

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
CỞ SỞ TP. HỒ CHÍ MINH
(2022 – 2023)



BỘ MÔN: PHÁT TRIỂN HỆ THỐNG THÔNG MINH



Báo cáo giữa kỳ

Đề tài: Xây dựng hệ thống đề xuất món ăn

Giảng viên: Nguyễn Ngọc Duy

Sinh viên: Nguyễn Hữu Phước - N19DCCN145

Hà Nam Thái - N19DCCN187

Nguyễn Ngọc Thanh Danh - N19DCCN027

Bùi Minh Quang - N19DCCN150

Nguyễn Tất Kỳ - N19DCCN083

12/2022

Mục lục

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI	1
1.1. Lý do chọn đề tài:	1
1.2. Mục tiêu nghiên cứu:	1
1.3. Đối tượng và phạm vi nghiên cứu:	1
1.4. Phương pháp nghiên cứu:	2
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	2
2.1 Giới thiệu về học máy & học sâu (cần bổ sung lý thuyết):	2
2.2 Giới thiệu về thuật toán tối ưu Stochastic gradient descent (cần bổ sung lý thuyết):	3
CHƯƠNG 4: THIẾT KẾ HỆ THỐNG	12
4.1 Thiết kế cơ sở dữ liệu:	12
4.1.1 Sơ đồ diagram:	12
4.1.2 Từ điển dữ liệu:	12
4.2 Giao diện hệ thống:	15
4.2.1 Giao diện đăng nhập, nhập tên người dùng:	15

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

1.1. Lý do chọn đề tài:

- Trong cuộc sống tập nập, hiện đại như bây giờ, việc chọn lựa đồ ăn và nấu ăn cũng là một vấn đề nan giải với một số bạn trẻ, người nội trợ. Đề tài “Xây dựng hệ thống đề xuất món ăn” được nhóm em xây dựng bằng cách cho máy tính học những tri thức liên quan dựa trên tốc độ xử lý ưu việt của máy tính. Việc đề xuất một món ăn phù hợp có rất nhiều tổ ảnh hưởng bởi vì có rất nhiều món ăn, mỗi món ăn cũng có những cách chế biến khác nhau và mỗi người lại có khẩu vị khác nhau,... Do đó, việc khai thác và phân tích dữ liệu có thể giúp nhóm hiểu hơn và đưa ra đề xuất chính xác hơn, phù hợp hơn với người dùng.
- Muốn máy tính đề xuất gợi ý theo mong muốn của người dùng hay không thì máy tính phải có những tri thức có đề cập đến các món ăn liên quan (món được chế biến như thế nào, có thể mua món này ở đâu,...). Các tri thức này được phân tích, gộp nhóm có chọn lọc. Bằng phương pháp gộp nhóm dữ liệu qua chọn lọc sẽ giúp hệ thống sẽ đưa ra những gợi ý liên quan một cách chính xác hơn giúp người dùng có thêm nhiều sự lựa chọn, không bị gò bó bởi một món ăn, một phương pháp truyền thống, giúp người dùng cảm thấy thoải mái sau một ngày làm việc mệt mỏi.
- Với những lí do nêu trên cùng với sự cho phép của Giảng viên Hướng dẫn Nguyễn Ngọc Duy, nhóm chúng em quyết định chọn đề tài “Xây dựng hệ thống đề xuất món ăn”.

1.2. Mục tiêu nghiên cứu:

- Nghiên cứu lý thuyết về mô hình thuật toán Stochastic gradient descent.
- Áp dụng thuật toán Stochastic gradient descent, bài toán gợi ý và bài toán trích xuất đặc trưng vào việc xử lý dữ liệu.
- Mục tiêu của hệ thống đề xuất món ăn là nó có thể tự động đưa ra món ăn dựa vào mong muốn gợi ý của người dùng, bao gồm cách chế biến, những món ăn nấu kèm hay gợi ý những địa điểm cửa hàng có món ăn đó.

1.3. Đối tượng và phạm vi nghiên cứu:

- Đối tượng nghiên cứu: dữ liệu về món ăn.
- Phạm vi nghiên cứu: Trước mắt là toàn bộ ẩm thực Việt Nam, sau đó mở rộng ra ẩm thực khu vực và thế giới.

1.4. Phương pháp nghiên cứu:

- Phương pháp thực nghiệm khoa học: Dựa vào tập dữ liệu nhận được, tiến hành tiền xử lý và áp dụng thuật toán Stochastic gradient descent để training, sau đó đánh giá hiệu quả của thuật toán dựa trên dữ liệu nhận được.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 Giới thiệu về học máy & học sâu (cần bổ sung lý thuyết):

➤ Giới thiệu:

- Học máy (tiếng Anh: machine learning) là một lĩnh vực của trí tuệ nhân tạo liên quan đến việc nghiên cứu và xây dựng các kỹ thuật cho phép các hệ thống "học" tự động từ dữ liệu để giải quyết những vấn đề cụ thể. Ví dụ như các máy có thể "học" cách phân loại thư điện tử xem có phải thư rác (spam) hay không và tự động xếp thư vào thư mục tương ứng. Học máy rất gần với suy diễn thống kê (statistical inference) tuy có khác nhau về thuật ngữ. Một nhánh của học máy là học sâu phát triển rất mạnh mẽ gần đây và có những kết quả vượt trội so với các phương pháp học máy khác. Học máy có liên quan lớn đến thống kê, vì cả hai lĩnh vực đều nghiên cứu việc phân tích dữ liệu, nhưng khác với thống kê, học máy tập trung vào sự phức tạp của các giải thuật trong việc thực thi tính toán. Nhiều bài toán suy luận được xếp vào loại bài toán NP-khó, vì thế một phần của học máy là nghiên cứu sự phát triển các giải thuật suy luận xấp xỉ mà có thể xử lý được.
- Học máy có hiện nay được áp dụng rộng rãi bao gồm máy truy tìm dữ liệu, chẩn đoán y khoa, phát hiện thẻ tín dụng giả, phân tích thị trường chứng khoán, phân loại các chuỗi DNA, nhận dạng tiếng nói và chữ viết, dịch tự động, chơi trò chơi và cử động rô-bốt (robot locomotion).

➤ Định nghĩa:

- Dưới góc nhìn của trí tuệ nhân tạo, động lực chính học máy bởi là nhu cầu thu nhận tri thức (knowledge acquisition). Thật vậy, trong nhiều trường hợp ta cần kiến thức chuyên gia là khan hiếm (không đủ chuyên gia ngồi phân loại lừa đảo thẻ tín dụng của tất cả giao dịch hàng ngày) hoặc chậm vì một số nhiệm vụ cần đưa ra quyết định nhanh chóng dựa trên xử lý dữ liệu khổng lồ (trong mua bán chứng khoán phải quyết định trong vài khoảnh khắc của giây chẳng hạn) và thiếu ổn định thì buộc phải cần đến máy tính. Ngoài ra,

đại đa số dữ liệu sinh ra ngày nay chỉ phù hợp cho máy đọc (computer readable) tiềm tàng nguồn kiến thức quan trọng. Máy học nghiên cứu cách thức để mô hình hóa bài toán cho phép máy tính tự động hiểu, xử lý và học từ dữ liệu để thực thi nhiệm vụ được giao cũng như cách đánh giá giúp tăng tính hiệu quả.

- Tom Mitchell, giáo sư nổi tiếng của Đại học Carnegie Mellon University - CMU định nghĩa cụ thể và chuẩn mực hơn như sau: "Một chương trình máy tính CT được xem là học cách thực thi một lớp nhiệm vụ NV thông qua trải nghiệm KN, đối với thang đo năng lực NL nếu như dùng NL ta đo thấy năng lực thực thi của chương trình có tiến bộ sau khi trải qua KN" (máy đã học).

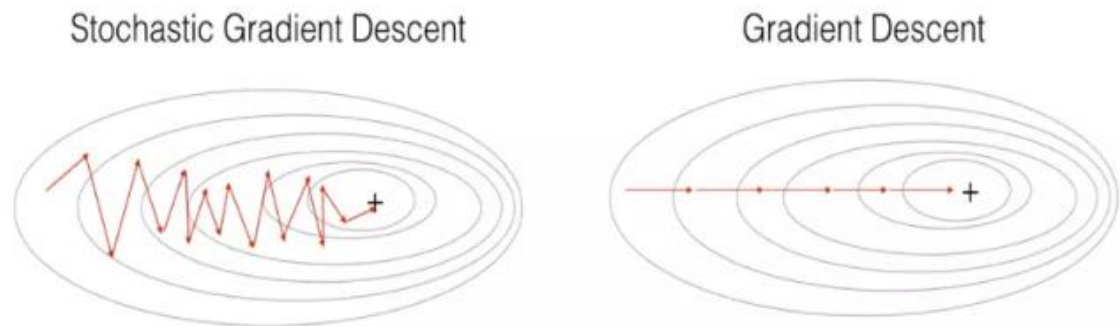
➤ **Biểu diễn:**

- Biểu diễn (tiếng Anh: representation) là một trong những vấn đề quan trọng của học máy. Biểu diễn ở đây có thể hiểu làm sao ghi mã (encode) những thông tin của thế giới thật giúp hoàn thành nhiệm vụ một cách hiệu quả và đầy đủ nhất có thể. Thông tin ở đây bao hàm cả thông tin về dữ liệu đầu vào, đầu ra hay các trạng thái của hệ thống; cũng như cách đánh giá hiệu quả của chương trình.
- Thông thường, trong học máy người ta hay xây dựng các mô hình sử dụng những **biến ngẫu nhiên** cho việc biểu diễn dữ liệu và nội trạng thái của hệ thống. Ví dụ: dùng biến ngẫu nhiên để biểu thị cho tính chất của email là spam (tương ứng giá trị 0) hay là bình thường (tương ứng 1). Mỗi tương quan giữa các biến ngẫu nhiên này có thể sử dụng ví dụ như **mô hình xác suất dạng đồ thị** để miêu tả. Mặt khác, để đo hiệu quả có thể dùng các **hàm thiệt hại** (hay **hàm tiện ích**, trong tiếng Anh là *loss function* và *utility function* tương ứng).

2.2 Giới thiệu về thuật toán tối ưu Stochastic gradient descent (cần bổ sung lý thuyết):

- Trước khi đi sâu vào vấn đề thì chúng ta cần hiểu thế nào là thuật toán tối ưu (optimizers). Về cơ bản, thuật toán tối ưu là cơ sở để xây dựng mô hình neural network với mục đích "học" được các features (hay pattern) của dữ liệu đầu vào, từ đó có thể tìm 1 cặp weights và bias phù hợp để tối ưu hóa model. Nhưng vấn đề là "học" như thế nào? Cụ thể là weights và bias được tìm như thế nào! Đâu phải chỉ cần random (weights, bias) 1 số lần hữu hạn và hy vọng ở 1 bước nào đó ta có thể tìm được lời giải. Rõ ràng là không khả thi và lãng phí tài nguyên! Chúng ta phải tìm 1 thuật toán để cải thiện weight và bias theo từng bước, và đó là lý do các thuật toán optimizer ra đời.

- Stochastic là 1 biến thể của Gradient Descent . Thay vì sau mỗi epoch chúng ta sẽ cập nhật trọng số (Weight) 1 lần thì trong mỗi epoch có N điểm dữ liệu chúng ta sẽ cập nhật trọng số N lần. Nhìn vào 1 mặt , SGD sẽ làm giảm đi tốc độ của 1 epoch. Tuy nhiên nhìn theo 1 hướng khác,SGD sẽ hội tụ rất nhanh chỉ sau vài epoch. Công thức SGD cũng tương tự như GD nhưng thực hiện trên từng điểm dữ liệu.



- Nhìn vào 2 hình trên, ta thấy SGD có đường đi khá là zig zắc, không mượt như GD. Dễ hiểu điều đó vì 1 điểm dữ liệu không thể đại diện cho toàn bộ dữ liệu. Đặt câu hỏi tại sao phải dùng SGD thay cho GD mặc dù đường đi của nó khá zig zắc? Ở đây, GD có hạn chế đối với cơ sở dữ liệu lớn (vài triệu dữ liệu) thì việc tính toán đạo hàm trên toàn bộ dữ liệu qua mỗi vòng lặp trở nên cồng kềnh. Bên cạnh đó GD không phù hợp với online learning. Vậy online learning là gì? online learning là khi dữ liệu cập nhật liên tục (ví dụ như thêm người dùng đăng ký) thì mỗi lần thêm dữ liệu ta phải tính lại đạo hàm trên toàn bộ dữ liệu => thời gian tính toán lâu, thuật toán không online nữa. Vì thế SGD ra đời để giải quyết vấn đề đó, vì mỗi lần thêm dữ liệu mới vào chỉ cần cập nhật trên 1 điểm dữ liệu đó thôi, phù hợp với online learning.
 - Một ví dụ minh họa: có 10.000 điểm dữ liệu thì chỉ sau 3 epoch ta đã có được nghiệm tốt, còn với GD ta phải dùng tới 90 epoch để đạt được kết quả đó.
 - Ưu điểm: Thuật toán giải quyết được đối với cơ sở dữ liệu lớn mà GD không làm được. Thuật toán tối ưu này hiện nay vẫn hay được sử dụng.
 - Nhược điểm: thuật toán vẫn chưa giải quyết được 2 nhược điểm lớn của gradient descent (learning rate, điểm dữ liệu ban đầu). Vì vậy ta phải kết hợp SGD với 1 số thuật toán khác như: Momentum, AdaGrad,... Các thuật toán này sẽ được trình bày ở phần sau.
- ❖ Giới thiệu về bài toán trích xuất đặc trưng
- Giới thiệu:

- Việc trích xuất đặc trưng này giúp chúng ta rất nhiều trong việc rút gọn vấn đề phức tạp thành vấn đề phổ biến, và áp dụng nhiều cách giải quyết khác nhau.
- Sau khi lấy được thông tin của đặc trưng, chúng ta có thể sử dụng thông tin đặc trưng đó cho việc gợi ý đối với từng tập user.

➤ Ý tưởng:

- Trong xử lý ngôn ngữ tự nhiên
 - Nhằm giải quyết vấn đề khó có thể trích xuất các biểu thức toán học trong các trường văn bản phi cấu trúc của tài liệu một cách tự động, nhanh chóng và hiệu quả, một phương pháp dựa trên Mô hình Markov ẩn (HMM) được đề xuất. Thứ nhất, phương pháp này đào tạo mô hình HMM thông qua việc sử dụng các đặc điểm kết hợp ký hiệu của các biểu thức toán học. Sau đó, một số công việc tiền xử lý như xóa nhãn và lọc từ được thực hiện. Cuối cùng, văn bản đã được xử lý trước được chuyển đổi thành một chuỗi quan sát làm đầu vào của mô hình HMM để xác định đâu là biểu thức toán học và trích xuất nó. Kết quả thực nghiệm cho thấy phương pháp đề xuất có thể trích xuất hiệu quả các biểu thức toán học từ các trường văn bản của tài liệu, đồng thời có tỷ lệ chính xác và tỷ lệ thu hồi tương đối cao, được tham khảo từ [nguồn này](#).
 - Neural Networks for Equation Extraction - Nhận dạng ký tự quang học chuyên giải mã ngôn ngữ tự nhiên từ một hình ảnh. Một số triển khai đẹp mắt của điều này đã hoạt động và được sử dụng rộng rãi để đạt được các nhiệm vụ khác nhau. Một trong những cái phổ biến hơn là thư viện python pytesseract. Đưa ra bất kỳ hình ảnh nào làm đầu vào, nó có thể giải mã chính xác thành văn bản bên dưới từ hình ảnh đó. Tuy nhiên, giải mã các phương trình từ một hình ảnh hoặc định dạng tệp pdf là một công việc tế nhị vì tệp không chứa bất kỳ thông tin được gắn thẻ nào về nội dung toán học của nó. Ngoài ra, pdf có thể là một phần tổng hợp của nhiều nguồn khác nhau như văn bản, hình ảnh, latex, v.v. Ngoài ra, các công thức toán học trong tài liệu PDF có thể được biểu thị một phần bằng các khối đồ họa được hiển thị trực tiếp từ các ký tự PDF, giữ nguyên hình dạng chính xác nhưng bỏ sót nghĩa nội dung.
- Classical AI Scheduling job apply Linear and Mixed-Integer Programming:
 - Lấy ý tưởng từ một dạng AI đời đầu, để giải quyết một vấn đề nào đó, chúng ta chuyển về dạng toán học và giải quyết nó một cách tối ưu nhất. được [tham khảo bởi](#).

➤ Các bước thực hiện:

- Đánh giá vấn đề và đưa ra phương diện và đưa ra công thức toán học.
 - Một trong các thuộc tính của món ăn thì chúng ta có thời gian nấu, calo của món ăn đó, hương vị
 - Chúng ta sẽ chọn hương vị món ăn làm thuộc tính quan trọng để trích xuất đặc trưng của món ăn đó.
 - Công thức mà ta sẽ áp dụng để tính đặc trưng

$$T_a \times F_a + T_b \times F_b + T_c \times F_c + \dots T_n \times F_n = T_{\text{food}} \times F_{\text{food}}$$

- Trong đó:
 - $T_{\langle \text{nguyên liệu} | \text{món ăn} \rangle}$: thông số hương vị chúng ta có thể biểu diễn dưới dạng ma trận 9 phần tử bao gồm và chúng ta có thể định nghĩa bằng tay áp dụng cho nguyên liệu và món ăn:
 - Ngọt
 - Mặn
 - Chua
 - Đắng
 - Béo vị phô mai
 - Nhạt
 - Cay nồng
 - Có vị tỏi
 - Vị xông khói
 - $F_{\langle \text{nguyên liệu} | \text{món ăn} \rangle}$: ma trận đặc trưng của nguyên liệu và món ăn là những tham số chúng ta cần đi tìm.

● Bởi vì với duy nhất một phương trình trên, chúng ta ko thể đi tìm được ma trận đặc trưng của cả các nguyên liệu và món ăn đó, nhưng bên cạnh đó thì chúng ta có những món ăn khác cũng xài lại nguyên liệu đó nên chúng ta có thể tổng hợp các phương trình lại và có thể hiểu định dạng đơn giản như thế này:

- $2x + 3y = 6$

- $7x - 8y = 0$

● Bởi vì mỗi món ăn được sử dụng với danh sách nguyên liệu khác nhau, để đơn giản hóa vấn đề chúng ta sẽ thêm một thông số để đại diện cho sự tồn tại của nguyên liệu đó trong một món ăn.

- $E_a \times T_a \times F_a + E_b \times T_b \times F_b + E_c \times T_c \times F_c + \dots E_n \times T_n \times F_n = T_{\text{food}} \times F_{\text{food}}$

- Ví dụ: $1 * \text{hành} + 0 * \text{rừng} + \dots = \text{lẩu}$

- với công thức trên thì lâu sẽ không bao gồm rùng.

- Sau khi định nghĩa được phương trình tìm các tham số, chúng ta cần phải có một objective function để tìm các tham số tối ưu nhất, chúng ta chỉ cần chuyển về

- $$(T_a \times F_a + T_b \times F_b + T_c \times F_c + \dots T_n \times F_n) / F_{\text{food}} - T_{\text{food}} = 0$$

- Có thể phương trình của chúng ta sao khi tính toán thì sẽ không bằng không, mục tiêu chúng ta cần phương trình đó tiến về không.

➤ Công nghệ áp dụng:

- Để giải được một phương trình phức tạp mà thực hiện đơn giản, chúng ta sử dụng thư viện pytorch để định nghĩa vài giải nhanh
- Thuật toán tối ưu sử dụng Stochastic gradient descent dùng để tối ưu các tham số cần tìm

❖ Giới thiệu về bài toán gợi ý:

➤ Giới thiệu:

- Collaborative Filtering (Lọc cộng tác) là thực hiện tư vấn (gợi ý) các sản phẩm, dịch vụ, nội dung cho một người dùng nào đó dựa trên mối quan tâm, sở thích (preferences) của những người dùng tương tự đối với các sản phẩm, dịch vụ, nội dung đó.

➤ Ý tưởng:

- Ý tưởng cơ bản của thuật toán này là dự đoán mức độ yêu thích của một user đối với một item dựa trên các users khác “gần giống” với user đang xét. Việc xác định độ “giống nhau” giữa các users có thể dựa vào mức độ quan tâm (rating) của các users này với các items khác mà hệ thống đã biết trong quá khứ.

➤ Ví dụ:

- Hai users A, B đều thích các phim về cảnh sát hình sự (tức là đều đánh giá các bộ phim thuộc thể loại này 4 -> 5 sao). Dựa vào lịch sử xem phim của B, ta thấy B thích bộ phim “Người phán xử”, vậy nhiều khả năng A cũng thích phim này, từ đó hệ thống sẽ đề xuất “Người phán xử” cho A.

➤ Có 2 hướng tiếp cận Collaborative Filtering:

- Một là xác định mức độ quan tâm của mỗi user tới một item dựa trên mức độ quan tâm của users gần giống nhau (similar users) tới item đó còn được gọi là User-user collaborative filtering.

- Hai là thay vì xác định user similarities, hệ thống sẽ xác định item similarities. Từ đó, hệ thống gợi ý những items gần giống với những items mà user có mức độ quan tâm cao.
- Trong đồ án này chúng ta sẽ thiết kế hệ thống theo hướng user-user.
- Các bước thực hiện
 - Bước 1: Khởi tạo ma trận dữ liệu
 - Đối với Collaborative filtering, chúng ta sử dụng 3 thành phần dữ liệu là user, movies và ratings (1), cụ thể:
 - Users: là danh sách người dùng
 - Foods: là danh sách các món ăn, mỗi món ăn có thể kèm theo thông tin mô tả.
 - Ratings: là số điểm user đánh giá cho các món ăn để lưu trữ được 3 thành phần dữ liệu trên, chúng ta tiếp tục sử dụng module read_csv của Pandas để lưu trữ, chúng ta gọi là ma trận user – items. Cụ thể, ma trận user - items biểu diễn mức độ quan tâm (rating) của user với mỗi item.
 - Tập hợp tất cả các *ratings*, bao gồm cả những giá trị chưa biết cần được dự đoán, tạo nên một ma trận gọi là *utility matrix*.
 - Ví dụ:

		users						
		u0	u1	u2	u3	u4	u5	u6
items	i0	5	5	2	0	1	?	?
	i1	4	?	?	0	?	2	?
	i2	?	4	1	?	?	1	1
	i3	2	2	3	4	4	?	4
	i4	2	0	4	?	?	?	5

- Ma trận này có rất nhiều các giá trị miss. Nhiệm vụ của hệ thống là dựa vào các ô đã có giá trị trong ma trận trên (dữ liệu thu được từ trong quá khứ), thông qua mô hình đã được xây dựng, dự đoán các ô còn trống (của user hiện hành), sau đó sắp xếp kết quả dự đoán (ví dụ, từ cao xuống thấp) và chọn ra Top-N items theo thứ tự rating giảm dần, từ đó gợi ý chúng cho người dùng.
- Bước 2: Chuẩn hóa ma trận dữ liệu
 - Để có thể sử dụng ma trận này vào việc tính toán, chúng ta cần thay những dấu ‘?’ bởi một giá trị. Đơn giản nhất có thể thay giá trị ‘0’ hay một cách

khác là '2.5' – giá trị trung bình giữa 0 và 5. Tuy nhiên, cách tính này có độ chính xác thấp vì những giá trị này sẽ hạn chế với những users dễ hoặc khó tính. Thay vào đó, ta sử dụng giá trị trung bình cộng ratings của mỗi user bằng cách trừ ratings của mỗi user cho giá trị trung bình ratings tương ứng của user đó và thay dấu '?' bằng giá trị 0.

trừ các giá trị của ma trận với giá trị trung bình

	users						
	u0	u1	u2	u3	u4	u5	u6
i0	5	5	2	0	1	?	?
i1	4	?	?	0	?	2	?
i2	?	4	1	?	?	1	1
i3	2	2	3	4	4	?	4
i4	2	0	4	?	?	?	5
↓	↓	↓	↓	↓	↓	↓	↓
trung bình i	3.2	2.75	2.5	1.33	2.5	1.5	3.33

thay dấu ? bằng 0

	users						
	u0	u1	u2	u3	u4	u5	u6
i0	1.75	2.25	-0.5	-1.33	-1.5	?	?
i1	0.75	?	?	-1.33	?	0.5	?
i2	?	1.2	-1.5	?	?	-0.5	-2.33
i3	-1.25	-0.75	0.5	2.67	1.5	?	0.67
i4	-1.25	-2.75	1.5	?	?	?	1.67

thay dấu ? bằng 0

	users						
	u0	u1	u2	u3	u4	u5	u6
i0	1.75	2.25	-0.5	-1.33	-1.5	0	0
i1	0.75	0	0	-1.33	0	0.5	0
i2	0	1.2	-1.5	0	0	-0.5	-2.33
i3	-1.25	-0.75	0.5	2.67	1.5	0	0.67
i4	-1.25	-2.75	1.5	0	0	0	1.67

- Mục đích của cách xử lý này là: – Phân loại ratings thành 2 loại: giá trị âm (user không thích item) và dương (user thích item). Các giá trị bằng 0 là những item chưa được đánh giá.
- Bước 3: Tính toán độ tương đồng
 - Sau khi chuẩn hóa ma trận Utility, ta tính toán độ tương đồng giữa các users. Chúng ta sử dụng hàm cosine similarity (hàm có sẵn của thư viện sklearn của Python). Dưới đây là công thức của cosine similarity:

$$\text{cosine_similarity}(\mathbf{u}_1, \mathbf{u}_2) = \cos(\mathbf{u}_1, \mathbf{u}_2) = \frac{\mathbf{u}_1^T \mathbf{u}_2}{\|\mathbf{u}_1\|_2 \cdot \|\mathbf{u}_2\|_2}$$

- Kết quả sau khi tính toán độ tương đồng (độ tương đồng là 1 con số trong khoảng -1 đến 1. Càng gần 1 thì càng tương đồng, càng gần -1 thì càng đối lập).

		users						
		u0	u1	u2	u3	u4	u5	u6
items	u0	1	0.83	-0.58	-0.79	-0.82	0.2	-0.38
	u1	0.83	1	-0.87	-0.4	-0.55	-0.23	-0.71
	u2	-0.58	-0.87	1	0.27	0.32	-0.47	0.96
	u3	-0.79	-0.4	0.27	1	0.87	-0.29	0.18
	u4	-0.82	-0.55	0.32	0.87	1	0	0.16
	u5	0.2	-0.23	0.47	-0.29	0	1	0.56
	u6	-0.38	-0.71	0.96	0.18	0.16	0.56	1

▪ Bước 4: Dự đoán ratings

- Ta sẽ dự đoán ratings của một user với mỗi item dựa trên k users gần nhất (neighbor users), tương tự như phương pháp K-nearest neighbors (KNN).

$$\hat{y}_{i,u} = \frac{\sum_{u_j \in \mathcal{N}(u,i)} \bar{y}_{i,u_j} \text{sim}(u, u_j)}{\sum_{u_j \in \mathcal{N}(u,i)} |\text{sim}(u, u_j)|}$$

- Trong đó, $\mathcal{N}(u, i)$ là tập k users gần nhất (có độ tương đồng cao nhất) với user u và đã từng đánh giá item i.
- Ví dụ: Chúng ta dự đoán normalized rating của user u1 cho item i1 với k = 2 là số users gần nhất.
 - ◆ Bước 1: Xác định các users đã rated cho i1, đó là u0, u3, u5.
 - ◆ Bước 2: Lấy similarities của u1 với u0, u3, u5. Kết quả lần lượt là: {u0, u3, u5: {0.83, -0.4, -0.23}}. Với k = 2. Chọn 2 giá trị lớn nhất là 0.83 và -0.23, tương ứng với các users u0, u5. Hai users này có normalized ratings với i1 là: {u0, u5} {0.75, 0.5}.
 - ◆ Bước 3: Tính normalized ratings theo công thức:

$$\hat{y}_{i_1, u_1} = \frac{0.83 \times 0.75 + (-0.23) \times 0.5}{0.83 + |-0.23|} \approx 0.48$$

- Thực hiện dự đoán cho các trường hợp missing ratings (chưa có dự đoán), ta sẽ thu được ma trận normalized ratings matrix như ví dụ: Cuối cùng,

cộng lại các giá trị ratings với ratings trung bình (ở bước chuẩn hóa) theo từng cột. Chúng ta sẽ thu được ma trận hoàn thiện.

		users						
		u0	u1	u2	u3	u4	u5	u6
items	i0	1.75	2.25	-0.5	-1.33	-1.5	0.18	-0.63
	i1	0.75	0.48	-0.17	-1.33	-1.33	0.5	0.05
	i2	0.91	1.2	-1.5	-1.84	-1.78	-0.5	-2.33
	i3	-1.25	-0.75	0.5	2.67	1.5	0.59	0.67
	i4	-1.25	-2.75	1.5	1.57	1.56	1.59	1.67

- Ta cũng có thể cộng các cột của ma trận hoàn thiện với giá trị rating trung bình của mỗi *user* như đã tính trong hình trước khi chuẩn hóa. (không cần tính cũng được)

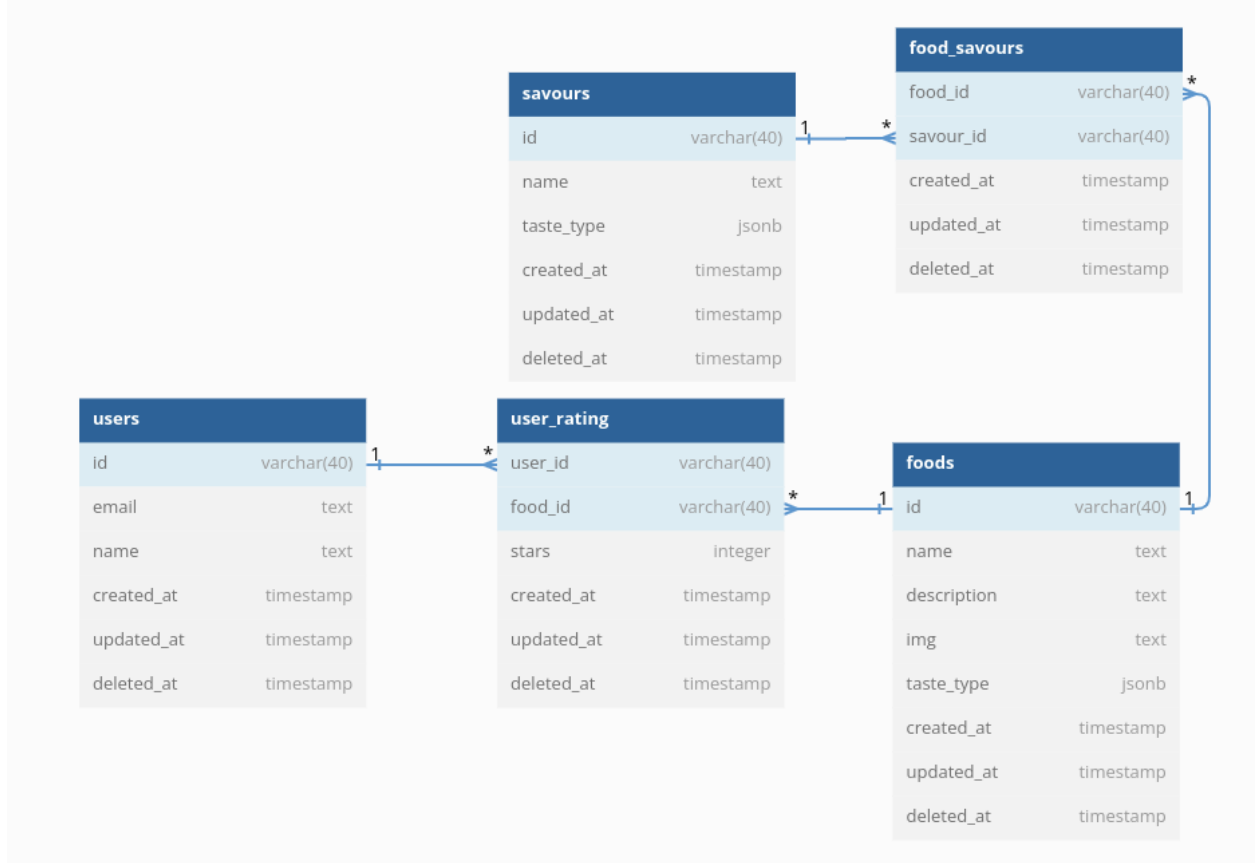
	u_0	u_1	u_2	u_3	u_4	u_5	u_6
i_0	5	5	2	0	1	1.68	2.70
i_1	4	3.23	2.33	0	1.67	2	3.38
i_2	4.15	4	1	-0.5	0.71	1	1
i_3	2	2	3	4	4	2.10	4
i_4	2	0	4	2.9	4.06	3.10	5

- Bước 5: Lấy ra top các items mà user sẽ thích.
 - Sau khi chúng tôi đã dự đoán được ratings của các món ăn mà người dùng sẽ đánh giá, chúng ta lấy ra được top món ăn mà user sẽ thích theo các bước:
 - Lấy id người dùng.
 - Nhận danh sách rating của người dùng đối với các món ăn chưa xem.
 - Sắp xếp danh sách các bộ giá trị rating đó.
 - Trả về các tiêu đề tương ứng với chỉ số của các phần tử trên cùng.

CHƯƠNG 4: THIẾT KẾ HỆ THỐNG

4.1 Thiết kế cơ sở dữ liệu:

4.1.1 Sơ đồ diagram:



4.1.2 Từ điển dữ liệu:

- Bảng **foods**:

STT	Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
1	id	varchar	40	Primary key	Identity
2	name	text			Tên món ăn
3	description	text			Mô tả món ăn
4	img	text			Ảnh món ăn
5	taste_type	jsonb			Loại hương vị
6	created_at	timestamp			Thời gian tạo món ăn
7	updated_at	timestamp			Thời gian cập nhật món ăn
8	deleted_at	timestamp			Thời gian xóa món ăn

- Bảng **food_savours**:

STT	Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
1	food_id	varchar(40)	40	Primary key	Identity
2	savour_id	varchar(40)	40	Foreign key	Identity hương vị
3	created_at	timestamp			Thời gian tạo hương vị cho món ăn
4	updated_at	timestamp			Thời gian cập nhật hương vị cho món ăn
5	deleted_at	timestamp			Thời gian xóa hương vị cho món ăn

- Bảng **savours**:

STT	Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
1	id	varchar(40)	40	Primary key	Identity
2	name	varchar(40)	40	Foreign key	Identity hương vị
3	taste_type	jsonb			Loại hương vị
4	created_at	timestamp			Thời gian tạo nguyên liệu
5	updated_at	timestamp			Thời gian cập nhật nguyên liệu
6	deleted_at	timestamp			Thời gian xóa nguyên liệu

- Bảng **user_rating**:

STT	Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
1	user_id	varchar(40)	40	Primary key	Identity người dùng
2	food_id	varchar(40)	40	Foreign key	Identity món ăn
3	stars	integer			Số sao đánh giá cho món ăn
4	created_at	timestamp			Thời gian thêm đánh giá cho món ăn
5	updated_at	timestamp			Thời gian cập nhật đánh giá

					cho món ăn
6	deleted_at	timestamp			Thời gian xóa đánh giá cho món ăn

- Bảng **users**:

STT	Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
1	id	varchar(40)	40	Primary key	Identity
2	email	text			Email đăng nhập của người dùng
3	name	text			Tên người dùng
4	created_at	timestamp			Thời gian tạo tài khoản
5	updated_at	timestamp			Thời gian cập nhật tài khoản
6	deleted_at	timestamp			Thời gian xóa tài khoản

4.2 Giao diện hệ thống:


4.2.1 Giao diện đăng nhập, nhập tên người dùng:

Đăng nhập

Nhập email

Nhập email của bạn

Đăng nhập



Hình 1: Giao diện người dùng đăng nhập vào hệ thống bằng email


Mô tả:

- + Người dùng đăng nhập vào hệ thống thông qua email, không cần mật khẩu, với lần đầu đăng nhập sẽ được tính là đăng ký tài khoản cho email người dùng nhập vào.
- + Với những lần đăng nhập sau người dùng chỉ cần nhập đúng email lần trước đã dùng để đăng nhập vào lại tài khoản của người dùng trong hệ thống.

Nhập tên

Nhập tên của bạn

Xác nhận



Hình 2: Giao diện nhập tên người dùng

Mô tả: Lần đầu đăng nhập, người dùng sẽ đặt tên cho tài khoản của họ và sẽ không được thay đổi trong suốt quá trình sử dụng.

Cùng tìm ra món ăn bạn yêu thích

Chọn hương vị

Chọn nguyên liệu

Chọn hương vị

Chọn nguyên liệu

MặnBéoĐắng

Thịt heo, Củ cải, Nước mắm

Tìm kiếm

M

Minh Quang


Đổi email

Đăng món ăn

Có thể bạn sẽ thích

Được nhiều người thích

chocolate *đặc biệt*




Kem Chocolate

NgọtBéo sữa

Đánh giá

★★★★☆16.5k

chocolate *đặc biệt*




Kem Chocolate

NgọtBéo sữa

Đánh giá

★★★★☆16.5k

chocolate *đặc biệt*




Kem Chocolate

NgọtBéo sữa

Đánh giá

★★★★☆16.5k

chocolate *đặc biệt*




Kem Chocolate

NgọtBéo sữa

Đánh giá

★★★★☆16.5k

chocolate *đặc biệt*



Kem Chocolate

NgọtBéo sữa

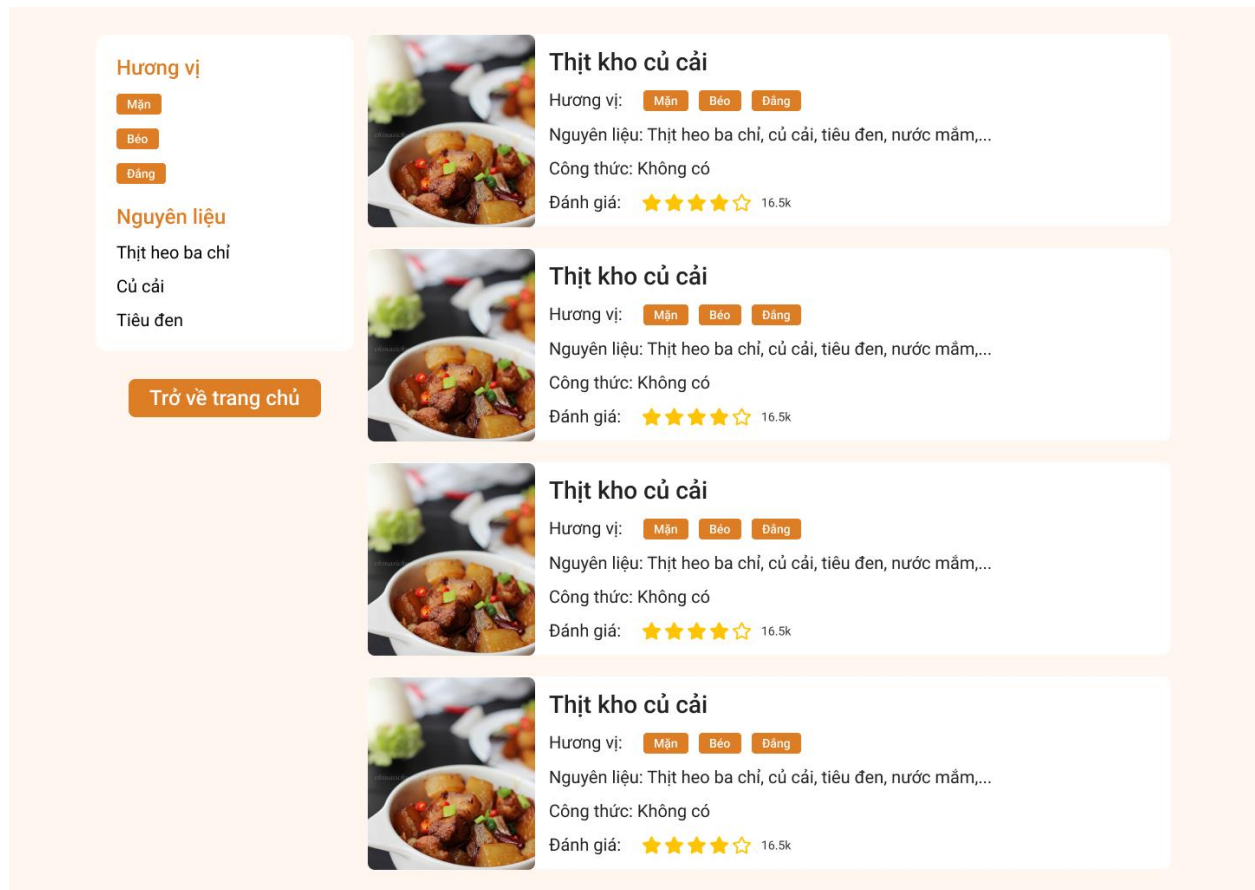
Đánh giá

★★★★☆16.5k

Hình 3: Giao diện trang chủ của hệ thống

Mô tả:

- + Giao diện trang chủ của hệ thống bao gồm chức năng chính phục vụ cho người dùng tìm kiếm món ăn và chức năng gợi ý những món ăn liên quan đến sở thích của người dùng hoặc những món ăn được nhiều người thích nhất.
- + Người dùng sẽ thực hiện các thao tác như chọn hương vị mình muốn và chọn nguyên liệu liên quan để hệ thống tiến hành phân tích và tìm ra những món ăn phù hợp với nhu cầu của người dùng.



Hình 4: Giao diện danh mục các món ăn đã được hệ thống lọc theo yêu cầu của người dùng

Mô tả:

- + Sau khi người dùng đã chọn các yêu cầu liên quan như hương vị và nguyên liệu thì hệ thống sẽ tiến hành phân tích và đưa ra những món ăn phù hợp với yêu cầu của người dùng.
- + Danh mục các món ăn sẽ được sắp xếp giảm dần theo số lượng sao do người dùng đánh giá đối với món ăn tức là hệ thống sẽ ưu tiên hiển thị những món ăn được nhiều người đánh giá cao nhất theo yêu cầu.



Thịt kho củ cải trắng

Hương Vị: Mặn Béo Đắng

Nguyên liệu: Thịt heo ba chỉ, củ cải trắng, nước mắm, đường, bột ngọt, tiêu đen, ớt trái, hành tím, dầu ăn, dầu hào, hạt nêm, bột ngọt

Ngày thêm: 12/12/2022

Được thêm bởi: Admin

Đánh giá: ★★★★★ 4.9 | 15.5k

Đánh giá của bạn: ☆☆☆☆☆

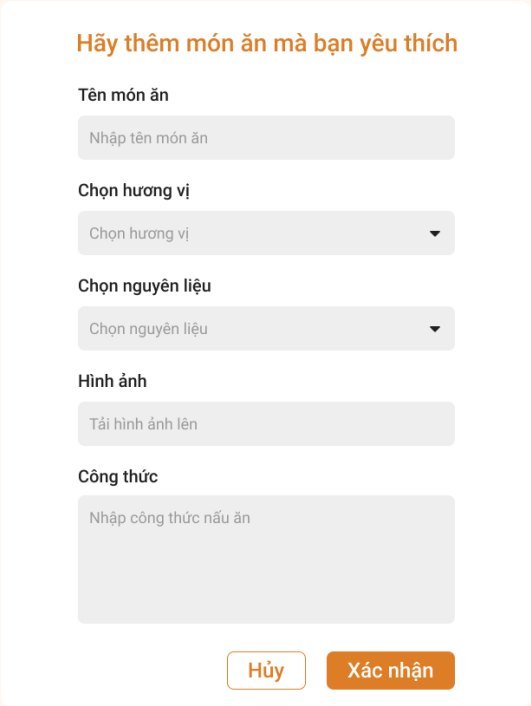
Công thức

Hiện tại chưa có công thức cho món ăn này

Hình 5: Giao diện thông tin món ăn

Mô tả:

- + Giao diện hiển thị các thông tin liên quan đến món ăn như hương vị, nguyên liệu, ngày thêm món ăn, người thêm món ăn, lượt đánh giá của món ăn đó.
- + Giao diện cho phép người dùng đánh giá số lượng sao cho món ăn tùy theo cảm nhận của người dùng và nếu món ăn có công thức thì sẽ hiển thị công thức nấu của món ăn.



Hãy thêm món ăn mà bạn yêu thích

Tên món ăn

Nhập tên món ăn

Chọn hương vị

Chọn hương vị

Chọn nguyên liệu

Chọn nguyên liệu

Hình ảnh

Tải hình ảnh lên

Công thức

Nhập công thức nấu ăn

Hủy Xác nhận

Hình 6: Form chức năng thêm món ăn mới của người dùng

Mô tả: Form dành cho người dùng thêm món ăn mới vào hệ thống, form cho phép người dùng nhập những thông tin như tên món ăn, hương vị, nguyên liệu