HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

NGUYỄN VĂN HÒA

NGHIÊN CỨU PHƯƠNG PHÁP TƯ VẤN CỘNG TÁC CHO CÁC CỔNG LẬP TRÌNH TRỰC TUYẾN

LUẬN VĂN THẠC SĨ KỸ THUẬT

(*Theo định hướng ứng dụng*)

HÀ NỘI - 2022

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

NGUYỄN VĂN HÒA

NGHIÊN CỨU PHƯƠNG PHÁP TƯ VẤN CỘNG TÁC CHO CÁC CỔNG LẬP TRÌNH TRỰC TUYẾN

Chuyên ngành: Hệ thống thông tin

Mã số: 8.48.01.04

LUẬN VĂN THẠC SĨ KỸ THUẬT

(*Theo định hướng ứng dụng*)

NGƯỜI HƯỚNG DẪN KHOA HỌC

TS. NGUYỄN DUY PHƯƠNG

HÀ NỘI - 2022

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi, kết quả đạt được trong luận văn là sản phẩm của riêng cá nhân, không sao chép lại của người khác. Trong toàn bộ nội dung của luận văn, những điều được trình bày hoặc là của cá nhân hoặc là được tổng hợp từ nhiều nguồn tài liệu. Tất cả các tài liệu tham khảo đều có xuất xứ rõ ràng và được trích dẫn hợp pháp. Các số liệu, kết quả nêu trong luận văn là trung thực.

Tác giả luận văn

Nguyễn Văn Hòa

LỜI CẢM ƠN

Em xin gửi lời cảm ơn tới thầy hướng dẫn TS. Nguyễn Duy Phương, thầy đã tận tình hướng dẫn giúp đỡ, chỉnh sửa và chỉ bảo em trong suốt quá trình nghiên cứu và hoàn thành luận văn.

Em cũng xin chân thành cảm ơn các thầy cô tại Học viện Công nghệ Bưu chính Viễn thông, đặc biệt các thầy cô khoa Công nghệ thông tin, đã tận tình dạy dỗ, giúp đỡ và tạo mọi điều kiện tốt nhất cho em trong suốt quãng thời gian em theo học tại học viện, để em có thể hoàn thành được luận văn này.

Mặc dù đã cố gắng hoàn thành luận văn nhưng chắc chắn sẽ không tránh khỏi những sai sót, em kính mong nhận được sự thông cảm và góp ý của các thầy cô và các bạn.

Em xin trân trọng cảm ơn.

Nguyễn Văn Hòa

MỤC LỤC

DANH MỤC CÁC KÝ HIỆU, CHỮ VIẾT TẮT

Viết tắt	Tiếng Anh	Tiếng việt
CF	Collaborative filtering	Lọc cộng tác
ICPC	International Collegiate Programming Contest	Cuộc thi lập trình quốc tế
MAE	Mean absolute error	Độ đo trung bình giá trị tuyệt đối lỗi

DANH MỤC CÁC BẢNG

DANH MỤC HÌNH ẢNH

LỜI MỞ ĐẦU

Ngày nay chúng ta đang chứng kiến một sự phát triển mạnh mẽ chưa từng có của các cổng lập trình trực tuyến. Các cổng lập trình cung cấp cho người dùng một môi trường lập trình đa ngôn ngữ để có thể lập trình, kết quả lập trình được chấm một cách tự động. Đồng hành cùng với người dùng trên cổng lập trình trực tuyến là các trường đại học, các tập đoàn kinh tế lớn nhằm mục tiêu đào tạo và tuyển dụng nguồn nhân lực chất lượng cao về công nghệ thông tin ví dụ như các tập đoàn lớn về công nghệ Microsoft, Google, Facebook, Apple, Samsung… Ở Việt Nam thì có FPT, Viettel, VNPT, các tập đoàn này không chỉ xây dựng riêng cho mình các cổng lập trình trực tuyến mà còn bảo trợ về tài chính và cung cấp kho nội dung số cho cổng lập trình trực tuyến của các trường đại học khác. Một số trường đại học như MIT, Stanford, Baylor xem cổng lập trình trực tuyến như một công cụ quan trọng trong giảng dạy, rèn luyện và đánh giá kỹ năng lập trình của kỹ sư ngành công nghệ thông tin. Nhận thức được tầm quan trọng và hiệu quả trong công nghệ lập trình trực tuyến, một số trường đại học ở Việt Nam như Đại học Quốc Gia Hà Nội, Đại học Quốc Gia thành phố HCM Đại học Bách Khoa, Học viện công nghệ Bưu Chính Viễn Thông đã xây dựng riêng cho mình các cổng lập trình trực tuyến và đã ứng dụng thành công trong giảng dạy và rèn luyện kỹ năng lập trình của sinh viên.

Có nhiều công nghệ khác nhau để xây dựng nên các cổng lập trình trực tuyến. Ở cấp độ trung học cơ sở hay phổ thông trung học hầu hết các quốc gia chọn công nghệ PC2 hoặc CMS trong giảng dạy, luyện tập và các tổ chức các kỳ thi lập trình quốc gia (NOI) hoặc quốc tế (IOI). Ở các cấp bậc cao hơn, các trường đại học thường lựa chọn các công nghệ Domjudge, Katis hoặc DMOJ trong giảng dạy, luyện tập và tổ chức các kỳ thi lập trình quốc gia hoặc quốc tế theo chuẩn ACM/ICPC. Sự khác biệt giữa các công nghệ này là khá nhỏ và chỉ được phân biệt khi ta triển khai ứng dụng dựa vào quy mô nhỏ hay lớn, nhiều hay ít người dùng, độ lớn dữ liệu của các test hoặc phương pháp đánh giá giải pháp của người lập trình theo mức từng phần hay toàn phần.

Đối với các cổng lập trình trực tuyến, tài nguyên quan trọng nhất là kho nội dung số được nạp bên trong mỗi cổng lập trình. Kho nội dung số được thể hiện dưới dạng tập các bài toán cùng với các bộ dữ liệu kiểm thử tương ứng. Mỗi bài toán cần được xây dựng nhiều bộ dữ liệu kiểm thử, mỗi bộ dữ liệu kiểm thử xác định một tính chất đúng mà giải pháp lập trình cần đạt được. Một giải pháp lập trình được xem là tốt nhất nếu nó

thỏa mãn được tất cả các bộ dữ liệu kiểm thử với thời gian và không gian nhớ xác định. Kho nội dung số được xây dựng bởi các chuyên gia, tổ chức sở hữu cổng lập trình trực tuyến. Kho nội dung số mà càng lớn thì càng thu hút được đông đảo người dùng tham gia.

Một yếu tố rất quan trọng tiếp theo của cổng lập trình trực tuyến là người dùng. Nếu chúng ta có một tập tài nguyên là các bài toán vô cùng đa dạng và phong phú mà lại thiếu mất đi những người tham gia giải quyết kho tài nguyên ấy thì cổng lập trình trực tuyến sẽ đánh mất đi mục đích chính của nó được tạo ra là nhằm phục vụ, nâng cao trình độ, kỹ năng lập trình của lập trình viên hay sinh viên tại các trường đại học. Tập người càng lớn chứng tỏ kho nội dung số của cổng lập trình trực tuyến rất có giá trị. Phần lớn trong số người dùng sử dụng cổng lập trình để học lập trình, một phần trong số họ tham gia sử dụng với mục đích nâng cao kỹ năng lập trình, phần còn lại để chứng tỏ bản thân, khẳng định khả năng của mình trong việc lập trình và cũng dựa vào đó để thu hút các nhà tuyển dụng. Đặc điểm chung nhất của các cổng lập trình trực tuyến là kho nội dung số và số lượng người dùng phong phú và đa dạng. Một số cổng lập trình trực tuyến như CodeForces, TopCoder, ICPC Baylor, CodeLearn, SPOJ,… đã thu hút hàng trăm ngàn lập trình viên trên toàn thế giới tham gia. Chính vì vậy, việc xây dựng một hệ thống gợi ý (Recommendation Systems) là một điều rất cần thiết để có thể tư vấn, gợi ý các bài toán phù hợp với khả năng lập trình của mỗi người dùng. Đây cũng là một điều tối quan trọng trong cổng lập trình trực tuyến vì nó giúp người dùng có thể giải quyết những bài toán phù hợp với khả năng của mình nhằm gây hứng thú với lập trình viên trong quá trình lập trình.

Một trong công nghệ lõi của cổng lập trình trực tuyến là hệ thống tư vấn kho nội dung số đến với người dùng. Một hệ tư vấn tốt có thể tư vấn, gợi ý những bài toán phù hợp với khả năng của lập trình viên với sai số thấp nhất. Trong những năm gần đây có rất nhiều phương pháp được đề xuất sử dụng trong hệ tư vấn nhưng đại đa số lại chỉ tập chung vào hai kiểu tư vấn:

- Tư vấn dựa theo nội dung (*Content – Based Systems*): Người dùng sẽ được tư vấn dựa theo những sản phẩm tương tự với những sản phẩm người dùng đó đã ưa thích. Có thể hiểu rằng phương pháp này đưa ra gợi ý dựa trên đặc tính của sản phẩm đó.
- Tư vấn cộng tác (*Collaborative filtering*): Ở phương pháp này hệ thống sẽ gợi ý sản phẩm dựa trên độ tương quan (Similarity) giữa các người dùng với sản phẩm (item)

hay người dùng (user) hoặc sản phẩm (item). Có thể hiểu đơn giản ở nhóm này một sản phẩm (item) được gợi ý tới một người dùng (user) dựa trên những người dùng (user) có hành vi tương tự với sản phẩm (item).

Mục đích nghiên cứu của đề tài là nghiên cứu phương pháp tư vấn cộng tác ứng dụng cho các cổng lập trình trực tuyến. Để thực hiện được những mục tiêu trên đề tài cần phải đạt được một số nhiệm vụ nghiên cứu sau:

- Nghiên cứu tổng quan về các cổng nghệ lập trình trực tuyến
- Nghiên cứu phương pháp lọc cộng tác ứng dụng cho hệ tư vấn
- Đề xuất phương pháp tư vấn cộng tác cho cổng lập trình trực tuyến

Đối tượng nghiên cứu của đề tài là các phương pháp tư vấn lọc cộng tác và các công nghệ xây dựng cổng lập trình trực tuyến. Phạm vi nghiên cứu là các phương pháp tư vấn cộng tác cho dữ liệu người dùng và dữ liệu sản phẩm của cổng lập trình trực tuyến, phương pháp nghiên cứu được chia làm 2 hướng:

- *Nghiên cứu lý thuyết*: Tập trung vào các công nghệ xây dựng cổng lập trình trực tuyến, đặc biệt quan tâm đến việc trích rút dữ liệu người dùng và kết quả lập trình trực tuyến của mỗi lập trình viên. Nghiên cứu các phương pháp tư vấn cộng tác áp dụng trên cổng lập trình trực tuyến dựa trên người dùng hoặc sản phẩm.
- *Nghiên cứu thực nghiệm*: Xây dựng bộ dữ liệu thử nghiệm riêng cho cổng lập trình trực tuyến của PTIT. Thực hiện thử nghiệm phương pháp đề xuất trên tập dữ liệu đã được xây dựng

Nội dung của luận văn bao gồm 3 chương với cấu trúc như sau:

CHƯƠNG 1: TỔNG QUAN VỀ CỔNG LẬP TRÌNH TRỰC TUYẾN VÀ HỆ TƯ VẤN CỘNG TÁC

Nội dung chính của chương 1 là tập trung nghiên cứu vào những vấn đề cơ bản của cổng lập trình trực tuyến và các phương pháp tư vấn cộng tác. Nội dung của chương bao gồm:

- **Giới thiệu về các công nghệ lập trình trực tuyến**: Ở phần này sẽ trình bày về các công nghệ lập trình trực tuyến, tài nguyên của cổng lập trình trực tuyến và những thách thức, khó khăn của cổng lập trình trực tuyến hiện nay
- **Giới thiệu về hệ tư vấn và một số vấn đề liên quan**: Trình bày về các hệ tư vấn, các phương pháp xây dựng nên hệ tư vấn và một số vấn đề liên quan

- **Phương pháp cộng tác baseline**: Trình bày các phương pháp tư vấn cộng tác cơ sở và được dùng trong việc so sánh, đánh giá với phương pháp đề xuất của luận văn
- **Một số vấn đề của phương pháp lọc cộng tác**: Trình bày một số vấn đề của phương pháp lọc cộng tác và một số nghiên cứu giải pháp
- **Kết luận chương**: Tóm tắt lại những kết quả nghiên cứu của chương

CHƯƠNG 2: PHƯƠNG PHÁP TƯ VẤN CỘNG TÁC CHO CỔNG LẬP TRÌNH TRỰC TUYẾN

Nội dung chính của chương trình bày phương pháp tư vấn cộng tác cho cổng lập trình trực tuyến. Dự kiến nội dung của chương bao gồm:

- **Phát biểu bài toán**: Mô hình hóa bài toán tư vấn cộng tác cho các cổng lập trình trực tuyến.
- **Phương pháp tư vấn cộng cho cổng lập trình trực tuyến**: Trình bày phương pháp tư vấn cộng tác bằng cách biểu diễn mối quan hệ giữa người dùng và các nội dung số trên cổng lập trình trực tuyến như một đồ thị hai phía.
- **Kết luận chương**: Tóm tắt lại những kết quả nghiên cứu của chương.

CHƯƠNG 3: THỰC NGHIỆM VÀ ĐÁNH GIÁ

Nội dung chính của chương trình bày phương pháp tư thử nghiệm và đánh giá kết quả cài đặt. Dự kiến nội dung của chương bao gồm:

- **Dữ liệu thực nghiệm**: Trình bày phương pháp thu thập dữ liệu từ cổng lập trình trực tuyến của Học viện Công nghệ BCVT.
- **Phương pháp thực nghiệm**: Trình bày các phương pháp đánh giá sai số dự đoán áp dụng cho hệ tư vấn cộng tác.
- **Kết quả thực nghiệm**: Đưa ra các kết quả thực nghiệm và so sánh, đánh giá kết quả so với các phương pháp khác.
- **Kết luận chương**: Tóm tắt lại các kết quả đã đạt được của chương

CHƯƠNG 1: TỔNG QUAN VỀ CỔNG LẬP TRÌNH TRỰC TUYẾN VÀ HỆ TƯ VẤN CỘNG TÁC

1.1. Giới thiệu về công nghệ lập trình trực tuyến

Hiện này cổng lập trình trực tuyến được rất nhiều các tập đoàn công nghệ lớn như Microsoft, Amazon, Facebook, Google, Samsung … hay các trường đại học đang đầu tư rất mạnh về xây dựng các cổng lập trình trực tuyến của riêng minh nhằm mục đích đào tạo các kỹ sư công nghệ thông tin. Chính vì vậy mà nền tảng công nghệ để xây dựng nên cổng lập trình trực tuyến là vô cùng quan trọng. Tùy vào quy mô của các cuộc thi nhỏ hay lớn mà được sử dụng các lõi công nghệ khác nhau. Phần này chúng ta sẽ tìm hiểu về các công nghệ thường được sử dụng trong cổng lập trình.

1.1.1. PC2

PC^2 (P-C-Squared) là một hệ thống phần mềm để quản lý các cuộc thi lập trình máy tính được phát triển tại Đại học bang California, Sacramento. Hệ thống PC^2 được sử dụng để hỗ trợ các hoạt động của các cuộc thi lập trình trên khắp thế giới bao gồm cuộc thi lập trình ICPC (International Collegiate Programming Contest) và các cuộc thi khu vực trên sáu lục địa. PC^2 có ưu điểm đáp ứng tất cả ràng buộc của kỳ thi ACM/ICPC trên toàn cầu, hoạt động tốt với số lượng người dùng nhỏ hơn 1500, cài đặt dễ dàng trên hệ thống mạng LAN hoặc Internet.

1.1.2. CMS

CMS là phần mềm tổ chức các cuộc thi lập trình tương tự như các cuộc thi nổi tiếng như IOI. Được viết và nhận được sự đóng góp bởi những người tổ chức các cuộc thi tương tự ở cấp quốc gia và quốc tế.CMS là một giải pháp được coi là hoàn chỉnh và đã được thử nghiệm và chứng minh tốt để quản lý các cuộc thi. Tuy nhiên nó chỉ cung cấp các công cụ hạn chế để phát triển dữ liệu, nhiệm vụ thuộc về cuộc thi, Hệ thống được tổ chức theo cách Modular với các dịch vụ khác nhau chạy trên các máy khác nhau và cung cấp khả năng mở rộng thông qua các bản sao dịch vụ trên một số thiết bị.

Trạng thái của cuộc thi được lưu trữ hoàn toàn trên cơ sở dữ liệu PostgreSQL các DBMS khác không được hỗ trợ vì CMS dựa trên tính năng LO của PostgreSQL, miền là cơ sở dữ liệu hoạt động chính xác, tất cả các dịch vụ khác có thể được khởi động và dừng một cách độc lập. Ví dụ quản trị viên có thể nhanh chóng thay thế một thiết bị hỏng bằng một máy giống hệt, máy sẽ đảm nhiệm vai trò của nó mà không cần phải di

chuyển thông tin từ máy bị hỏng đây cũng là một trong những ưu điểm của CMS. Nhược điểm của hệ thống chính là phụ thuộc hoàn toàn vào cơ sở dữ liệu để chạy.

1.1.3. Domjudge

DOM JUDGE là một hệ thống giám khảo tự động để điều hành các cuộc thi lập trình. Nó có một cơ chế để gửi các giải pháp vấn đề để chúng được đánh giá hoàn toàn tự động và cung cấp giao diện cho các nhóm, giám khảo.

DOMjudge chủ yếu tập trung để sử dụng trong các cuộc thi lập trình như ACM/ICPC hay IOI, nơi các nhóm thi đấu, làm việc tại chỗ và có một bộ vấn đề và khung thời gian cố định. Tuy nhiên, nó cũng có thể được điều chỉnh cho phù hợp với các bối cảnh khác.

Hệ thống có quy mô khá tốt với số lượng người dùng lớn nhưng lại cần một tài nguyên server đủ mạnh. Nó có đầy đủ các tính năng của một OPJ bao gồm:

- Máy test là một hệ thống chạy độc lập so với giao diện
- Sandbox: Máy chấm của DMOJ được thiết kế dựa trên Sandbox Package. Sandbox là tính năng cho phép chạy một chương trình trong môi trường cô lập nhờ có nó mà chương trình test có thể được chạy an toàn và không ảnh hưởng tới máy chủ.
- Hệ dịch: là một phần của máy chấm và đã được triển khai, cài đặt hơn 10 ngôn ngữ chính bao gồm C, Java, PHP, Pascal, Python, JS,..
- Test case: Là một loạt các bộ dữ liệu đầu vào và đầu ra tương ứng để nhập vào và so sánh với kết quả chạy.
- Hệ đánh giá: Hệ đánh giá trong DMOJ chấp nhận nhiều dạng kết quả đầu ra như là dạng chữ, dạng số, dạng số thập phân thậm chí là dạng hàm hoặc luật.
- Giao diện: Giúp người dùng có thể can thiệp vào hệ thống.

DOM JUDGE được Đại học Bách Khoa Hà Nội sử dụng trong xây dựng website nội bộ cho luyện tập lập trình và luyện thi Olympic Tin học,.

1.1.4. Kattis

Đây là một bộ công cụ khá mới nhưng càng ngày tỏ ra uy tín trong công việc tổ chức các contest chấm thi ACM/ICPC. Website này có đội ngũ chuyên gia đánh giá đề bài rất khắt khe và chuyên nghiệp. Điểm mạnh của Kattis là tích hợp công cụ kiểm tra đạo văn giúp cải thiện tính trung thực của điểm số trong thi cử đối với các trường Đại học và được cài đặt, tích hợp hơn 10 ngôn ngữ lập trình có sẵn như Java, Python, PHP, Javascript,… chính vì vậy đã được rất nhiều trường đại học quốc tế lớn như KTH Royal

Institute of Technology, National University of Singapore, Lund University,.. sử dụng trong tổ chức thi, kiểm tra và tập luyện cho sinh viên. Từ năm 2016, kỳ thi lập trình ACM/ICPC World Final đã sử dụng bộ công cụ này

1.1.5. SPOJ

Đây là một website luyện tập trực tuyến rất lớn. Website này cung cấp các API mở cho phép các trường hoặc các tổ chức trên toàn cầu đăng ký sử dụng các site con và công cụ chấm thi trên đó. Tại Việt Nam đã có rất nhiều trường Đại học sử dụng API mở này để xây dựng nên website luyện tập lập trình riêng như Học viện công nghệ Bưu Chính Viễn Thông có website luyện tập lập trình sử dụng trình chấm SPOJ từ năm 2012 và đã thu hút rất nhiều các lập trình viên trong nước tham gia, hội tin học Việt Nam sử dụng trình chấm SPOJ để tạo website luyện tập cho cộng đồng học sinh, sinh viên, hướng tới các kỳ thi cấp Quốc gia, Quốc tế về tin học.

1.1.6. DLab

Đây là hệ thống cổng lập trình trực tuyến của khoa Công nghệ thông tin 1 thuộc Học viện Công nghệ Bưu chính Viễn thông với mục đích phục vụ quá trình học tập và luyện tập cho sinh viên trong Học viện. Ngoài ra Dlab còn từng được sử dụng để sử dụng để tổ chức các cuộc thi lập trình theo chuẩn ICPC như cuộc thi ICPC - PTIT, ICPC miền Bắc,… với số lượng người dùng lớn. Dữ liệu trong luận văn được sử dụng là dữ liệu thực trên hệ thống Dlab. Dữ liệu này cũng được công bố bởi nhóm tác giả Nguyễn Mạnh Sơn, Nguyễn Duy Phương qua bài báo “Một phương pháp tư vấn cộng tác cho các cổng lập trình trực tuyến”.

1.1.7. Tổng kết

Việc xây dựng nên một cổng lập trình trực tuyến hoàn chỉnh bao gồm bộ công cụ kiểm thử, website quản lý, hệ thống tư vấn cho sinh viên luôn là một nhu thiết yếu tại các trường đại học trên thế giới nói chung cũng như Việt Nam nói riêng, nhất là tại các trường chuyên đào tạo nhân lực trong lĩnh vực CNTT. Ở phần này chúng ta đã cùng tìm hiểu qua về các công cụ kiểm thử tự động trong cổng lập trình trực tuyến, một điều tối quan trọng là khi chúng ta đã có một hệ thống kiểm thử tự động thì làm thế nào ta có thể tìm được tập bài toán phù hợp với kiến thức, sở trường của mình. Đây chính là nhiệm vụ của hệ thống tư vấn trong cổng lập trình trực tuyến. Hệ tư vấn đang được ứng dụng rất rộng rãi trong thương mại điện tử và sẽ được giới thiệu trong mục *1.2*

1.2. Giới thiệu về hệ tư vấn

Với sự phát triển mạnh mẽ về thương mại điện tử hay các nền tảng mạng xã hội thì không chỉ cổng lập trình trực tuyến cần sử dụng hệ thống tư vấn để gợi ý các bài toán phù hợp với trình độ của lập trình viên mà ngay cả đối với các sàn thương mại điện tử lớn như Shopee, Lazada, Amazon,.. cũng đều rất cần phải có hệ tư vấn để tư vấn sản phẩm đến người dùng ngay cả thay vì chờ đợi họ tìm kiếm thông tin sản phẩm và đây cũng chính là yếu tố quan trọng góp vào doanh thu khổng lồ của các sàn thương mại điện tử này cũng như tiết kiệm được thời gian của khách hàng. Các trang mạng xã hội lớn cũng vậy điển hình là Facebook, họ cũng có một hệ thống tư vấn cực kỳ mạnh mẽ, mạnh đến nỗi dường như chúng ta có thể cảm thấy họ đang đọc suy nghĩ hay nghe lén chúng ta vậy.

Hệ tư vấn là một dạng của hệ hỗ trợ ra quyết định, cung cấp các giải pháp mang tính cá nhân hóa mà không phải trải qua quá trình tìm kiếm phức tạp. Hệ tư vấn học từ khách hàng và gợi ý các sản phẩm tốt nhất trong số các sản phẩm phù hợp. Hiện nay, các hệ tư vấn được xem như phương pháp hiệu quả trong lọc thông tin và đóng vai trò quan trọng trong các hệ thống thương mại điện tử là điều tất yếu nhằm tiết kiệm thời gian, công sức chi phí cho khách hàng, giúp học tìm ra sản phẩm ưng ý nhất để mua.

Hệ tư vấn sử dụng các tri thức về sản phẩm, các tri thức của chuyên gia hay tri thức khai phá học được từ hành vi của người tiêu dùng để đưa ra các dự đoán về các hành vi mua hàng trong tương lai của chính khách hàng đó. Các dạng tư vấn như: tư vấn các sản phẩm tới người tiêu dùng, các thông tin sản phẩm mang tính cá nhân hóa, tổng kết các ý kiến cộng đồng, và cung cấp các chia sẻ, các phê bình, đánh giá mang tính cộng đồng liên quan đến yêu cầu, mục đích của người sử dụng đó.

Một hệ tư vấn tốt có thể đưa ra gợi ý có độ chính xác cao ví dụ như một hệ tư vấn cho cổng lập trình được coi là tốt có thể đưa ra gợi ý các bài toán đến với lập trình viên nếu họ có thể làm được đa số thì chứng tỏ hệ thống gợi ý của cổng lập trình hoạt động tốt. Một trong những yếu tố quan trọng ảnh hưởng đến hiệu quả của hệ tư vấn chính là phương pháp hay thuật toán bên trong, và trong những năm lại đây, hệ tư vấn được chia làm 2 loại chính là lọc nội dung và lọc cộng tác.

1.2.1. Phương pháp lọc nội dung (Content Filtering)

Lọc nội dung là kỹ thuật lọc dựa trên sự phân tích về nội dung của dữ liệu chứ không phải là tìm hiểu về nguồn gốc của nó hay các tiêu chí khác. Lọc theo nội dung được thực hiện trên cơ sở so sánh nội dung thông tin hay mô tả bài toán để tìm ra những bài toán tương tự với những gì mà sinh viên đã từng quan tâm để giới thiệu cho họ những bài toán này. Lọc dựa trên nội dung thực hiện hiệu quả trên các đối tượng dữ liệu biểu diễn dưới dạng văn bản và được sử dụng rộng rãi trên internet để lọc email và truy cập các trang web.

Trong phương pháp tư vấn dựa trên nội dung, hàm tiện ích ***u(c,s)*** của item (bài toán) *s* ứng với người dùng *c* được đánh giá dựa trên những hàm ước lượng ***u(c,s$_i$)*** được gán bởi người dùng ***c*** với những item ***s$_i$ ∈ S*** tương tự với item ***s***.

Phương pháp tiếp cận dựa trên nội dung để tư vấn bắt nguồn từ việc truy vấn thông tin và những nghiên cứu về kỹ thuật lọc thông tin. Do tầm quan trọng cũng như là những thuận lợi của việc truy vấn thông tin, các phương diện của kỹ thuật lọc, các ứng dụng dựa trên một vài văn bản text nên nhiều hệ thống dựa trên nội dung hiện thời đều tập trung vào những sản phẩm tư vấn chứa thông tin theo đúng nguyên bản, như những tài liệu Websites (URLs), và những thông điệp mới của người dùng. Việc cải tiến dựa trên phương pháp truy vấn thông tin truyền thống từ hồ sơ cá nhân của người dùng thường chứa thông tin về sở thích, nhu cầu và thị hiếu của người dùng. Các thông tin này rõ ràng có thể được suy ra từ những người dùng ví dụ qua các câu hỏi hoặc nằm ẩn trong các hành vi giao dịch.

Giống như trước, ta xem ***Content(s)*** là một thông tin riêng của sản phẩm, nghĩa là một tập các đặc tính đặc trưng cho sản phẩm ***s***. Nó thường được tính toán thông qua việc trích rút từ tập các đặc tính của items (nội dung của nó) và ứng với mỗi mục đích tư vấn sẽ xác định ra item thích hợp. Dễ dàng nhận thấy nội dung của những hệ thống này được mô tả như là một từ khóa (keyword). Chẳng hạn, hệ thống Syskill & Webert đưa ra tài liệu với 128 từ cung cấp nhiều thông tin nhất. "Tầm quan trọng" (việc cung cấp nhiều thông tin) của từ k_j trong tài liệu d_j được xác định bằng độ đo trọng lượng w_{ij} định nghĩa qua một vài phương pháp khác nhau.

Một trong những thước đo phổ biến để xác định mức độ quan trọng của từ khóa trong việc truy vấn thông tin là đo tần suất xuất hiện của mục từ trong tài liệu (term

frequency - **tf**) và tần số nghịch đảo của tần suất xuất hiện các tài liệu (inverse document frequency - ***idf***)

Giới hạn của lọc nội dung

Những kỹ thuật dựa trên nội dung thường bị giới hạn bởi chính những đặc trưng được kết hợp giữa các đối tượng mà hệ thống đó tư vấn. Vì thế, để có một tập các đặc trưng đầy đủ, nội dung hoặc phải là một dạng mà máy tính có thể tự động phân tích được (ví dụ như văn bản text) hoặc phải được quy về thành những sản phẩm vận hành bằng tay được. Nhưng trong thực tế, kỹ thuật truy vấn thông tin thường chỉ làm việc tốt với cách trích rút các đặc trưng từ tài liệu text còn đối với một số lĩnh vực khác nó lại vấp phải nhiều vấn đề. Chẳng hạn, những phương pháp trích rút đặc trưng tự động sẽ gặp rất nhiều khó khăn khi áp dụng với dữ liệu đa phương tiện như ảnh đồ họa, luồng audio và video. Hơn nữa, nó thường không được thực hành bằng tay để ấn định những thuộc tính do những giới hạn về tài nguyên.

Một vấn đề khác liên quan đến giới hạn về phân tích nội dung là nếu hai item khác nhau được biểu diễn cùng một tập đặc trưng thì chúng không thể phân biệt được. Vì vậy khi những tài liệu dựa trên văn bản thường được biểu diễn dưới những từ khóa quan trọng, thì những hệ thống dựa trên nội dung không thể phân biệt được cái nào hợp, cái nào không hợp nếu chúng cùng sử dụng một thuật ngữ.

1.2.2. Phương pháp lọc cộng tác (Collaborative Filtering)

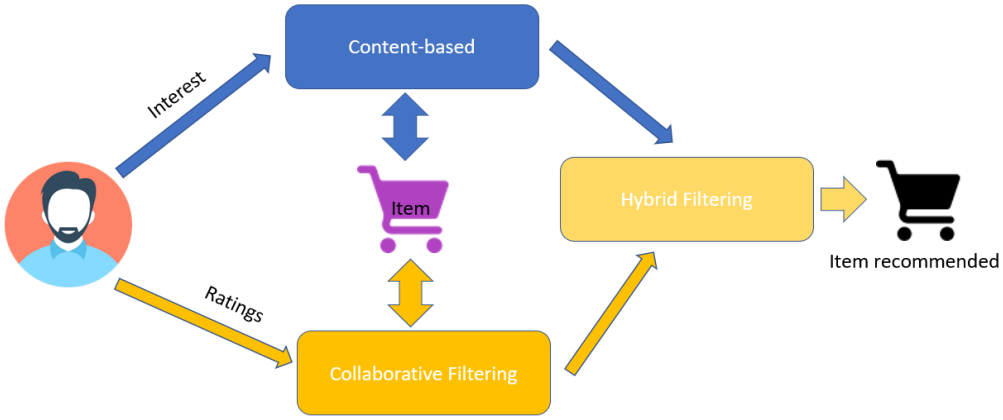
Lọc cộng tác hoạt động bằng cách thu thập phản hồi (Ratings) của người dùng dưới dạng xếp hạng cho các mặt hàng, sản phẩm trong một miền giá trị nhất định có thể từ (-1,1) hay (0,5) tùy thuộc vào bài toán của nhà phát triển. Với giá trị đánh giá này, hệ thống sẽ khai thác các điểm tương đồng về hành vi xếp hạng giữa một số người dùng để xác định cách đề xuất một sản phẩm đến với người dùng. Kỹ thuật lọc cộng tác hiện nay đã ứng dụng thành công và rộng rãi trong nhiều lĩnh vực như thương mại điện tử, mạng xã hội,.. điển hình như Amazon, Shopee, Lazada,… So với lọc theo nội dung, lọc cộng tác có thể lọc được mọi loại thông tin mà không cần mô tả dưới dạng văn bản. Hiện nay đã có nhiều nghiên cứu cho thấy phương pháp lọc cộng tác cho kết quả cao hơn lọc nội dung.

Các phương pháp lọc cộng tác (CF) có thể được chia nhỏ hơn nữa thành hai phương pháp sau:

- Neighborhood-based (Memory-based): Trong kỹ thuật này, một tập hợp con người dùng được chọn dựa trên sự tương đồng của họ với người dùng đang hoạt động và kết hợp có trọng số các xếp hạng của họ được sử dụng để đưa ra các dự đoán cho người dùng này.
- Model-based: Kỹ thuật này đưa ra gợi ý bằng cách ước lượng các tham số của mô hình thống kê cho xếp hạng của người dùng, Phương pháp liên quan đến việc xây dựng các mô hình dự đoán trên dữ liệu thu nhập được trong quá khứ, Kỹ thuật này phân tích ma trận người dùng – sản phẩm để nhận diện các mối quan hệ giữa các sản phẩm, những mối quan hệ này được dùng để so sánh danh sách những gợi ý top-N.

1.2.3. Phương pháp lọc kết hợp (Hybrid Filtering)

Để tận dụng thế mạnh của hai phương pháp lọc cộng tác và lọc nội dung đã có một số phương pháp lai được đề xuất kết hợp cả hai. Một cách tiếp cận đơn giản là cho phép cả phương pháp lọc cộng tác và dựa trên nội dung để tạo ra các danh sách đề xuất được xếp hạng riêng biệt và sau đó hợp nhất các kết quả của chúng để tạo ra một danh sách cuối cùng.



Hình 1: *Mô tả phương pháp Hybrid Filtering*

Việc kết hợp lọc cộng tác và lọc dựa trên nội dung với nhau có thể giúp khắc phục nhược điểm mà chúng ta đang gặp phải khi sử dụng chúng một cách riêng biệt và cũng có thể hiệu quả hơn trong một số trường hợp. Có một số nghiên cứu so sánh hiệu quả của các phương pháp tiếp cận thông thường và các phương pháp lai và cho thấy rằng bằng cách sử dụng phương pháp kết hợp cho kết quả tốt hơn trong nhiều trường hợp.

1.3. Phương pháp tư vấn cộng tác based-line

Trong hầu hết các hệ thống lọc cộng tác, dữ liệu đầu vào chính là sự đánh giá về các sản phẩm của một số người dùng. Trong hệ thống tư vấn bài toán cho cổng lập trình trực tuyến, dữ liệu đầu vào là sự đánh giá về bài toán lập trình của sinh viên. Về mặt biểu diễn, cho *U* là tập người dùng (sinh viên), và *P* là tập các item (bài toán lập trình), cơ sở dữ liệu *D* là 1 tập hợp của các bộ (*i,x,r*). Trong đó $i \in U$, $x \in P$, và $r \in R$ là giá trị đánh giá của bài toán *x* bởi người dùng *i*. Để đơn giản hóa, chúng ta biểu diễn là r_{ix} là đánh giá của bài toán *x* bởi người dùng *i*, $r_{ix}=(i,x,r)$. Hơn nữa, cho người dùng $i \in U$, chúng ta biểu diễn $P_i \in P$ là tập các bài toán đã được đánh giá bởi người dùng *i*. Ngược lại chúng ta biểu diễn U_x là tập người dùng mà đã đánh giá bài toán *x*.

Phương pháp lọc cộng tác dựa trên bộ nhớ có đặc điểm sử dụng tất cả dữ liệu có sẵn để dự đoán đánh giá về một bài toán mới bởi một số người dùng cụ thể. Phương pháp này được áp dụng thành công cho nhiều ứng dụng thực tế khác nhau. Phương pháp lọc cộng tác dựa trên bộ nhớ thường thấy, sử dụng các bước sau: tính toán sự tương tự hay trọng số (w_{ij}), nó phản ánh khoảng cách, độ tương quan hay trọng số giữa 2 người dùng hay 2 bài toán *i* và *j*. Phương pháp đưa ra dự đoán tích cực cho người dùng bằng cách lấy trọng số trung bình của tất cả các đánh giá của người dùng hay các bài toán trên một người dùng hay một bài toán nào đó.

Để tính toán sự tương tự giữa 2 người dùng *u* và *v* hay tương tự giữa 2 bài toán *i* và *j* chúng ta có thể tính toán dựa trên độ tương quan Pearson hay vector tương tự cosin.

Vector tương tự cosin - đặc điểm cơ bản của kỹ thuật này là xem xét sự tương tự giữa 2 tài liệu văn bản. Kỹ thuật này coi từng tài liệu như là một vector tần số và tính toán cosin của góc được hình thành bởi các vector tần số. Áp dụng vào lọc cộng tác, các người dùng và các bài toán được thay thế cho tài liệu và xếp hạng tần số. Nếu *R(m x n)* là ma trận tương tác người dùng - bài toán, thì sự tương tự(trọng số) giữa 2 bài toán *i* và *j* được định nghĩa là cosin của các vector *n* chiều tương ứng với cột thứ *i* và thứ *j* của ma trận *R*. Tương tự ta tính được sự tương tự giữa 2 người dùng *u* và *v*. Về biểu diễn, vector tương tự cosin giữa 2 bài toán *i* và *j* như sau:

$$W_{i,j} = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \bullet \vec{j}}{||\vec{i}|| * ||\vec{j}||} \quad (1)$$

Ví dụ cho *n* bài toán và ma trận *R(n x n)* đã được tính, vector $A(x_1,y_1)$ và vector $B(x_2,y_2)$. Vector tương tự cosin giữa bài toán A và bài toán B là:

$$W_{A,B}=\cos(\vec{A},\vec{B})=\frac{\vec{A}\bullet\vec{B}}{||\vec{A}||*||\vec{B}||}=\frac{x_1x_2+y_1y_2}{\sqrt{x_1^2+y_1^2}\sqrt{x_2^2+y_2^2}} \quad (2)$$

Như vậy chúng ta thấy việc tính tương tự theo vector tương tự cosin rất đơn giản, tuy nhiên kỹ thuật này gặp phải vấn đề khi người dùng sử dụng những đánh giá khác nhau (người dùng có thể đánh giá bài toán ở nhiều mức). Để giải quyết vấn đề này, kỹ thuật cosin đã điều chỉnh và sử dụng công thức như độ tương quan Pearson. Nói ở khía cạnh khác, chúng ta có thể hiểu là độ tương quan Pearson chính là sự cải tiến của vector tương tự cosin khi thêm vào trung bình đánh giá của một người dùng trên các bài toán hay trung bình đánh giá của các người dùng trên một bài toán cụ thể. Độ tương quan Pearson sẽ được nói rõ trong phần phương pháp Item-based và phương pháp User-based.

1.3.1. Phương pháp tư vấn cộng tác User-based

Phương pháp User-based tình toán sự tương tự giữa 2 người dùng *x* và *y* sử dụng công thức độ tương quan Person hoặc dựa trên Vector Cosin sau đó hệ thống sẽ xác định tập các giá trị trọng số của người dùng trên tất cả các bài toán dựa vào độ tương tự giữa các người dùng từ đó lấy k bài toán có trọng số hay đánh giá cao nhất để đưa ra gợi ý.

Độ tương quan Pearson giữa người dùng *u* và người dùng *v* là:

$$W_{u,v}=\frac{\sum_{i\in I}(r_{u,i}-\bar{r}_u)(r_{v,i}-\bar{r}_v)}{\sqrt{\sum_{i\in I}(r_{u,i}-\bar{r}_u)^2}\sqrt{\sum_{i\in I}(r_{v,i}-\bar{r}_v)^2}} \quad (3)$$

Độ tương tự giữa người dùng *u* và ngươi dùng *v* dựa trên vector tương tự Cosin:

$$W_{u,v}=\frac{\sum_{i\in I}(r_{u,i})(r_{v,i})}{\sqrt{\sum_{i\in I}(r_{u,i})^2}\sqrt{\sum_{i\in I}(r_{v,i})^2}} \quad (4)$$

Trong đó:

- I là tập hợp tất cả các bài toán mà cả hai người dùng *u* và *v* đã đánh giá.
- $\bar{r}_u\bar{r}_u$ là đánh giá trung bình của các mục mà được người dùng *u* đánh giá.
- $\bar{r}_v\bar{r}_v$ là đánh giá trung bình của các mục mà được người dùng *v* đánh giá.
- $r_{u,i}$ là đánh giá của người dùng *u* trên bài toán *i*.
- $r_{v,i}$ là đánh giá của người dùng **v** trên bài toán ***i***.

Trọng số đánh giá trung bình của người dùng **a** trên bài toán **i**:

$$P_{a,i} = \bar{r_a} + \frac{\sum_{u \in U}(r_{u,i} - \bar{r}_u) w_{a,u}}{\sum_{u \in U} |w_{a,u}|} \quad (5)$$

Trong đó:

- U là tập tất cả các người dùng đã đánh giá bài toán i.
- $\bar{r}_a \bar{r}_a$ là trung bình đánh giá của tập $P\backslash\{i\}$(tập các bài toán ngoại trừ bài toán i).
- $w_{a,u}$ là độ tương quan Pearson giữa người dùng a và người dùng u.

Phương pháp lọc cộng tác dựa trên người dùng (User-Based) sử dụng *độ tương quan Person* đưa ra gợi ý cho người dùng a được thể hiện như sau:

Bước 1. Xác định ma trận đánh giá $\{r_{ij}\}$ với công thức:

$$r_{ij} = \begin{cases} a & \text{nếu } u_i \text{ có đánh giá } p_i \text{ với mức là } a \\ \emptyset & \text{nếu } u_i \text{ chưa có đánh giá } p_i \end{cases}$$

Bước 2. Xác định tập các bài toán mà người dùng a chưa đánh giá (N).

Bước 3. Với mỗi bài toán $i \in N$, xác định tập người dùng U đã đánh giá bài toán i

Bước 4. Với $u \in U$, tính trọng số tương tự giữa người dùng a và người dùng u:

$$W_{a,u} = \frac{\sum_{i \in I}(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I}(r_{a,i} - \bar{r}_a)^2} \sqrt{\sum_{i \in I}(r_{u,i} - \bar{r}_u)^2}} \quad (6)$$

Bước 5. Với mỗi bài toán $i \in N$, tính trọng số đánh giá trung bình của người dùng a trên bài toán i:

$$P_{a,i} = \bar{r_a} + \frac{\sum_{u \in U}(r_{u,i} - \bar{r}_u) w_{a,u}}{\sum_{u \in U} |w_{a,u}|} \quad (7)$$

Bước 6. Sắp xếp các bài toán của tập N theo trọng số đánh giá trung bình tính được ở **Bước 5**. Sau đó lấy ra k bài toán có trọng số đánh giá trung bình cao nhất để tư vấn cho người dùng a.

Ví dụ 1: Tính trọng số đánh giá của người dùng cho bài toán theo phương pháp lọc cộng tác dựa trên người dùng sử dụng *độ tương quan Person*

Ta có ma trận đánh giá người dùng:

Người dùng	*Bài toán*						
	p_1	p_2	p_3	p_4	p_5	p_6	p_7
u_1	4	∅	0	5	∅	2	∅
u_2	∅	3	2	2	1	∅	3
u_3	2	4	3	1	∅	∅	2
u_4	0	1	1	∅	2	1	3
u_5	?	2	?	0	4	5	?

Bảng 1: *Ma trận đánh giá Người dùng – Bài toán*

Ta có thể tính trọng số đánh giá của người dùng u_5 trên bài toán p_1 như sau:

$$P_{5,1} = \overline{r_5} + \frac{\sum_{u \in U}(r_{u,1} - \overline{r_u}) w_{5,u}}{\sum_{u \in U} |w_{5,u}|}$$

$$= \underline{r_5} + \frac{(r_{1,1} - \underline{r_1}) w_{5,1} + (r_{3,1} - \underline{r_3}) w_{5,3} + (r_{4,1} - \underline{r_4}) w_{5,4}}{|w_{5,1}| + |w_{5,3}| + |w_{5,4}|}$$

Tính trọng số tương tự giữa U_5 và các User còn lại

$$W_{5,1} = \frac{(r_{5,4} - r_5)(r_{5,4} - r_5) + (r_{1,4} - r_1)(r_{5,4} - r_5)}{\sqrt{(r_{5,4} - r_5)^2 + (r_{5,6} - r_5)^2} \sqrt{(r_{1,4} - r_1)^2 + (r_{1,6} - r_1)^2}}$$

$$= \frac{(0 - 2.75)(5 - 2.4) + (5 - 2.75)(2 - 2.4)}{\sqrt{(0 - 2.75)^2 + (5 - 2.75)^2} \cdot \sqrt{(5 - 2.4)^2 + (2 - 2.4)^2}}$$

$$= -0.861$$

$$W_{5,3} = \frac{(r_{5,4} - r_5)(r_{3,4} - r_3) + (r_{5,2} - r_5)(r_{3,2} - r_3)}{\sqrt{(r_{5,4} - r_5)^2 + (r_{5,2} - r_5)^2} \sqrt{(r_{3,4} - r_3)^2 + (r_{3,2} - r_3)^2}}$$

$$= \frac{(0 - 2.75)(1 - 2.4) + (3 - 2.75)(4 - 2.4)}{\sqrt{(0 - 2.75)^2 + (3 - 2.75)^2} \cdot \sqrt{(1 - 2.4)^2 + (4 - 2.4)^2}}$$

$$= 0.701$$

$$W_{5,4} = \frac{(r_{5,2} - r_5)(r_{4,2} - r_4) + (r_{5,5} - r_5)(r_{4,5} - r_4) + (r_{5,6} - r_5)(r_{4,6} - r_4)}{\sqrt{(r_{5,5} - r_5)^2 + (r_{5,2} - r_5)^2 + (r_{5,6} - r_5)^2} \sqrt{(r_{4,2} - r_4)^2 + (r_{4,5} - r_4)^2 + (r_{4,6} - r_4)^2}}$$

$$= \frac{(3 - 2.75)(1 - 1.6) + (3 - 2.75)(3 - 1.6) + (4 - 2.75)(1 - 1.6)}{\sqrt{(3 - 2.75)^2 + (3 - 2.75)^2 + (4 - 2.75)^2} \cdot \sqrt{(1 - 1.6)^2 + (3 - 1.6)^2 + (1 - 1.6)^2}} = -0.258$$

➔ $P_{5,1}$ = 2.75 – 0.281 = 2.4689

Ngoài phương pháp tính độ tương đồng giữa hai user bằng độ tương quan Person ra chúng ta còn một cách khác bằng cách tính cosin của góc giữa hai vector u_1 và u_2, Độ tương quan của hai vector là 1 số trong đoạn [-1, 1]. Giá trị bằng 1 thể hiện hai vector hoàn toàn giống nhau. Hàm số cos của một góc bằng 1 nghĩa là góc giữa hai vector bằng 0, tức một vector bằng tích của một số dương với vector còn lại. Giá trị coscos bằng -1 thể hiện hai vector này hoàn toàn trái ngược nhau. Điều này cũng hợp lý, tức khi hành vi của hai users là hoàn toàn ngược nhau thì độ tương quan giữa hai vector đó là thấp nhất.Thuật toán lọc cộng tác trên người dùng(User-Based) sử dụng *vector tương tự Cosin* đưa gợi ý cho người dùng *a* được thể hiện qua các bước sau:

Bước 1. Xác định ma trận đánh giá $\{r_{ij}\}$ với công thức:

$$r_{ij} = \begin{cases} a & nếu\ u_i\ có\ đánh\ giá\ p_i\ với\ mức\ là\ a \\ \emptyset & nếu\ u_i\ chưa\ có\ đánh\ giá\ p_i \end{cases} \quad (8)$$

Bước 2. Xác định tập các bài toán mà người dùng *a* chưa đánh giá (*N*).

Bước 3. Với mỗi bài toán $i \in N$, xác định tập người dùng U đã đánh giá bài toán i

Bước 4. Với $u \in U$, tính trọng số tương tự giữa người dùng *a* và người dùng *u*:

$$W_{a,u} = \frac{\sum_{i \in I}(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I}(r_{a,i} - \bar{r}_a)^2}\sqrt{\sum_{i \in I}(r_{u,i} - \bar{r}_u)^2}} \quad (9)$$

Bước 5. Với mỗi bài toán $i \in N$, tính trọng số đánh giá trung bình của người dùng *a* trên bài toán *i*:

$$P_{a,i} = \bar{r}_a + \frac{\sum_{u \in U}(r_{u,i} - \bar{r}_u)w_{a,u}}{\sum_{u \in U}|w_{a,u}|} \quad (10)$$

Bước 6. Sắp xếp các bài toán của tập *N* theo trọng số đánh giá trung bình tính được ở **Bước 5**. Sau đó lấy ra *k* bài toán có trọng số đánh giá trung bình cao nhất để tư vấn cho người dùng *a*.

Ví dụ 2: Xét ma trận đánh giá hệ thống cho ở Bảng 1, trọng số đánh giá của người dùng U_5 lên bài toán P_1 được tính như sau:

Người dùng	*Bài toán*						
	p_1	p_2	p_3	p_4	p_5	p_6	p_7
u_1	4	∅	0	5	∅	2	∅
u_2	∅	3	2	2	1	∅	3
u_3	2	4	3	1	∅	∅	2
u_4	0	1	1	∅	2	1	3
u_5	?	2	?	0	4	5	?

$$P_{5,1} = \overline{r_5} + \frac{\sum_{u \in U}(r_{u,1} - \overline{r_u}) w_{5,u}}{\sum_{u \in U} |w_{5,u}|}$$

$$= \overline{r_5} + \frac{(r_{1,1} - \overline{r_1}) w_{5,1} + (r_{3,1} - \overline{r_3}) w_{5,3} + (r_{4,1} - \overline{r_4}) w_{5,4}}{|w_{5,1}| + |w_{5,3}| + |w_{5,4}|}$$

$$= 2.5 + \frac{(4 - {}^{7}/_{3}) \cdot 0.371 + (2 - 2.5) \cdot 0.970 + (0 - 1.6) \cdot 0.894}{0.371 + 0.970 + 0.894}$$

$$= 0.999$$

Trong đó:

$$W_{5,1} = \frac{r_{5,4} r_{1,4} + r_{5,6} r_{1,6}}{\sqrt{(r_{5,4})^2 + (r_{1,4})^2} \sqrt{(r_{1,4})^2 + (r_{1,6})^2}}$$

$$= \frac{5*0 + 5*2}{\sqrt{(0)^2 + (5)^2} \cdot \sqrt{(5)^2 + (2)^2}}$$

$$= 0.371$$

$$W_{5,3} = \frac{r_{5,2} r_{3,2} + r_{5,4} r_{3,4}}{\sqrt{(r_{5,4})^2 + (r_{5,2})^2} \sqrt{(r_{3,4})^2 + (r_{3,2})^2}}$$

$$= \frac{2*4 + 1*0}{\sqrt{(0)^2 + (2)^2} \cdot \sqrt{(4)^2 + (1)^2}}$$

$$= 0.97$$

$$W_{5,4} = \frac{r_{5,2} r_{4,2} + r_{5,5} r_{4,5} + r_{5,6} r_{4,6}}{\sqrt{(r_{5,5})^2 + (r_{5,2})^2 + (r_{5,6})^2} \sqrt{(r_{4,6})^2 + (r_{4,5})^2 + (r_{4,2})^2}}$$

$$= \frac{1*2 + 2*4 + 5*1}{\sqrt{(5)^2 + (4)^2 + 2^2} \cdot \sqrt{(1)^2 + (2)^2 + 1}}$$

$$= 0.894$$

1.3.2. Phương pháp tư vấn cộng tác Item-based

Phương pháp Item-based tính toán sự tương tự (*similarity*) giữa 2 bài toán *i* và *j*. Sau đó trên cơ sở người dùng cần tư vấn, hệ thống sẽ lấy ra *k* bài toán mà người dùng có khả năng quan tâm nhất. Sau đó hệ thống tư vấn sẽ xếp hạng *k* bài toán này dựa trên trọng số đã tính toán được để đưa ra quyết định kiến nghị cho người dùng. Để tính sự tương đồng giữa 2 bài toán và trọng số đánh giá giữa người dùng *u* trên bài toán *i*, chúng ta sử dụng độ tương quan Pearson hoặc vector tương tự Cosin để tính toán.

Độ tương quan Pearson giữa bài toán *i* và bài toán *j*:

$$W_{i,j}=\frac{\sum_{u\in U}(r_{u,i}-\bar{r}_i)(r_{u,j}-\bar{r}_j)}{\sqrt{\sum_{u\in U}(r_{u,i}-\bar{r}_i)^2}\sqrt{\sum_{u\in U}(r_{u,j}-\bar{r}_j)^2}} \quad (11)$$

Tương tự giữa bài toán i và bài toán j dựa trên vector tương tự Cosin:

$$W_{i,j}=\frac{\sum_{u\in U} r_{u,i} r_{u,j}}{\sqrt{\sum_{u\in U}(r_{u,i})^2}\sqrt{\sum_{u\in U}(r_{u,j})^2}} \quad (12)$$

Trong đó:

- U là tập hợp những người có đánh giá cả 2 bài toán ***i*** và ***j***.
- $r_{u,i}$ là đánh giá của người dùng *u* trên bài toán *i*.
- $r_{u,j}$ là đánh giá của người dùng *u* trên bài toán *j*.
- $\bar{r}_i\bar{r}_i$ là đánh giá trung bình của mục *i* bởi các người dùng.
- $\bar{r}_j\bar{r}_j$ là đánh giá trung bình của mục *j* bởi các người dùng.

Trọng số đánh giá trung bình của người dùng **u** trên bài toán **i**:

$$P_{u,i} = \frac{\sum_{n\in N} r_{u,n} w_{i,n}}{\sum_{n\in N} |w_{i,n}|} \quad (13)$$

Trong đó:

- N là tập các bài toán do người dùng *u* đã đánh giá.
- $r_{u,n}$ là đánh giá của người dùng *u* trên bài toán,
- $w_{i,n}$ là độ tương quan Pearson (trọng số tương tự) giữa bài toán *i* và bài toán *n*.

Thuật toán lọc cộng tác dựa trên sản phẩm(bài toán) (Item-Based) sử dụng độ tương quan Pearson đưa ra tư vấn cho người dùng **u** được thể hiện theo các bước sau:

Bước 1. Xác định ma trận đánh giá $\{r_{ij}\}$ với công thức:

$$r_{ij} = \begin{cases} a & \text{nếu } u_i \text{ có đánh giá } p_i \text{ với mức là } a \\ \emptyset & \text{nếu } u_i \text{ chưa có đánh giá } p_i \end{cases}$$

Bước 2. Xác định tập các bài toán do người dùng u đã đánh giá (*N*).

Bước 3. Với mỗi bài toán *n* ϵ *N*, *i* ϵ *P**N*(*P* là tập tất cả các bài toán), tính trọng số tương tự giữa bài toán *i* và bài toán *n*:

$$W_{i,n} = \frac{\sum_{u \in U}(r_{u,i} - \bar{r}_n)(r_{u,n} - \bar{r}_n)}{\sqrt{\sum_{u \in U}(r_{u,i} - \bar{r}_n)^2}\sqrt{\sum_{u \in U}(r_{u,n} - \bar{r}_n)^2}} \quad (14)$$

Bước 4. Với *i* ϵ *P**N*, tính trọng số đánh giá trung bình của người dùng *u* trên bài toán *i*:

$$P_{u,i} = \frac{\sum_{n \in N} r_{u,n} w_{i,n}}{\sum_{n \in N} |w_{i,n}|} \quad (15)$$

Bước 5. Sắp xếp các bài toán thuộc tập *P**N* theo trọng số đánh giá trung bình tính được ở **Bước 4**. Sau đó lấy ra *k* bài toán có trọng số đánh giá trung bình cao nhất để tư vấn cho người dùng *u*.

Ví dụ 3: Tính trọng số đánh giá của người dùng cho bài toán theo phương pháp lọc cộng tác dựa trên bài toán sử dụng *độ tương quan Person*

Giả sử ta có bảng ma trận trọng số sau :

Người dùng	*Bài toán*						
	p_1	p_2	p_3	p_4	p_5	p_6	p_7
u_1	4	∅	0	5	∅	2	∅
u_2	∅	3	2	2	1	∅	3
u_3	2	4	3	1	∅	∅	2
u_4	0	1	1	∅	2	1	3
u_5	?	2	?	0	4	5	?

Bảng 2: *Ma trận đánh giá R theo độ tương quan Pearson*

Trọng số đánh giá (*ratings*) của người dùng U_5 trên bài toán P_1 được tính như sau :

$$P_{5,1} = \frac{\sum_n r_{5,n} w_{1,n}}{\sum_n |w_{1,n}|} = \frac{r_{5,2} w_{1,2} + r_{5,4} w_{1,4} + r_{5,5} w_{1,5} + r_{5,6} w_{1,6}}{|w_{1,6}| + |w_{1,4}| + |w_{1,5}| + |w_{1,2}|}$$

Trong đó $w_{1,6}$, $w_{1,5}$, $w_{1,2}$, $w_{1,4}$ lần lượt là độ tương tự giữa bài toán 1 với bài toán 6, 5, 2, 4 và được tính dựa trên công thức độ tương quan person như sau:

$$W_{1,2} = \frac{(r_{3,1}-r_1)(r_{3,2}-r_2)+(r_{4,1}-r_1)(r_{4,2}-r_2)}{\sqrt{(r_{3,1}-r_1)^2+(r_{4,1}-r_1)^2}\sqrt{(r_{3,2}-r_2)^2+(r_{4,2}-r_2)^2}}$$

$$= \frac{(1-\frac{4}{3})(4-\frac{11}{4})+(0-\frac{4}{3})(1-\frac{11}{4})}{\sqrt{(1-\frac{4}{3})^2+(0-\frac{4}{3})^2}\cdot\sqrt{(4-\frac{11}{4})^2+(1-\frac{11}{4})^2}}$$

$$= 0.648$$

Tương tự ta có $w_{1,4} = 0.949$, $w_{1,5} = 1$, $w_{1,6} = -0.111$

Ta đây ta có :

$$P_{5,1} = \frac{3\cdot 0.648+0\cdot 0.949+3\cdot 1+4\cdot(-0.111)}{0.648+0.949+1+0.111}$$

$$= 1.662$$

Thuật toán lọc cộng tác dựa trên sản phẩm(bài toán) (*Item-based)* sử dụng *vector tương tự cosin* về cơ bản các bước làm tương tự như sử dụng *độ tương quan pearson* chỉ khác công thức sử dụng ở bước 3 thay vì sử dụng công thức *độ tương quan person* thì chúng ta sẽ tính bằng công thức *vector tương tự cosin* như sau:

$$W_{i,n}=\frac{\sum_{u\in U}(r_{u,i})(r_{u,n})}{\sqrt{\sum_{u\in U}(r_{u,i})^2}\sqrt{\sum_{u\in U}(r_{u,n})^2}} \qquad (16)$$

Ví dụ 4: Với ma trận trọng có được ở *ví dụ 3* trọng số đánh giá(*ratings*) của người dùng U_5 trên bài toán P_1 sử dụng công thức *vector tương tự cosin* được tính như sau:

$$P_{5,1} = \frac{\sum_n r_{5,n}\cdot w_{1,n}}{\sum_n |w_{1,n}|}$$

$$= \frac{r_{5,2}\cdot w_{1,2}+r_{5,4}\cdot w_{1,4}+r_{5,5}\cdot w_{1,5}+r_{5,6}\cdot w_{1,6}}{|w_{1,2}|+|w_{1,4}|+|w_{1,5}|+|w_{1,6}|}$$

$$= \frac{3\cdot 0.970+0\cdot 0.997+3\cdot 1+4\cdot 0.707}{0.970+0.997+1+0.707}$$

$$= 2.378$$

Trong đó:

$$W_{1,2} = \frac{r_{3,1} \cdot r_{3,2} + r_{4,1} \cdot r_{4,2}}{\sqrt{(r_{3,1})^2 + (r_{4,1})^2} \cdot \sqrt{(r_{3,2})^2 + (r_{4,2})^2}}$$

$$= \frac{1 \cdot 4 + 0 \cdot 1}{\sqrt{1^2 + 0} \cdot \sqrt{4^2 + 1^2}}$$

$$= 0.970$$

Tương tự ta có $w_{1,4} = 0.997$, $w_{1,5} = 1$, $w_{1,6} = 0.707$

1.4. Một số vấn đề của tư vấn cộng tác

Trong phần này, tôi sẽ trình bày một số trở ngại phổ biến trong tư vấn lọc cộng tác cũng như trình bày một số nghiên cứu giải quyết chúng

Sự thưa thớt (*Sparsity***)**: Nói một cách đơn giản, hầu hết người dùng không xếp hạng hầu hết các bài toán, mặt hàng, sản phẩm, và do đó ma trận đánh giá của người dùng thường rất thưa thớt. Đây là một vấn đề rất lớn đối với hệ thống lọc cộng tác vì nó làm giảm xác suất tìm thấy một nhóm người dùng có xếp hạng. Sự cố này thường xảy ra khi hệ thống có tỷ lệ rất cao bài toán với người dùng hay hệ thống đang trong giai đoạn ban đầu sử dụng. Nhiều nhà nghiên cứu đã cố gắng giảm nhẹ vấn đề rời rạc. Sarwar và các cộng sự đã đề xuất một phương pháp tiếp cận dựa trên bài toán để giải quyết vấn đề mở rộng và vấn đề ít dữ liệu đánh giá. Một cách tiếp cận khác, đó là cắt giảm số chiều, nhằm mục đích giảm số chiều của ma trận tương tác người dùng - bài toán. Một chiến lược đơn giản là phân cụm các mặt hàng hay người dùng để đưa ra các kiến nghị. Nhiều kỹ thuật tiên tiến được áp dụng để làm giảm số chiều. Ví dụ như kỹ thuật thống kê như là nguyên lý phân tích thành phần (Principal Component Analysis-PCA) và kỹ thuật phục hồi thông tin như là lập chỉ mục ngữ nghĩa tiềm ẩn (Latent Semantic Indexing). Nghiên cứu thực nghiệm cho thấy rằng việc cắt giảm số chiều có thể cải thiện được chất lượng kiến nghị 1 cách đáng kể trong một số ứng dụng. Việc cắt giảm số chiều giải quyết được vấn đề rời rạc bởi việc loại bỏ những người dùng ko đại diện hoặc người dùng vô nghĩa. Các nhà nghiên cứu cũng đã cố gắng kết hợp phương pháp lọc lọc cộng tác với phương pháp kiến nghị dựa trên nội dung để làm giảm vấn đề rời rạc. Phương pháp như vậy xem xét sự tương quan giữa người dùng - bài toán nhưng cũng tương tự như khách hàng và các bài toán bắt nguồn từ đặc tính hay tính chất nội tại của nó. Chúng ta để cập tới phương pháp này như là một phương pháp lai. Hầu hết những nghiên cứu đầu tiên sử dụng phương pháp lai đã chứng tỏ được việc cải thiện được chất lượng của

kiến nghị so với phương pháp dựa trên người dùng đã nói ở trên. Tuy nhiên, phương pháp tiếp cận lại đòi hỏi có thêm thông tin về bài toán và một độ đo để tính toán sự giống nhau giữa chúng. Trong thực tế, thông tin bài toán có thể khó hay chi phí tốn kém để có được thông tin.

Cold-start: Các mặt hàng mới và người dùng mới đặt ra một thách thức đáng kể cho các hệ thống tư vấn, Gọi chung những vấn đề này được gọi là *cold-start problem.* Vấn đề đầu tiên trong số vấn đề này nảy sinh trong lọc cộng tác hệ thống, trong đó một mục không thể được đề xuất trừ khi một số người dùng đã xếp hạng nó trước đó. Vấn đề này không chỉ áp dụng cho các bài toán mới mà còn cho các bài toán bị che khuất đặc biệt gây bất lợi cho những người dùng có thị hiếu chiết trung. Như vậy vấn đề *new-item* cũng thường được gọi là *first-rater problem.*Vì dựa trên nội dung phương pháp tiếp cận không dựa vào xếp hạng từ những người dùng khác chúng có thể được sử dụng để đưa ra các đề xuất cho tất cả các bài toán, các thuộc tính được cung cấp cho các mặt hàng là có sẵn. Trên thực tế, các dự đoán dựa trên nội dung của những người dùng tương tự cũng có thể được sử dụng để cải thiện hơn nữa các dự đoán dựa trên nội dung của những người dùng tương tự cũng có thể được sử dụng để cải thiện hơn nữa các dự đoán cho người dùng đang hoạt động

Các vấn đề của *người dùng mới* (*new-user)* là tương đối khó khăn để giải quyết vì không có sở thích trước của một người dùng, không thể tìm thấy những người dùng tương tự hoặc để xây dựng một hồ sơ dựa trên nội dung. Do đó, nghiên cứu trong lĩnh vực này chủ yếu tập trung vào việc lựa chọn một cách hiệu quả các mặt hàng được người dùng xếp hạng để nhanh chóng cải thiện hiệu suất tư vấn với ít phản hồi của người dùng nhất. Trong bối cảnh này, các kỹ thuật cổ điển từ *active learning* có thể được tận dụng để giải quyết vấn đề này.

Gian lận (Fraud): Vì hệ thống tư vấn ngày càng được áp dụng trên các trang web, sàn thương mại điện tử, chúng đã bắt đầu đóng một vai trò quan trọng trong việc ảnh hưởng đến lợi nhuận của người bán. Điều này đã dẫn đến nhiều nhà cung cấp thiếu đạo đức tham gia vào các hình thức khác nhau gian lận đối với hệ thống giới thiệu vì lợi ích của họ ngay cả trong cổng lập trình trực tuyến cũng có các hành vi sao chép code để vượt qua các contest điều này sẽ ảnh hưởng rất lớn đến chất lượng dự đoán tiếp theo cho lập trình viên đó. Thông thường, họ cố gắng để tăng khả năng mong muốn được nhận thức về sản phẩm của chính họ (*push attacks)* hoặc hạ thấp xếp hạng của đối thủ cạnh

tranh của họ (*nuck attacks*). Những kiểu tấn công này đã được nghiên cứu rộng rãi như *shilling attacks* hoặc *profile injection attacks*. Ví dụ trên một sàn thương mại điện tử, kẻ tấn công chỉ định hay đánh giá các sản phẩm của đối phương nhằm hạ thấp xếp hạng của các mặt hàng này xuống và các sản phẩm có xếp hạng cao sẽ được đẩy lên. Các nghiên cứu đã chỉ ra rằng các cuộc tấn công hay gian lận này có thể gây bất lợi cho xếp hạng dự đoán, mặc dù lọc cộng tác dựa trên sản phẩm (item-based) có xu hướng bị chịu tác động nhiều hơn chính vì vậy mà các phương pháp dựa trên nội dung (content-based) chỉ dựa vào xếp hạng của người dùng trong quá khứ, không bị ảnh hưởng bởi các cuộc tấn công đưa vào hồ sơ.

1.5. Kết luận

Mặc dù các phương pháp dựa trên nội dung thuần túy tránh được một số cạm bẫy đã thảo luận ở trên, lọc cộng tác vẫn có một số ưu điểm chính so với chúng. Thứ nhất, CF có thể thực hiện trong các miền không có nhiều nội dung được liên kết với các mục hoặc trong đó máy tính khó phân tích nội dung, chẳng hạn như ý tưởng, quan điểm, v.v,.. Thứ hai, hệ thống CF có khả năng cung cấp các tư vấn có tính bất ngờ, tức là nó có thể đề xuất các mục có liên quan đến người dùng, nhưng không chứa nội dung từ hồ sơ của người dùng. Chính vì vậy mà phương pháp lọc cộng tác được đề xuất làm phương pháp trong hệ thống tư vấn của cổng lập trình trực tuyến cũng một phần do tính đặc trưng của cổng lập trình trực tuyến khó có thể gian lận trong đánh giá vấn đề mà lọc cộng tác rất khó giải quyết với những ratings không chính xác. Ở chương 2, luận văn sẽ trình bày về phương pháp lọc cộng tác cho cổng lập trình trực tuyến nhằm nâng cao hiệu quả tư vấn.

CHƯƠNG 2: PHƯƠNG PHÁP TƯ VẤN CỘNG TÁC CHO CỔNG LẬP TRÌNH TRỰC TUYẾN

2.1. Phát biểu bài toán

Tư vấn lọc cộng tác (*collaborative filtering recommendation*) là phương pháp phổ biến trong xây dựng các hệ tư vấn được ứng dụng rộng rãi trong thương mại điện tử, Phương pháp được xây dựng từ tập người dùng $U = \{u_1, u_2, \ldots, u_n\}$ và tập các sản phẩm $P = \{p_1, p_2, \ldots, p_m\}$. Mỗi người dùng $u_i \in U$ đưa ra đánh giá của mình cho một số sản phẩm $p_x \in P$ bằng một số $r_{ix} \in \Omega$ (Ω có thể là tập các số nguyên hoặc tập các số thực). Ma trận đánh giá R là đầu vào duy nhất của các phương pháp tư vấn cộng tác [5]. Để thuận tiện trong trình bày, thay bằng viết $u_i \in U$ và $p_x \in P$ ta viết ngắn gọn thành $i \in U$ và $x \in P$. Dựa vào ma trận đánh giá $R = \{r_{ix}: i=1, 2,.., n; x = 1, 2,.., m\}$, các phương pháp tư vấn lọc cộng tác khai thác những khía cạnh liên quan đến cộng đồng đồng người dùng có cùng chung sở thích để cung cấp cho người dùng này những sản phẩm phù hợp nhất với họ. Tư tưởng chủ đạo của lọc cộng tác là những người dùng có sở thích tương tự nhau trong quá khứ thì họ có thể có chung sở thích trong tương lai. Mỗi người dùng trong hệ tư vấn lọc cộng tác là độc lập với người dùng còn lại.

Bài toán tư vấn nội dung số cho mỗi người dùng trên cổng lập trình trực tuyến có thể được xem xét như bài toán tư vấn lọc cộng tác điển hình. Cụ thể luận văn này xây dựng hệ tư vấn bài toán cho người dùng trên cổng lập trình trực tuyến Dlab sử dụng phương pháp lọc cộng tác. Gọi tập người dùng là $U = \{u_1, u_2,.., u_n\}$, tập bài toán là $P = \{p_1, p_2,.., p_m\}$. Tập U và P lần lượt là tập người dùng và bài tập là dữ liệu thực được thu thập trên cổng lập trực tuyến trình Dlab. Kết quả lập trình của mỗi người dùng $i \in U$ giải quyết bài toán $x \in P$ được hệ thống ghi nhận một cách tự động bằng một số r_{ix}. Trong đó, r_{ix} được ghi nhận giá trị *1* nếu giải pháp lập trình của người dùng $i \in U$ thỏa mãn tất cả các bộ dữ liệu kiểm thử đối với bài toán $x \in P$, r_{ix} được ghi nhận giá trị *-1* nếu giải pháp lập trình của người dùng $i \in U$ chưa thỏa mãn đầy đủ các bộ dữ liệu kiểm thử đối với bài toán $x \in P$, r_{ix} được ghi nhận giá trị *0* nếu người dùng $i \in U$ chưa giải quyết bài toán $x \in P$. Nhiệm vụ của phương pháp tư vấn lọc cộng tác là cung cấp tập các bài toán phù hợp với khả năng lập trình của mỗi người dùng trên cổng lập trình trực tuyến Dlab.

Có nhiều đề xuất khác nhau để giải quyết bài toán tư vấn lọc cộng tác, tuy vậy ta có thể phân loại các phương pháp thành hai cách tiếp cận chính: tư vấn lọc cộng tác dựa

vào bộ nhớ (Memory-Based) và tư vấn lọc cộng tác dựa vào mô hình(Model-Based). Trong luận văn này tập trung vào phương pháp tư vấn dựa vào bộ nhớ. Tư vấn lọc cộng tác dựa trên bộ nhớ được tiếp cận theo hai phương pháp chính: Phương pháp tư vấn lọc cộng tác dựa vào người dùng (UserBased) và tư vấn lọc cộng tác dựa vào sản phẩm(bài toán) (ItemBased). Mỗi phương pháp đều có những ưu điểm riêng khai thác khía cạnh liên quan đến người dùng hoặc bài toán. Đặc điểm chung của cả hai phương pháp này là sử dụng toàn bộ tập dữ liệu đánh giá để dự đoán quan điểm của người dùng cần được tư vấn về các bài toán mà họ chưa hề biết đến. Về bản chất, đây là phương pháp học lười hay học dựa trên ví dụ được sử dụng trong học máy. Ngoài hai phương pháp tư vấn lọc cộng tác dựa vào người dùng (UserBased) và tư vấn lọc cộng tác dựa vào sản phẩm (ItemBased) đã được trình bày ở chương 1 thì trong luận văn sẽ trình bày thêm phương pháp tư vấn lọc cộng được thực hiện bằng cách biểu diễn mối quan hệ giữa người dùng và các nội dung số trên cổng lập trình trực tuyến như một đồ thị hai phía (GraphBased). Phương pháp này sẽ được trình bày chi tiết ở mục 2.2.

2.2. Phương pháp tư vấn cộng tác cho cổng lập trình trực tuyến

Để nâng cao chất lượng tư vấn, trong mục này luận văn sẽ trình bày đề xuất một thuật toán tư vấn lọc cộng tác cho cổng lập trình trực tuyến. Phương pháp được thực hiện bằng cách biểu diễn mối quan hệ giữa người dùng và các nội dung số trên cổng lập trình trực tuyến như một đồ thị hai phía. Lợi dụng tính chất này, luận văn xem xét bài toán cần giải quyết như một vấn đề tìm kiếm trên đồ thị. Phương pháp sẽ được tiến hành như dưới đây.

2.2.1. Phương pháp ước lượng mức độ phù hợp của người dùng đối với bài toán

Giả sử ta có hệ n người dùng $U = \{u_1, u_2,.., u_n\}$, m bài toán $P = \{p_1, p_2,,., p_m\}$. Trong đó, tập người dùng U được thu thập một cách tự động ngay từ khi người dùng đăng ký tham gia cổng lập trình trực tuyến. Tập bài toán P là tập các bài toán cùng với các bộ dữ liệu kiểm thử đã được nạp trong cổng lập trình trực tuyến. Ma trận đánh giá $R = \{r_{ix}: i=1, 2,.., n; x = 1, 2,,.m\}$ được ghi nhận tự động trong cổng lập trình trực tuyến theo công thức (17).

$$r_{ix} = \begin{cases} 1 & nếu\ người\ dùng\ i \in U\ submit\ đúng\ bài\ toán\ x \in P \\ 0 & nếu\ người\ dùng\ i \in U\ chưa\ submit\ \ bài\ toán\ x \in P \\ -1 & nếu\ người\ dùng\ i \in U\ submit\ sai\ bài\ toán\ x \in P \end{cases} \quad (17)$$

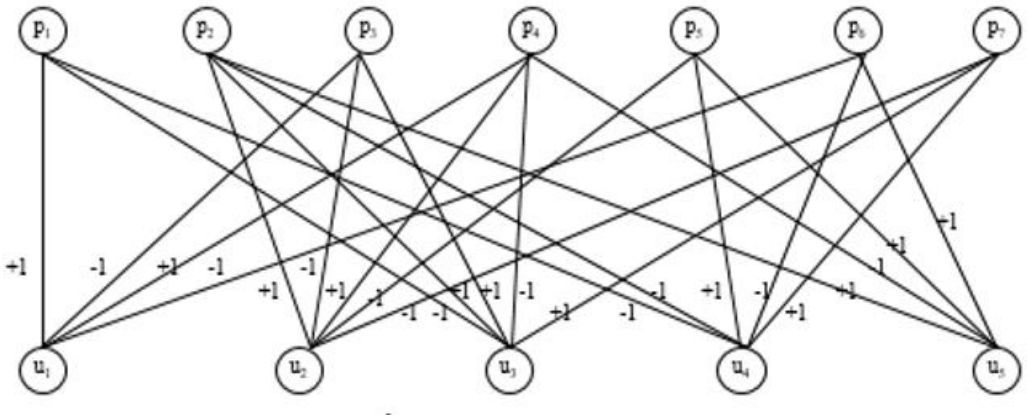
Dễ dàng nhận thấy, ma trận đánh giá R được xác định theo (17) được biểu diễn như một đồ thị hai phía (bipart graph). Một phía là tập người người dùng U, phía còn lại

là tập bài toán *P* trong cổng lập trình trực tuyến. Tập cạnh của đồ thị chỉ bao gồm các cạnh có dạng $e = (i, x)$, trong đó $i \in U$ và $x \in P$. Không tồn tại các cạnh nối các đỉnh ở cùng một phía. Nói cách khác, không tồn tại các cạnh nối giữa đỉnh người dùng và đỉnh người dùng và không tồn tại các cạnh nối giữa đỉnh bài toán và đỉnh bài toán. Ví dụ với ma trận đánh giá của hệ gồm 5 người dùng và 7 bài toán như trong Bảng 3 sẽ cho ta biểu diễn đồ thị hai phía tương ứng trong Hình 2. Trong đó, giá trị $r_{ix} = 1$ thể hiện người dùng *i* lập trình đúng bài toán *x*, $r_{ix} = -1$ thể hiện người dùng *i* lập trình chưa đúng bài toán *x*, $r_{ix} = 0$ thể hiện người dùng *i* chưa lập trình bài toán *x*.

Người dùng	*Bài toán*						
	p_1	p_2	p_3	p_4	p_5	p_6	p_7
u_1	1	0	-1	1	0	-1	0
u_2	0	1	-1	1	-1	0	-1
u_3	-1	1	1	-1	0	0	1
u_4	-1	-1	0	0	1	-1	1
u_5	0	1	0	-1	1	1	0

Bảng 3: *Ví dụ với ma trận đánh giá của hệ gồm 5 người dùng và 7 bài toán*

Theo tính chất của đồ thị hai phía, đường đi từ đỉnh người dùng $i \in U$ đến đỉnh bài toán $x \in P$ luôn có độ dài *L* lẻ ($L=1, 3, 5, 7...$). Ví dụ các đường đi u_1-p_4-u_3-p_2, u_1-p_3-u_2-p_7, u_1-p_1-u_2-p_2 là những đường đi độ dài 3, các đường đi u_1-p_4-u_2-p_2-u_3-p_7, u_2-p_3-u_1-p_6-u_4-p_1, u_1-p_1-u_3-p_7-u_4-p_2 là những đường đi độ dài 5. Tập tất cả các đường đi từ đỉnh người dùng $i \in U$ đến đỉnh bài toán $x \in P$ được chia thành 3 loại: loại 1 bao gồm tập các đường đi chỉ đi qua các cạnh có trọng số dương (+1), loại 2 bao gồm tập các đường đi chỉ đi qua các cạnh có trọng số âm (-1), loại 3 bao gồm tập các đường đi đi qua các cạnh có trọng số hoặc (+1) hoặc(-1). Ví dụ: u_1-p_4-u_3-p_2, u_1-p_4-u_2-p_2-u_3-p_7 là các đường đi loại 1; u_1-p_3-u_2-p_7, u_2-p_3-u_1-p_6-u_4-p_1 là các đường đi loại 2; u_1-p_1-u_2-p_2, u_1-p_1-u_3-p_7-u_4-p_2 là các đường đi loại 3.



Hình 2: *Đồ thị hai phía tương ứng với ma trận đánh giá của hệ gồm 5 người dùng và 7 bài toán*

Bằng trực quan ta dễ dàng quan sát được, nếu số lượng đường đi độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$ thuộc loại 1 chiếm đa số trong cả ba loại đường đi thì việc dự đoán bài toán x phù hợp với người dùng i có khả năng đúng cao nhất. Trong trường hợp này, ta dự đoán quan điểm của người dùng i đối với bài toán mới x có giá trị $r_{ix} = 1$. Nếu số lượng đường đi độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$ thuộc loại 2 chiếm đa số trong cả ba loại đường đi thì việc dự đoán bài toán x không phù hợp với người dùng i cũng có khả năng đúng cao nhất. Trong trường hợp này, ta dự đoán quan điểm của người dùng i đối với bài toán mới x có giá trị $r_{ix} = -1$. Trường hợp cuối cùng, nếu số lượng đường đi độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$ thuộc loại 3 chiếm đa số trong cả ba loại đường đi thì chúng ta không dự đoán được bài toán x có phù hợp hay không đối với người dùng i. Trong trường hợp này, ta dự đoán quan điểm của người dùng i đối với bài toán mới x có giá trị $r_{ix} = 0$. Dựa trên nhận xét này, chúng tôi đề xuất phương pháp tính toán mức độ phù hợp của người dùng đối với các bài toán mới như sau.

Gọi $W = \{w_{ix}: i = 1, 2, .., n; x = 1, 2, , ., m\}$ là ma trận liền kề biểu diễn đồ thị hai phía của ma trận đánh giá R được xác định theo công thức(2), $W^1 = \{w_{ix}^1: i = 1, 2, , ., n; x = 1, 2, , ., m\}$ là ma trận liền kề biểu diễn đồ thị hai phía cho các giá trị đánh giá dương được xác định theo công thức(3), $W^2 = \{w_{ix}^2: i = 1, 2, , ., n; x = 1, 2, , ., m\}$ là ma trận liền kề biểu diễn đồ thị hai phía cho các giá trị đánh giá âm được xác định theo công thức (4). Nói cách khác ta thực hiện tách đồ thị hai phía biểu diễn ma trận đánh giá R thành hai đồ thị con, đồ thị con W^1 chỉ bao gồm các cạnh có trọng số dương (+1), đồ thị con W^2 chỉ bao gồm các cạnh có trọng số âm (-1).

$$W = \begin{cases} 1 & nếu\ r_{ix} \neq 0 \\ 0 & otherwise \end{cases} \quad (18)$$

$$W^1 = \begin{cases} 1 & nếu\ r_{ix} > 0 \\ 0 & otherwise \end{cases} \quad (19)$$

$$W^2 = \begin{cases} 1 & nếu\ r_{ix} < 0 \\ 0 & otherwise \end{cases} \quad (20)$$

Khi đó, số lượng tất cả các đường đi có độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$ trên toàn bộ ma trận đánh giá R được xác định theo công thức (21). Đây cũng chính là số lượng của cả ba loại đường đi từ đỉnh $i \in U$ đến đỉnh $x \in P$. Số lượng đường đi loại 1(đường đi qua các cạnh có trọng số 1) từ đỉnh $i \in U$ đến đỉnh $x \in P$ được xác định theo công thức (22) . Số lượng đường đi loại 2(đường đi qua các cạnh có trọng số -1) từ đỉnh $i \in U$ đến đỉnh $x \in P$ được xác định theo công thức (23) . Số lượng đường đi loại 3(đường đi qua các cạnh có trọng số 1 hoặc -1) từ đỉnh $i \in U$ đến đỉnh $x \in P$ được xác định theo công thức (24). Trong đó, W^T là ma trận chuyển vị của W, $(W^1)^T$ là ma trận chuyển vị của W^1, $(W^2)^T$ là ma trận chuyển vị của W^2.

$$W^L = \begin{cases} W & nếu\ \ L = 1 \\ W.W^T.W^{L-2} & nếu\ L = \ 3, 5, \ldots \end{cases} \quad (21)$$

$$(W^1)^L = \begin{cases} W^1 & nếu\ \ L = 1 \\ W^1.(W^1)^T.(W^1)^{L-2} & nếu\ L = \ 3, 5, \ldots \end{cases} \quad (22)$$

$$(W^2)^L = \begin{cases} W^2 & nếu\ \ L = 1 \\ W^1.(W^2)^T.(W^1)^{L-2} & nếu\ L = \ 3, 5, \ldots \end{cases} \quad (23)$$

$$(W^3)^L = \ W^L - (W^1)^L - (W^2)^L \quad (24)$$

Như vậy ta đã xác định được số lượng của từng loại đường đi có độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$, Gọi MAX là số lượng đường đi có độ dài L lớn nhất từ đỉnh $i \in U$ đến đỉnh $x \in P$ được xác định theo (25). Khi đó, phương pháp dự đoán mức độ phù hợp của người dùng $i \in U$ đối với các bài toán mới $x \in P$ được xác định theo công thức (26).

$$MAX = max\{w_{ix}^L : i = 1, 2, .., n; x = 1, 2, .., m\} \quad (25)$$

$$r_{ix} = \begin{cases} 1 & nếu\ \frac{(w^1)_{ix}^L}{MAX} > 0.5 \\ 0 & nếu\ \frac{(w^3)_{ix}^L}{MAX} > 0.5 \\ -1 & nếu\ \frac{(w^2)_{ix}^L}{MAX} > 0.5 \end{cases} \quad (26)$$

Trong công thức dự đoán, để hạn chế các cặp (i, x) có số lượng đường đi độ dài L nhỏ nhưng có số lượng đường đi độ dài L thuộc mỗi loại vẫn chiếm đại đa số so với w_{ix}^{L}, Chính vì vậy, chúng tôi so sánh với giá trị lớn nhất trong ma trận W^L để tiến hành so sánh. Giá trị dự đoán quan điểm của người dùng i đối với bài toán mới x là r_{ix}=1 khi tỉ số $\frac{(w^1)_{ix}^{L}}{MAX} > 0.5$. Điều này có nghĩa số lượng đường đi độ dài L từ đỉnh i đến đỉnh x phải đạt giá trị đủ lớn và số lượng đường đi độ dài L đi qua các cạnh có trọng số dương chiếm đại đa số. Tương tự như vậy, giá trị dự đoán r_{ix} =-1 khi số lượng đường đi độ dài L từ đỉnh i đến đỉnh x phải đạt giá trị đủ lớn và số lượng đường đi độ dài L đi qua các cạnh có trọng số âm chiếm đại đa số. Giá trị dự đoán r_{ix} =0 khi số lượng đường đi độ dài L từ đỉnh i đến đỉnh x phải đạt giá trị đủ lớn và số lượng dường đi độ dài L đi qua các cạnh có trọng số cả âm lẫn dương chiếm đại đa số. Dựa vào phương pháp dự đoán đã được xây dựng theo (26) luận văn đề xuất thuật toán tư vấn cộng tác cho cổng lập trình trực tuyến như trong Mục 2.2.2.

2.2.2. Thuật toán tư vấn cộng tác cho cổng lập trình trực tuyến

Thuật toán tư vấn cộng tác cho cổng lập trình trực tuyến (ký hiệu là GraphBased) được thực hiện thông qua 4 bước như trong Hình 6. Tại bước 1 của thuật toán ta tiến hành xây dựng các đồi thị hai phía. Đồ thị W dùng để xác định số lượng các đường đi có độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$. Đồ thị W^1 dùng để xác định số lượng các đường đi có độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$ chỉ đi qua các cạnh có trọng số dương. Đồ thị W^2 dùng để xác định số lượng các đường đi có độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$ chỉ đi qua các cạnh có trọng số âm. Trong đó, giá trị L được xác định thông qua thử nghiệm. Trong bài báo này, chúng thử nghiệm và lấy $L=7$ đã cho kết quả tốt nhất.

Tại bước 2 của thuật toán, chúng ta tiến hành tìm số lượng đường đi có độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$ trên đồ thị W. Kết quả nhận được là ma trận W^L ghi nhận số lượng tất cả các loại đường đi có độ dài L từ đỉnh $i \in U$ đến đỉnh $x \in P$. Tiếp đến ta tìm được số lượng đường đi độ dài L đi qua các cạnh có trọng số dương trong ma trận $(W^1)^L$ và số lượng đường đi độ dài L đi qua các cạnh có trọng số dương trong ma trận $(W^2)^L$. Cuối cùng ta xác định được ma trận $(W^4)^L$ ghi lại số lượng đường đi độ dài L đi qua các cạnh có trọng số cả âm cả dương.

Tại bước 3 của thuật toán chúng ta tiến hành điền các giá trị dự đoán r_{ix} theo công thức (20). Trong đó, một số nhãn lúc đầu có r_{ix} = 0 được thay thế bằng giá trị 1, một số

khác được thay thế bằng giá trị -1, phần còn lại có giá trị 0 tương ứng với việc ta chưa thể đưa ra dự đoán. Thuật toán chi tiết được thể hiện trong Hình 6.

Thuật toán GraphBased:

Đầu vào:

- Ma trận đánh giá *R* được xác định theo công thức (17).

Đầu ra:

- Danh sách *k* bài toán mới $x \in P$ phù hợp nhất đối với người dùng $i \in U$.

Các bước tiến hành:

Bước 1. Xây dựng các đồ thị hai phía từ ma trận đánh giá *R*:

1.1. Xây dựng ma trận kề biểu diễn đồ thị hai phía trên toàn bộ *R* theo công thức (18):

$$W = \begin{cases} 1 & nếu\ r_{ix} \neq 0 \\ 0 & otherwise \end{cases}$$

1.2. Xây dựng ma trận kề biểu diễn đồ thị hai phía cho các đánh giá có giá trị 1 theo công thức (19):

$$W^1 = \begin{cases} 1 & nếu\ r_{ix} > 0 \\ 0 & otherwise \end{cases}$$

1.3. Xây dựng ma trận kề biểu diễn đồ thị hai phía cho các đánh giá có giá trị -1 theo công thức (20):

$$W^2 = \begin{cases} 1 & nếu\ r_{ix} < 0 \\ 0 & otherwise \end{cases}$$

Bước 2. Tìm số lượng đường đi độ dài *L* của mỗi loại:

2.1. Tìm số lượng đường đi độ dài *L* từ đỉnh $i \in U$ đến đỉnh $x \in P$ trên toàn bộ *R* theo công thức (21):

$$W^L = \begin{cases} W & nếu\ \ L = 1 \\ W.W^T.W^{L-2} & nếu\ L = \ 3, 5, \ldots \end{cases}$$

2.2. Tìm số lượng đường đi loại 1 có độ dài L từ đỉnh i∈U đến đỉnh x∈P theo công thức (22):

$$(W^1)^L = \begin{cases} W^1 & nếu\ \ L = 1 \\ W^1.(W^1)^T.(W^1)^{L-2} & nếu\ L = \ 3, 5, \ldots \end{cases}$$

2.3. Tìm số lượng đường đi loại 2 có độ dài L từ đỉnh i∈U đến đỉnh x∈P theo công thức (23):

$$(W^2)^L = \begin{cases} W^2 & nếu\ \ L = 1 \\ W^2.(W^2)^T.(W^2)^{L-2} & nếu\ L = \ 3, 5, \ldots \end{cases}$$

2.4. Tìm số lượng đường đi loại 3 có độ dài L từ đỉnh i∈U đến đỉnh x∈P theo công thức (24):

$$(W^3)^L = \ W^L - (W^1)^L - (W^2)^L$$

Bước 3. Dự đoán quan điểm của $i \in U$ đối với các bài toán mới $x \in P$:

3.1. Tìm số lượng đường đi độ dài *L* từ đỉnh $i \in U$ đến đỉnh $x \in P$ trên toàn bộ *R* theo công thức (25):

$$MAX = max\{w_{ix}^L : i = 1, 2, , ., n; x = 1, 2, , ., m\}$$

3.2. Sinh ra dự đoán quan điểm $i \in U$ đối với các bài toán mới $x \in P$ theo công thức (26):

$$r_{ix} = \begin{cases} 1 & nếu\ \frac{(w^1)_{ix}^L}{MAX} > 0.5 \\ 0 & nếu\ \frac{(w^3)_{ix}^L}{\text{MAX}} > 0.5 \\ -1 & nếu\ \frac{(w^2)_{ix}^L}{MAX} > 0.5 \end{cases}$$

Bước 4. Tạo nên tư vấn cho người dùng $i \in U$ các bài toán mới $x \in P$**:**

4.1. Sắp xếp r_{ix} theo thứ tự tăng dần của trọng số.

4.2. Chọn k bài toán mới đầu tiên có $r_{ix}=1$ để tư vấn cho người dùng i.

Hình 2: *Thuật toán GraphBased.*

Ví dụ: Xét ma trận trọng số sau:

	U_1	U_2	U_3	U_4	U_5	U_6
P_1	1	0	-1	-1	0	1
P_2	0	1	1	-1	1	0
P_3	-1	-1	1	0	0	1
P_4	1	1	-1	0	-1	-1
P_5	0	-1	0	1	1	0
P_6	-1	0	0	-1	1	1

Bảng 4: *Ma trận đánh giá R*

Bước 1: Xây dựng các đồ thị hai phía

- Ma trận kề biểu diễn đồ thị hai phía trên toàn bộ R:

$$\begin{vmatrix} 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 \end{vmatrix}$$

- Ma trận kề biểu diễn cho các đánh giá có giá trị 1:

$$\begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{vmatrix}$$

- Ma trận kề biểu diễn cho các đánh giá có giá trị -1:

$$\begin{vmatrix} 0 & 0 & 1 & 1 & 0 & 0 \end{vmatrix}$$

$$\begin{vmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \end{vmatrix}$$

Bước 2: Xác định số lượng đường đi độ dài L của mỗi loại, ở đây chúng ta xét L = 5.

- Tìm số lượng đường di độ dài L = 5 trên toàn bộ R:

$$W^5 = W.W^T.W.W^T.W$$

$$= \begin{vmatrix} 196 & 171 & 191 & 161 & 169 & 196 \\ 175 & 175 & 180 & 162 & 175 & 075 \\ 199 & 178 & 197 & 159 & 171 & 199 \\ 241 & 222 & 240 & 201 & 217 & 241 \\ 125 & 129 & 129 & 122 & 132 & 125 \\ 188 & 169 & 183 & 163 & 172 & 188 \end{vmatrix}$$

- Tìm số lượng đường đi độ dài L = 5 có giá trị 1:

$$(\boldsymbol{W^1})^5 = \boldsymbol{W^1}.\boldsymbol{W^{1T}}.\boldsymbol{W^1}.\boldsymbol{W^{1T}}.\boldsymbol{W^1}$$

$$= \begin{vmatrix} 11 & 7 & 8 & 1 & 9 & 17 \\ 8 & 18 & 19 & 6 & 26 & 16 \\ 7 & 8 & 13 & 2 & 14 & 18 \\ 10 & 11 & 7 & 1 & 8 & 8 \\ 2 & 7 & 8 & 6 & 17 & 8 \\ 8 & 9 & 13 & 5 & 20 & 19 \end{vmatrix}$$

- Tìm số lượng đường đi độ dài L = 5 có giá trị 1:

$$(\boldsymbol{W^2})^5 = \boldsymbol{W^2}.\boldsymbol{W^{2T}}.\boldsymbol{W^2}.\boldsymbol{W^{2T}}.\boldsymbol{W^2}$$

$$= \begin{vmatrix} 6 & 1 & 12 & 16 & 5 & 5 \\ 5 & 1 & 5 & 11 & 1 & 1 \\ 10 & 9 & 1 & 6 & 0 & 0 \\ 1 & 0 & 15 & 7 & 10 & 10 \\ 4 & 5 & 0 & 1 & 0 & 0 \\ 11 & 5 & 6 & 16 & 1 & 1 \end{vmatrix}$$

- Tìm số lượng đường đi loại 3 độ dài L = 5:

$$(\boldsymbol{W^3})^5 = \boldsymbol{W^5} - (\boldsymbol{W^1})^5 - (\boldsymbol{W^2})^5$$

$$= \begin{vmatrix} 179 & 163 & 171 & 144 & 155 & 174 \\ 162 & 156 & 156 & 145 & 148 & 158 \\ 182 & 161 & 183 & 151 & 157 & 181 \\ 230 & 211 & 218 & 193 & 199 & 223 \\ 119 & 117 & 121 & 115 & 115 & 117 \\ 170 & 155 & 164 & 142 & 151 & 168 \end{vmatrix}$$

Bước 3: Sinh dự đoán

Ví dụ ta sinh dự đoán của U_2 lên bài toán P_1

$$\boldsymbol{r_{1,2} = 0} \text{ vì } \frac{\boldsymbol{W^3}_{1,2}}{\boldsymbol{Max}} = \frac{\boldsymbol{163}}{\boldsymbol{171}} > \boldsymbol{0.5}$$

Điều đó có nghĩa là chưa thể dự đoán được U_2 có thể làm được bài toán P_1 hay không.

2.3. Kết luận

Chương này tôi đã đề xuất 3 phương pháp cho cổng lập trinh trực tuyến, ở hai phương pháp *User-based* và *Item-based,* đây là hai phương pháp phổ biến và chúng khắc phục những nhược điểm của nhau, ở phương pháp *User-based* sẽ không có tính hiệu quả, thực tế cho thấy nếu cổng lập trình trực tuyến có một lượng người dùng lớn hơn rất nhiều so với số lượng bài toán thì lúc này hệ thống sẽ gặp rất nhiều khó khăn trong khâu tính toán ma trận tương tự giữa các cặp người dùng vì lúc này ma trận tương quan là rất lớn còn về phương pháp *Item-based* cho thấy hiệu quả hơn ở phương diện người dùng và bài toán bởi trên thực tế số lượng người dùng đa phần sẽ lớn hơn rất nhiều so với bài toán cũng chính vì điều này sẽ làm ma trận đánh giá của *Item-based* sẽ

được đầy đủ và có được những đánh giá chất lượng hơn từ nhiều người dùng không những thế với số lượng bài toán ít hơn rất nhiều người dùng sẽ giúp hệ thống tính toán và đưa ra gợi ý nhanh hơn so với phương pháp *User-based.* Tuy vậy cả hai phương pháp lại có nhược điểm là nếu dữ liệu thưa thớt sẽ rất khó để đưa ra phán đoán chính xác bởi dữ liệu thưa thớt sẽ ảnh hưởng rất lớn đến khâu tính toán độ tương quan như tôi đã trình bày hai phương pháp khi sử dụng công thức tương quan cosin, để tăng độ chính xác phải cần một lượng dữ liệu lớn điều này rất khó với hệ thống mới chạy giai đoạn đầu chính vì thế mà luận văn đã đề xuất phương pháp lọc cộng tác dựa trên mô hình đồ thị, phương pháp này mạnh mẽ ở điểm có thể đưa ra dự đoán ngay khi dữ liệu không đủ lớn bằng cách tính toán số đường theo các loại mà bài toán đưa ra từ số lượng đường đi tính được ta có thể phán đoán quan điểm của người dùng và bài toán theo tỷ lệ số loại đường đi mà cặp người dùng và bài toán có thể đi được. Ở chương sau, luận văn sẽ thực nghiệm trên cổng lập trình trực tuyến của Học Viện Công Nghệ Bưu Chính Viễn Thông để thấy được sự ưu việt của phương pháp lọc cộng tác theo mô hình đồ thị hai phía so với hai phương pháp còn lại.

CHƯƠNG 3: THỰC NGHIỆM VÀ ĐÁNH GIÁ

3.1. Phương pháp thực nghiệm

Thuật toán dựa trên đồ thị (GraphBased) đề xuất được tiến hành thử nghiệm trên tập dữ liệu được thu thập từ cổng lập trình trực tuyến Dlab của Học viện Công nghệ Bưu chính Viễn thông. Phương pháp xây dựng bộ dữ liệu và kết quả thử nghiệm được đánh giá và so sánh với các phương pháp khác theo thủ tục được mô tả như dưới đây.

Trước tiên, toàn bộ tập người dùng được chia thành hai phần, một phần U_{tr} được sử dụng làm dữ liệu huấn luyện, phần còn lại U_{te} được sử dụng để kiểm tra. Tập dữ liệu huấn luyện dùng để xây dựng mô hình theo các thuật toán lọc được sử dụng. Với mỗi người dùng $u \in U_{te}$, các đánh giá $r_{u,p} \neq \varnothing$ được chia thành hai phần O_u và P_u. O_u được coi là đã biết, trong khi đó P_u là đánh giá cần dự đoán từ dữ liệu huấn luyện và O_u. Giả sử phương pháp lọc đưa ra dự đoán cho người dùng trong tập P_u là P'_u, Khi đó, sai số dự đoán được thực hiện bằng cách so sánh các đánh giá trong hai tập P_u là P'_u,

Có nhiều phương pháp đánh giá sai số phân loại khác nhau đã được đề xuất. Phương pháp phổ biến nhất được sử dụng trong lọc nội dung và lọc kết hợp là đánh giá sai số phân loại thông qua độ chính xác, độ nhạy. Phương pháp đánh giá sai số phân loại thông dụng nhất trong lọc cộng tác là trung bình giá trị tuyệt đối lỗi.

Độ đo trung bình giá trị tuyệt đối lỗi

Đánh giá sai số phân loại trung bình giá trị tuyệt đối lỗi (MAE) được Breese đề xuất năm 1998 được xem là phương pháp tiêu chuẩn cho lọc cộng tác. Sai số dự đoán MAE_u cho mỗi người dùng u thuộc tập dữ liệu kiểm tra được tính bằng trung bình giá trị tuyệt đối giữa hiệu số giá trị dự đoán và giá trị thực đối với tất cả mặt hàng thuộc tập P_u.

$$MAE_u = \frac{1}{|P_u|} \sum_{y \in P_u} \left| \hat{r}_y^{\,u} - r_y^{\,u} \right| \qquad (27)$$

Sai số dự đoán trên toàn tập dữ liệu kiểm tra được tính bằng trung bình cộng sai số dự đoán cho mỗi khách hàng thuộc U_{te} được tính toán theo công thức (27). Giá trị MAE càng nhỏ, phương pháp dự đoán càng chính xác.

$$MAE = \frac{\sum_{u \in U_{te}} MAE_u}{|U_{te}|} \qquad (28)$$

3.2. Dữ liệu thực nghiệm

3.2.1. Dữ liệu đầu vào

Phương pháp tư vấn cộng tác do nhóm nghiên thu thập trực tiếp từ cổng lập trình trực tuyến của Học viện Công nghệ Bưu chính Viễn thông, Tập dữ liệu thu thập từ tháng 8/2020 đến tháng 6 năm 2021 được 6435 người dùng đăng ký tham gia cổng lập trình trực tuyến. Kho nội dung số được xây dựng trong vòng 1 năm với 1245 bài toán để người học có thể lập trình và chấm bài tự động. Tổng số lượt giải bài (submission) của 6435 người dùng ghi nhận trong cổng lập trình trực tuyến đến ngày 20/6/2021 là 1,151,000. Trong đó, người dùng *i* lập trình đúng bài toán *x* cổng lập trình ghi lại giá trị 1, người dùng *i* lập trình chưa đúng bài toán *x* cổng lập trình ghi lại giá trị -1, người dùng *i* chưa giải bài toán *x* cổng lập trình ghi lại giá trị 0. Mỗi người lập trình có thể submit một bài nhiều lần và hệ thống chỉ ghi nhận giá trị 1, -1 cho kết quả cuối cùng. Trong số 6435 người dùng, luận văn lọc ra được 6120 người dùng đã tham gia lập trình ít nhất 20 bài dù đúng hoặc sai để tiến hành thử nghiệm. Dữ liệu đầu vào được minh họa ở hình 3.

1	1	1
1	2	1
1	3	1
1	9	1
1	10	1
1	11	1
1	17	1
1	20	1
1	21	1
1	27	1
1	33	1
1	34	1
1	45	-1
2	1	1
2	2	1
2	3	1
2	8	-1
2	9	-1
2	11	1
2	12	-1
2	17	1
2	21	1
2	33	1
2	34	-1
2	45	-1
2	67	1

Hình 3: *Ảnh minh họa dữ liệu đầu vào*

Ở hình 3, cột đầu tiên là thể hiện id của người dùng, cột thứ 2 là id của bài tập, cột thứ 3 là trạng thái giải bài của người dùng đối với bài tập.

3.2.2. Xử lý dữ liệu

Toàn bộ tập người dùng được chia thành hai phần:

- $\boldsymbol{U_{tr}}$ được sử dụng làm dữ liệu huấn luyện.
- $\boldsymbol{U_{te}}$ được sử dụng để kiểm tra.

Chọn ngẫu nhiên 2000, 3000, và 4000 trong tập 6120 người dùng làm dữ liệu huấn luyện. Chọn ngẫu nhiên 400, 600, và 800 người trong số còn lại làm tập dữ liệu kiểm tra. Để thử nghiệm khả năng của phương pháp mới đề xuất so với những phương pháp khác trong trường hợp có ít dữ liệu, chúng tôi thay đổi số lượng bài lập trình của mỗi người dùng trong tập kiểm tra sao cho số lượng bài đã lập trình lần lượt là 5, 10 và 20, phần còn lại là những đánh giá cần dự đoán.

Dữ liệu đầu vào được chuyển về dạng ma trận với mỗi cột ứng với một bài toán, mỗi hàng ứng với một người dùng. Với mỗi phương pháp sẽ ứng với một ma trận đầu vào, nhưng với phương pháp Gragh-based ma trận đầu vào sẽ được biến đổi thành 3 ma trận ứng với mỗi loại đường đi.

3.3. Kết quả thực nghiệm

Phương pháp GraphBased đề xuất trong Mục 2.2 được thử nghiệm và so sánh với những phương pháp sau:

- Phương pháp Userbased sử dụng độ tương quan Person, đây là phương pháp lọc cộng tác dựa trên người dùng được trình bày ở mục 1.3.1
- Phương pháp Itembased sử dụng độ tương quan Person, đây là phương pháp lọc cộng tác dựa trên bài toán được trình bày ở mục 1.3.2

Giá trị MAE trong Bảng 10 được ước lượng từ trung bình của 10 lần thử nghiệm ngẫu nhiên. Kết quả thử nghiệm cho thấy phương pháp đề xuất đều cho giá trị MAE nhỏ hơn phương pháp UseBassed và ItemBased trong tất cả các trường hợp dữ liệu biết trước là 5, 10 hay 20 đánh giá. Cụ thể, trong trường hợp dữ liệu rất thưa với số lượng đánh giá biết trước trong tập dữ liệu kiểm tra là 5 thì giá trị MAE của phương pháp Usebased, ItemBased, GraphBased lần lượt là 0.2158, 0.2172, 0.208 trên tập dữ liệu huấn luyện 2000 người dùng. Giá trị MAE của các phương pháp Userbased, ItemBased, GraphBased có xu hướng nhỏ dần khi kích cỡ tập dữ liệu huấn luyện tăng lên 3000 và 4000 người dùng. Tuy nhiên, phương pháp GraphBased vẫn có giá trị MAE nhỏ hơn

các phương pháp còn lại. Điều này chứng tỏ phương pháp đề xuất phát huy được hiệu quả ngay cả trường hợp dữ liệu thưa của lọc cộng tác.

Trong trường hợp có tương đối đầy đủ dữ liệu, cụ thể với số lượng đánh giá biết trước là 20 thì giá trị MAE của phương pháp GraphBased có giá trị nhỏ hơn hẳn các phương pháp còn lại. Giá trị MAE của phương pháp GraphBased là 0.1812, 0.1792, 0.1712, hai phương pháp Userbased-Graph, Itembased-Graph cũng cho 1 kết quả tương đối tốt không kém Graph-Based với MAE trong khoảng [0.17,0.19] trong khi đó các phương pháp UserBased và ItemBased đều cho kết quả MAE >1.820. Điều này chỉ có thể lý giải các phương pháp tư vấn dựa trên độ tương quan thực hiện ước lượng mức độ giữa các cặp người dùng hoặc bài toán trực tiếp trên tập giao các đánh giá về bài toán hay người dùng. Tuy nhiên, với phương pháp dựa vào đồ thị, chúng ta có thể suy diễn mức độ trên tập các đường đi có trọng số dương và tập các đường đi có trọng số âm đồng thời không đưa ra dự đoán với các đường đi có cả trọng số âm lẫn dương. Điều này cho phép ta tận dụng được các mối liên hệ gián tiếp vào kết quả dự đoán.

Mặc dù kết quả của các phương pháp Graph cho ra 1 kết quả tốt hơn hẳn những phương pháp thông thường nhưng lại gặp 1 vấn đề về sự liên thông giữa các đỉnh trong các đồ thị tức sẽ có 1 cặp đỉnh không có cạnh nối và hoàn toàn bị tách ra khỏi đồ thị nếu chúng ta có 1 tập người dùng và bài toán khổng lồ nhưng dữ liệu lại cực kì thưa thớt điều này sẽ dẫn đến khi ta tách mô hình đồ thị tổng quát thành các đồ thị con thể hiện các đường âm và dương sẽ dẫn đến đồ thị này sẽ không thể liên thông, khi đồ thị không liên thông sẽ ảnh hưởng rất cao đến kết quả khi ta tăng độ dài $L = 5,7,9 \ldots$

Để khắc phục nhược điểm này tôi đã tăng cường lọc dữ liệu cho tập training, có thể thấy rằng vấn đề đồ thị không liên thông là do sự thưa thớt dữ liệu giữa các đỉnh với nhau chính vì vậy mà tôi đã chọn ra 6000 người dùng đã làm ít nhất 20 bài toán và mỗi bài toán phải tối thiểu có 20 người đã làm điều này không thể chắc chắn sẽ cho chúng ta một đồ thị liên thông nhưng nó có thể hạn chế bớt sự không liên thông giữa các đỉnh trong đồ thị. Có thể tăng dữ liệu lên 30, 50 để khắc phục vấn đề này. Một hướng khác là chúng ta có thể thấy tôi đã thử nghiệm mỗi phương pháp 10 lần để lấy ra kết quả trung bình, điều này cũng nhằm hạn chế trong 1 vài trường hợp có thể xuất hiện đồ thị không liên thông nên lấy trung bình kết quả cũng là 1 cách có thể giảm thiểu sự sai số khi gặp đồ thị không liên thông. Để mà có thể chắc chắn khi tách thành các đồ thị con thể hiện đường đi âm và dương là 1 đồ thị liên thông thì yếu tố dữ liệu rất quan trọng, với 2

phương pháp trên chỉ 1 phần nào hạn chế được sự không liên thông xảy ra chứ không hoàn toàn khắc phục triệt để vấn đề này.

Kích thước tập dữ liệu huấn luyện	Phương pháp	Số lượng đánh giá biết trước		
		5	10	20
2000 người dùng	UserBased	0.2158	0.2117	0.2042
	ItemBased	0.2172	0.2146	0.1992
	GraphBased	**0.2068**	**0.1984**	**0.1872**
3000 người dùng	UserBased	0.2147	0.2012	0.1924
	ItemBased	0.2145	0.2116	0.1967
	GraphBased	**0.2043**	**0.1877**	**0.1792**
4000 người dùng	UserBased	0.2037	0.2117	0.1942
	ItemBased	0.2012	0.2146	0.1820
	GraphBased	**0.1987**	**0.1784**	**0.1712**

Bảng 5*: Giá trị MAE của các phương pháp*

3.4. Phân tích và đánh giá thực nghiệm

Phương pháp User-based sẽ không có tính hiệu quả, thực tế cho thấy nếu cổng lập trình trực tuyến có một lượng người dùng lớn hơn rất nhiều so với số lượng bài toán thì lúc này hệ thống sẽ gặp rất nhiều khó khăn trong khâu tính toán ma trận tương tự giữa các cặp người dùng vì lúc này ma trận tương quan là rất lớn.

Còn về phương pháp Item-based cho thấy hiệu quả hơn ở phương diện người dùng và bài toán bởi trên thực tế số lượng người dùng đa phần sẽ lớn hơn rất nhiều so với số lượng bài toán. Cũng chính vì điều này sẽ làm ma trận đánh giá của Item-based sẽ được đầy đủ và có được những đánh giá chất lượng hơn từ nhiều người dùng. Không những thế với số lượng bài toán ít hơn người dùng sẽ giúp hệ thống đưa ra gợi ý nhanh hơn so với phương pháp User-based.

Cả hai phương pháp vừa nêu đã rất thành công trong việc cố gắng nắm bắt những điểm tương đồng của hai người dùng hay hai bài. Tuy nhiên, chúng ta có thể thấy rằng mô hình này lại có nhược điểm là nếu dữ liệu thưa thớt sẽ rất khó để đưa ra phán đoán chính xác bởi dữ liệu thưa thớt sẽ ảnh hưởng rất lớn đến khâu tính toán độ tương quan như đã trình bày. Hai phương pháp khi sử dụng công thức tương quan cosin, để tăng độ chính xác phải cần một lượng dữ liệu lớn điều này rất khó với hệ thống mới chạy giai đoạn đầu .Từ kết quả, chúng ta có thể thấy rằng mô hình Gragh-based đã thành công

trong việc nắm bắt những điểm tương đồng giữa hai User tốt hơn User-based và Item-based. Gragh-based không dễ bị ảnh hưởng bởi các hiệu ứng phân cụm, không bị giới hạn trong việc chỉ đề xuất lẫn nhau là người dùng giống nhau nhất và không thể khám phá để xem xét các mục khác. Do đó, chúng ta có thể kết luận rằng đối với tập dữ liệu có ma trận xếp hạng thưa thớt, mô hình Gragh-based có thể hoạt động tốt hơn các mô hình truyền thống.

3.5. Kết luận

Nghiên cứu đã đề xuất một phương pháp tư vấn lọc cộng tác trên cổng lập trình trực tuyến của Học viện Công nghệ Bưu chính Viễn thông nhằm cung cấp cho tập bài toán phù hợp với khả năng lập trình cho mỗi người dùng. Mô hình đề xuất phát huy hiệu quả ngay cả trong trường hợp dữ liệu thưa mà các mô hình tư vấn lọc cộng tác khác thường gặp phải khó khăn. Phương pháp tư vấn cộng tác đề xuất đã đem lại hiệu quả thiết thực cho người học lập trình trực tuyến của Học viện Công nghệ Bưu chính Viễn thông.

TÀI LIỆU THAM KHẢO

[1] Wasik, S.; Antczak, M.; Badura, J.; Laskowski, A.; Sternal, T. *A Survey on Online Judge Systems and Their Applications*. ACM Comput. Surv.(CSUR) 2018, 51, 1–34.

[2] Liu, J.; Zhang, S; Yang, Z.; Zhang, Z.; Wang, J.; Xing, X. *Online Judge System Topic Classification*. In Proceedings of the 2018 14th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Huangshan, China, 28–30 July 2018; pp. 993–1000.

[3] Paolo, A.; C, Estler.; Durica, N.; Marco, P.; Bertrand, M (2015). *An Incremental Hint System ´ For Automated Programming Assignments*. In Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education. ACM, 320–325.

[4] Joeran, B.; Bela, G.; Stefan, L.; Corinna, B. *Research-paper recommender systems: a literature survey*. International Journal on Digital Libraries; 17 (2016), 4. - S. 305-338.

[5] Nirav, R.; Vijayshri, K. *A Review Paper On Collaborative Filtering Based Moive Recommedation System*. International Jounal of Sciencetific & Technology Research, Volume 8, 2019.

[6] Cui, Bei-Bei.(2017). *Design and Implementation of Movie Recommendation System Based on Knn Collaborative Filtering Algorithm*. ITM Web of Conferences. 12. 04008. 10.1051/itmconf/20171204008.

[7] Zhao, Zhi-Dan & Shang, Ming Sheng (2010). *UserBased Collaborative-Filtering Recommendation Algorithms on Hadoop*. 3rd International Conference on Knowledge Discovery and Data Mining, WKDD 2010. 478-481.

[8] Kharita, M. K., Kumar, A., & Singh, P.(2018). *ItemBased Collaborative Filtering in Movie Recommendation in Real-time*. 2018 First International Conference on Secure Cyber Computing and Communication(ICSCCC).

[9] Thakkar, Priyank & Varma(Thakkar), Krunal & Ukani, Vijay & Mankad, Sapan &Tanwar, Sudeep.(2019). *Combining User-Based and Item-Based Collaborative Filtering Using Machine Learning*: Proceedings of ICTIS 2018, Volume 2. 978-981

[10] Tu Minh Phuong, Do Thi Lien, Nguyen Duy Phuong (2019). *Graph-based Context-aware Collaborative Filtering.* Expert System and Application, Vol: 126, pp: 9-19.

[11] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Trans. on Knowl. and Data Eng., 17(6):734–749, 2005

[12] Marko Balabanovic and Yoav Shoham. Fab: Content-based, collaborative recommendation. Communications of the Association for Computing Machinery, 40(3):66–72, 1997.

[13] Abhay S. Harpale and Yiming Yang. Personalized active learning for collaborative filtering. In SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pages 91–98, New York, NY, USA, 2008. ACM

[14] Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing, 7(1):76–80, 2003

[15] Rong Pan and Martin Scholz. Mind the gaps: Weighting the unknown in large-scale one-class collaborative filtering. In 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), 2009

[16] Alexandrin Popescul, Lyle Ungar, David M. Pennock, and Steve Lawrence. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In Proceedings of the Seventeenth Conference on Uncertainity in Artificial Intelligence, 2001

[17] Nguyễn Mạnh Sơn, Nguyễn Duy Phương: "MỘT PHƯƠNG PHÁP TƯ VẤN CỘNG TÁC CHO CÁC CỔNG LẬP TRÌNH TRỰC TUYẾN". Kỷ yếu Hội nghị KHCN Quốc gia lần thứ XIV về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR), TP. HCM, ngày 23-24/12/2021 DOI: 10.15625/vap.2021.0039

Danh mục các Website tham khảo:

[1]. Information-filtering http://dsv.su.se/jpalme/select/information-filtering.pdf

[2]. AWS Machine Learning Blog (amazon.com)

[3]. The TensorFlow Blog

[4]. https://www.code.ptit.edu.vn.