.

### 4.2.1 Phép biến đổi nhúng đỉnh

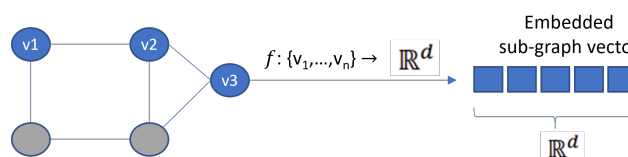
Phép biến đổi nhúng để biến đổi một đỉnh  $v \in V$  vào một không gian nhúng  $d$  chiều để tạo ra các véc-tơ nhúng đỉnh trong không gian mới  $\forall v \in \mathbb{R}^d$ , được minh họa như Hình 4.1.



Hình 4.1: Phép biến đổi nhúng đỉnh

### 4.2.2 Phép biến đổi nhúng đồ thị con

Phép biến đổi nhúng đồ thị con là phép biến đổi một nhóm đỉnh có liên quan với nhau vào một không gian nhúng  $d$  chiều để tạo ra các véc-tơ nhúng trong không gian mới  $\forall v \in \mathbb{R}^d$ , được minh họa như Hình 4.2.



Hình 4.2: Phép biến đổi nhúng đồ thị con

## 4.3 Đề xuất cải tiến mô hình GNN.K

### 4.3.1 Chuyển đổi bài toán đa nhãn thành nhị phân

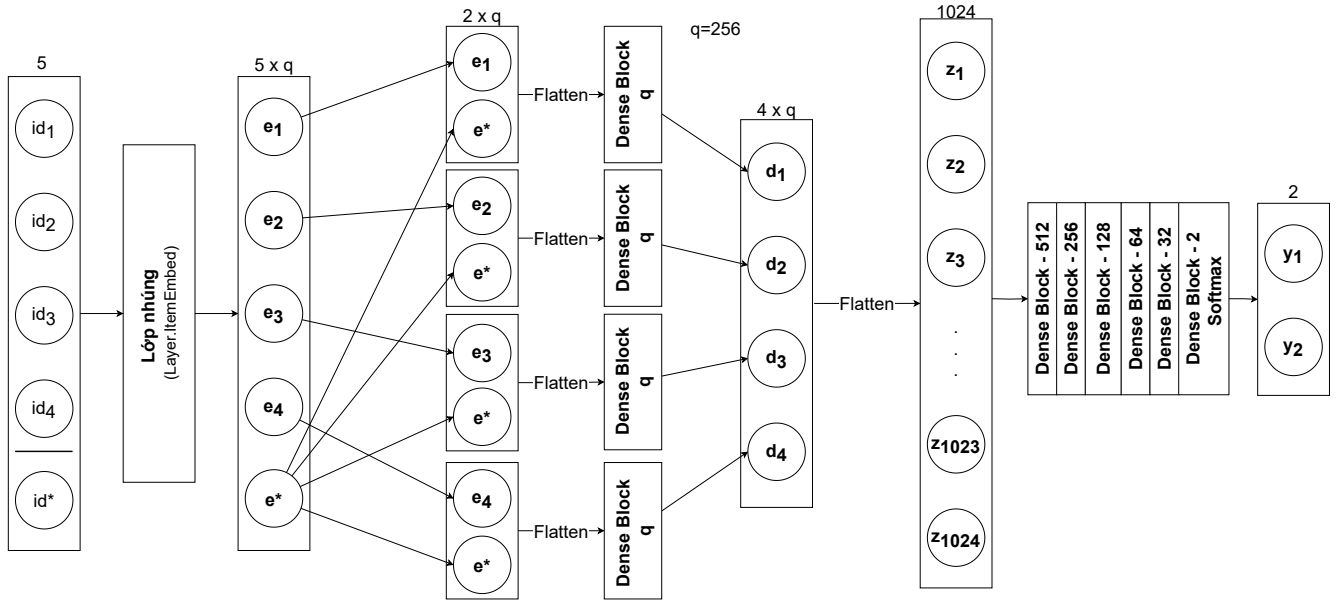
Tác giả đề xuất thêm mô hình nhị phân để đánh giá thêm mức độ hiệu quả giữa mô hình đa nhãn và mô hình nhị phân. Để biến đổi một mô hình đa nhãn thành mô hình nhị phân chúng ta đưa nhãn vào đầu vào để mô hình trả lời "có" hoặc "không" với nhãn đó.

### 4.3.2 Đề xuất mạng nơ-ron truyền thẳng nhị phân

Tác giả đề xuất chuyển đổi thành mô hình nhị phân thông qua việc tiếp tục sử dụng lớp nhúng sản phẩm `Layer.ItemEmbed` như mô hình FNN cơ sở tuy nhiên có điểm khác biệt là đưa thêm



thành phần nhãn  $id^*$  và kết hợp chéo với từng thành phần  $id_i$  của dữ liệu đầu vào. Mô hình đề xuất được mô tả như ở Hình 4.3.



Hình 4.3: Mô hình FNN nhị phân ( $FNN.bin$ )

### 4.3.3 Đề xuất mô hình nhúng đồ thị $\mathcal{K}$ nhị phân

#### a. Đề xuất lớp nhúng phiên kết hợp

Trước tiên, luận án đề xuất kỹ thuật nhúng đồ thị biểu diễn phiên làm việc bằng cách kết hợp mô hình  $FNN.bin$  (Hình 4.3) sử dụng lớp nhúng sản phẩm  $Layer.ItemEmbed$  và lớp nhúng đồ thị  $\mathcal{K}$ , trong đó lớp nhúng đồ thị  $\mathcal{K}$  cũng sử dụng kỹ thuật nhúng chéo kết hợp nhãn  $id^*$  với từng thành phần  $id_i$ . Lớp nhúng phiên đề xuất với tên gọi  $Layer.SessionEmbed$  được thiết kế như Hình 4.4.

#### b. Đề xuất mô hình

Mô hình đề xuất có tính phức tạp vì tích hợp nhiều cải tiến qua những mô hình thử nghiệm để xử lý cho bài toán đa nhãn có không gian nhãn lớn bao gồm: (1) biến đổi nhị phân; (2) biểu diễn đồ thị; (3) nhúng đồ thị kết hợp với nhúng nhãn. Mô hình gợi ý được đề xuất có cấu trúc nhị phân như Hình 4.5.

## 4.4 Kỹ thuật thực nghiệm

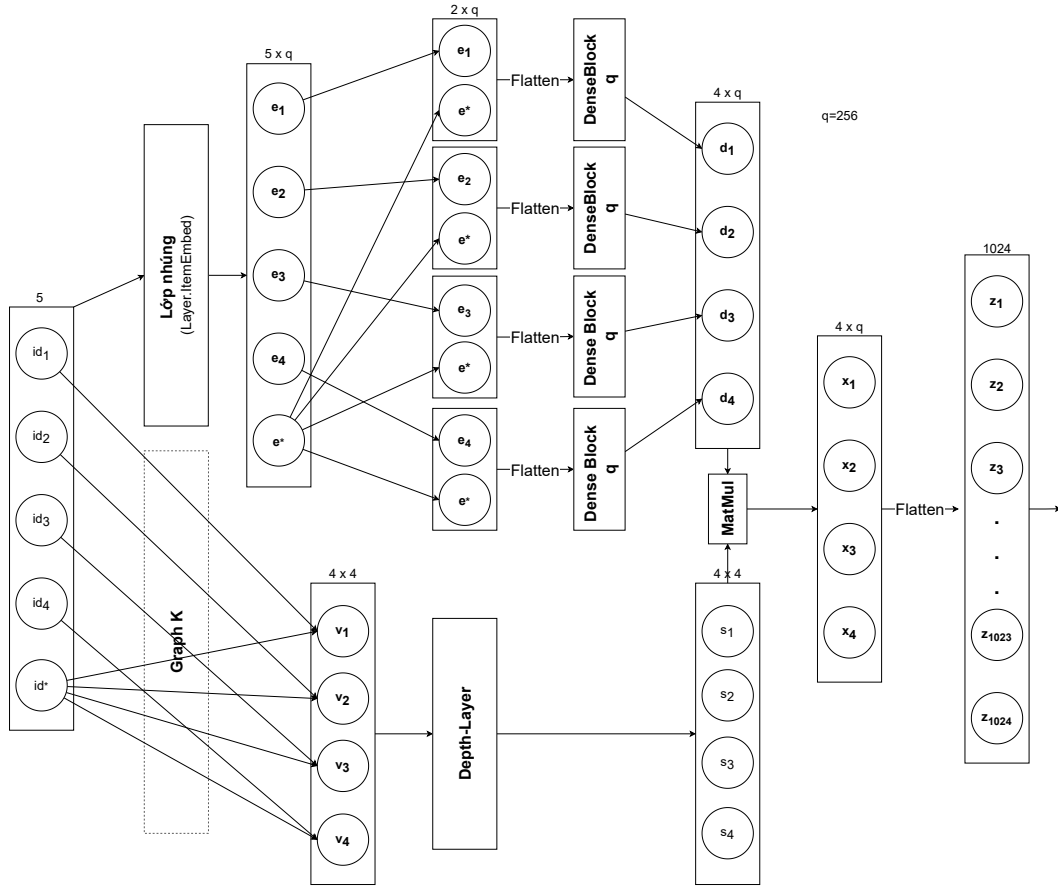
### 4.4.1 Chuẩn hóa dữ liệu huấn luyện

Thuật toán chuẩn hóa dữ liệu huấn luyện được mô tả như sau cho mỗi phiên ứng với đồ thị  $\mathcal{K}$  được mô tả tại Thuật toán 4.1:

## 4.5 Kết quả và nhận xét

### 4.5.1 Kết quả thực nghiệm

Hình 4.6 biểu diễn kết quả tổng hợp của  $k \in [1, 5, 10, 20]$  trong cùng một biểu đồ để tiện so sánh kết quả. Kết quả cho thấy mô hình nhúng với đồ thị  $\mathcal{K}$  ( $GNN.Bin.K$ ) cao hơn hết các mô hình dùng mạng nơ-ron khác.


 Hình 4.4: Lớp nhúng phiên với đồ thị  $\mathcal{K}$  ( $Layer.SessionEmbed$ )

**Algorithm 4.1:** Thuật toán NORM.GNN.Bin:

Chuẩn hóa dữ liệu huấn luyện cho mô hình GNN nhị phân

**Input:**

$s = \{id_1, id_2, \dots, id_c\}$  //phiên lựa chọn

$n_{id^*}$  //số lượng đỉnh cần cần quan sát xem có phải là nhãn không

**Output:** Dữ liệu đầu vào huấn luyện là  $x$  và đầu ra huấn luyện  $y$

1  $c' \leftarrow c;$

2 **while**  $c' < 5$  **do**

3     Thêm vào cuối  $s$  một nhập  $None$ ;

4      $c' \leftarrow c' + 1;$

5  $Z \leftarrow id_5, id_6, \dots, id_{c'};$

6  $I \leftarrow$  tập chứa  $n_{id^*}$  đỉnh kề của các đỉnh ngẫu nhiên trong phiên, ưu tiên đỉnh có trong các  $\{id_5, id_6, \dots, id_{c'}\}$ ; //lưu ý bỏ các đỉnh có giá trị là  $None$ .

7 **for** đỉnh  $o \in I$  **do**

8      $x^o \leftarrow \{v_1^o, v_2^o, v_3^o, v_4^o\}$  với  $v_i^o$  là trọng số cạnh nối từ đỉnh  $id_i$  đến đỉnh  $o$ ;

9      $y^o \leftarrow \{0, 1\}$ ; //nhãn true

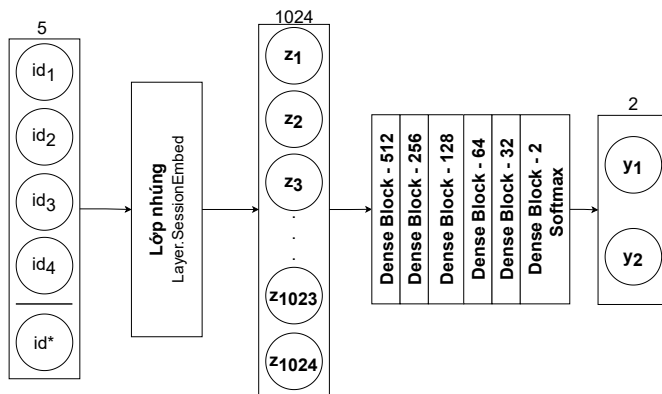
10     **if**  $o \notin Z$  **then**

11          $y^o \leftarrow \{1, 0\}$  //nhãn false

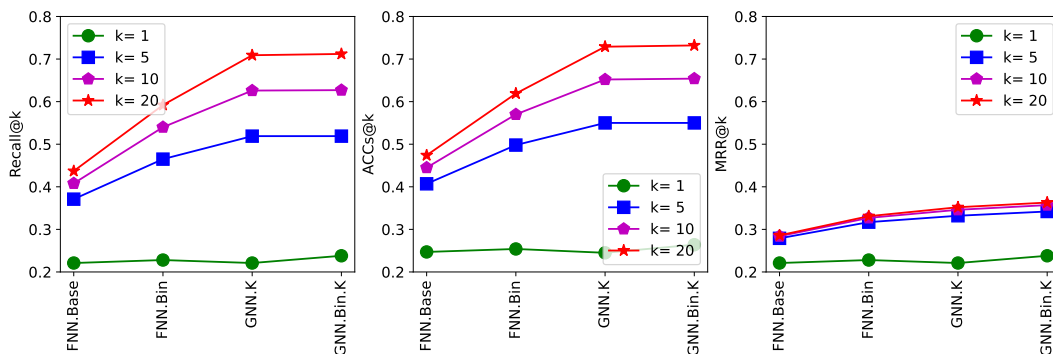
12  $\mathbf{x} \leftarrow \{x^o | o \in I\} \in \mathbb{R}^{n_{id^*} \times 4};$

13  $\mathbf{y} \leftarrow \{y^o | o \in I\} \in \mathbb{R}^{n_{id^*} \times 2};$

14 **return**  $\mathbf{x}, \mathbf{y};$



Hình 4.5: Mô hình nhúng nhị phân với đồ thị  $\mathcal{K}$  ( $GNN.Bin.K$ )



Hình 4.6: So sánh  $GNN.Bin.K$  với các mô hình khác

## 4.6 Kết luận chương

Kết luận phép biến đổi nhúng đồ thị là kỹ thuật quan trọng để xây dựng hệ thống gợi ý  $top-k$ , đặc biệt với các bài toán liên quan đến việc biểu diễn mối tương tác giữa người dùng khi lựa chọn sản phẩm trong phiên làm việc dưới dạng đồ thị. Bằng cách học cách biểu diễn đồ thị sang một chiều không gian nhúng mới để nắm bắt các đặc trưng tiềm ẩn của các véc-tơ nhúng phiên, mô hình gợi ý  $top-k$  hoạt động hiệu quả hơn.

Kết quả thực nghiệm trong chương này đã chứng minh mô hình đề xuất đạt được hiệu suất tốt với 3 cải tiến gồm (1) chuyển đổi mô hình nhị phân, (2) đề xuất lớp nhúng đồ thị biểu diễn phiên làm việc và (3) thiết kế kết hợp nhúng nhãn.

# Kết luận

## 1 Kết luận chung

Luận án đề xuất sử dụng đồ thị để biểu diễn dữ liệu chuỗi sự kiện nhấp chuột mua hàng, gồm ba đồ thị  $\mathcal{G}$ ,  $\mathcal{H}$ , và  $\mathcal{K}$  với các độ phức tạp khác nhau để đánh giá mức độ hiệu quả của các mô hình dự báo  $top - k$ . Với các đồ thị biểu diễn dữ liệu, tác giả đề xuất sử dụng mạng nơ-ron đồ thị GNN để làm mô hình dự báo.

## 2 Kết quả đạt được

Một số nhận xét về kết quả của mình so với các nghiên cứu trước đây:

- ✓ Luận án nghiên cứu và đề xuất mô hình mạng nơ-ron học sâu cho Bài toán 1 và mạng nơ-ron đồ thị cho Bài toán 2. Trong đó Bài toán 1 là bài toán nhị phân và Bài toán 2 là bài toán đa nhãn  $top - k$ .
- ✓ Luận án này sử dụng cả tập dữ liệu huấn luyện và kiểm thử từ bộ dữ liệu gốc với số lượng sản phẩm, tức số lượng nhãn, lên tới hơn 52 nghìn.
  - Các nghiên cứu trước đây không sử dụng bộ dữ liệu kiểm thử riêng biệt, mà trích ra từ tập dữ liệu huấn luyện.
- ✓ Luận án này đề xuất và xây dựng mô hình GNN có tính mở rộng cao khi hoạt động với đồ thị với hơn 52 nghìn đỉnh. Luận án đề xuất thiết kế đồ thị  $\mathcal{G}$  với khái niệm nhấp kế, đồ thị  $\mathcal{H}$  sử dụng trọng số cạnh là đường đi giữa các nhấp trong phiên làm việc, và đồ thị  $\mathcal{K}$  với trọng số cạnh là một véc-tơ  $c$  chiều.
  - Một số nghiên cứu liên quan trình bày không thể chạy được mô hình với bộ dữ liệu đầy đủ, do đó họ phải thực nghiệm với bộ dữ liệu nhỏ hơn với số lượng nhãn thậm chí còn ít hơn.
- ✓ Mô hình đề xuất cho kết quả  $Recall@20$  là 0,712 và  $MRR@20$  là 0,363
  - Kết quả trên là tốt hơn nghiên cứu của Kiewan có  $Recall@20$  là 0,691 và của Tan có  $Recall@20$  là 0,680 và tốt hơn hẳn nghiên cứu đầu tiên của Balázs Hidas với  $Recall@20$  là 0,632.

## 3 Các đóng góp chính của luận án

Luận án này có các đóng góp chính sau:

- Sử dụng đồ thị để mô hình hóa hành vi mua sắm của khách hàng thông qua chuỗi nhấp chuột trong phiên làm việc, bao gồm cả đồ thị đơn và đa quan hệ
- Đề xuất mô hình mạng nơ-ron học sâu cho Bài toán 1 và mạng nơ-ron đồ thị cho Bài toán 2. Với Bài toán 2, luận án đề xuất thiết kế ba đồ thị  $\mathcal{G}$ ,  $\mathcal{H}$  và  $\mathcal{K}$ . Với đồ thị đa quan hệ  $\mathcal{K}$  sử dụng trọng số cạnh là một véc-tơ, luận án đề xuất sử dụng thêm một lớp học sâu tuyến tính cho phép mạng GNN có thể học được đồ thị này hiệu quả hơn.
- Đề xuất thuật toán nhúng đồ thị cho phép mô hình GNN có thể học được các thuộc tính ẩn của hành vi của người dùng trong quá trình lựa chọn các danh mục sản phẩm trong phiên làm việc hiện tại.

# DANH MỤC CÁC BÀI BÁO ĐÃ XUẤT BẢN LIÊN QUAN ĐẾN LUẬN ÁN

1. **Khang Nguyen**, Anh V. Nguyen, Lan N. Vu, Nga T. Mai, and Binh P. Nguyen, "An Efficient Deep Learning Method for Customer Behaviour Prediction Using Mouse Click Events", Proceedings of the 11th National Conference on Fundamental and Applied Information Technology Research (FAIR'2028), 2018, pp.10, Vietnam, doi = 10.15625/vap.2018.0002.
2. **Khang Nguyen**, Nga T. Mai, An H. Nguyen, and Binh P. Nguyen, "Prediction of Wart Treatment Using Deep Learning with Implicit Feature Engineering", Soft Computing for Biomedical Applications and Related Topics, Springer International Publishing, 2020, pp.153–168, doi = 10.1007/978-3-030-49536-7\_14.
3. **Nguyễn Tuấn Khang**, Nguyễn Việt Việt, Nguyễn Hải An, Mai Sơn, Mai Thúy Nga, và Nguyễn Việt Anh, "Phát hiện giao dịch thẻ gian lận sử dụng mô hình học sâu", hội thảo quốc gia lần thứ XXIII, 2020, pp.335
4. **Nguyễn Tuấn Khang**, Mai Thúy Nga, Nguyễn Hải An, và Nguyễn Việt Anh, "Phân Tích Hành Vi Khách Hàng Với Mô Hình Mạng Học Sâu Đồ Thị", hội thảo quốc gia lần thứ XXIV, 2021, p.439
5. **Nguyễn Tuấn Khang**, Nguyễn Tú Anh, Mai Thúy Nga, Nguyễn Hải An, và Nguyễn Việt Anh, "Hệ Gợi Ý Mua Sắm Dựa Theo Phân Loại Việc Với Mô Hình Mạng Học Sâu Đồ Thị", chuyên san Các công trình nghiên cứu, phát triển và ứng dụng CNTT và Truyền thông, Bộ Thông tin và Truyền thông, 2022, vol. 2022, no. 02.
6. **Khang Nguyen**, Viet V. Nguyen, Nga T. Mai, An H. Nguyen, and Anh V. Nguyen, "Behavioral gait recognition using hybrid Convolutional Neural Networks", Journal of Computer Science and Cybernetics, 2023
7. **Khang Nguyen**, Nga T. Mai, An H. Nguyen, and Anh V. Nguyen, "A Computational Model for Predicting Customer Behaviors Using Transformer Adapted with Tabular Features", International Journal of Computational Intelligence Systems, vol. 16, no. 1, pp. 1–8, 2023, doi = 10.1007/s44196-023-00307-5.
8. **Khang Nguyen**, Anh T. Nguyen, Nga T. Mai, An H. Nguyen, and Anh V. Nguyen, "Developing Advanced Product Recommendation System using Embedding Graph Neural Networks", Applied Intelligence, Springer, 2023 (bài đang nộp)