Hệ thống khuyến nghị các khoá học trực tuyến trên Coursera

Phạm Thảo Nhi[1,2] and Trịnh Linh Chi[1,2]

[1]University of Information Technology - VNUHCM, Ho Chi Minh City, Viet Nam.
[2]University of Information Technology - VNUHCM, Ho Chi Minh City, Viet Nam.

Contributing authors: 19520815@gm.uit.edu.vn, 19521285@gm.uit.edu.vn;

Tóm tắt nội dung

Hệ thống khuyến nghị (Recommender Systems – RS) thường được sử dụng để dự đoán sở thích của người dùng dựa vào những phản hồi của họ nhằm gợi ý các sản phẩm (item) mà người dùng có thể thích. RS hiện đang được ứng dụng ở rất nhiều lĩnh vực khác nhau như thương mại điện tử (bán hàng trực tuyến), giải trí (âm nhạc, phim ảnh,...), giáo dục (gợi ý nguồn tài nguyên học tập như khoá học, sách, báo,...). Trong bài báo cáo này, chúng tôi sẽ xây dựng Hệ thống khuyến nghị khoá học trực tuyến trên Coursera thông qua các phương pháp tiếp cận là lọc dựa trên nội dung - Content-based filtering (CBF), lọc cộng tác – Collaborative filtering (CF) và phương pháp Bayesian Personalized Ranking (BPR).

Keywords: Courses Recommender System, Content-based filtering, Collaborative filtering, Bayesian Personalized Ranking

1 Giới thiệu

Ngày nay, với sự bùng nổ và phát triển mạnh mẽ của Internet, mọi người có thể dễ dàng truy cập vào các website để tự do tìm kiếm các mặt hàng, dịch vụ mà họ yêu thích hay có nhu cầu sử dụng. Tuy nhiên, khối lượng thông tin khổng lồ có sẵn trên web dẫn đến vấn đề quá tải thông tin. Việc xác định sở thích của mọi người và giúp họ đưa ra quyết định trở thành một thách thức

lớn để xây dựng các hệ thống tư vấn đối phó với tình trạng quá tải thông tin. Hệ thống khuyến nghị (Recommender Systems) sẽ làm cho quy trình lựa chọn sản phẩm của người dùng trở nên dễ dàng và thuận tiện hơn.

Trong những năm gần đây, sự xuất hiện của các nền tảng giáo dục trực tuyến và các khóa học trực tuyến mở lớn (MOOCs) đã thu hút sự quan tâm rộng rãi của người dùng. Tính đến cuối năm 2019, hơn 900 trường đại học đã cung cấp MOOC với 13500 khóa học có sẵn và khoảng 110 triệu sinh viên đăng ký [1]. Việc thành lập các nền tảng MOOC khác nhau bao gồm XuetangX, Coursera, Udacity, EdX,... đã cung cấp một nền giáo dục thuận tiện cho hơn 100 triệu người dùng trên khắp thế giới và mang đến cơ hội tiếp cận các khóa học xuất sắc ở nhiều trường đại học hàng đầu với chi phí thấp. Học trực tuyến tuy mang lại nhiều tiện ích nhưng cũng dẫn đến vấn đề quá tải thông tin ngày càng nghiêm trọng khiến cho mọi người dần trở nên khó khăn trong việc lựa chọn các khóa học phù hợp để học. Vì thế, trong bài báo cáo này, chúng tôi sẽ xây dựng Hệ thống khuyến nghị các khoá học trực tuyến trên Coursera. Chúng tôi sẽ sử dụng ba phương pháp thử nghiệm là lọc dựa trên nội dung - Content-based filtering (CBF), lọc cộng tác – Collaborative filtering (CF) và Bayesian Personalized Ranking (BPR) để gợi ý khoá học cho người dùng.

Phần còn lại của bài báo cáo này được tổ chức như sau. Phần 2 trình bày tổng quan về các nghiên cứu liên quan. Phần 3 mô tả chi tiết về bộ dữ liệu. Phần 4 trình bày các hướng tiếp cận đối với bài toán. Thực nghiệm và kết quả được trình bày ở phần 5. Cuối cùng, trong phần 6, chúng tôi đưa ra kết luận và hướng phát triển cho các nghiên cứu trong tương lai.

2 Nghiên cứu liên quan

Hầu hết các nghiên cứu trước đây cũng đã sử dụng các phương pháp khuyến nghị truyền thống bao gồm lọc dựa trên nội dung (CBF), lọc cộng tác (CF) cùng với các thuật toán kết hợp (hybrid) để đề xuất các khóa học cho người dùng. Cụ thể là, Boratto và cộng sự [2] đã sử dụng phương pháp lọc cộng tác, hệ số hóa ma trận (matrix factorization) và đề xuất kết hợp (hybrid) để tính toán các đề xuất cho người học. Mục đích của công việc này là để xác định ảnh hưởng của mức độ phổ biến của một mục (item) đối với kết quả đề xuất và nghiên cứu cho thấy các mục phổ biến ảnh hưởng đến danh sách đề xuất của hệ thống đề xuất. Yanhui và cộng sự [3] đã sử dụng các hệ thống đề xuất lọc cộng tác và lọc dựa trên nội dung để phân loại các khóa học dựa trên thông tin khóa học, sau đó các khóa học và người học được chia thành các cụm mờ (fuzzy clusters) với lọc cộng tác được sử dụng để phân cụm. Mỗi khóa học và người học có thể được liên kết trong nhiều cụm và nếu một người học và khóa học thuộc nhiều cụm tương đồng, điều đó có nghĩa là chúng tương đồng nhau và người học có thể quan tâm đến khóa học đó. Symeonidis và cộng sự [4] đã thiết kế một hệ thống đề xuất khóa học sử dụng ma trận nhiều chiều, chẳng hạn như hệ số hóa ma trận với bộ lọc cộng tác được gọi là xSVD++. Thuật toán này sử dụng thông tin từ các nguồn bên ngoài, kỹ năng người học và đặc điểm của khóa học để dự đoán xu hướng và xếp hạng của khóa học, sau đó

sử dụng xếp hạng khóa học, kỹ năng của người học và kỹ năng của khóa học để thực hiện các đề xuất.

3 Dataset

3.1 Mô tả bộ dữ liệu

3.1.1 Bộ dữ liệu Course Reviews trên kaggle

Course Reviews [1] bao gồm hai tập dữ liệu: Coursera courses và Coursera reviews.

- Coursera courses chứa danh sách 622 khoá học trực tuyến có trên Coursera, gồm 4 thuộc tính:
  - name: tên khoá học.
  - institution: tên tổ chức, đối tác cung cấp khoá học.
  - course_url: URL tới trang chủ của khoá học.
  - course_id: mã khoá học.

name	institution	course_url	course_id
Machine Learning	Stanford University	https://www.coursera.org/learn/machine-learning	machine-learning

Bảng 1 Ví dụ về dữ liệu trong Coursera courses.

- Coursera reviews chứa danh sách 1454711 đánh giá cùng xếp hạng của người dùng đối với các khoá học, gồm 5 thuộc tính:
  - reviews: đánh giá của người dùng đối với các khoá học.
  - reviewers: tên của người dùng đã viết đánh giá.
  - date_reviews: ngày đánh giá được đăng.
  - rating: xếp hạng của người dùng với khoá học, có giá trị trên thang điểm từ 1 – 5.
  - course_id: mã khoá học.

reviews	reviewers	date_reviews	rating	course_id
Good course but difficult for Non Science background people. It has quite a bit of Maths in it.	By FARHAAN H	9-May-20	4	negotiation

Bảng 2 Ví dụ về dữ liệu trong Coursera reviews.

3.1.2 Bộ dữ liệu Crawled Courses

Crawled Courses bao gồm 10255 khoá học trực tuyến được chúng tôi crawl trên website Coursera [2], gồm 10 thuộc tính:

- name: tên khoá học.

[1] https://www.kaggle.com/datasets/imuhammad/course-reviews-on-coursera
[2] https://www.coursera.org/directory/courses

- url: URL tới trang chủ của khoá học.
- category: thể loại của khoá học.
- subcategory: thể loại con của khoá học.
- instructors: tên người hướng dẫn khoá học.
- description: mô tả khoá học.
- enrollment: số lượng người dùng đã đăng ký khoá học.
- views: số lượng lượt xem của khoá học.
- rating: xếp hạng trung bình của khoá học.
- raters: số lượng người đã xếp hạng khoá học.

name	course_id	category	instructors	description	rating
Create a Budget with Google Sheets	Business	Finance	Jamie Schroeder	By the end of this project...	4.7

Bảng 3 Ví dụ về dữ liệu trong Courses.

3.2 Tiền xử lý dữ liệu

Sau khi thu thập hai bộ dữ liệu được nêu ở phần 3.1, chúng tôi tiến hành kết hợp chúng lại với nhau để tạo ra bộ dữ liệu mới là Courses và Ratings.

Đầu tiên, chúng tôi sẽ tạo một tập dữ liệu với tên là Courses bằng cách merge tập Coursera courses và tập Crawled Courses bằng thuộc tính ('course_url' = 'url') để lấy ra những khoá học chung có ở cả hai bộ dữ liệu. Sau đó đổi tên thuộc tính 'rating' thành 'avg_rating' và thêm một cột thuộc tính mới là 'courseId' vào bộ dữ liệu mới (trong đó, 'courseId' là mã khoá học đã được mã hoá từ 'course_id' thành số). Tập dữ liệu Courses vừa mới được tạo sẽ chứa thông tin của các khoá học trực tuyến trên Coursera, bao gồm 588 khoá học và 13 thuộc tính (course_url, course_id, name, category, subcategory, instructors, description, enrollment, views, avg_rating, raters, courseId).

name	course_id	courseId	instructors	avg_rating
Stanford University	machine-learning	334	Andrew Ng, Eddy Shyu, Aarti Bagul, Geoff Ladwig	4.9

Bảng 4 Ví dụ về dữ liệu trong Courses.

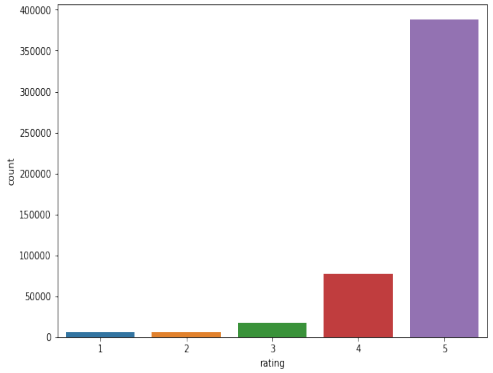
Tiếp theo, chúng tôi sẽ sẽ tạo thêm một tập dữ liệu mới với tên là Ratings bằng cách merge tập Coursera reviews và tập Courses bằng thuộc tính 'course_id' (mã khoá học) và thêm một cột thuộc tính mới là 'userId' vào bộ dữ liệu mới (trong đó, 'userId' là mã người dùng đã được mã hoá từ 'reviewers' thành số). Sau khi thực hiện các bước nêu trên, chúng tôi nhận thấy rằng, có nhiều đánh giá khoá học bị trùng lặp trong bộ dữ liệu, vì vậy chúng tôi tiến hành loại bỏ các dòng dữ liệu trùng bằng lệnh drop_duplicates(). Tiếp tục quan sát bộ dữ liệu, thấy được rằng có một số người dùng đã đánh giá một khoá học nhiều lần, để xử lý vấn đề này, chúng tôi quyết định sẽ chỉ giữ lại dòng dữ liệu có ngày đánh giá được đăng (date_reviews) mới nhất và loại bỏ các dòng còn lại. Tập dữ liệu Ratings vừa mới được tạo sẽ chứa thông tin của các đánh giá và xếp hạng của người dùng đối với các khoá học,

bao gồm 493274 dòng dữ liệu và 6 thuộc tính (reviews, reviewers, userId, courseId, date_reviews, rating).

reviewers	userId	date_reviews	rating	course_id
By Jon M	108516	1-Apr-16	4	565

Bảng 5 Ví dụ về dữ liệu trong Ratings.

Hình 1 là biểu đồ phân bố xếp hạng khoá học của người dùng (rating) theo các mức từ 1 - 5 sao. Có thể thấy dữ liệu rating phân bố không đồng đều, tập trung vào các mức 4 và 5, trong khi đó các mức 1, 2, 3 được đánh giá rất ít.



Hình 1 Biểu đồ phân bố xếp hạng khoá học của người dùng (rating).

4 Phương pháp tiếp cận

4.1 Content-based filtering

Lọc dựa trên nội dung là một hệ thống đề xuất cho người dùng những items tương tự với nhứng items đã được người dùng đánh giá cao từ trước đó.

Tính năng lọc dựa trên nội dung đưa ra các đề xuất bằng cách sử dụng các từ khóa và thuộc tính của các đối tượng trong cơ sở dữ liệu và đối sánh chúng với hồ sơ người dùng. Hồ sơ người dùng được tạo dựa trên dữ liệu thu được từ các hành vi của người dùng, ví dụ như mua hàng, xếp hạng (thích hoặc không thích), download, các item được tìm kiếm trên trang web hoặc được đặt trong giỏ hàng,... Dựa vào đó, hệ thống đề xuất sẽ tạo ra một mô hình duy nhất cho sở thích của từng người dùng. Mô hình bao gồm các thuộc tính mà người dùng có thể thích hoặc không thích dựa trên các hành vi trong quá khứ. Các mô hình người dùng sau đó được so sánh với tất cả các đối tượng trong cơ sở dữ liệu, sau đó được tính toán điểm tương đồng dựa trên sự tương đồng của chúng với hồ sơ người dùng.

Một số ưu điểm của lọc dựa trên nội dung:

- Không cần dữ liệu từ những người dùng khác để đưa ra đề xuất. Sau khi người dùng đã tìm kiếm và duyệt qua một số item, hệ thống lọc dựa trên nội dung có thể bắt đầu đưa ra các đề xuất có liên quan.

- Khuyến nghị có độ liên quan cao đến người dùng. Các đề xuất dựa trên nội dung có thể được điều chỉnh phù hợp với sở thích của người dùng, bao gồm các đề xuất cho các mục thích hợp.
- Tránh được vấn đề "cold start". Mặc dù tính năng lọc dựa trên nội dung cần một số thông tin đầu vào ban đầu từ người dùng để bắt đầu đưa ra các đề xuất, nhưng chất lượng của các đề xuất ban đầu thường tốt hơn so với hệ thống lọc cộng tác.

Một số hạn chế của lọc dựa trên nội dung:

- Các đề xuất thiếu sự mới lạ và đa dạng.
- Khả năng mở rộng là một thách thức.
- Các thuộc tính có thể không chính xác hoặc không nhất quán.

4.2 Collaborative filtering

Phương pháp lọc cộng tác là phương pháp phân tích dữ liệu người dùng để tìm ra mối tương quan giữa các đối tượng người dùng. Lọc cộng tác hoạt động bằng cách xây dựng một cơ sở dữ liệu, lưu trữ dưới dạng ma trận người dùng (users) - sản phẩm (items) và mỗi dòng của nó là một vectơ. Có nhiều cách tiếp cận để giải quyết bài toán lọc cộng tác: Cách tiếp cận dựa trên bộ nhớ (memory-based); Cách tiếp cận dựa trên mô hình (model-based); Kết hợp nhiều cách tiếp cận,...

Phương pháp lọc cộng tác với cách tiếp cận dựa trên bộ nhớ có đặc trưng cơ bản là thường sử dụng toàn bộ dữ liệu đã có để dự đoán đánh giá của một người dùng nào đó về sản phẩm mới. Cách tiếp cận dựa trên bộ nhớ thường được chia làm 2 loại: dựa trên người dùng (user-based) và dựa trên sản phẩm (item-based).

Một số ưu điểm của lọc cộng tác:

- Dự đoán được sở thích và nhu cầu của người dùng để đưa ra gợi ý các sản phẩm phù hợp với từng khách hàng mà không cần hiểu sản phẩm.
- Gợi ý dựa trên trải nghiệm của người dùng tương tự khác nên có thể gợi ý được những sản phẩm mới phù hợp sở thích mới.
- Phù hợp với những hệ thống lớn có nhiều đánh giá từ phía người dùng.

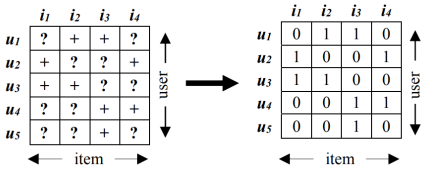
Một số hạn chế của lọc cộng tác:

- Không thể gợi ý nếu khách hàng chưa có dữ liệu về lịch sử tương tác mặt hàng.
- Khi lượng sản phẩm lớn và số lượng khách hàng đánh giá không nhiều thì phương pháp này không hiệu quả.
- Không thể gợi ý được các sản phẩm mới hoặc những sản phẩm chưa được ai đánh giá.

4.3 Bayesian Personalized Ranking (BPR)

Bayesian Personalized Ranking (BPR) là một hướng tiếp cận để tối ưu tham số của các mô hình hệ khuyến nghị thông dụng từ việc sử dụng thông tin phản hồi tiềm ẩn được đề xuất bởi Rendle và cộng sự [5].

Thông thường, các hệ thống khuyến nghị sử dụng phản hồi tiềm ẩn từ người dùng thường chỉ chứa dữ liệu tương tác/quan sát hay gọi là sự phản hồi tích cực/dương (positive feedback). Còn những mục tin mà người dùng chưa quan sát/tương tác là sự trộn lẫn giữa giá trị phản hồi âm (negative feedback – người dùng không thích mục này) và những giá trị thiếu (missing values – người dùng có thể thích mục này trong tương lai do họ chưa thấy/tương tác với chúng) như biểu diễn bên tay trái trong Hình 2. Do đó, cách thông thường để tạo ra tập dữ liệu huấn luyện là với những cặp (u, i) ∈ S thuộc lớp dương (positive class) sẽ được gán giá trị 1, phần còn lại thuộc về lớp âm (negative class) được gán giá trị 0, như bên tay phải của Hình 2 [6].

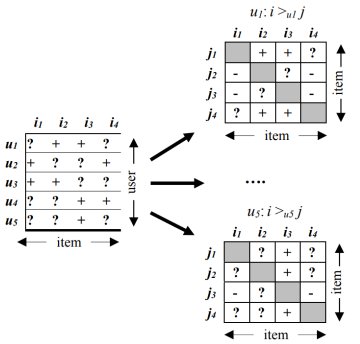


Hình 2 Biểu diễn dữ liệu dưới dạng nhị phân.

Tuy nhiên, bất lợi lớn của phương pháp này là trong suốt quá trình huấn luyện, các mô hình sẽ không phân biệt được đâu là phản hồi âm (negative feedbacks – tức người dùng không thích) và đâu là giá trị cần dự đoán do cả missing values và negative values đều được xem là giá trị 0. Để giải quyết vấn đề nêu trên, Rendle và cộng sự [5] đã đề xuất sử dụng phương pháp so sánh từng đôi (pair-wise ranking).

Cụ thể, từ dữ liệu đã có S tiến hành xây dựng lại tập dữ liệu huấn luyện D_S dựa vào mối quan hệ "thích hơn – prefer" giữa các khoá học cho mỗi người dùng u ($>_u$). Giả sử rằng nếu một khoá học i được học bởi người dùng u ($(u, i) \in S$) thì người dùng thích khoá học này hơn tất cả các khoá học chưa học khác. Ví dụ trong Hình 3 bên trái, biểu diễn quan hệ "thích hơn" cho các khoá học của người dùng u_1 như sau: $i_2 >_{u1} i_1$; $i_2 >_{u1} i_4$,... Với những khoá học mà người dùng đã học (như i_2 và i_3 ứng với người dùng u_1 trong Hình 3) và những khoá học mà người dùng chưa học (như i_1 và i_4) hệ thống cũng không sử dụng quan hệ "thích hơn" do chưa có thông tin.

Bên phải trong Hình 3 là cách biểu diễn dữ liệu được sử dụng trong hệ thống. Dấu + thể hiện người dùng thích khoá học i hơn j (bản chất i_1 và j_1 là như nhau), dấu – thể hiện người dùng thích khoá học j hơn i và dấu ? là những cặp các khoá học cần được xếp hạng sau này. Như vậy, hệ thống gợi ý khoá học cần dự đoán cho các giá trị ? trong các ma trận bên phải của 3, sau đó sắp xếp chúng từ cao xuống thấp và chọn ra N khoá học cần gợi ý. Tập D_S được biểu diễn một cách hình thức như sau (I_u^+ là các khoá học mà u đã học và có phản hồi dương – thích):



Hình 3 Biểu diễn dữ liệu bằng phương pháp so sánh từng đôi.

$$D_S := \{(u, i, j) \mid i \in I_u^+ \wedge j \in I \setminus I_u^+\}$$

Trong bài báo cáo này, chúng tôi sẽ sử dụng BPR-MF (dùng tiêu chuẩn tối ưu BPR cho kỹ thuật phân rã ma trận Matrix Factorization – MF) được trình bày trong [5].

5 Thực nghiệm và Kết quả

5.1 Content-based Filtering

5.1.1 Cài đặt

Trong phần này, chúng tôi đã cài đặt phương pháp lọc dựa trên nội dung để xây dựng một hệ thống đề xuất khóa học online.

Đầu tiên, chúng tôi lọc ra các người dùng có ít nhất 20 xếp hạng, từ đó thu được tập dữ liệu bao gồm 27837 đánh giá, 588 khóa học và 880 người dùng. Để cung cấp thông tin nội dung cho khóa học, chúng tôi đã thu thập các cột name, description, category, và instructors cho từng khóa học trong bộ dữ liệu, sau đó gộp chúng lại thành một thuộc tính mới là 'tag'. Thuộc tính 'tag' này tiếp tục được tiền xử lý bao gồm các bước: loại bỏ các dòng không phải tiếng Anh, loại bỏ stopwords, đưa về từ gốc (stemming), chuyển về ký tự thường, sau đó được vectơ hóa sử dụng tf-idf và word2vec. Tập dữ liệu lúc này được chia thành hai tập train và test với tỉ lệ 8/2, tương đương 22269 dòng dữ liệu cho tập train và 5568 dòng cho tập test. Từ đó, hồ sơ của mỗi người dùng được xây dựng với vectơ đặc trưng văn bản của các khóa học đã được xếp hạng. Để đánh giá kết quả của phương pháp này, chúng tôi sử dụng độ đo Recall@N (R@N).

5.1.2 Kết quả

Sau khi thực nghiệm phương pháp lọc dựa trên nội dung đã nêu phía trên, chúng tôi thu được bảng kết quả 6.

Với độ đo Recall@10, mô hình Content-based (tf-idf) cho kết quả 0.264, mô hình Content-based (Word2vec) cho kết quả 0.114. Trong hai mô hình, có thể thấy mô hình sử dụng word embbeding cho kết quả cao hơn mô hình còn

Model	Recall@10
Content-based (tf-idf)	**0.264**
Content-based (Word2vec)	0.114

Bảng 6 Kết quả thực nghiệm của các mô hình Content-based filtering.

lại. Lý do là vì Word2vec là một mạng neural nên đòi hỏi một lượng dữ liệu khá lớn để thu thập thông tin ngữ nghĩa và cú pháp của từ một cách chính xác. Trong khi đó, mô hình tf-idf có thể có độ chính xác cao ngay cả với một lượng dữ liệu nhỏ. Ngoài ra, một điều cần lưu ý, vì dữ liệu được tổng hợp từ nhiều thuộc tính khác nhau, nên các câu trong văn bản có cấu trúc ngữ pháp không đầy đủ và chính xác, từ đó làm cho việc sử dụng Word2vec bị hạn chế.

Tóm lại, mặc dù mô hình content-based sử dụng tf-idf cho kết quả tốt hơn nhưng nhìn chung mô hình lọc dựa trên nội dung có hiệu suất không được khả quan. Điều này cho thấy rằng có thể người dùng không chỉ tham gia các khóa học cố định giống các khóa học trước mà tham gia đa dạng các các khóa học dựa theo việc thay đổi sở thích hoặc xu hướng lúc bấy giờ.

	Historical enrolled Course
1	Crash course on Python
2	Add Ragdoll Effect to a Character in Unity
3	Programming for Everybody (Getting Started with Python)
4	Add Gore to Your Game in Unity

	Recommended Course
1	The Bits and Bytes of Computer Networking
2	HTML CSS and Javascript for Web Developers
3	Create Jumping Mechanics with C# in Unity

Hình 4 Ví dụ về top 3 khóa học được đề xuất với mẫu các khóa học đã tham gia trước đó.

5.1.3 Ablation Study

Trong phần này, chúng tôi tìm hiểu hiệu quả của việc sử dụng các kỹ thuật tiền xử lý khác nhau bằng cách loại bỏ chúng. Kết quả được minh họa trong bảng 7.

Model	Recall@10
Content-based (tf-idf)	0.264
w/o stopwords removing	0.220
w/o stemming	0.263
Content-based (Word2vec)	0.114
w/o stopwords removing	0.112
w/o stemming	0.118

Bảng 7 Kết quả thực nghiệm của các mô hình Content-based filtering sau khi loại bỏ các kỹ thuật tiền xử lý

Nhìn chung, ngoại trừ trường hợp loại bỏ stopwords cho Content-based (tf-idf), việc sử dụng các kỹ thuật tiền xử lý như loại bỏ stopwords và stemming không mang lại hiệu suất vượt trội, thậm chí còn làm giảm hiệu suất mô hình (Content-based (Word2vec) áp dụng stemming). Lý do là vì, thứ nhất, TF-IDF hoạt động dựa trên tần suất xuất hiện của các từ trong văn bản nên nó

có thể cho trọng số cao hơn với các stopwords nếu chúng xuất hiện thường xuyên trong văn bản. Điều này có thể dẫn đến việc stopwords gây mất cân bằng đến các vecto đại diện của văn bản, từ đó có thể gây bất lợi cho hiệu suất của mô hình. Ngoài ra, Stemming là kỹ thuật dùng để biến đổi 1 từ về dạng gốc (được gọi là stem hoặc root form) bằng cách cực kỳ đơn giản là loại bỏ 1 số ký tự nằm ở cuối từ mà nó nghĩ rằng là biến thể của từ. Bởi vì nguyên tắc hoạt động này khá đơn giản nên kết quả stem đôi khi không được như mong muốn và có thể gặp phải các lỗi như Over-stemming và Under-stemming.

5.2 Collaborative filtering

Đối với mô hình Collaborative filtering, chúng tôi sẽ thực hiện theo hai hướng tiếp cận là user-based và item-based. Trong phần này, bộ dữ liệu Ratings sẽ được sử dụng. Tiến hành lọc ra những người dùng đã đánh giá trên 20 khoá học, chúng tôi thu được 880 người dùng và 27837 đánh giá của người dùng với khoá học. Sau đó, để có thể đánh giá được hiệu suất mô hình, chúng tôi chia bộ dữ liệu thành 2 tập train, test với test_size = 0.01. Đầu vào của mô hình sẽ bao gồm các thuộc tính userId, rating, kết quả mô hình sẽ khuyến nghị cho người dùng các thông tin về khoá học mà người dùng có thể sẽ quan tâm (coursesId, name, rating).

Cài đặt:

- Bước 1: Tính toán độ tương đồng giữa các người dùng đối với phương pháp user-based và giữa các sản phẩm với items-based.
- Bước 2: Dự đoán giá trị rating của active user đối với một sản phẩm nào đó dựa trên thông tin từ bước 1.
- Bước 3: Gợi ý top-N sản phẩm cho người dùng.

Có hai công thức để tính độ tương đồng mà nhóm đã xem xét trong việc thiết lập mô hình là Cosine và Pearson. Công thức Cosine được biểu diễn như sau:

$$cosine\_similarity(\vec{a},\ \vec{b}) = \frac{\vec{a}.\vec{b}}{\parallel \vec{a} \parallel . \parallel \vec{b} \parallel} \tag{1}$$

Công thức Pearson được biểu diễn như sau:

- Item-based

$$pearson\_correlation(\vec{a},\ \vec{b}) = \frac{\sum_{u \in U} (r_{u,a} - \bar{r}_a)(r_{u,b} - \bar{r}_b)}{\sqrt{\sum_{u \in U} (r_{u,a} - \bar{r}_a)^2} \sqrt{\sum_{u \in U} (r_{u,b} - \bar{r}_b)^2}} \tag{2}$$

- User-based

$$pearson\_correlation(\vec{a},\ \vec{b}) = \frac{\sum_{i \in I} (r_{a,i} - \bar{r}_a)(r_{b,i} - \bar{r}_b)}{\sqrt{\sum_{i \in I} (r_{a,i} - \bar{r}_a)^2} \sqrt{\sum_{i \in I} (r_{b,i} - \bar{r}_b)^2}} \tag{3}$$

Trong đó:

– I: tập các khoá học cả hai người dùng a và b cùng đánh giá.
– $r_{a,i}$: rating của người dùng a đối với khoá học i.
– $r_{b,i}$: rating của người dùng b đối với khoá học i.
– U: tập các users cùng đánh giá hai khoá học a và b.
– $r_{u,a}$: rating của người dùng u đối với khoá học a.
– $r_{u,b}$: rating của người dùng u đối với khoá học b.

Sau khi thực nghiệm các phương pháp Collaborative filtering đã nêu phía trên, chúng tôi thu được kết quả ở bảng 8. Bảng kết quả cho thấy, mô hình Collaborative filtering (User-based) sử dụng độ đo tương đồng Cosine đạt hiệu suất tốt nhất với RMSE là 1.1296 và NMAE là 0.5878. Nhìn chung mô hình Collaborative filtering (User-based) đạt kết quả tốt hơn mô hình Collaborative filtering (Item-based) trên cả hai độ đo Cosine và Pearson. Tuy nhiên, hiệu suất của các mô hình lọc cộng tác mà chúng tôi thực nghiệm vẫn chưa thực sự tốt, điều này một phần có thể là do bộ dữ liệu chúng tôi sử dụng có số lượng rating ở mức 5 chiếm đa số, khiến cho mô hình khó có thể phân biệt được đâu mới thực sự là khoá học mà người dùng quan tâm. Ngoài ra, mô hình lọc cộng tác vẫn còn hạn chế khi không thể đề xuất nếu người dùng chưa có dữ liệu về lịch sử tương tác với các khoá học và đồng thời không thể đề xuất được các khoá học mới và các khoá học chưa được ai đánh giá.

Model	RMSE	MAE
Collaborative filtering (Item – based) + Pearson	1.7331	0.9351
Collaborative filtering (Item – based) + Cosine	1.6478	0.9175
Collaborative filtering (User – based) + Pearson	1.5310	0.8602
Collaborative filtering (User – based) + Cosine	**1.1296**	**0.5878**

Bảng 8 Kết quả thực nghiệm của các mô hình Collaborative filtering.

name	courseId	rating
Introduction to Marketing	571	5.0
Introduction to Financial Accounting	560	5.0
Introduction to Typography	531	5.0
Teamwork Skills: Communicating Effectively in Groups	516	5.0
Privacy Law and Data Protection	417	5.0

Bảng 9 Top 5 khoá học được khuyến nghị cho người dùng có userId = 8747 dựa trên CF (User – based) + Cosine.

5.3 Bayesian Personalized Ranking (BPR)

Đối với mô hình Bayesian Personalized Ranking, chúng tôi sử dụng bộ dữ liệu Ratings, chuyển rating của người dùng đối với các khoá học về hai nhãn 0 hoặc 1 (các rating $>$ 3 được chuyển về 1, các rating $<=$ 3 được chuyển về 0). Sau đó, chúng tôi chia bộ dữ liệu thành 2 tập train, test với test_size = 0.2. Đầu vào của mô hình sẽ bao gồm các thuộc tính userId, rating, kết quả mô hình sẽ khuyến nghị cho người dùng các thông tin về khoá học mà người dùng có thể sẽ quan tâm (coursesId, name).

Sau khi thực nghiệm, mô hình Bayesian Personalized Ranking đạt hiệu suất với độ đo AUC là 0.784202.

courseId	name
9	AI For Everyone
307	Introduction to TensorFlow for Artificial Intelligence, Machine Learning, and Deep Learning
137	Diversity and inclusion in the workplace
486	Social Psychology
412	Positive Psychology: Martin E. P. Seligman's Visionary Science

Bảng 10 Top 5 khoá học được khuyến nghị cho người dùng có userId = 28 dựa trên BPR.

6 Kết luận

Trong đồ án này, nhóm chúng tôi đã sử dụng tập dữ liệu Courses để thực nghiệm dự đoán đánh giá khoá học của người dùng và đề xuất, gợi ý khoá học cho từng người dùng theo ba phương pháp là Content-based Filtering, Collaborative filtering và Bayesian Personalized Ranking (BPR). Mỗi mô hình đều có ưu và nhược điểm riêng, nhìn chung mô hình Lọc cộng tác (User-based) cho đề xuất tốt nhất. Trong tương lai, nhóm mong muốn có thể cải thiện các mô hình đồng thời làm phong phú thêm dữ liệu cũng như các online course websites khác để đề xuất khoá học được hiệu quả hơn.

Tài liệu

[1] Khalid, A., Lundqvist, K., Yates, A., Ghzanfar, M.A.: Novel online Recommendation algorithm for Massive Open Online Courses (NoR-MOOCs) (2021).

[2] Boratto, L., Fenu, G., Marras, M.: The Effect of Algorithmic Bias on Recommender Systems for Massive Open Online Courses, p.457–472 (2019).

[3] Yanhui, D., Dequan, W., Yongxin, Z., Lin, L.: A Group Recommender System for Online Course Study. In: 7th International Conferenceon Information Technology in Medicine and Education (ITME),p.318–320 (2015).

[4] Symeonidisa, P., Malakoudis, D.: Multi-Modal Matrix Factorization with Side Information for Recommending Massive Open Online Courses. Expert Systems with Applications (2018).

[5] Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: BPR: Bayes Personalized Ranking from implicit feedback. In Proceedings of the 25th International Conference on Uncertainty in Articial Intelligence, AUAI Press (2009).

[6] Hu, Y., Koren, Y., Volinsky, C.:Collaborative filtering for implicit feedback datasets, in IEEE International Conference on Data Mining, pages 263-272 (2008).