

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM



NGUYỄN LÊ DUY

**KỸ THUẬT MATRIX FACTORIZATION TRONG
XÂY DỰNG HỆ TƯ VẤN**

LUẬN VĂN THẠC SĨ

Chuyên ngành : Công nghệ thông tin

Mã số ngành: 60480201

TP. HỒ CHÍ MINH, tháng 10 năm 2015

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM



NGUYỄN LÊ DUY

**KỸ THUẬT MATRIX FACTORIZATION TRONG
XÂY DỰNG HỆ TƯ VẤN**

LUẬN VĂN THẠC SĨ

Chuyên ngành : Công nghệ thông tin

Mã số ngành: 60480201

CÁN BỘ HƯỚNG DẪN KHOA HỌC: PGS. TS. QUẢN THÀNH THƠ

TP. HỒ CHÍ MINH, tháng 10 năm 2015

**CÔNG TRÌNH ĐƯỢC HOÀN THÀNH TẠI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM**

Cán bộ hướng dẫn khoa học : PGS. TS. Quản Thành Thơ

Luận văn Thạc sĩ được bảo vệ tại Trường Đại học Công nghệ TP. HCM
ngày 17 tháng 10 năm 2015

Thành phần Hội đồng đánh giá Luận văn Thạc sĩ gồm:

TT	Họ và tên	Chức danh Hội đồng
1	PGS. TSKH. Nguyễn Xuân Huy	Chủ tịch
2	TS. Trần Đức Khánh	Phản biện 1
3	TS. Lư Nhật Vinh	Phản biện 2
4	TS. Võ Đình Bảy	Ủy viên
5	TS. Nguyễn Thị Thúy Loan	Ủy viên, Thư ký

Xác nhận của Chủ tịch Hội đồng đánh giá Luận văn sau khi Luận văn đã được
sửa chữa (nếu có).

Chủ tịch Hội đồng đánh giá LV

TRƯỜNG ĐH CÔNG NGHỆ TP. HCM
PHÒNG QLKH – ĐTSĐH

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

TP. HCM, ngày 15 tháng 09 năm 2015

NHIỆM VỤ LUẬN VĂN THẠC SĨ

Họ tên học viên: NGUYỄN LÊ DUY

Giới tính: NAM

Ngày, tháng, năm sinh: 27/11/1985

Nơi sinh: TP HCM

Chuyên ngành: Công nghệ thông tin

MSHV: 1341860036

I- Tên đề tài:

KỸ THUẬT MATRIX FACTORIZATION TRONG XÂY DỰNG HỆ TƯ VẤN.

II- Nhiệm vụ và nội dung:

Nghiên cứu kỹ thuật Matrix Factorization và ứng dụng xây dựng hệ tư vấn phim.

III- Ngày giao nhiệm vụ: 03/04/2015

IV- Ngày hoàn thành nhiệm vụ: 15/09/2015

V- Cán bộ hướng dẫn: PGS. TS. QUẢN THÀNH THƠ

CÁN BỘ HƯỚNG DẪN

KHOA QUẢN LÝ CHUYÊN NGÀNH

PGS. TS. QUẢN THÀNH THƠ

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi. Các số liệu, kết quả nêu trong Luận văn là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Tôi xin cam đoan rằng mọi sự giúp đỡ cho việc thực hiện Luận văn này đã được cảm ơn và các thông tin trích dẫn trong Luận văn đã được chỉ rõ nguồn gốc.

Học viên thực hiện Luận văn

Nguyễn Lê Duy

LỜI CẢM ƠN

Luận văn được thực hiện tại Khoa Công nghệ thông tin - Trường Đại học Công nghệ TP HCM, dưới sự hướng dẫn khoa học của PGS. TS. Quản Thành Thơ.

Trước tiên tôi xin bày tỏ lòng biết ơn sâu sắc tới PGS. TS. Quản Thành Thơ đã đưa tôi đến với lĩnh vực nghiên cứu này. Thầy đã tận tình giảng dạy, hướng dẫn giúp tôi tiếp cận và đạt được thành công trong công việc nghiên cứu của mình. Thầy đã luôn tận tâm động viên, khuyến khích và chỉ dẫn giúp tôi hoàn thành được luận văn này.

Tôi xin bày tỏ lòng biết ơn tới các Thầy Cô thuộc Khoa Công nghệ thông tin và cán bộ phòng Quản lý khoa học và đào tạo sau đại học - Trường Đại học Công nghệ TP HCM, đã tạo mọi điều kiện thuận lợi giúp đỡ tôi trong quá trình học tập và nghiên cứu tại trường.

Tôi xin chân thành cảm ơn các Thầy Cô trong Hội đồng đánh giá luận văn Thạc Sĩ đã đóng góp ý kiến quý báu giúp tôi hoàn thiện bản luận án.

Học viên thực hiện Luận văn

Nguyễn Lê Duy

TÓM TẮT

Trong xu thế công nghệ thông tin phát triển, việc sử dụng mạng Internet đã trở thành một công việc không thể thiếu trong thời gian biểu hằng ngày của con người hiện đại. Có rất nhiều thông tin được đưa lên mạng, từ thông tin cá nhân, đánh giá, nhận xét một dịch vụ, sự kiện,...

Cuộc sống con người ngày càng bận rộn, chúng ta không thể đọc hết mọi thông tin trên Internet để rút ra được một quyết định cho riêng mình. Do đó hệ tư vấn đang trở thành xu hướng phát triển. Hệ tư vấn giúp con người tham khảo và ra quyết định dựa trên tập hợp ý kiến, số liệu từ những kinh nghiệm của người đi trước.

Thấy được sự cần thiết và lợi ích của hệ tư vấn cũng như muốn khai thác nguồn thông tin vô tận và vô giá của Internet, tác giả quyết định chọn đề tài “Kỹ thuật Matrix Factorization trong xây dựng hệ tư vấn”.

Bên cạnh việc nghiên cứu lý thuyết, tác giả còn lập trình một phần mềm mẫu áp dụng kỹ thuật trên để xây dựng một hệ tư vấn nhằm đánh giá và hỗ trợ người dùng chọn những bộ phim mình nên xem (vì thời gian của chúng ta là có hạn và nguồn phim thì vô tận).

ABSTRACT

Nowaday, with devepoment of the information technology, using the internet has been a dispensable part of a modern man's daily work. You can find many types of information on the Internet, such as private information of someone, some comments about a service, events, etc.

Of course we cant check every information on the internet to make an our own decision. Because of this matter, recommendation systems has been a global inevitable tendency. A recommendation systems gathers and presents data from a wide range of sources, thus helps us with refering and making a decision.

Because of benefit and necessary characteristics of recommendation systems, as well as hoping to take advantage of an unlimited and invaluable information source from the internet, the writer chose the thesis on "recommendation systems with Matrix Factorization technology".

Not only theoretical research, the writer also developed a demonstration software using a Matrix Factorization technology to develop a recommendation systems to evaluate and support users in choosing movies.

MỤC LỤC

LỜI CAM ĐOAN	i
LỜI CẢM ƠN	ii
TÓM TẮT	iii
ABSTRACT	iv
DANH MỤC TỪ VIẾT TẮT.....	vii
DANH MỤC CÁC BẢNG.....	viii
DANH MỤC HÌNH VẼ, ĐỒ THỊ	ix
CHƯƠNG 1 GIỚI THIỆU	1
1.1. LÝ DO CHỌN ĐỀ TÀI:	1
1.2. MỤC TIÊU CỦA ĐỀ TÀI:	1
1.3. CẤU TRÚC CỦA LUẬN VĂN:.....	1
CHƯƠNG 2 CÁC NGHIÊN CỨU LIÊN QUAN	3
2.1. TỔNG QUAN VỀ HỆ TƯ VẤN.....	3
2.1.1. GIỚI THIỆU HỆ TƯ VẤN	3
2.2. KIẾN TRÚC CƠ BẢN MỘT HỆ TƯ VẤN	5
2.3. MÔ TẢ BÀI TOÁN TƯ VẤN	6
2.4. PHÂN LOẠI HỆ TƯ VẤN	8
2.4.1. PHƯƠNG PHÁP DỰA TRÊN NỘI DUNG (CONTENT-BASED)	8
2.4.2. PHƯƠNG PHÁP CHỌN LỌC CỘNG TÁC (COLLABORATIVE FILTERING)	13
2.4.3. PHƯƠNG PHÁP DỰA TRÊN NHÂN KHẨU HỌC (DEMOGRAPHIC- BASED)	18
2.4.4. PHƯƠNG PHÁP DỰA TRÊN TRI THỨC (KNOWLEDGE-BASED)..	20
2.4.5. PHƯƠNG PHÁP DỰA TRÊN CỘNG ĐỒNG (COMMUNITY-BASED)..	23

2.4.6. PHƯƠNG PHÁP LAI (HYBRID).....	25
2.4.7. VẤN ĐỀ RAMP-UP.....	30
CHƯƠNG 3 CƠ SỞ LÝ THUYẾT	32
3.1. MÔ HÌNH PHÂN RÃ MA TRẬN.....	32
3.2. CÁC THUẬT TOÁN HỌC (<i>Learning Algorithms</i>)	35
3.2.1. PHƯƠNG PHÁP GIẢM GRADIENT NGẪU NHIÊN (STOCHASTIC GRADIENT DESCENT).....	35
3.2.2. PHƯƠNG PHÁP ALTERNATING LEAST SQUARE (ASL)	37
3.3. HỆ SỐ BIAS.....	38
3.4. REGULARIZATION	39
3.5. PHÂN RÃ MA TRẬN KHÔNG ÂM (NMF).....	40
CHƯƠNG 4 HỆ THỐNG ĐỀ NGHỊ.....	42
4.1. MÔ HÌNH HỆ THỐNG:	42
4.2. GIẢI THUẬT CHƯƠNG TRÌNH	42
4.2.1. ĐẦU VÀO:	42
4.2.2. ĐẦU RA:.....	43
4.2.3. LƯU ĐỒ THUẬT TOÁN:.....	43
CHƯƠNG 5 HIỆN THỰC VÀ THÍ NGHIỆM	44
5.1. HIỆN THỰC	44
5.1.1. GIỚI THIỆU	44
5.2. MỘT SỐ THỬ NGHIỆM VÀ NHẬN XÉT VỀ THUẬT TOÁN	46
5.2.1. TẬP DỮ LIỆU: 10 Người dùng, 712 Hạng mục.....	47
5.2.2. TẬP DỮ LIỆU: 50 Người dùng, 1084 Hạng mục.....	51
5.2.3. TẬP DỮ LIỆU: 100 Người dùng, 1238 Hạng mục.....	54
CHƯƠNG 6 KẾT LUẬN.....	57
TÀI LIỆU THAM KHẢO.....	59

DANH MỤC TỪ VIẾT TẮT

Ký hiệu	Tiếng Anh	Tiếng Việt
NMF	Non-negative Matrix Factorization	Phân rã ma trận không âm
UI	User Interface	Giao diện người dùng
DB	Database	Cơ sở dữ liệu
MinErr	Min Error	Lỗi tối thiểu

DANH MỤC CÁC BẢNG

Bảng 2.1. Kết quả đánh giá của người xem phim với một số phim khảo sát	8
Bảng 2.2. Các kỹ thuật tư vấn	10
Bảng 2.3. Thông tin nhân khẩu học được sử dụng trong hệ tư vấn	19
Bảng 2.4. Các phương pháp lai	26
Bảng 3.1. Ma trận điểm đánh giá	33
Bảng 5.1. Thay đổi số lần học	48
Bảng 5.2. Thay đổi hệ số alpha	49
Bảng 5.3. Thay đổi hệ số beta	50
Bảng 5.4. Thay đổi số lần học	51
Bảng 5.5. Thay đổi hệ số alpha	52
Bảng 5.6. Thay đổi hệ số beta	53
Bảng 5.7. Thay đổi số lần học	54
Bảng 5.8. Thay đổi hệ số alpha	55
Bảng 5.9. Thay đổi hệ số beta	56

DANH MỤC HÌNH VẼ, ĐỒ THỊ

Hình 2.1. Ví dụ về một hệ tư vấn trên website thương mại	5
Hình 2.2. Kiến trúc cơ bản một hệ tư vấn	6
Hình 2.3. Ví dụ về một hệ tư vấn trên website thương mại	9
Hình 2.4. Hoạt động của hệ tư vấn chọn lọc hợp tác	13
Hình 2.5. Mô hình hoạt động của hệ tư vấn dựa trên hạng mục của Amazon.....	16
Hình 2.6. Mô hình hệ tư vấn dựa trên nhân chủng học.....	19
Hình 2.7. Vòng tròn suy luận của kỹ thuật dựa trên trường hợp.	22
Hình 2.8. Mô hình tư vấn dựa trên cộng đồng	24
Hình 2.9. Mô hình một hệ tư vấn lai	27
Hình 3.1. Mô hình Phân rã ma trận	34
Hình 3.2. Ảnh hưởng của bước nhảy trong quá trình tiệm tiến đến cực tiểu.....	36
Hình 4.1. Mô hình hệ tư vấn dựa trên kỹ thuật Phân rã ma trận.....	42
Hình 4.2. Lưu đồ thuật toán NMF.....	43
Hình 5.1. Giao diện chương trình thử nghiệm	46
Hình 5.2. Thay đổi số lần học.	48
Hình 5.3. Thay đổi hệ số alpha	49
Hình 5.4. Thay đổi hệ số beta	50
Hình 5.5. Thay đổi số lần học	51
Hình 5.6. Thay đổi hệ số alpha	52
Hình 5.7. Thay đổi hệ số beta	53
Hình 5.8. Thay đổi số lần học	54
Hình 5.9. Thay đổi hệ số alpha	55
Hình 5.10. Thay đổi hệ số beta	56

CHƯƠNG 1 GIỚI THIỆU

1.1. LÝ DO CHỌN ĐỀ TÀI:

Có thể nói rằng hiện nay có đủ mọi loại thông tin trên mạng internet; và với sự giúp sức của các công cụ tìm kiếm thông tin hiện hữu, người ta có thể dễ dàng tìm thấy những gì mà họ quan tâm. Mặt khác, do có quá nhiều thông tin, nên đã gây không ít khó khăn, lúng túng cho việc định hướng chọn lựa hoặc ra quyết định từ phía người sử dụng các thông tin trên. Các hệ tư vấn (recommendation systems) ra đời không nằm ngoài mục đích hỗ trợ cho người dùng trong các lựa chọn và/hoặc ra quyết định.

Có nhiều hướng tiếp cận để xây dựng một hệ tư vấn. Tùy thuộc vào nguồn thông tin có được, nhu cầu tư vấn thực tế, đặc thù riêng của dịch vụ tư vấn cần cung cấp, v.v... mà mỗi hệ tư vấn sẽ có phương pháp và thuật toán phù hợp cho riêng mình. Kỹ thuật phân rã ma trận (Matrix Factorization) là một trong số các kỹ thuật được sử dụng để xây dựng một hệ tư vấn dựa trên các dữ liệu đánh giá. Kỹ thuật này được đánh giá cao nhờ khả năng cải thiện độ chính xác của các thuật toán tư vấn khác, tính linh hoạt, thời gian thực thi thấp, v.v...

1.2. MỤC TIÊU CỦA ĐỀ TÀI:

- + Tìm hiểu các hệ tư vấn (recommendation systems).
- + Tìm hiểu các khái niệm, phương pháp xây dựng 1 hệ tư vấn.
- + Tìm hiểu kỹ thuật phân rã ma trận.
- + Áp dụng phương pháp phân rã ma trận để xây dựng 1 hệ tư vấn về phim.

1.3. CẤU TRÚC CỦA LUẬN VĂN:

- + **Chương 1:** Giới thiệu đề tài.
- + **Chương 2:** Các nghiên cứu liên quan.
- + **Chương 3:** Cơ sở lý thuyết.
- + **Chương 4:** Hệ thống đề nghị.

- + **Chương 5:** Hiện thực và thí nghiệm.
- + **Chương 6:** Tổng kết.

CHƯƠNG 2 CÁC NGHIÊN CỨU LIÊN QUAN

2.1. TỔNG QUAN VỀ HỆ TƯ VẤN

2.1.1. GIỚI THIỆU HỆ TƯ VẤN

Hệ tư vấn (*Recommender system*) được xem là một hệ thống chọn lọc thông tin nhằm tìm kiếm và dự báo các đánh giá (*rating*)/ưa chuộng (*preferences*) mà người sử dụng sẽ dành cho các hạng mục / sản phẩm (*items*) mà trước đó họ chưa quan tâm đến.

Hệ tư vấn được sử dụng nhiều trên các website thương mại của nhiều công ty kinh doanh lớn nhằm dự báo và giúp người sử dụng hướng đến các sản phẩm phù hợp với nhu cầu và sở thích của họ hơn, từ đó nâng cao uy tín, doanh thu và lợi nhuận của công ty. Có thể kể đến một vài hệ tư vấn của một số công ty nổi tiếng như:

- ✓ Hệ tư vấn trên Amazon.com: khi xem một sản phẩm trên website của hãng này, hệ thống sẽ đề nghị một danh sách các sản phẩm cộng thêm dựa trên ma trận sản phẩm mà những người mua hàng trước đó đã mua kèm với sản phẩm đang được xem.
- ✓ Netflix lại quan tâm đến việc dự báo những phim người tiêu dùng thích xem dựa trên kết quả bình chọn trước đó, thói quen xem phim và các đặc tính của phim (thể loại phim chẳng hạn).
- ✓ Last.fm cung cấp dịch vụ radio trên internet miễn phí. Hệ tư vấn của Last.fm dựa trên các băng tần mà người sử dụng đã nghe cũng như thói quen nghe đài của họ.
- ✓ Ngoài ra, còn có khá nhiều hệ tư vấn của nhiều công ty thuộc nhiều lĩnh vực kinh doanh khác nhau như: Yahoo, YouTube, MovieLens, Morse, Polylens, Gab, Fab, v.v...

Hình 2.1 minh họa một hệ tư vấn trên website thương mại www.cduniverse.com.

Đặc điểm của các tư vấn là mang tính chất cá nhân hóa nghĩa là nó chỉ phù hợp với một số người dùng (hay nhóm người dùng) có cùng một số đặc tính đã được khảo sát trước đó. Điều này cũng phù hợp với thực tế bởi lẽ không thể có được một lời khuyên chung nhất cho mọi đối tượng.

Các hệ tư vấn ngày càng đóng vai trò quan trọng trong việc thúc đẩy phát triển các giao dịch trên mạng. Vai trò đó được thể hiện qua một số chức năng cơ bản sau đây:

- ✓ Gia tăng doanh số bán hàng: nhờ tư vấn mà người kinh doanh có thể bán được một nhóm sản phẩm phù hợp với nhu cầu và thị hiếu của người dùng so với việc không tư vấn chỉ bán được các sản phẩm đơn lẻ.
- ✓ Gia tăng thỏa mãn khách hàng: khách hàng càng được thỏa mãn, họ càng gia tăng việc sử dụng các tiện ích khác của công ty (nếu có).
- ✓ Tăng độ trung thành của khách hàng.

Đối với người sử dụng, hệ tư vấn mang lại một số lợi ích sau đây:

- ✓ Giúp họ giới hạn phạm vi tìm kiếm trong số vô vàn các thông tin thực và ảo.
- ✓ Tự tin và quyết đoán hơn khi quyết định mua sản phẩm được tư vấn.
- ✓ Nắm bắt kịp thời các xu hướng sử dụng hiện hành.

Monty Python and the Holy Grail DVD
Special Edition; Widescreen
★★★★★ 15 Customer Reviews
Add to Cart
Our Price: \$12.99
Usually ships in 1-2 c

Other Versions
BLU-RAY \$16.15 BUY IT

This is a Region 1 DVD
It will not play in a DVD
Vietnam.

Category Comedies, Action/Adventure, Recommended, Essential Cir Satire, Spoof, On-The-Road, Witches And Wizards, Poverty Classic Fight Scenes

Starring Monty Python Cast, John Cleese, Eric Idle, Terry Gilliam, T Chapman, Carol Cleveland, Connie Booth, Bee Duffell, Jo Terry Gilliam

Director Terry Gilliam

Cinematographer Terry Bedford

Similar Movies
Amazon Women on the Moon, Excalibur, Fawlty Towers - Series Three, Fierce Cr World: Part 1, Jeeves and Wooster: The Complete Third Season, Kids in the Hall 5, Secret Policeman's Ball, Start the Revolution Without Me, Tristram Shandy: A

Customers Who Bought Monty Python and the Holy Grail Also Bought
Led Zeppelin
Led Zeppelin - DVD (2003) ★★★★★ Top Seller
DVD \$24.29 BUY IT DETAILS
Very few Led Zeppelin performances were caught on camera but the footage that has been preserved for posterity is colle running time of over five-and-a-half hours, this is the definit

Also Bought
Pirates of the Caribbean: The Curse of the Black Pearl
★★★★★ Top Seller
DVD \$12.99 BUY IT DETAILS Buena Vista Ho
DVD \$16.79 BUY IT DETAILS Walt Disney Ho
BLU-RAY \$16.55 BUY IT DETAILS
Loosely inspired by the time-honored Disney theme-park rid CARIBBEAN: THE CURSE OF THE BLACK PEARL is a swashbuck by Gore Verbinski. When Captain Jack Sparrow (Johnny Depp arrives at Port Royal, ...

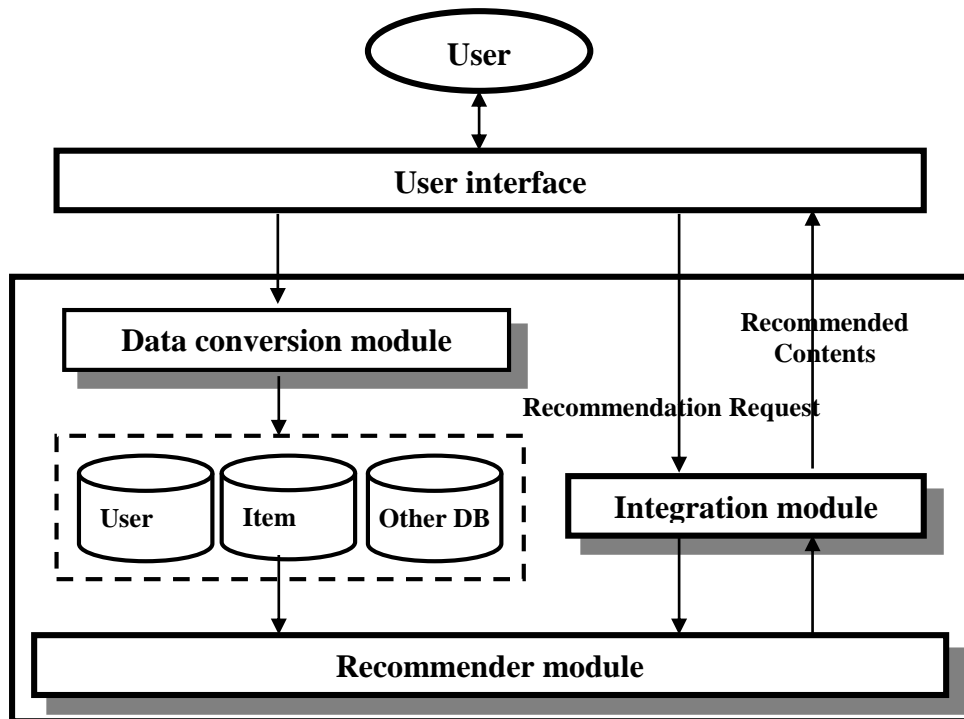
Hình 2.1. Ví dụ về một hệ tư vấn trên website thương mại

2.2. KIẾN TRÚC CƠ BẢN MỘT HỆ TƯ VẤN

Về cơ bản, một hệ tư vấn thường bao gồm các thành phần sau (Hình 2.2.):

- 1. Giao diện người dùng (UI):** gồm giao diện trên ứng dụng hoặc giao diện Web. Trên đó, yêu cầu về tư vấn có thể do người dùng đề xuất trực tiếp cho hệ thống hoặc được tự động chuyển đổi thông qua các yêu cầu truy cập, chẳng hạn yêu cầu tìm kiếm thông tin về sản phẩm trên website. Các yêu cầu tư vấn này được chuyển thành các truy vấn cho hệ thống. Giao diện còn là nơi hệ thống trả về kết quả tư vấn cho người dùng.
- 2. Hệ thống cơ sở dữ liệu:** gồm cơ sở dữ liệu về các người dùng, các hạng mục. Chẳng hạn như dữ liệu điểm đánh giá của các người dùng đánh giá các hạng mục, thông tin nhân khẩu học của người dùng, đặc tính của hạng mục, quan hệ giữa người dùng và hạng mục, v.v... Hệ thống cơ sở dữ liệu còn có module chuyển đổi dữ liệu từ UI, lưu trữ vào hệ thống bổ sung cho nguồn dữ liệu sẵn có.

3. **Recommender module:** nhận các yêu cầu truy vấn từ UI thông qua một module tích hợp trung gian và cơ sở dữ liệu từ hệ thống, thực hiện triển khai các thuật toán tư vấn, xử lý truy vấn trả kết quả về UI cho người dùng.



Hình 2.2. Kiến trúc cơ bản một hệ tư vấn

2.3. MÔ TẢ BÀI TOÁN TƯ VẤN

Một cách hình thức, bài toán tư vấn có thể được mô tả như sau:

Gọi U là tập tất cả người dùng, I là tập tất cả các hạng mục có thể tư vấn được. Không gian U và I có thể từ vài trăm đến vài ngàn, thậm chí lên đến hàng triệu. Khi đó hàm $R(u,i)$ đo lường mức độ hữu ích của hạng mục i đối với người dùng u được gọi là hàm hữu dụng và $R: U \times I \rightarrow P$, với P là một tập có thứ tự.

Khi đó, với mỗi người dùng $u \in U$, ta sẽ tìm các hạng mục $i' \in I$ để tối ưu hóa tính hữu dụng cho người dùng u . Nghĩa là ta sẽ tìm i_u' sao cho:

$$\forall u \in U, \quad i_u' = \arg \max_{i \in I} R(u, i) \quad (2.1)$$

Tính hữu dụng của một hạng mục đối với người dùng trong các hệ tư vấn được đo lường bằng điểm đánh giá, nó biểu thị mức độ ưa chuộng của người dùng đối với một hạng mục nào đó. Hàm hữu dụng R tổng quát có thể là một hàm bất kỳ, hoặc do người dùng đặc tả hoặc được tính toán bởi một ứng dụng cụ thể.

Mỗi người dùng trong không gian U có thể được xác định thông qua profile, chứa nhiều thông tin đặc trưng cho người dùng như tuổi, giới tính, hôn nhân, thu nhập, ... hay đơn giản hơn chỉ là mã số người dùng (userId). Tương tự, mỗi hạng mục trong tập I cũng được xác định bởi tập các đặc trưng cho hạng mục đó.

Trong các hệ tư vấn, hàm R không được xác định trên toàn bộ không gian $U \times I$ mà chỉ xác định trên một miền nhỏ của không gian trên. Điều này dẫn đến việc hàm R phải được ngoại suy trong không gian $U \times I$. Tính hữu dụng trong hệ tư vấn được biểu diễn bằng điểm đánh giá và thường chỉ được bắt đầu tính cho một số hạng mục trước đó đã được người dùng bình chọn (thường là rất nhỏ). Thang điểm đánh giá có thể là 10 hoặc 5. Với các hạng mục chưa được người dùng bình chọn thì dùng ký hiệu \emptyset để biểu diễn. Điểm hữu dụng giữa người dùng và hạng mục được biểu diễn dưới dạng ma trận như ví dụ trong *Bảng 2.1*. Từ những thông tin đó, hệ tư vấn sẽ ước lượng (dự đoán) điểm đánh giá giữa các cặp người dùng / hạng mục chưa được bình chọn, từ đó đưa ra các khuyến nghị thích hợp nhất. Việc ngoại suy từ điểm đánh giá đã biết trước sang các điểm đánh giá chưa biết thường được thực hiện bởi (i) đặc tả các heuristics để định nghĩa hàm hữu dụng và đánh giá hiệu năng theo kinh nghiệm và (ii) ước lượng hàm hữu dụng dựa trên các phương pháp xấp xỉ tối ưu như MSE (*mean square error*) chẳng hạn.

Bảng 2.1. Kết quả đánh giá của người xem phim với một số phim khảo sát

	Harry Potter	Star trek	Xmen	Transformer
A	2	∅	5	4
B	5	2	2	∅
C	∅	∅	1	1
D	4	3	∅	4

2.4. PHÂN LOẠI HỆ TƯ VẤN

Phân loại hệ tư vấn chủ yếu dựa trên kỹ thuật mà hệ sử dụng để tiếp cận. Có 6 hướng tiếp cận tư vấn chính là:

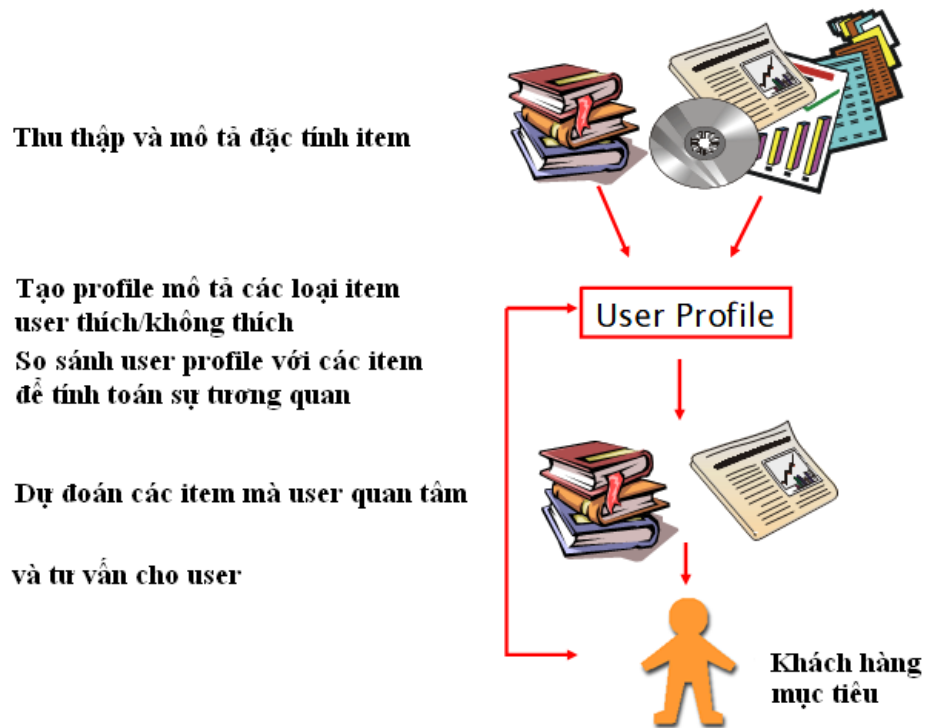
- Phương pháp dựa trên nội dung (*content-based*).
- Phương pháp chọn lọc cộng tác (*collaborative filtering*).
- Phương pháp dựa trên nhân khẩu học (*demographic-based*).
- Phương pháp dựa trên tri thức (*knowledge-based*).
- Phương pháp dựa trên cộng đồng (*community-based*).
- Phương pháp lai (*hybrid*).

Bảng 2.2. tóm tắt các kỹ thuật cơ bản dùng để xây dựng một hệ tư vấn. Thiết kế hệ tư vấn dựa vào kỹ thuật nào tùy thuộc vào lĩnh vực và đặc trưng của dữ liệu có sẵn.

2.4.1. PHƯƠNG PHÁP DỰA TRÊN NỘI DUNG (CONTENT-BASED)

Trong phương pháp này, hàm hữu dụng $R(u,i)$ của hạng mục i đối với người dùng u được ước lượng dựa trên giá trị hữu dụng $R(u,i_k)$ của hạng mục $i_k \in I$ mà người dùng u đánh giá cao và được cho là tương tự với hạng mục i . Ví dụ để

giới thiệu phim cho người dùng u , hệ thống sẽ tìm ra những nét phổ biến nhất trong số các phim được người dùng u đánh giá cao trong quá khứ (dựa vào điểm đánh giá). Sau đó chỉ những phim nào có điểm tương đồng cao với các đặc điểm phổ biến này mới được đề cử cho người dùng u . Các bước cơ bản triển khai một hệ tư vấn CB được mô tả trong Hình 2.3.



Hình 2.3. Ví dụ về một hệ tư vấn trên website thương mại

Phương pháp này dựa trên nền tảng các nghiên cứu về rút trích thông tin (*information retrieval*) và chắt lọc thông tin (*information filtering*). Nhờ những thành tựu trong các nghiên cứu này mà hiện nay nhiều hệ tư vấn loại này tập trung vào tư vấn các đối tượng chứa thông tin dạng văn bản như tài liệu, websites, tin tức, v.v... Những tiến bộ trong rút trích thông tin đã cải thiện đáng kể profile của người dùng như thông tin về sở thích, nhu cầu, thói quen,...), các thông tin này có được một cách tường minh thông qua các bảng câu hỏi gợi ý cho người dùng,

hoặc không tường minh thông qua khai phá thông tin từ những ứng xử và giao tiếp của họ.

Bảng 2.2. Các kỹ thuật tư vấn

Kỹ thuật	Dữ liệu cơ bản	Đầu vào	Xử lý
Chọn lọc cộng tác	Điểm đánh giá của các người dùng trong tập U cho các hạng mục trong tập I .	Điểm đánh giá của người dùng u cho các hạng mục trong tập I .	Xác định các người dùng trong tập U tương đồng với u , và ngoại suy từ điểm đánh giá của các người dùng này cho hạng mục i .
Theo nội dung	Đặc tính của các hạng mục trong I .	Điểm đánh giá của người dùng u cho các hạng mục trong I	Phân loại sự phù hợp giữa thái độ và việc sử dụng hạng mục i của người dùng u .
Nhân khẩu học	Thông tin nhân khẩu học của các người dùng trong U và điểm đánh giá của họ cho các hạng mục trong tập I .	Thông tin nhân khẩu học của người dùng u .	Xác định các người dùng tương đồng về nhân khẩu học với u , và ngoại suy từ điểm đánh giá của họ cho i .

Dựa theo cộng đồng	Đặc tính của các hạng mục trong I .	Hàm hữu dụng cho các hạng mục trong I mô tả mức độ ưu chuộng của u .	Áp dụng hàm cho các hạng mục và xác định hạng của i .
Dựa trên tri thức	Đặc tính của các hạng mục trong I . Tri thức về cách thức thỏa mãn nhu cầu người dùng của các hạng mục này.	Mô tả nhu cầu hoặc quan tâm của u .	Suy luận sự phù hợp giữa i và nhu cầu của u .

Nếu gọi $Content(i)$ là tập các thuộc tính đặc trưng cho hạng mục i , các thuộc tính này thường được mô tả bằng các từ khóa (*keywords*) thì mức độ quan trọng của các thuộc tính được biểu diễn bằng trọng số w , và do đó:

$$Content(i) = (w_{1i}, w_{2i}, \dots, w_{ki}), \text{ với } k \text{ là số thuộc tính đặc trưng cho } i.$$

Đặt $Profile(u)$ là tập các thói quen, sở thích của người dùng u , được thu thập bằng cách phân tích nội dung của các hạng mục đã từng được u cho điểm đánh giá trước đó. $Profile(u)$ cũng có thể được định nghĩa như một vector trọng số:

$Profile(u) = (w_{u1}, w_{u2}, \dots, w_{uk})$, trong đó mỗi trọng số w_{ui} biểu thị mức độ quan trọng của từ khóa k_i đối với người dùng u .

Khi đó hàm hữu dụng $R(u, i)$ được định nghĩa như sau:

$$R(u, i) = score(Profile(u), Content(i)) \quad (2.2)$$

Cả $Profile(u)$, $Content(i)$ đều có thể được biểu diễn bằng vector trọng số \bar{w}_u, \bar{w}_i nên có thể đo độ tương đồng giữa chúng bằng công thức đo cosin như sau:

$$R(u,i) = \cos(\vec{w}_u, \vec{w}_i) = \frac{\vec{w}_u \cdot \vec{w}_i}{\|\vec{w}_u\|_2 \times \|\vec{w}_i\|_2} = \frac{\sum_{j=1}^K w_{j,u} w_{j,i}}{\sqrt{\sum_{j=1}^K w_{j,u}^2} \sqrt{\sum_{j=1}^K w_{j,i}^2}} \quad (2.3)$$

với K là số từ khóa có trong hệ thống.

▪ **Ưu và khuyết điểm:**

○ **Ưu điểm:**

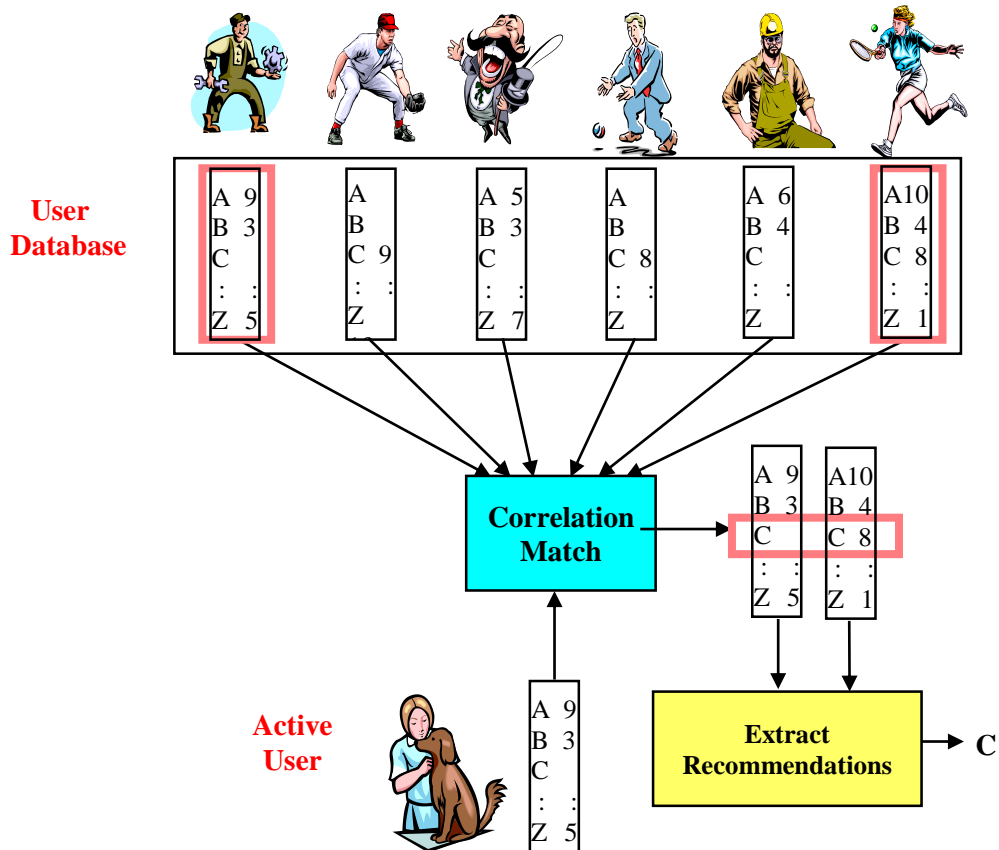
- ✓ *Sự độc lập của người dùng:* CB chỉ khai thác điểm đánh giá được cung cấp do người sử dụng chủ động để xây dựng profile.
- ✓ *Hạng mục mới:* CB có thể khuyến nghị những hạng mục chưa được người dùng đánh giá.

○ **Khuyết điểm:**

- ✓ *Phân tích nội dung bị giới hạn:* không có hệ thống CB nào có thể tư vấn tốt nếu nội dung không đầy đủ thông tin để phân biệt những hạng mục nào được ưa thích với những hạng mục không được ưa thích.
- ✓ *Chuyên biệt hóa quá mức:* các hệ CB không có phương pháp tìm nội dung ngoài mong đợi. Hệ thống chỉ tư vấn những hạng mục có điểm số đánh giá cao. Khuyết điểm này gọi là hiện tượng cầu may (*serendipity*).
- ✓ *Người dùng mới:* các điểm đánh giá phải được thu thập đầy đủ trước khi hệ thống có thể hiểu các ưa chuộng của người dùng và đưa ra những tư vấn chính xác. Vì thế khi có ít điểm đánh giá, như đối với trường hợp một người dùng mới, khuyến nghị của hệ thống thường không đáng tin cậy.

2.4.2. PHƯƠNG PHÁP CHỌN LỌC CỘNG TÁC (COLLABORATIVE FILTERING)

Trong phương pháp này, hệ tư vấn sẽ dự đoán tính hữu dụng của các hạng mục đối với một người dùng cụ thể nào đó dựa trên các item đã được các người dùng khác bình chọn trước đó. Hàm hữu dụng $R(u,i)$ của hạng mục i đối với người dùng u được ước tính dựa trên các giá trị hữu dụng $R(u_j,i)$ của hạng mục i đã được các người dùng u_j , được xem là đồng sở thích với người dùng u , đánh giá trước đó. Ví dụ, để đề cử phim cho người dùng u , hệ tư vấn CF sẽ tìm kiếm các người dùng khác có cùng sở thích phim ảnh như u , sau đó những bộ phim được họ đánh giá cao sẽ được dùng để tư vấn cho u . Hình 2.4. minh họa tiến trình hoạt động của một hệ tư vấn chọn lọc hợp tác.



Hình 2.4. Hoạt động của hệ tư vấn chọn lọc hợp tác

Dựa vào thuật toán cài đặt cho ứng dụng, người ta chia hệ tư vấn loại này thành 2 nhóm: dựa trên kinh nghiệm (*memory-based* hoặc *heuristic-based*) và dựa trên mô hình (*model-based*).

2.4.2.1. Hệ thống dựa trên kinh nghiệm

Các thuật toán dựa trên kinh nghiệm chủ yếu là các thuật toán heuristic đánh giá các hạng mục dựa trên thu thập toàn bộ các hạng mục đã được đánh giá bởi người dùng. Khi đó giá trị đánh giá $r_{u,i}$ (chưa biết) của người dùng u đối với hạng mục i được xem là giá trị tích hợp (*aggregated*) từ đánh giá về i của những người dùng có nhiều sở thích chung nhất với u (thường là N người có sở thích tương đồng nhất):

$$r_{u,i} = \text{aggr}(r_{u',i}) \quad (2.4)$$

$u' \in \hat{U}$

trong đó \hat{U} là tập N người dùng tương đồng nhất với người dùng u và đã đánh giá hạng mục i .

Dưới đây là ví dụ về một số hàm tích hợp (*aggregation function*):

$$(i) \quad r_{u,i} = \frac{1}{N} \sum_{u' \in \hat{U}} r_{u',i}$$

$$(ii) \quad r_{u,i} = k \sum_{u' \in \hat{U}} \text{sim}(u, u') \times r_{u',i}$$

$$(iii) \quad r_{u,i} = \bar{r}_u + k \sum_{u' \in \hat{U}} \text{sim}(u, u') \times (r_{u',i} - \bar{r}_{u'})$$

với: k là hệ số chuẩn hóa và thường lấy giá trị $k = \frac{1}{\sum_{u' \in \hat{U}} |\text{sim}(u, u')|}$

$\text{sim}(u, u')$ là độ tương đồng giữa người dùng u và người dùng u' .

\bar{r}_u là đánh giá trung bình của người dùng u .

Độ tương đồng giữa người dùng u và người dùng u' , $sim(u, u')$ được sử dụng như trọng số và có nhiều cách để tính trọng số này. Hai phương pháp phổ biến nhất là dựa trên độ tương quan (*correlation-based*) và dựa trên cosin (*cosine-based*).

Đặt $S_{xy} = \{i \in I \mid r_{x,i} \neq \emptyset \ \& \ r_{y,i} \neq \emptyset\}$ là tập các hạng mục đều được người dùng x và y đánh giá.

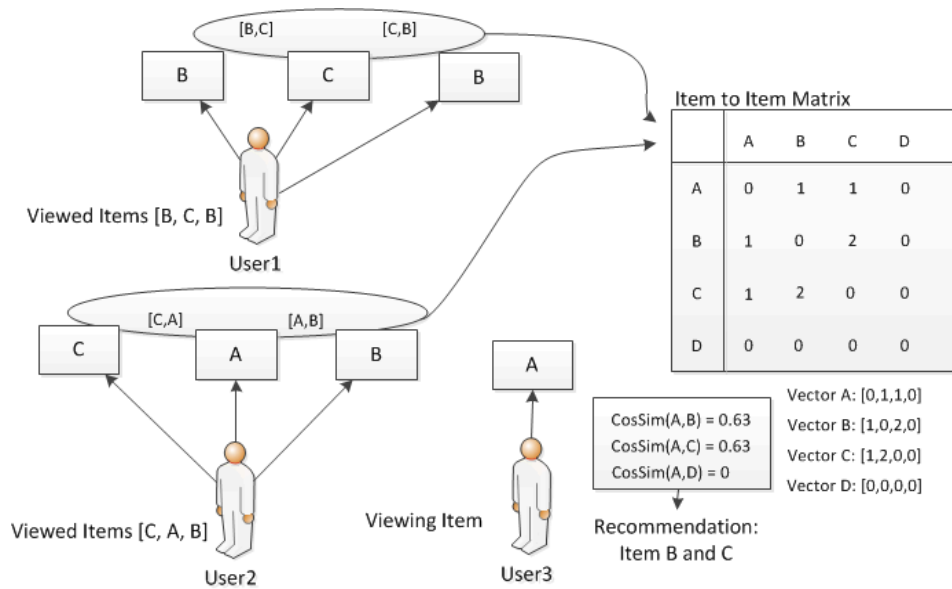
Với phương pháp dựa trên độ tương quan, hệ số tương quan Pearson sẽ là:

$$sim(x, y) = \frac{\sum_{s \in S_{xy}} (r_{x,s} - \bar{r}_x)(r_{y,s} - \bar{r}_y)}{\sqrt{\sum_{s \in S_{xy}} (r_{x,s} - \bar{r}_x)^2 \sum_{s \in S_{xy}} (r_{y,s} - \bar{r}_y)^2}} \quad (2.5)$$

Với phương pháp dựa trên cosin, x và y được xem như 2 vector m chiều, với $m = |S_{xy}|$; độ tương đồng giữa 2 vector chính là cosin của góc giữa 2 vector này:

$$sim(x, y) = \cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\|_2 \times \|\vec{y}\|_2} = \frac{\sum_{s \in S_{xy}} r_{x,s} r_{y,s}}{\sqrt{\sum_{s \in S_{xy}} r_{x,s}^2} \sqrt{\sum_{s \in S_{xy}} r_{y,s}^2}} \quad (2.6)$$

Một biến thể khác của mô hình này là tư vấn dựa trên hạng mục do Amazon phát triển. Thuật toán dùng cho mô hình này được gọi là thuật toán Amazon dựa trên chọn lọc từ hạng mục đến hạng mục (*algorithm based on item-to-item filtering*). Lý do Amazon phát triển thuật toán này vì dữ liệu của họ về hạng mục ít hơn nhiều so với dữ liệu về người dùng. Do đó thay vì tìm độ tương đồng giữa người dùng-người dùng, tìm độ tương đồng hạng mục-hạng mục sẽ lợi hơn nhiều về chi phí thời gian tính toán. Hình 2.5. mô tả vắn tắt cách thức hoạt động của mô hình này.



Hình 2.5. Mô hình hoạt động của hệ tư vấn dựa trên hạng mục của Amazon

Mô hình cho thấy hoạt động tìm vết của hệ thống từ việc thu thập dữ liệu của người dùng và đưa vào ma trận hạng mục. Dựa trên thứ tự các hạng mục đã được người dùng1 và người dùng2 xem, hệ thống thiết lập ma trận hạng mục-hạng mục với các giá trị biểu diễn cho quan hệ giữa các cặp hạng mục là số tổ hợp của các cặp hạng mục này. Từ ma trận hạng mục, hệ thống tính toán độ tương đồng cosin của từng cặp vector hạng mục. Khi người dùng3 (khách hàng mới) xem hạng mục A, hệ thống sẽ dựa trên độ tương đồng hạng mục giữa vector A với các vector hạng mục khác để tìm ra các sản phẩm gần với A nhất và đề xuất tư vấn cho người dùng3.

2.4.2.2. Hệ thống dựa trên mô hình

Các thuật toán dựa trên mô hình thu thập các dữ liệu đã đánh giá để học máy cách xây dựng nên một mô hình, từ đó dự báo các hạng mục chưa được

đánh giá. Breese đã đề xuất phương pháp tiếp cận xác suất để lọc cộng tác (*collaborative filtering*) ước lượng các đánh giá hạng mục theo công thức sau:

$$r_{u,i} = E(r_{u,i}) = \sum_{j=0}^n j \times \Pr(r_{u,i} = j | r_{u,i'}, i' \in I_u) \quad (2.7)$$

với giả thiết là miền giá trị đánh giá từ 0 – n.

Có thể thực hiện hệ thống trên nền học máy với một số kỹ thuật học máy như mạng neuron nhân tạo kết hợp với kỹ thuật trích chọn đặc trưng như kỹ thuật phân rã ma trận (kỹ thuật Khử trị đơn – *Singular Value Decomposition*). Kỹ thuật phân rã ma trận sẽ được trình bày chi tiết trong phần sau của luận văn này.

Ngoài ra, còn nhiều hướng tiếp cận khác như mô hình thống kê, mô hình Bayesian, mô hình entropy cực đại, mô hình hồi quy tuyến tính, ...

Ưu điểm của hệ tư vấn cộng tác so với hệ thống dựa trên nội dung là nó có thể xử lý mọi loại dữ liệu và tư vấn cho mọi loại sản phẩm, kể cả những sản phẩm hoàn toàn mới. Tuy nhiên, với các sản phẩm chưa có điểm đánh giá thì hệ tư vấn cộng tác không thể xử lý được.

▪ **Ưu và khuyết điểm:**

○ **Ưu điểm:**

- ✓ *Tư vấn nhiều loại sản phẩm:* hệ tư vấn CF có khả năng tư vấn nhiều loại sản phẩm trong cùng một hệ thống từ phim ảnh, nghệ thuật, đến các hạng mục văn bản.

○ **Khuyết điểm:**

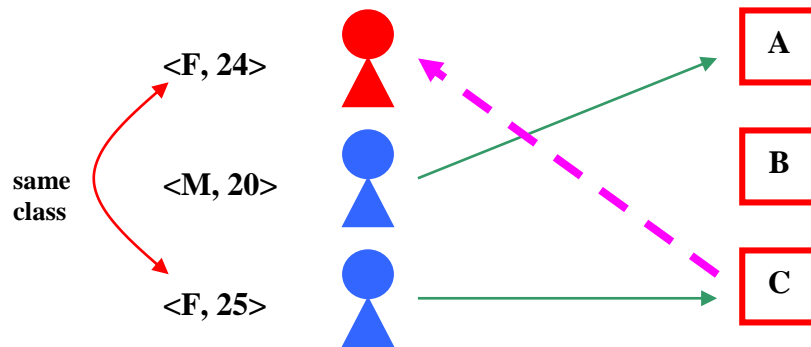
- ✓ *Người dùng mới:* để tư vấn chính xác, hệ phải được học trước các ưa chuộng của người dùng từ điểm đánh giá do họ đưa ra.

- ✓ *Hạng mục mới*: hệ thống sẽ không thể tư vấn các hạng mục không đủ số đông người dùng cho điểm đánh giá.
- ✓ *Tính thưa thớt (sparsity)*: số lượng điểm đánh giá có được thường rất nhỏ so với số điểm đánh giá cần được dự báo nên để hệ thống tư vấn thành công cần phải có số lượng lớn người dùng. Để khắc phục tình trạng này hệ sẽ sử dụng profile của người dùng để tính toán sự tương đồng giữa các người dùng. Hai người dùng được gọi là tương đồng không chỉ khi họ cùng cho điểm đánh giá cho cùng một hạng mục mà còn phải chung lớp nhân khẩu học.
- ✓ *Người dùng không thường xuyên (hiện tượng Grey Sheep)*: đó là các cá nhân có quan điểm bất nhất đối với nhóm người dùng. Các người dùng này hiếm khi nhận được khuyến nghị chính xác từ hệ thống.
- ✓ *Vấn đề dữ liệu kích thước lớn (scalability)*: nguồn dữ liệu càng lớn hệ càng dễ tìm những người dùng tương đồng với nhau về sở thích.
- ✓ *Thiếu minh bạch*: hệ CF được xem là những hộp đen, nghĩa là người dùng không được cung cấp những đầu mối để quyết định khi nào thì một lời tư vấn đáng tin, khi nào thì nghi ngờ. Vì thế hệ CF không được chấp nhận trong những lĩnh vực có nội dung nguy cơ thấp.

2.4.3. PHƯƠNG PHÁP DỰA TRÊN NHÂN KHẨU HỌC (DEMOGRAPHIC-BASED)

Hệ thống loại này sử dụng dữ liệu profile về nhân khẩu học của người dùng (như ngôn ngữ, nghề nghiệp, tuổi tác, địa chỉ...) để phân loại người dùng thành các lớp nhân khẩu. Các sản phẩm gợi ý đều gắn liền với các lớp nhân khẩu này. Phương pháp này không đòi hỏi dữ liệu đánh giá từ phía người dùng mà chỉ cần thông tin nhân khẩu học của các người dùng và thuộc tính đã được định nghĩa

trước của mỗi lớp nhân khẩu. Điểm đánh giá chỉ hữu dụng trong trường hợp các hạng mục có điểm đánh giá cao nhất từ những người dùng thuộc cùng một lớp nhân khẩu. *Hình 2.6.* minh họa cho một hệ thống tư vấn dựa trên nhân chủng học.



Hình 2.6. Mô hình hệ tư vấn dựa trên nhân chủng học

Việc phân lớp nhân khẩu có thể thực hiện bằng các phương pháp máy học, hoặc kỹ thuật phân loại văn bản (*text classification*). Phương pháp đơn giản được sử dụng phổ biến là phân chia các người dùng thành các lớp nhân khẩu và khai thác các tri thức về nhân khẩu học của các lớp này để xác định các quy luật cá nhân hóa (*personalization rules*). *Bảng 2.3.* là ví dụ về một phương pháp nhân khẩu học.

Việc dự báo (tư vấn) có thể được thực hiện bằng các cơ chế học máy như *k*-nearest neighbors, phân lớp naïve Bayes,... và chỉ thực hiện một lần cho một sản phẩm.

Bảng 2.3. Thông tin nhân khẩu học được sử dụng trong hệ tư vấn

	gender	age	area code	education	employed	Dolce
Karen	F	15	714	HS	F	+
Lynn	F	17	714	HS	F	-
Chris	M	35	714	C	T	+
Mike	F	40	714	C	T	-
Jill	F	10	714	E	F	?

Product (e.g., restaurant)

Do việc thu thập thông tin gặp nhiều khó khăn do các người dùng ngại chia sẻ thông tin thật về họ nên hiện nay có rất ít hệ tư vấn sử dụng phương pháp này. Những năm gần đây, nhờ sự phát triển của các mạng xã hội và sự phát triển không ngừng của platforms Web 2.0, các thông tin nhân khẩu học được chia sẻ cũng tương đối đáng tin cậy hơn, nhưng đáng tiếc vẫn không có mấy hệ tư vấn loại này thành công.

▪ **Ưu và khuyết điểm:**

○ **Ưu điểm:**

- ✓ *Không cần số liệu đánh giá:* do đó có thể tư vấn cho các người dùng mới chưa từng cho điểm đánh giá cho hạng mục nào.
- ✓ *Không cần thông tin về các đặc tính của hạng mục:* các hạng mục có thể thuộc nhiều lĩnh vực khác nhau.

○ **Khuyết điểm:**

- ✓ *Xâm phạm quyền riêng tư:* nên khó thu thập dữ liệu.
- ✓ *Tư vấn dễ bị cá nhân hóa:* chẳng hạn không phải tất cả nữ tuổi 20, có việc làm đều thích xem cùng một bộ phim.
- ✓ *Xung đột giữa tính ổn định và tính mềm dẻo:* khó thay đổi profile của người dùng khi xu hướng nhu cầu của họ thay đổi.

2.4.4. PHƯƠNG PHÁP DỰA TRÊN TRI THỨC (KNOWLEDGE-BASED)

Điểm khác biệt cơ bản giữa hệ tư vấn loại này so với các hệ tư vấn trên là tri thức chức năng (*functional knowledge*): đó là tri thức về một sản phẩm cụ thể nào đó đáp ứng như thế nào với nhu cầu cụ thể của người tiêu dùng, từ đó có thể suy luận về mối quan hệ giữa nhu cầu và khuyến nghị có thể có. Hệ tư vấn loại này thực hiện kỹ thuật suy luận dựa trên trường hợp (*case-based reasoning* –

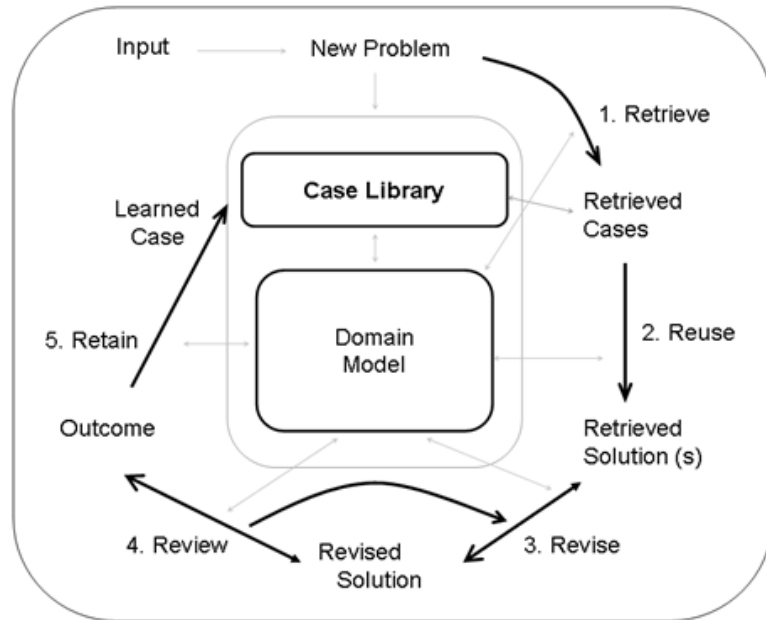
CBR), theo đó hệ sẽ tìm kiếm các vấn đề đã được giải quyết trong quá khứ để giải quyết một vấn đề mới tương tự. Bốn bước cơ bản trong kỹ thuật này là: rút trích (*retrieve*), tái sử dụng (*reuse*), mô phỏng (*adaptation*) và duy trì (*retain*).

- *Rút trích*: tìm kiếm trong cơ sở tri thức các trường hợp tương tự như vấn đề mới đang cần giải quyết.
- *Tái sử dụng*: rút ra các giải pháp đã được sử dụng để giải quyết các trường hợp đó.
- *Mô phỏng*: được chia thành 2 bước nhỏ là tu chỉnh (*revise*) và xem lại (*review*). Tu chỉnh là bước mô phỏng giải pháp sao cho phù hợp với các ràng buộc cụ thể của vấn đề mới. Trong bước xem lại, giải pháp tái dựng sẽ được đánh giá bằng cách áp dụng vào tình huống mới, tìm hiểu tại sao thất bại và thất bại ở đâu, đồng thời có những điều chỉnh thích hợp.
- *Duy trì*: lưu trữ tình huống mới đã được mô phỏng lại vào thư viện tri thức.

Hình 2.7. mô tả các bước giải quyết vấn đề của kỹ thuật suy luận dựa trên trường hợp.

Hệ thống loại này không sinh ra khám phá mới nào vì tác vụ chính của thuật toán tư vấn chỉ là rút ra trường hợp tương đồng nhất với vấn đề đang cần được giải quyết.

Hệ thống không gặp phải vấn đề ramp-up (sẽ bàn luận ở phần sau) vì hệ không phụ thuộc vào cơ sở điểm đánh giá của người dùng.



Hình 2.7. Vòng tròn suy luận của kỹ thuật dựa trên trường hợp.

- **Hệ tư vấn dựa trên tính hữu dụng (utility-based):** Cùng đặc điểm với hệ tư vấn dựa trên tri thức, còn có một hình thức tư vấn khác là tư vấn dựa trên tính hữu dụng. Trong đó, thay vì tính toán sự thỏa mãn nhu cầu của một hạng mục đối với người dùng như thế nào, thì hàm hữu dụng trong hệ tư vấn dựa trên tính hữu dụng sẽ tính toán tính hữu dụng của mỗi hạng mục đối với một đối tượng người dùng cụ thể. Nói cách khác hàm hữu dụng trong hệ này là một ánh xạ từ các hạng mục sang các con số mô tả mức độ hữu dụng của người dùng đối với các hạng mục.

Một hạng mục được liệt kê bởi danh sách các thuộc tính: p_1, \dots, p_m (chẳng hạn như số lượng phòng, số đo diện tích, ...) có giá trị thực hoặc giá trị boolean, với giả định là giá trị của thuộc tính càng cao mức độ hữu dụng càng cao. Hàm hữu dụng của người dùng được mô hình bằng một tập các trọng số w_1, \dots, w_m ($w_i \in [0,1]$):

$$U(w_1, \dots, w_m, p_1, \dots, p_m) = \sum_{j=1}^m w_j p_j \quad (2.8)$$

Mục tiêu là tìm (rút ra) được hạng mục có tính hữu dụng lớn nhất bằng cách cực đại hóa hàm tuyến tính trên.

Hoạt động của hệ thường dựa trên tương tác với người dùng theo cơ chế sau: (i) người dùng đưa ra các yêu cầu; (ii) hệ thống đề xuất các giải pháp; (iii) nếu giải pháp phù hợp thì dừng; (iv) ngược lại, người dùng sẽ chọn giải pháp; điều chỉnh trọng số để hệ thống tính toán lại; (v) trở lại (ii) đến khi được giải pháp thích hợp.

▪ **Ưu và khuyết điểm:**

○ **Ưu điểm:**

- ✓ *Không mắc phải vấn đề ramp-up:* hệ tư vấn không sử dụng cơ sở dữ liệu điểm đánh giá của người dùng.
- ✓ *Không cần thu thập thông tin người dùng cần tư vấn:* vì hệ tư vấn chỉ cần dựa vào các trường hợp tương tự có sẵn để xử lý cho trường hợp hiện tại.

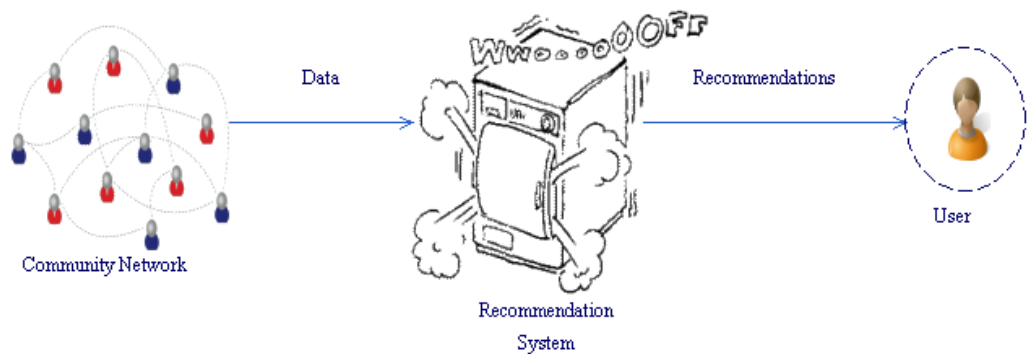
○ **Khuyết điểm:**

- ✓ *Không sinh ra tri thức mới:* tư vấn chỉ thực hiện được cho những trường hợp hiện tại tương tự các trường hợp đã có trong quá khứ.

2.4.5. PHƯƠNG PHÁP DỰA TRÊN CỘNG ĐỒNG (COMMUNITY-BASED)

Hệ tư vấn loại này gợi ý các hạng mục dựa trên sở thích của bạn bè của người sử dụng theo triết lý “Hãy nói với tôi về bạn bè của bạn, tôi sẽ nói bạn là ai” (*Tell me who your friends are and tell you who you are*). Thực tế cho thấy rằng con người có xu hướng nghe theo lời khuyên của bạn bè nhiều hơn là nghe

theo lời khuyên tương tự từ những người không quen biết. Hệ thống dựa vào cộng đồng phát triển cùng với sự phổ biến rộng rãi của các mạng xã hội mở. Mô hình này cần dữ liệu về các mối quan hệ xã hội giữa các người dùng và sở thích của bạn bè của người dùng. Việc tư vấn dựa trên điểm đánh giá được thu thập từ bạn bè của người dùng (Hình 2.8).



Hình 2.8. Mô hình tư vấn dựa trên cộng đồng

Các nghiên cứu về phương pháp này còn trong giai đoạn đầu và hiệu quả của phương pháp vẫn chưa rõ ràng. Chẳng hạn, lời khuyên dựa trên mạng xã hội thường không chính xác bằng lời khuyên được rút ra từ các phương pháp tiếp cận CF truyền thống, trừ một số trường hợp đặc biệt, như điểm đánh giá cho một hạng mục cụ thể nào đó dao động nhiều (các hạng mục còn gây tranh cãi), hoặc với các hạng mục ít được quan tâm (*cold-start*), là các hạng mục mà người dùng không cung cấp đầy đủ điểm đánh giá để tính toán sự tương đồng với các người dùng khác. Ngược lại cũng có những lời khuyên dựa trên dữ liệu từ mạng xã hội tốt hơn lời khuyên rút ra từ những dữ liệu profile tương tự, và khi đưa dữ liệu mạng xã hội này vào các phương pháp tiếp cận CF đã cải thiện đáng kể kết quả tư vấn.

- **Ưu và khuyết điểm:**

- **Ưu điểm:**

- ✓ *Không giới hạn hạng mục tư vấn:* trong môi trường cộng đồng, mối quan tâm là vô hạn.
- ✓ *Có thể tư vấn người dùng:* chẳng hạn tư vấn về tác giả một video clip mà người dùng cần tư vấn quan tâm.

- **Khuyết điểm:**

- ✓ *Hiện tượng ramp-up:* không thể tư vấn các hạng mục ít được cộng đồng quan tâm.
- ✓ *Ảnh hưởng bởi tâm lý đám đông:* tâm lý đám đông thường tạo ra những xu hướng thiên lệch nhất thời, phản ánh không chính xác bản chất của cộng đồng.

2.4.6. PHƯƠNG PHÁP LAI (HYBRID)

Các hệ tư vấn lai được xây dựng dựa trên kết hợp các kỹ thuật đã nêu ở trên. Một hệ thống lai kết hợp kỹ thuật A và B sẽ tận dụng những ưu điểm của A để khắc phục những nhược điểm của B. Ví dụ, các phương pháp tiếp cận CF không thể khuyến nghị được đối với các hạng mục mới (hạng mục không có điểm đánh giá). Nhưng với hệ dựa vào nội dung thì điều này không bị hạn chế vì phương pháp chủ yếu dựa vào các mô tả (đặc tính) có sẵn của hạng mục. Tùy thuộc vào ngữ cảnh mà có thể linh hoạt kết hợp 2 hoặc nhiều kỹ thuật tư vấn cơ bản trong một hệ thống lai. Chẳng hạn, trong ngữ cảnh thời gian, các khuyến nghị về các tour nghỉ đông sẽ phải khác với các tour nghỉ hè, hoặc tư vấn nhà hàng cho bữa ăn tối thứ bảy với bạn bè phải khác với tư vấn bữa ăn trưa với các đồng nghiệp.

Có 3 mô hình kết hợp thông tin ngữ cảnh trong một hệ thống lai là: dựa trên suy luận (*reduction-based*), còn gọi là lọc trước (*pre-filtering*); lọc sau theo

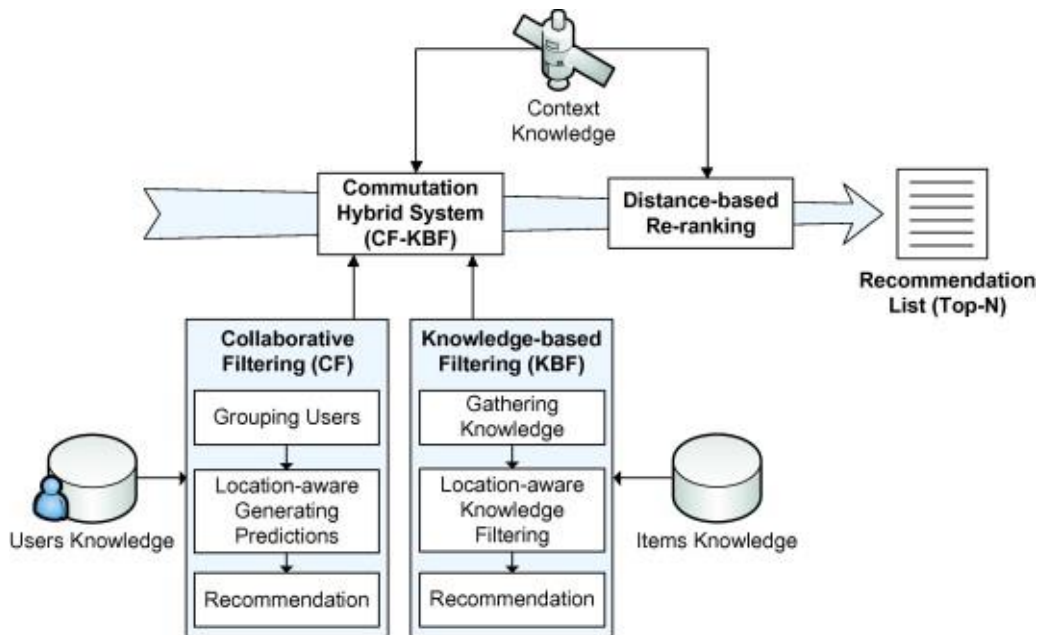
ngữ cảnh (*contextual post filtering*); và mô hình ngữ cảnh (*context modeling*). Trong phương pháp lọc trước, chỉ những thông tin phù hợp với ngữ cảnh hiện tại, là những điểm đánh giá của các hạng mục được đánh giá trong cùng ngữ cảnh, mới được sử dụng để đưa vào hệ thống xử lý. Trong phương pháp lọc sau theo ngữ cảnh, thuật toán tư vấn sẽ bỏ qua thông tin về ngữ cảnh. Sau đó đầu ra của thuật toán sẽ được lọc / điều chỉnh sao cho để chỉ những khuyến nghị có liên quan đến ngữ cảnh mục tiêu mới được đưa ra. Trong mô hình ngữ cảnh, là phương pháp phức tạp nhất, dữ liệu ngữ cảnh được sử dụng một cách tường minh trong mô hình. *Hình 2.9.* minh họa một hệ tư vấn lai giữa tư vấn dựa trên chọn lọc cộng tác (CF) và tư vấn dựa trên tri thức (KB) được sử dụng để tư vấn cho du khách về hệ thống thông tin địa lý 3D trên thiết bị di động (*mobile 3D-GIS hybrid recommender system for tourism*).

Có nhiều phương pháp lai kết hợp các kỹ thuật tư vấn cơ bản trong một hệ tư vấn lai. *Bảng 2.4.* liệt kê các phương pháp lai phổ biến đang được sử dụng trên các website thương mại hiện nay.

Bảng 2.4. Các phương pháp lai

Phương pháp lai	Mô tả
Weighted	Điểm đánh giá (hoặc bình chọn) của các kỹ thuật tư vấn được kết hợp lại với nhau để sinh ra một lời khuyên duy nhất.
Switching	Hệ thống tự động chuyển đổi qua lại giữa các kỹ thuật tư vấn tùy thuộc vào tình huống cụ thể hiện tại.
Mixed	Các khuyến nghị từ nhiều kỹ thuật tư vấn khác nhau được thể hiện ra cùng một lúc.

Feature combination	Các đặc tính từ các nguồn dữ liệu khác nhau dùng cho tư vấn được đưa vào cho một thuật toán tư vấn duy nhất xử lý.
Cascade	Một thuật toán tư vấn điều chỉnh kết quả tư vấn từ một thuật toán tư vấn khác.
Feature augmentation	Đầu ra từ một kỹ thuật tư vấn được sử dụng làm đặc tính đầu vào cho một kỹ thuật tư vấn khác.
Meta-level	Mô hình được học bởi một kỹ thuật tư vấn được dùng làm đầu vào cho một kỹ thuật tư vấn khác.



Hình 2.9. Mô hình một hệ tư vấn lai

- **Phương pháp lai có trọng số (*weighted hybrid*):** được thực hiện như sau:

Đặt $S_A(p)$ là điểm đánh giá dự báo cho sản phẩm p được tính bởi thuật toán A , $S_B(p)$ là điểm đánh giá dự báo cho sản phẩm p được tính bởi thuật toán B , thì điểm đánh giá dự báo cho sản phẩm p được tính bởi thuật toán lai H được tính theo công thức sau:

$$S_H(p) = \alpha S_A(p) + (1-\alpha)S_B(p) \quad (2.9)$$

với α là trọng số (hay xác suất xảy ra) cho $S_A(p)$.

- **Phương pháp lai chuyển đổi (*switch hybrid*):** hệ thống sử dụng một số tiêu chí để chuyển đổi qua lại giữa các kỹ thuật tư vấn. Chẳng hạn hệ tư vấn The DailyLearner sử dụng phương pháp lai content/collaborative, trong đó phương pháp dựa vào nội dung được thực hiện trước. Nếu hệ thống dựa vào nội dung không thể cho được kết quả tư vấn đáng tin cậy, thì hệ sẽ chuyển qua thực hiện kỹ thuật cộng tác. Mức độ tin cậy của kết quả tư vấn như thế nào là tùy thuộc vào chủ định của người thiết lập hệ thống. Nếu điều kiện chuyển đổi tốt và hợp lý thì sẽ cho kết quả tốt.
- **Phương pháp lai hỗn hợp (*mixed hybrid*):** các khuyến nghị từ nhiều kỹ thuật tư vấn sẽ thể hiện ra cùng một lúc trong hệ thống. Mỗi kỹ thuật trong hệ thống sẽ tạo ra một danh sách các khuyến nghị cùng với điểm xếp hạng của từng khuyến nghị. Sau đó thuật toán tính điểm của hệ thống sẽ so sánh các khuyến nghị và hợp nhất điểm xếp hạng của các khuyến nghị này vào chung một danh sách đã xếp hạng. Chẳng hạn cách hợp nhất đơn giản nhất là cộng điểm xếp hạng cho khuyến nghị của từng kỹ thuật như sau: $CN_Rank(2) + CF_Rank(3) = Mixed_Rank(5)$. Do đưa ra danh sách xếp hạng hỗn hợp nên hệ thống lai loại này tránh tư vấn cho các sản phẩm mới.
- **Phương pháp lai tổ hợp đặc tính (*feature combination*):** thường kết hợp content/collaborative, trong đó các thông tin của kỹ thuật cộng tác (điểm đánh giá của người dùng) được xử lý như là dữ liệu đặc tính đơn giản bổ sung đi kèm với các hạng mục và sử dụng kỹ thuật dựa vào nội dung để xử lý tập dữ liệu đã được bổ sung này. Phương pháp lai này cho phép hệ thống tham khảo dữ liệu cộng tác mà không phụ thuộc vào kết quả xử lý của kỹ thuật cộng tác, do đó sẽ giảm thiểu độ nhạy cảm của hệ thống đối với số lượng người dùng đã tham gia đánh giá một

hạng mục. Hệ thống sẽ cho thông tin về tính tương đồng vốn có của các hạng mục đã bị hệ thống cộng tác che mờ.

- **Phương pháp lai bậc thang (*cascade*):** một kỹ thuật tư vấn được thực hiện trước để tạo ra xếp hạng thô cho các ứng viên. Sau đó kỹ thuật thứ hai sẽ tinh chỉnh lại kết quả tư vấn từ tập các ứng viên này. Phương pháp bậc thang giúp hệ thống tránh việc sử dụng kỹ thuật thứ cấp (có độ ưu tiên thấp) cho các hạng mục đã phân hóa tốt trong kỹ thuật thứ nhất. Để hệ thực hiện tốt cần sắp xếp hợp lý thứ tự của các kỹ thuật thực hiện.
- **Phương pháp lai bổ sung đặc tính (*feature augmentation*):** tạo điểm đánh giá hoặc phân loại một hạng mục và thông tin đó được đưa vào tiến trình xử lý của kỹ thuật thứ hai. Chẳng hạn, hệ tư vấn Libra tư vấn sách bằng kỹ thuật dựa vào nội dung với dữ liệu tìm thấy trên Amazon.com trên nền tảng phân loại văn bản naïve Bayes. Các dữ liệu dạng văn bản trên Amazon.com bao gồm thông tin về tác giả liên quan và tựa sách liên quan, lại được hệ tư vấn trên Amazon sinh ra bằng kỹ thuật cộng tác.
- **Phương pháp lai siêu cấp (*meta-level*):** sử dụng mô hình được sinh ra từ một bởi một hệ thống làm đầu vào cho một hệ thống khác. Ví dụ: hệ tư vấn cộng tác thông qua nội dung, trong đó mô hình được sinh ra bởi phương pháp tiếp cận dựa vào nội dung (winnow – mô hình 1), và được dùng để biểu diễn cho các người dùng bằng phương pháp lọc cộng tác (mô hình 2). Phân biệt giữa phương pháp lai meta-level và phương pháp lai feature augmentation ở chỗ: trong phương pháp lai feature augmentation ta sử dụng một mô hình đã được học để tạo ra các đặc tính làm đầu vào cho thuật toán thứ hai; còn trong phương pháp lai meta-level, toàn bộ mô hình trở thành đầu vào cho thuật toán thứ hai.

2.4.7. VẤN ĐỀ RAMP-UP

Ramp-up là từ dùng để mô tả một vấn đề mà các kỹ thuật tư vấn cơ bản thường gặp phải, nhất là đối với hệ tư vấn dựa trên nội dung. Ramp-up gồm một hoặc cả hai vấn đề sau: Early rater và Sparsity.

- **Early rater/cold-start:** các hạng mục mới và người dùng mới thường gây ra vấn đề này cho các hệ tư vấn. Early rater là hiện tượng một hạng mục mới sẽ không được tư vấn trừ phi hạng mục này đã được một số người dùng cho điểm đánh giá trước đó. Vấn đề này thường xảy ra với hệ tư vấn CF. Hệ tư vấn CB không bị ảnh hưởng bởi hiện tượng này vì CB không dựa vào điểm đánh giá mà dựa vào thuộc tính của tất cả hạng mục để tư vấn. Early rater cũng có thể xảy ra với các hạng mục ít được quan tâm. Cold-start lại xảy ra với một người dùng mới, do không có những thông tin trước đó nên hệ thống không thể tìm được các người dùng có cùng quan tâm để tư vấn cho người dùng mới này.
- **Sparsity:** nói một cách đơn giản, đa số người dùng không đánh điểm đánh giá cho đa số hạng mục và do đó ma trận dữ liệu rất thưa. Đây là vấn đề của hệ thống tư vấn CF, vì làm giảm xác suất tìm tập các người dùng có điểm đánh giá tương tự nhau. Vấn đề này xảy ra khi dữ liệu có tỉ lệ hạng mục/người dùng quá cao hoặc dữ liệu mới được khởi tạo.

Nói cách khác, một hệ thống chỉ hữu dụng đối với đa số người dùng khi có đủ lớn một số lượng người dùng được biết thói quen của họ; và hệ thống chỉ hữu dụng đối với một đối tượng người dùng cụ thể khi thu thập đủ một số lượng hạng mục được đánh giá.

Ngoại trừ kỹ thuật tư vấn dựa trên tri thức, các kỹ thuật tư vấn còn lại đều có thể gặp phải vấn đề ramp-up.

Bên cạnh những vấn đề ramp-up nêu trên gây khó khăn cho hệ tư vấn về mặt kỹ thuật, một khó khăn khác mang tính chủ quan là sự gian lận. Người bán hàng vì muốn bán được hoặc tẩy chay một mặt hàng nào đó đã tạo ra các số liệu gian lận như thổi phồng nhu cầu cho mặt hàng của mình hoặc hạ thấp nhu cầu mặt hàng của đối thủ cạnh tranh để hệ thống tư vấn cho người dùng theo hướng có lợi cho mục đích của họ. Các kiểu gian lận như vậy đều có hại cho khả năng dự đoán điểm đánh giá của hệ tư vấn.

Nhìn chung, có nhiều phương pháp xây dựng một hệ tư vấn. Không có phương pháp nào là tối ưu cho mọi trường hợp. Mỗi phương pháp đều có ưu và khuyết điểm riêng. Phương pháp lai kết hợp các phương pháp khác nhau nhằm tận dụng ưu điểm của phương pháp này để khắc phục khuyết điểm của phương pháp kia và ngược lại. Lựa chọn phương pháp nào là tùy thuộc vào mục đích xây dựng hệ tư vấn, tính đặc thù của dịch vụ tư vấn, hình thức thu thập dữ liệu, tính chất dữ liệu, chi phí triển khai, v.v...

CHƯƠNG 3 CƠ SỞ LÝ THUYẾT

Kỹ thuật Phân rã ma trận được xem là kỹ thuật tiên bộ nhất trong số các kỹ thuật dùng trong phương pháp CF vì nó cho phép ta khám phá ra những đặc tính tiềm ẩn nằm bên dưới các tương tác giữa người dùng và hạng mục.

3.1. MÔ HÌNH PHÂN RÃ MA TRẬN

Về cơ bản, phân rã ma trận đặc trưng cho cả người dùng và hạng mục bởi các vector các yếu tố được suy diễn từ các mô hình đánh giá hạng mục. Sự tương đồng cao giữa các yếu tố người dùng và hạng mục sẽ tạo ra tư vấn. Phương pháp này trở nên phổ biến trong những năm gần đây do xử lý tốt dữ liệu có kích thước lớn và cho ra các khuyến nghị chính xác, cũng như tính linh động khi tạo mô hình cho nhiều tình huống trong đời sống thực.

Dữ liệu đầu vào cho các hệ tư vấn có nhiều kiểu và thường được biểu diễn trong một ma trận 2 chiều với một chiều biểu diễn cho các người dùng và chiều còn lại là các hạng mục được người dùng quan tâm. Có 2 phương pháp thu thập dữ liệu. Thu thập dữ liệu trực tiếp qua thông tin phản hồi trực tiếp (*explicit feedback*) từ phía người dùng về quan tâm của họ dành cho sản phẩm. Các quan tâm này thường được đặc trưng bởi các con số được gọi là điểm đánh giá. Ma trận điểm đánh giá từ thu thập trực tiếp thường là các ma trận thưa vì bất kỳ một người dùng nào cũng thường có xu hướng chỉ đánh giá một tỉ lệ nhỏ các hạng mục có sẵn. Trường hợp không thu thập dữ liệu trực tiếp được, hệ tư vấn sẽ phải thu thập dữ liệu gián tiếp (*implicit feedback*) để suy diễn ra các quan tâm của người dùng bằng cách quan sát thái độ của người dùng trong quá khứ từ các dữ liệu lịch sử đi mua, duyệt các dữ liệu này để tìm kiếm các mô hình, ... Dữ liệu gián tiếp thường là có hoặc không có một sự kiện nào đó nên ma trận dữ liệu là một ma trận dày đặc. Điểm mạnh của phương pháp phân rã ma trận là khả năng cho phép kết hợp thông tin bổ sung, tránh hiện tượng ramp-up cho hệ tư vấn.

Giả sử ta có mỗi người dùng đã cho điểm đánh giá cho một số hạng mục trong hệ thống, ta sẽ dự báo xem các người dùng sẽ cho điểm đánh giá như thế nào cho các hạng mục mà họ chưa đánh giá, nghĩa là ta sẽ tư vấn (dự báo) cho các người dùng này đánh giá các hạng mục chưa được họ đánh giá. Các thông tin đánh giá sẽ được lưu trữ trong một ma trận. *Bảng 3.1* là một ví dụ về ma trận điểm đánh giá gồm có 5 người dùng và 4 hạng mục.

Tác vụ dự báo được xem như là công việc lấp đầy cho ma trận. Ta có thể khám phá các đặc tính (yếu tố) tiềm ẩn qua việc dự báo điểm đánh giá mà một người dùng nào đó đánh giá một hạng mục nào đó, vì các đặc tính mà người dùng quan tâm sẽ trùng khớp các đặc tính của hạng mục đó.

Bảng 3.1. Ma trận điểm đánh giá

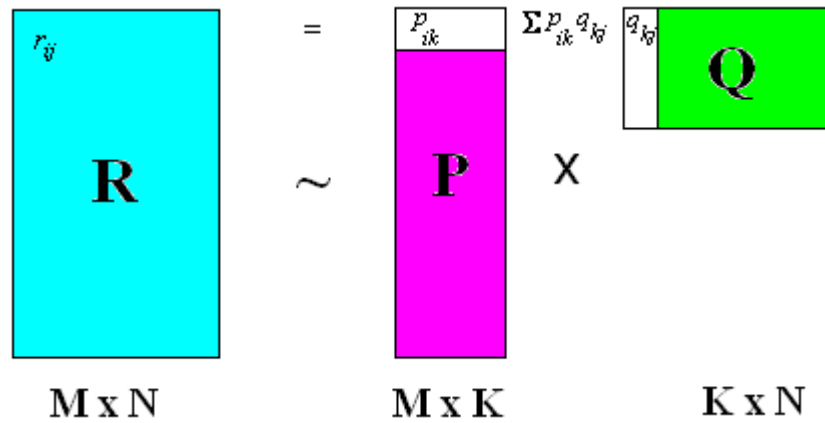
	D1	D2	D3	D4
U1	5	3	-	1
U2	4	-	-	1
U3	1	1	-	5
U4	1	-	-	4

Dấu (-) có nghĩa là người dùng U_i chưa cho điểm đánh giá hạng mục D_j .

Phân rã ma trận ánh xạ người dùng và hạng mục sang một không gian có hướng f các yếu tố tiềm ẩn kết hợp với nhau, nghĩa là tương quan người dùng-hạng mục được mô hình thành tích vô hướng bên trong không gian đó. Đặt U là tập các người dùng, D là tập các hạng mục, khi đó ma trận \mathbf{R} có kích thước $|U| \times |D|$ là ma trận chứa tất cả các điểm đánh giá thu thập được mà các người dùng đã đánh giá các hạng mục.

Ý tưởng chính của mô hình phân rã ma trận trong xây dựng hệ tư vấn là xem ma trận điểm đánh giá \mathbf{R} là kết quả nhân từ 2 ma trận có hạng nhỏ hơn \mathbf{P} và \mathbf{Q} . Ma trận \mathbf{P} gọi là ma trận cơ sở người dùng với mỗi hàng trong \mathbf{P} tượng trưng cho một người dùng.

Các giá trị p_{ik} trong vector hàng i của \mathbf{P} biểu thị mức độ quan tâm của người dùng i đến đặc tính k của hạng mục. Ma trận \mathbf{Q} là ma trận đặc tính của hạng mục với mỗi cột trong \mathbf{Q} tương ứng cho một hạng mục. Các giá trị q_{kj} trong cột j biểu thị mức độ thuộc về của đặc tính k với hạng mục j . Hình 3.1 minh họa cho ý tưởng này. Như vậy từ ma trận \mathbf{R} ban đầu, ta có thể phân rã ra thành 2 ma trận \mathbf{P} và \mathbf{Q} có hạng thấp hơn. Sau đó tìm \mathbf{P} và \mathbf{Q} sao cho phép nhân $\mathbf{P}\mathbf{Q}$ xấp xỉ với \mathbf{R} .



Hình 3.1. Mô hình Phân rã ma trận

Giả sử ta cần khám phá K đặc tính tiềm ẩn ($K < |U|, |D|$), khi đó ta sẽ tìm 2 ma trận $\mathbf{P}_{|U| \times K}$ và $\mathbf{Q}_{|D| \times K}$ sao cho tích $\mathbf{P} \times \mathbf{Q}$ xấp xỉ được ma trận \mathbf{R} , nghĩa là:

$$\mathbf{R} \approx \mathbf{P} \times \mathbf{Q}^T = \hat{\mathbf{R}} \quad (3.1)$$

Mỗi hàng của \mathbf{P} biểu diễn mức độ quan tâm của người dùng với các đặc tính. Mỗi hàng của \mathbf{Q} biểu diễn mức độ thuộc về của các đặc tính trong các hạng mục. Để dự báo điểm đánh giá của người dùng u_i dành cho hạng mục d_j , ta sẽ tính tích vô hướng của hai vector tương ứng với u_i và d_j :

$$\hat{r}_{ij} = p_i q_j^T = \sum_{k=1}^K p_{ik} q_{kj} \quad (3.2)$$

Để tìm P và Q , đầu tiên khởi tạo trị ban đầu cho P và Q , tính $M = PQ$ rồi tối thiểu hóa độ lệch giữa ma trận M với R . Mỗi lần lặp là một lần điều chỉnh P và Q để tối thiểu hóa độ lệch giữa M và R .

3.2. CÁC THUẬT TOÁN HỌC (*Learning Algorithms*)

Có nhiều phương pháp để giảm lỗi đến cực tiểu trong mô hình phân rã ma trận. Áp dụng các phương pháp này đồng nghĩa với việc tối ưu hóa hàm mục tiêu (*objective function*). Phương pháp được sử dụng phổ biến trong kỹ thuật phân rã ma trận là giảm gradient ngẫu nhiên (*stochastic gradient descent*).

3.2.1. PHƯƠNG PHÁP GIẢM GRADIENT NGẪU NHIÊN (STOCHASTIC GRADIENT DESCENT)

Phương pháp này nhằm tìm điểm cực tiểu cục bộ của độ lệch bình phương giữa 2 đối tượng. Hàm mục tiêu giảm độ lệch bình phương giữa 2 đối tượng A và B được định nghĩa như sau:

$$\min f(A||B) = (A - B)^2 \quad (3.3)$$

Độ lệch được gọi là lỗi giữa điểm đánh giá ước lượng với điểm đánh giá thực tế, có thể được tính toán bằng công thức sau cho mỗi cặp người dùng-hạng mục:

$$e_{ij}^2 = (r_{ij} - \hat{r}_{ij})^2 = (r_{ij} - \sum_{k=1}^K p_{ik} q_{kj})^2 \quad (3.4)$$

Ta xét lỗi bình phương vì điểm đánh giá ước lượng có thể có lúc cao hơn, có lúc thấp hơn điểm đánh giá thực tế gây ra hiện tượng bù trừ khi tính tổng lỗi.

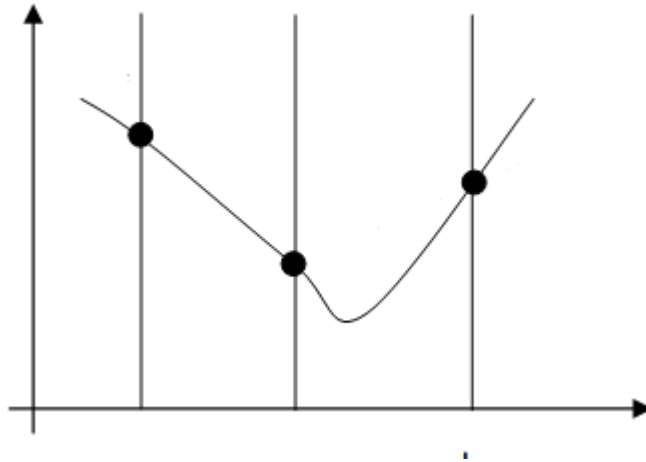
Để cực tiểu lỗi, ta phải biết hướng để chỉnh sửa các giá trị của p_{ik} và q_{kj} . Nghĩa là ta cần phải biết được gradient của các giá trị hiện tại. Lấy đạo hàm riêng cho công thức (3.4) theo các biến p_{ik} và q_{kj} , ta được:

$$\begin{aligned}\frac{\partial}{\partial p_{ik}} e_{ij}^2 &= -2(r_{ij} - \hat{r}_{ij})(q_{kj}) = -2e_{ij} q_{kj} \\ \frac{\partial}{\partial q_{kj}} e_{ij}^2 &= -2(r_{ij} - \hat{r}_{ij})(p_{ik}) = -2e_{ij} p_{ik}\end{aligned}\quad (3.5)$$

Cập nhật gradient cho các p_{ik} và q_{kj} , ta có:

$$\begin{aligned}p'_{ik} &= p_{ik} + \alpha \frac{\partial}{\partial p_{ik}} e_{ij}^2 = p_{ik} + 2\alpha e_{ij} q_{kj} \\ q'_{kj} &= q_{kj} + \alpha \frac{\partial}{\partial q_{kj}} e_{ij}^2 = q_{kj} + 2\alpha e_{ij} p_{ik}\end{aligned}\quad (3.6)$$

trong đó α , còn gọi là bước nhảy, là hằng số mà giá trị của nó xác định tốc độ dần đến điểm cực tiểu. Giá trị α là tùy chọn và thường khá nhỏ (khoảng 0.0002). Nếu chọn bước nhảy quá lớn để tiến đến cực tiểu có thể vượt khỏi điểm cực tiểu, và dao động quanh điểm cực tiểu. Ngược lại, nếu chọn α quá nhỏ tốc độ dần đến cực tiểu quá chậm. *Hình 3.2* cho thấy ảnh hưởng của bước nhảy trong quá trình dần đến điểm cực tiểu.



Hình 3.2. Ảnh hưởng của bước nhảy trong quá trình tiệm tiến đến cực tiểu

Vấn đề nảy sinh ra cho mô hình phân rã ma trận là nếu tìm được các ma trận \mathbf{P} và \mathbf{Q} sao cho $\mathbf{P} \times \mathbf{Q}$ xấp xỉ được \mathbf{R} thì dự báo cho các đánh giá chưa thấy có gần về zero hết hay không? Thực tế, mô hình không đi tìm \mathbf{P} và \mathbf{Q} để có được \mathbf{R} một cách chính xác, mà chỉ tối thiểu hóa lỗi của các cặp người dùng-hạng mục quan sát được. Nói cách khác, nếu đặt \mathbf{T} là tập các bộ có dạng (u_i, d_j, r_{ij}) , tức \mathbf{T} chứa tất cả các cặp người dùng-hạng mục cùng với điểm đánh giá, ta sẽ cố gắng cực tiểu lỗi e_{ij} cho mỗi bộ $(u_i, d_j, r_{ij}) \in \mathbf{T}$ (\mathbf{T} chính là tập dữ liệu huấn luyện). Đối với các bộ chưa đầy đủ (thiếu r_{ij}), mô hình sẽ xác định các giá trị này một khi mối quan hệ giữa người dùng, hạng mục và các đặc tính được học.

Sử dụng công thức (3.5) lặp đi lặp lại nhiều lần cho đến khi lỗi hội tụ về điểm cực tiểu. Có thể kiểm tra lỗi tổng thể sau mỗi lần tính toán theo công thức sau để xác định khi nào có thể dừng được:

$$E = \sum_{(u_i, d_j, r_{ij}) \in \mathbf{T}} e_{ij} = \sum_{(u_i, d_j, r_{ij}) \in \mathbf{T}} \left(r_{ij} - \sum_{k=1}^K p_{ik} q_{kj} \right)^2 \quad (3.7)$$

3.2.2. PHƯƠNG PHÁP ALTERNATING LEAST SQUARE (ALS)

Do cả hai vector p_{ik} và q_{kj} đều chưa được xác định, hàm (3.4) không lõm. Tuy nhiên, nếu ta cố định một trong hai vector trên thì hàm (3.4) trở thành hàm bậc hai và có thể giải được bài toán tối ưu. Kỹ thuật ALS lần lượt luân phiên cố định p_{ik} và q_{kj} . Khi cố định p_{ik} , hệ thống sẽ tính toán lại q_{kj} bằng cách giải quyết vấn đề bình phương tối thiểu và ngược lại. Như vậy mỗi bước thực hiện sẽ giảm e_{ij} xuống cho đến khi hội tụ.

Nhìn chung, phương pháp giảm gradient ngẫu nhiên dễ cài đặt và chạy nhanh hơn phương pháp ALS, nhưng phương pháp ALS có ưu thế hơn trong ít nhất 2 trường hợp sau:

- (1) Khi hệ thống có thể thực hiện song song hóa. Khi sử dụng phương pháp ASL, hệ thống sẽ tính mỗi q_{kj} và mỗi p_{ik} một cách độc lập. Điều này có thể làm phát sinh một lượng lớn tính toán song song.
- (2) Khi hệ thống tập trung vào xử lý dữ liệu gián tiếp. Trong trường hợp này tập dữ liệu huấn luyện thường rời rạc, nếu thực hiện phương pháp giảm gradient ngẫu nhiên có thể xuất hiện hiện tượng loop trên mỗi trường hợp huấn luyện. Phương pháp ASL có thể xử lý tốt trường hợp này.

3.3. HỆ SỐ BIAS

Một ưu điểm của hệ thống phân rã ma trận là tính linh động về khả năng xử lý dữ liệu thuộc nhiều lĩnh vực khác nhau cũng như các yêu cầu có tính đặc thù của ứng dụng. Tuy nhiên, nhiều biến đổi đánh giá quan sát được chịu ảnh hưởng từ phía người dùng hoặc hạng mục, ảnh hưởng này được gọi là thiên hướng (*bias*) và tồn tại một cách độc lập với quan hệ chủ quan giữa người dùng và hạng mục. Thiên hướng này được thể hiện (trong dữ liệu) xu hướng có tính hệ thống, trong đó một số người dùng sẽ cho điểm đánh giá cao hơn số khác và / hoặc một số hạng mục được chấp nhận tốt hơn (hoặc kém hơn) so với các hạng mục khác.

Hệ tư vẫn sẽ xác định tỉ lệ các điểm đánh giá chịu ảnh hưởng bởi thiên hướng. Xấp xỉ thiên hướng theo thứ tự ưu tiên ảnh hưởng lên điểm đánh giá như sau: Đặt b_{ij} là hệ số bias ảnh hưởng lên điểm đánh giá r_{ij} ,

$$b_{ij} = \mu + b_i + b_j \quad (3.8)$$

trong đó μ là điểm đánh giá trung bình chung cho mọi hạng mục trong hệ thống

b_i và b_j là độ lệch quan sát được của người dùng i và hạng mục j so với

μ .

Ví dụ, giả sử ta muốn ước lượng hệ số bias của người dùng A đối với hạng mục B, biết rằng điểm đánh giá trung bình chung của hệ thống là $\mu = 3.7$. Ngoài ra, hạng mục B có điểm đánh giá cao hơn trung bình là $b_j = 0.5$, A là người thích các hạng mục có đặc tính mà đa số các hạng mục trong hệ thống không có nên điểm đánh giá của A sẽ thấp hơn trung bình là $b_i = 0.3$. Khi đó, điểm đánh giá của A dành cho hạng mục B sẽ là 3.9 ($3.7 + 0.5 - 0.3$). Công thức (3.2) được viết lại như sau:

$$\hat{r}_{ij} = \mu + b_i + b_j + p_i q_j^T \quad (3.9)$$

Như vậy một điểm đánh giá gồm có 4 thành phần: trung bình chung của hệ thống, hệ số bias của người dùng, hệ số bias của hạng mục và tương quan người dùng-hạng mục. Mỗi thành phần chỉ giải thích phần thuộc tính liên quan, nhờ đó mà độ chính xác của hệ thống có sử dụng hệ số bias sẽ cao hơn.

3.4. REGULARIZATION

Khi thực hiện tối ưu lỗi trong quá trình học của thuật toán có thể xảy ra hiện tượng overfitting trong dữ liệu huấn luyện. Đó là hiện tượng “tối ưu quá” khiến mô hình không đủ tổng quát để mô hình hóa dữ liệu mới. Regularization là kỹ thuật để cân bằng hiện tượng này. Có nhiều phương pháp regularization, đơn giản nhất là thêm vào một tham số β , công thức tính lỗi bình phương (II.4) được viết lại như sau:

$$e_{ij}^2 = (r_{ij} - \sum_{k=1}^K p_{ik} q_{kj})^2 + \frac{\beta}{2} \sum_{k=1}^K (\|P\|^2 + \|Q\|^2) \quad (3.10)$$

Tham số β được dùng để kiểm soát độ lớn của các vector đặc trưng của người dùng và vector đặc trưng của hạng mục sao cho \mathbf{P} và \mathbf{Q} xấp xỉ tốt tới \mathbf{R} mà không chứa các số lớn. Trong thực tế, β là một tập các giá trị trong khoảng 0.02. Công thức cập nhật gradient cho các vector p_{ik} và q_{kj} với lỗi bình phương (3.6) được viết lại như sau:

$$p'_{ik} = p_{ik} + \alpha \frac{\partial}{\partial p_{ik}} e_{ij}^2 = p_{ik} + \alpha(2e_{ij}q_{kj} - \beta p_{ik}) \quad (3.11)$$

$$q'_{kj} = q_{kj} + \alpha \frac{\partial}{\partial q_{kj}} e_{ij}^2 = q_{kj} + \alpha(2e_{ij}p_{ik} - \beta q_{kj}) \quad (3.12)$$

Thuật toán phân rã ma trận được tóm tắt như sau:

Thuật toán II.1. Thuật toán cập nhật nhân tử (Lee & Seung)

Input: ma trận \mathbf{R} , α , β , ε (ngưỡng lỗi tối thiểu chấp nhận được)

1: Khởi tạo ngẫu nhiên \mathbf{P} và \mathbf{Q}

2: while $k < \text{iter}$ OR $e < \varepsilon$

3: cập nhật \mathbf{P} theo công thức (3.11)

4: cập nhật \mathbf{Q} theo công thức (3.12)

5: tính lại e theo công thức (3.10)

Output: ma trận $\hat{\mathbf{R}} = \mathbf{P}\mathbf{Q}$

3.5. PHÂN RÃ MA TRẬN KHÔNG ÂM (NMF)

Khi các giá trị trong ma trận đánh giá không âm, ta có mô hình NMF. Ưu điểm của NMF là cho ma trận kết quả mang ý nghĩa trực quan.

Vì không có phần tử nào âm nên ma trận kết quả của quá trình nhân ma trận để xấp xỉ về ma trận ban đầu sẽ không sinh ra số âm và có thể xem đây là một tiến trình sinh ra dữ liệu ban đầu bằng các tổ hợp tuyến tính các đặc tính tiềm ẩn. Phần lớn các dữ liệu đánh giá thường là các ma trận không âm nên mô hình NMF được sử dụng phổ biến.

Về nguyên tắc, thuật toán NMF cũng được thực hiện theo các quy luật cập nhật (II.11) và (II.12), tuy nhiên trong một số trường hợp cần đưa vào các ràng buộc $\mathbf{P} \geq 0$ và $\mathbf{Q} \geq 0$. Thuật toán II.1 được viết lại như sau:

Thuật toán II.2. Thuật toán cập nhật nhân tử -NMF (Lee & Seung)

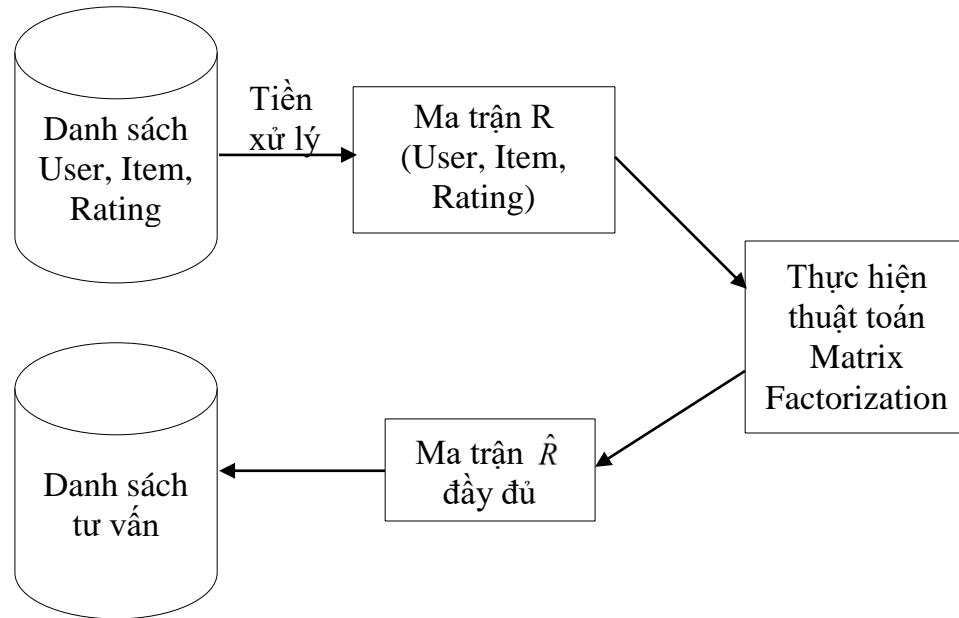
Input: ma trận \mathbf{R} , α , β , ε (ngưỡng lỗi tối thiểu chấp nhận được)

- 1: Khởi tạo ngẫu nhiên \mathbf{P} và \mathbf{Q}
- 2: while $k < \text{iter}$ OR $e < \varepsilon$
- 3: cập nhật \mathbf{P} theo công thức (3.11)
- 4: cập nhật \mathbf{Q} theo công thức (3.12)
- 5: tính lại e theo công thức (3.10)
- 6: if $p_{ij} < 0$ OR $q_{ij} < 0$ then
- 7: $p_{ij} := 0$; $q_{ij} := 0$

Output: ma trận $\hat{\mathbf{R}} = \mathbf{P}\mathbf{Q}$

CHƯƠNG 4 HỆ THỐNG ĐỀ NGHỊ

4.1. MÔ HÌNH HỆ THỐNG:



Hình 4.1. Mô hình hệ tư vấn dựa trên kỹ thuật Phân rã ma trận

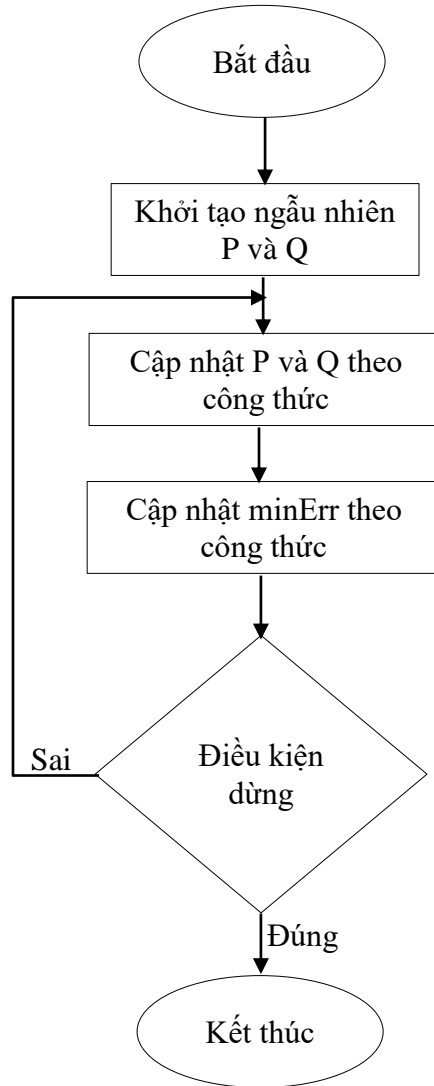
4.2. GIẢI THUẬT CHƯƠNG TRÌNH

4.2.1. ĐẦU VÀO:

- + Ma trận R: Người dùng, Hạng mục, Đánh giá.
- + Số lần học.
- + Số đặc tính quan tâm.
- + Lỗi tối thiểu cho phép.
- + Hệ số alpha.
- + Hệ số beta.

4.2.2. ĐẦU RA:

- + Ma trận \hat{R}
- + Lời tư vấn

4.2.3. LƯU ĐỒ THUẬT TOÁN:**Hình 4.2.** Lưu đồ thuật toán NMF

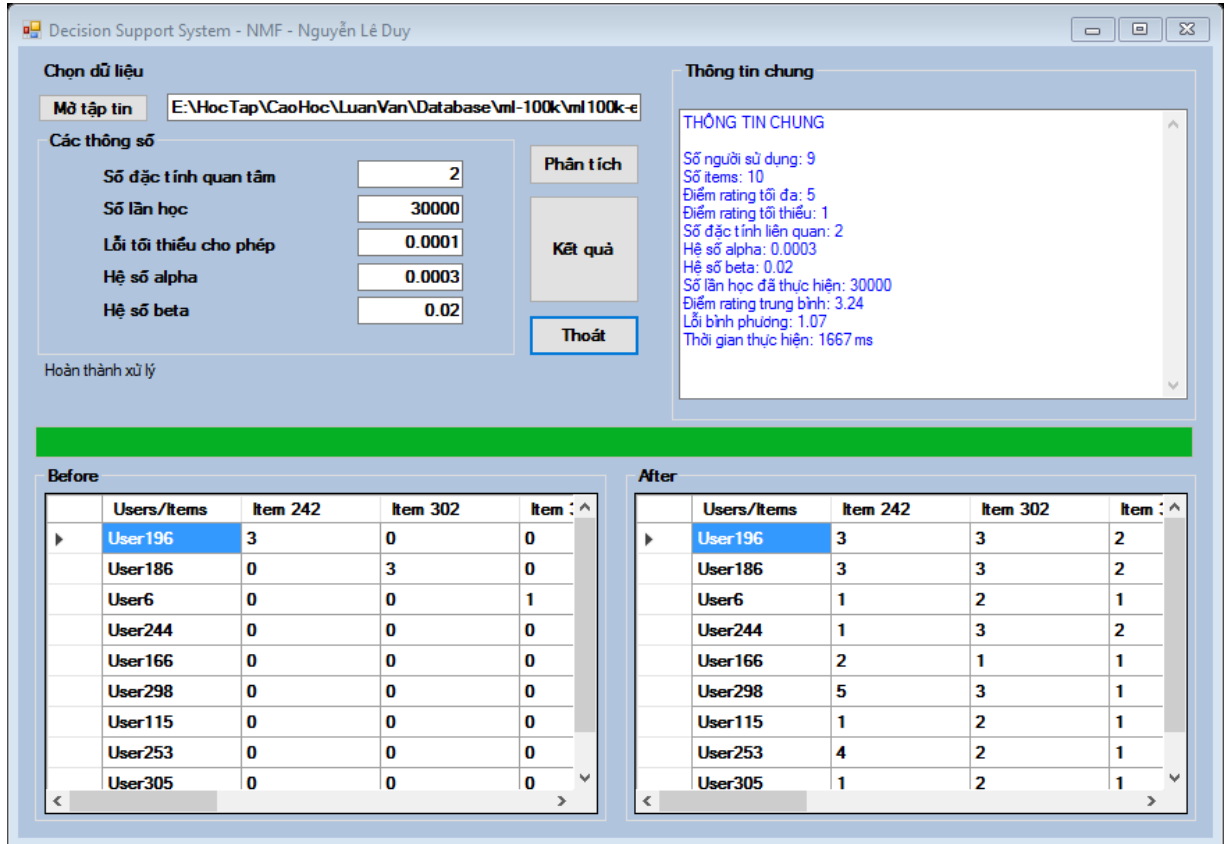
CHƯƠNG 5 HIỆN THỰC VÀ THÍ NGHIỆM

5.1. HIỆN THỰC

5.1.1. GIỚI THIỆU

- Ngôn ngữ lập trình: Microsoft Visual Studio 2012.
- Tổ chức và thuộc tính dữ liệu: dữ liệu được tiền xử lý và lưu trữ dưới dạng database bằng phần mềm Microsoft Access 2013 bao gồm các trường UserId, MovieId, Đánh giá. Chương trình sẽ tiến hành đọc dữ liệu và tổ chức thành ma trận đánh giá. Trong ma trận đánh giá, cột của ma trận đại diện cho một hạng mục, hàng của ma trận đại diện cho một người dùng. Giá trị của các phần tử trong ma trận là điểm đánh giá của người dùng đánh giá hạng mục tương ứng. Các hạng mục chưa được người dùng cho điểm đánh giá có giá trị của phần tử tương ứng là 0. Chương trình áp dụng thuật toán NMF nên điểm đánh giá trong ma trận đánh giá không âm. Giá định các giá trị đánh giá là số nguyên (điểm đánh giá thu thập qua website chẳng hạn). Giá trị đánh giá dương thấp nhất là 1, giá trị đánh giá dương cao nhất là giá trị lớn nhất trong số các giá trị của các phần tử trong ma trận (thường là 5).
- Các tham số: bước nhảy α , tham số regularization β , số lần học *steps*, số đặc tính quan tâm k , lỗi tối thiểu cho phép *minErr* có thể tùy chỉnh. Giá trị mặc định của các tham số này là các giá trị chỉ có tính tham khảo.
- Để đơn giản, thuật toán không sử dụng hệ số bias, độ tin cậy của dữ liệu. Chương trình cài đặt mang tính thử nghiệm nên cũng không ánh xạ các thành phần của ma trận đánh giá sang các file thông tin về người dùng, thông tin về các hạng mục, thông tin về các đặc tính cụ thể (chẳng hạn thể loại phim, diễn viên chính đóng trong phim, ...).

- Điều kiện dừng thuật toán: sau khi đã thực hiện đủ số lần lặp được xác định trong tham số *steps*, hoặc lỗi bình phương < lỗi tối thiểu cho phép được lưu trữ trong tham số *minErr*.
- Ma trận kết quả được xem là xấp xỉ tốt khi các giá trị của các thành phần trong ma trận kết quả, tương ứng với các thành phần đã có giá trị trong ma trận ban đầu, chênh lệch nhau không nhiều. Nếu sau một số lần học, kết quả xấp xỉ chưa được tốt có thể điều chỉnh tham số *steps*.
- Điểm đánh giá dự báo (của các người dùng chưa cho điểm đánh giá cho các hạng mục ban đầu) sẽ được làm tròn trong phần diễn giải kết quả bằng ngôn ngữ tự nhiên. Các giá trị dự báo cao hơn giá trị đánh giá tối đa sẽ được cho điểm đánh giá tối đa. Các giá trị dự báo thấp hơn giá trị đánh giá tối thiểu sẽ được cho điểm đánh giá tối thiểu.
- *Hình 5.1* là giao diện của chương trình cài đặt thử nghiệm.



Hình 5.1. Giao diện chương trình thử nghiệm

trong đó: *dữ liệu đánh giá* là ma trận dữ liệu R ban đầu

kết quả dự đoán là ma trận xấp xỉ $\hat{R} = PQ$

các hệ số tùy chỉnh *Số đặc tính quan tâm*, *Số lần học*, v.v... đã được đề cập ở trên.

5.2. MỘT SỐ THỬ NGHIỆM VÀ NHẬN XÉT VỀ THUẬT TOÁN

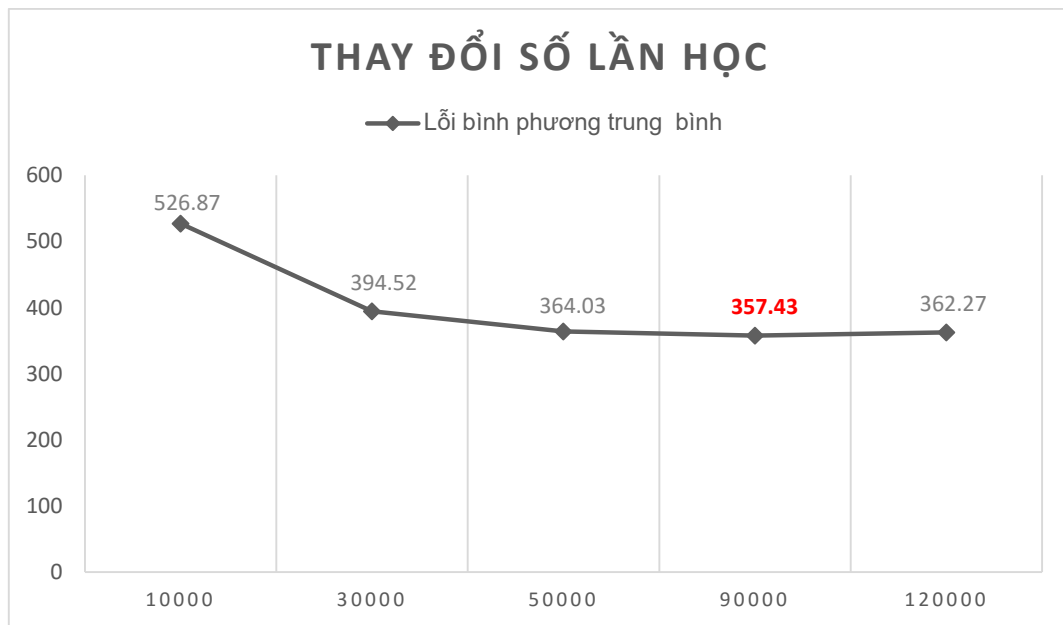
- Thuật toán xấp xỉ tốt các giá trị > 0 trong ma trận dữ liệu R ban đầu. Các phần tử có giá trị bằng 0 trong R đều đã được lấp đầy bởi các giá trị xấp xỉ.

- Giá trị của các phần tử trong ma trận kết quả tương ứng với các phần tử có điểm đánh giá trong ma trận ban đầu có thể xấp xỉ lớn hơn hoặc xấp xỉ nhỏ hơn.
- Điều kiện dừng $\min \text{Err}$ rất khó đạt được vì tổng lỗi tăng tỷ lệ thuận với độ lớn tập dữ liệu. Thuật toán chỉ dừng khi chạy đủ số lần học.
- Lỗi bình phương không phụ thuộc vào số lần học. Do đó cần chọn số lần học phù hợp với tập dữ liệu để có lỗi bình phương tốt nhất.
- Lỗi bình phương không phụ thuộc vào hệ số alpha. Do đó cần chọn số lần học phù hợp với tập dữ liệu để có lỗi bình phương tốt nhất.
- Lỗi bình phương giảm dần khi ta giảm hệ số Beta, như vậy lỗi bình phương phụ thuộc vào hệ số Beta.
- Tùy độ lớn tập dữ liệu, chúng ta sẽ quyết định chọn các thông số phù hợp để tối ưu chi phí tính toán và kết quả.
- Ta thực hiện một số thử nghiệm sau:

5.2.1. TẬP DỮ LIỆU: 10 Người dùng, 712 Hàng mục

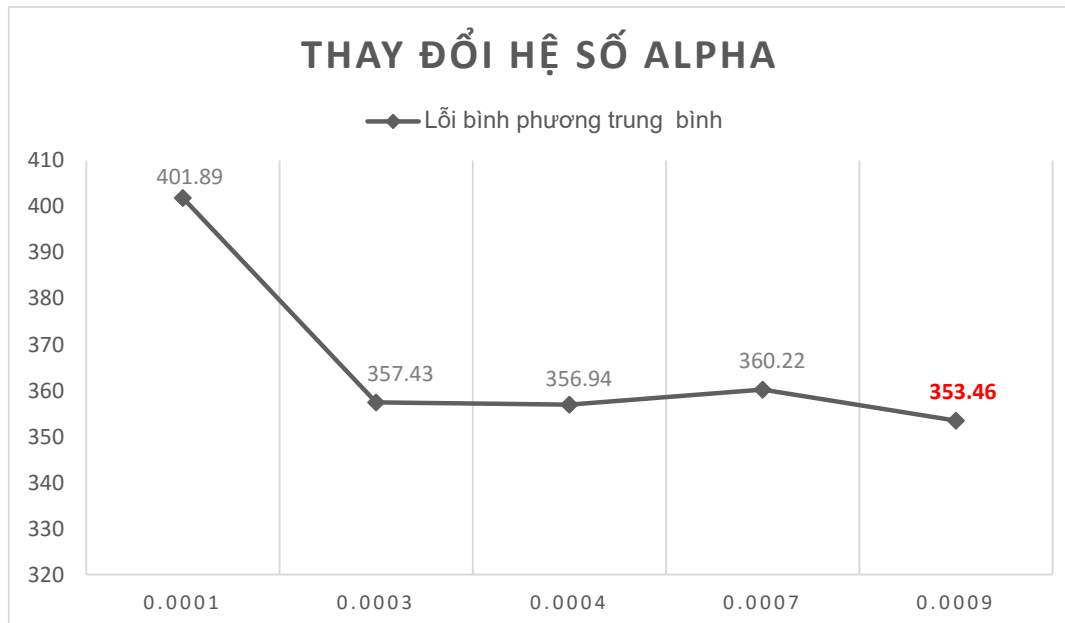
Bảng 5.1. Thay đổi số lần học

Lần thực hiện	Số lần học	Hệ số alpha	Hệ số Beta	Điểm đánh giá trung bình	Lỗi bình phương	Thời gian thực hiện (ms)	Lỗi bình phương trung bình
1	10,000	0.0003	0.02	3.55	547.57	102,333	526.87
2				3.56	543.33	105,091	
3				3.53	489.70	106,271	
1	30,000	0.0003	0.02	3.53	411.91	153,936	394.52
2				3.47	396.99	150,188	
3				3.37	374.67	152,242	
1	50,000	0.0003	0.02	3.37	368.86	195,342	364.03
2				3.50	360.14	195,993	
3				3.52	363.10	193,644	
1	90,000	0.0003	0.02	3.46	363.44	342,734	357.43
2				3.33	357.27	345,090	
3				3.48	351.59	341,960	
1	120,000	0.0003	0.02	3.49	361.94	447,167	362.27
2				3.46	356.67	447,764	
3				3.41	368.21	447,480	

**Hình 5.2.** Thay đổi số lần học.

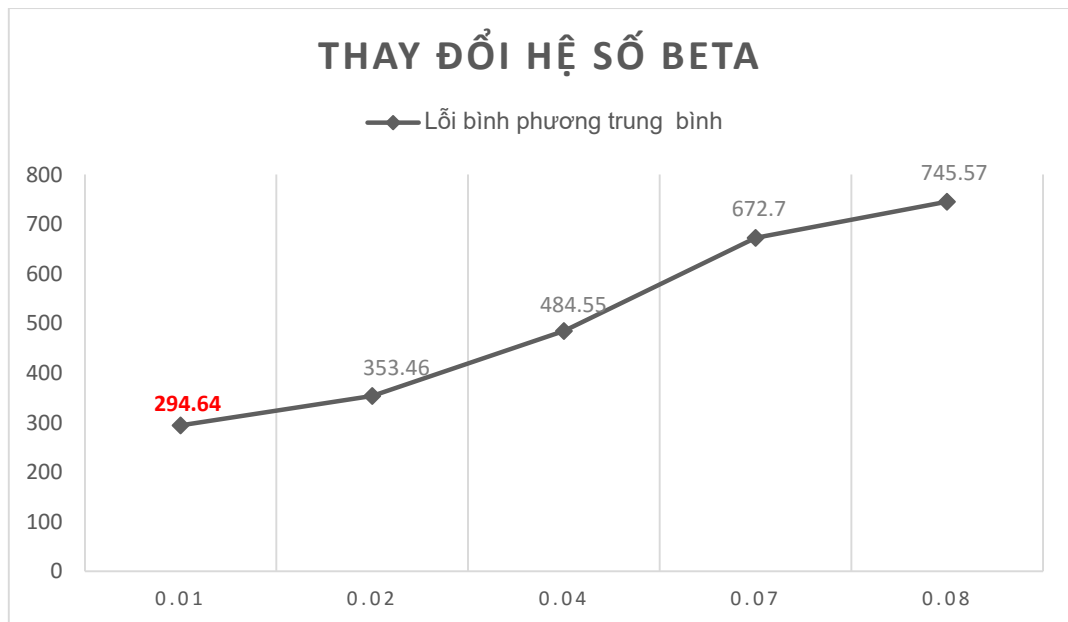
Bảng 5.2. Thay đổi hệ số alpha

Lần thực hiện	Số lần học	Hệ số alpha	Hệ số Beta	Điểm đánh giá trung bình	Lỗi bình phương	Thời gian thực hiện (ms)	Lỗi bình phương trung bình
1	90,000	0.0001	0.02	3.48	368.76	350,603	401.89
2				3.46	416.93	352,471	
3				3.51	419.97	351,596	
1		0.0003	0.02	3.46	363.44	342,734	357.43
2				3.33	357.27	345,090	
3				3.48	351.59	341,960	
1		0.0004	0.02	3.40	355.70	329,350	356.94
2				3.27	352.41	328,679	
3				3.42	362.71	326,688	
1		0.0007	0.02	3.34	365.59	342,856	360.22
2				3.22	354.31	342,584	
3				3.42	360.76	342,728	
1	0.0009	0.02	3.33	348.83	341,579	353.46	
2			3.35	360.44	341,835		
3			3.26	351.10	341,492		

**Hình 5.3.** Thay đổi hệ số alpha

Bảng 5.3. Thay đổi hệ số beta

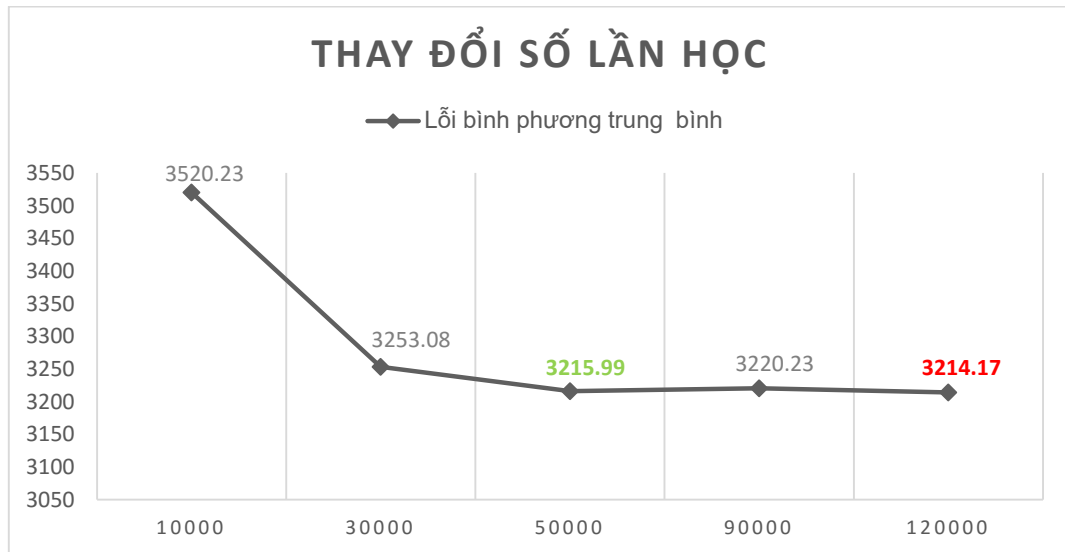
Lần thực hiện	Số lần học	Hệ số alpha	Hệ số Beta	Điểm đánh giá trung bình	Lỗi bình phương	Thời gian thực hiện (ms)	Lỗi bình phương trung bình
1	90,000	0.0009	0.01	3.40	287.30	323,282	294.64
2				3.42	306.16	323,545	
3				3.41	290.46	323,715	
1			0.02	3.33	348.83	341,579	353.46
2				3.35	360.44	341,835	
3				3.26	351.10	341,492	
1			0.04	3.25	481.67	321,031	484.55
2				3.24	485.48	319,749	
3				3.19	486.49	322,149	
1			0.07	3.15	675.38	315,008	672.70
2				3.23	667.90	314,934	
3				3.20	674.82	314,746	
1			0.08	3.19	749.16	354,009	745.57
2				3.2	749.1	356,158	
3				3.16	738.45	355,798	

**Hình 5.4.** Thay đổi hệ số beta

5.2.2. TẬP DỮ LIỆU: 50 Người dùng, 1084 Hạng mục

Bảng 5.4. Thay đổi số lần học

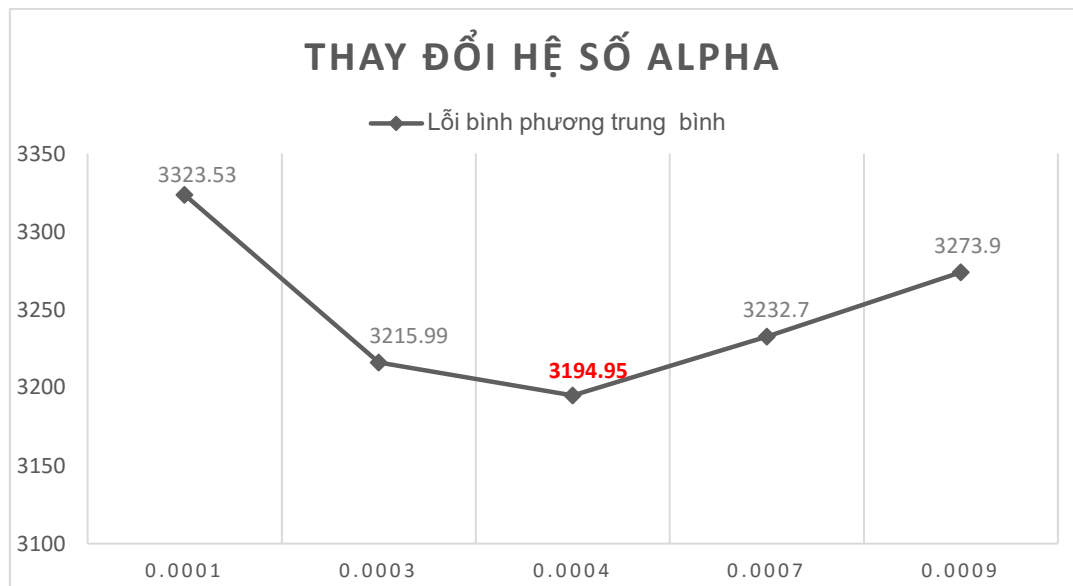
Lần thực hiện	Số lần học	Hệ số alpha	Hệ số Beta	Điểm đánh giá trung bình	Lỗi bình phương	Thời gian thực hiện (ms)	Lỗi bình phương trung bình
1	10,000	0.0003	0.02	3.33	3719.27	560,368	3520.23
2				3.27	3397.26	564,744	
3				3.28	3444.17	568,375	
1	30,000	0.0003	0.02	3.27	3304.99	916,401	3253.08
2				3.29	3219.59	906,253	
3				3.28	3234.66	917,026	
1	50,000	0.0003	0.02	3.26	3217.15	1,268,164	3215.99
2				3.23	3191.09	1,269,136	
3				3.26	3239.73	1,265,584	
1	90,000	0.0003	0.02	3.26	3258.33	1,960,338	3220.23
2				3.25	3204.74	1,973,512	
3				3.22	3197.61	1,970,660	
1	120,000	0.0003	0.02	3.25	3202.26	2,499,054	3214.17
2				3.24	3216.94	2,505,429	
3				3.24	3223.31	2,501,122	



Hình 5.5. Thay đổi số lần học

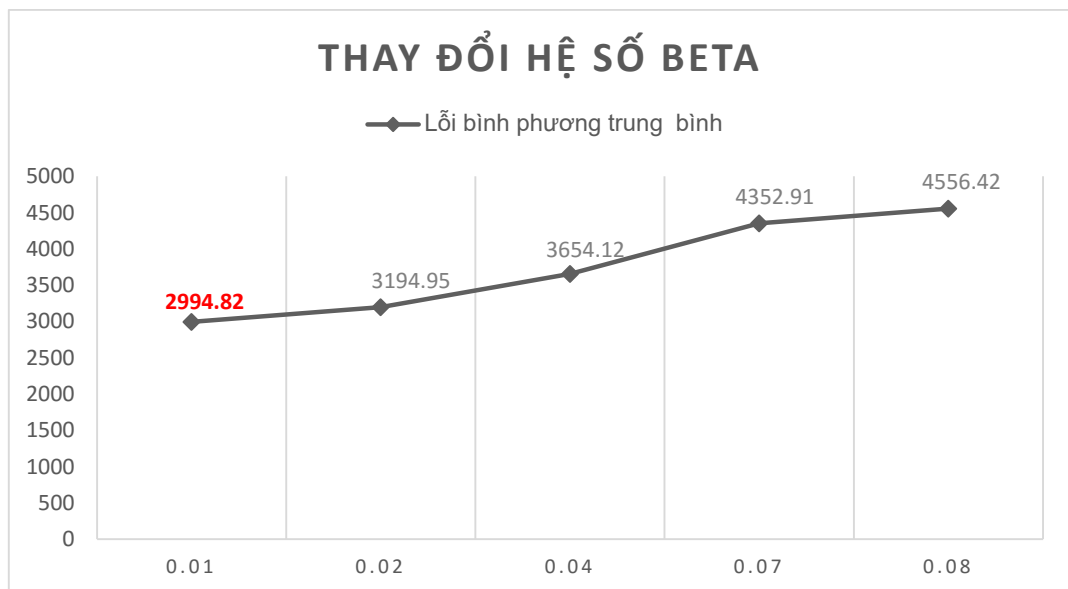
Bảng 5.5. Thay đổi hệ số alpha

Lần thực hiện	Số lần học	Hệ số alpha	Hệ số Beta	Điểm đánh giá trung bình	Lỗi bình phương	Thời gian thực hiện (ms)	Lỗi bình phương trung bình
1	50,000	0.0001	0.02	3.29	3366.80	1,260,412	3323.53
2				3.31	3226.97	1,274,021	
3				3.29	3376.81	1,272,879	
1		0.0003	0.02	3.26	3217.15	1,268,164	3215.99
2				3.23	3191.09	1,269,136	
3				3.26	3239.73	1,265,584	
1		0.0004	0.02	3.23	3209.22	1,274,019	3194.95
2				3.27	3186.24	1,275,907	
3				3.24	3189.39	1,274,080	
1		0.0007	0.02	3.27	3260.79	1,270,552	3232.70
2				3.26	3222.51	1,268,264	
3				3.24	3214.79	1,269,675	
1		0.0009	0.02	3.23	3302.35	1,276,261	3273.90
2				3.24	3298.30	1,277,904	
3				3.23	3221.04	1,274,781	

**Hình 5.6.** Thay đổi hệ số alpha

Bảng 5.6. Thay đổi hệ số beta

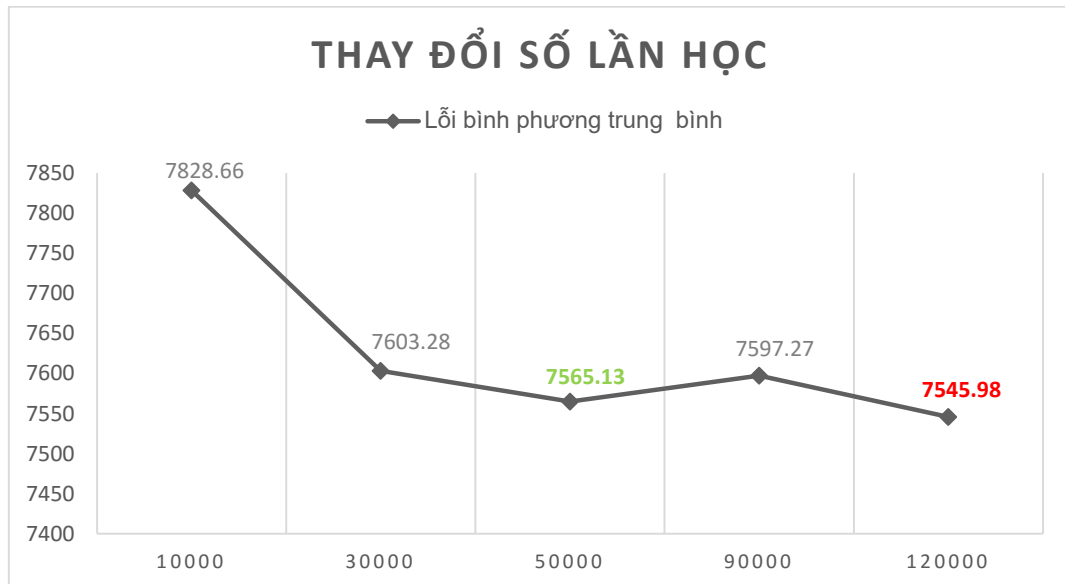
Lần thực hiện	Số lần học	Hệ số alpha	Hệ số Beta	Điểm đánh giá trung bình	Lỗi bình phương	Thời gian thực hiện (ms)	Lỗi bình phương trung bình
1	50,000	0.0004	0.01	3.27	3000.51	1,273,171	2994.82
2				3.26	2987.57	1,274,104	
3				3.27	2996.37	1,271,919	
1			0.02	3.23	3209.22	1,274,019	3194.95
2				3.27	3186.24	1,275,907	
3				3.24	3189.39	1,274,080	
1			0.04	3.19	3638.62	1,274,988	3654.12
2				3.21	3656.29	1,277,150	
3				3.20	3667.46	1,266,903	
1			0.07	3.18	4421.71	1,290,834	4352.91
2				3.18	4300.36	1,300,757	
3				3.16	4336.66	1,289,963	
1			0.08	3.16	4546.89	1,273,829	4556.42
2				3.14	4594.22	1,285,163	
3				3.16	4528.15	1,275,604	

**Hình 5.7.** Thay đổi hệ số beta

5.2.3. TẬP DỮ LIỆU: 100 Người dùng, 1238 Hàng mục

Bảng 5.7. Thay đổi số lần học

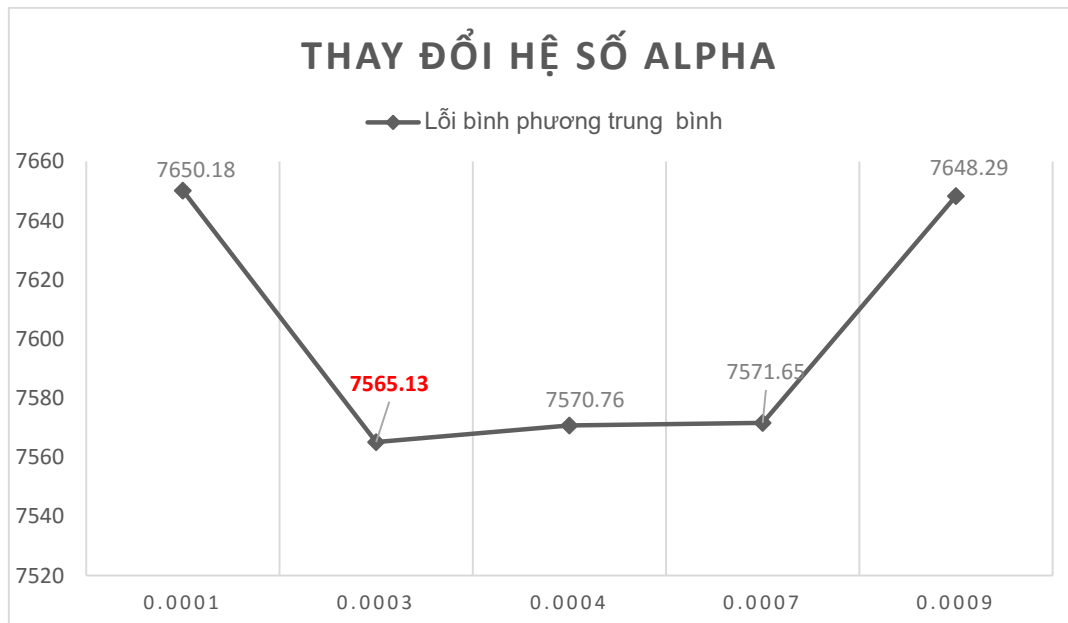
Lần thực hiện	Số lần học	Hệ số alpha	Hệ số Beta	Điểm đánh giá trung bình	Lỗi bình phương	Thời gian thực hiện (ms)	Lỗi bình phương trung bình
1	10,000	0.0003	0.02	3.26	7736.24	2,039,131	7828.66
2				3.24	7797.87	2,038,733	
3				3.29	7951.87	2,162,972	
1	30,000	0.0003	0.02	3.25	7623.24	2,777,156	7603.28
2				3.24	7565.75	2,763,741	
3				3.25	7620.86	2,889,829	
1	50,000	0.0003	0.02	3.21	7572.28	3,565,742	7565.13
2				3.23	7561.10	3,539,292	
3				3.21	7562.00	3,652,869	
1	90,000	0.0003	0.02	3.24	7673.57	5,118,899	7597.27
2				3.20	7583.24	5,080,978	
3				3.23	7534.99	5,199,998	
1	120,000	0.0003	0.02	3.21	7563.87	6,277,840	7545.98
2				3.21	7538.96	6,368,697	
3				3.21	7535.12	6,247,157	



Hình 5.8. Thay đổi số lần học

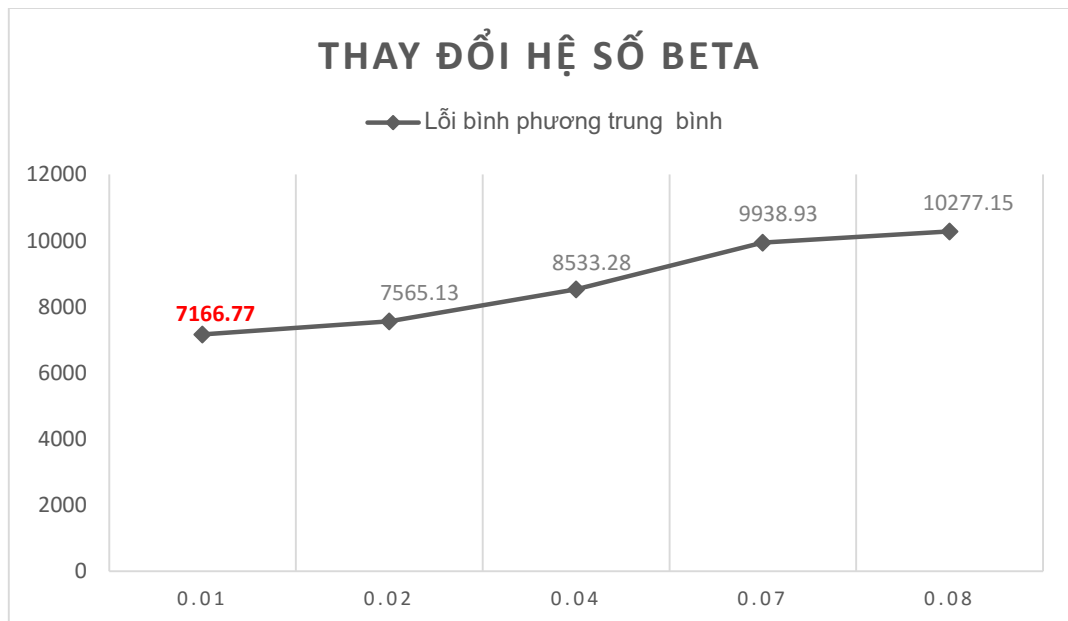
Bảng 5.8. Thay đổi hệ số alpha

Lần thực hiện	Số lần học	Hệ số alpha	Hệ số Beta	Điểm đánh giá trung bình	Lỗi bình phương	Thời gian thực hiện (ms)	Lỗi bình phương trung bình
1	50,000	0.0001	0.02	3.25	7559.34	3,564,284	7650.18
2				3.26	7623.18	3,551,323	
3				3.28	7768.03	3,659,491	
1		0.0003	0.02	3.21	7572.28	3,565,742	7565.13
2				3.23	7561.10	3,539,292	
3				3.21	7562.00	3,652,869	
1		0.0004	0.02	3.27	7566.37	3,552,408	7570.76
2				3.23	7573.28	3,569,563	
3				3.23	7572.62	3,652,290	
1	0.0007	0.02	3.22	7578.35	3,575,039	7571.65	
2			3.21	7579.46	3,540,954		
3			3.23	7557.14	3,663,336		
1	0.0009	0.02	3.25	7708.95	3,573,138	7648.29	
2			3.24	7676.51	3,548,942		
3			3.21	7559.40	3,665,732		

**Hình 5.9.** Thay đổi hệ số alpha

Bảng 5.9. Thay đổi hệ số beta

Lần thực hiện	Số lần học	Hệ số alpha	Hệ số Beta	Điểm đánh giá trung bình	Lỗi bình phương	Thời gian thực hiện (ms)	Lỗi bình phương trung bình
1	50,000	0.0003	0.01	3.25	7244.33	3,327,010	7166.77
2				3.25	7105.93	3,334,796	
3				3.26	7150.06	3,314,516	
1			0.02	3.21	7572.28	3,565,742	7565.13
2				3.23	7561.10	3,539,292	
3				3.21	7562.00	3,652,869	
1			0.04	3.22	8509.02	3,662,597	8533.28
2				3.27	8599.95	3,654,556	
3				3.21	8490.88	3,632,343	
1			0.07	3.21	9894.75	3,753,987	9938.93
2				3.24	9983.35	3,744,621	
3				3.23	9938.69	3,732,557	
1			0.08	3.17	10294.01	3,803,275	10277.15
2				3.19	10247.75	3,802,530	
3				3.19	10289.70	3,777,018	

**Hình 5.10.** Thay đổi hệ số beta

CHƯƠNG 6

KẾT LUẬN

Kỹ thuật phân rã ma trận đã đóng góp đáng kể về mặt phương pháp luận trong việc phát triển các hệ thống tư vấn CF. Các thử nghiệm của nhiều nghiên cứu đều đánh giá kỹ thuật phân rã ma trận cho độ chính xác cao hơn so với các phương pháp đã được áp dụng trước đó như kỹ thuật k-nearest neighbor, v.v... Kỹ thuật phân rã ma trận tương đối dễ cài đặt. Quá trình thử nghiệm trên chương trình kiểm thử cho thấy để kỹ thuật cho được kết quả tốt cần có các hệ số thích hợp.

Dù đã cố gắng tìm tòi nhiều nhưng chắc hẳn luận văn này không tránh khỏi những khiếm khuyết. Qua thực hiện đề tài này, tác giả đã thu được những kết quả sau:

- Tìm hiểu một cách cơ bản về các hệ tư vấn.
 - Kỹ thuật phân rã ma trận và các vấn đề liên quan.
 - Rút ra một số nhận xét cơ bản liên quan đến các hệ số trong kỹ thuật phân rã ma trận thông qua chạy thử nghiệm chương trình cài đặt.
- **Hạn chế của ứng dụng:**
- Chỉ có ý nghĩa về mặt phương pháp luận mà chưa thể xem là một ứng dụng trong thực tế do chưa thực hiện được chức năng ánh xạ giữa quan tâm của user và các thuộc tính vốn có của các item (hệ số features trong hệ thống).
 - Do hạn chế về khả năng xử lý của máy tính nên việc xử lý các nguồn dữ liệu có kích thước lớn của các công ty như Netflix, MovieLens, v.v... chạy rất chậm.
 - Việc tìm kết quả trung bình của n lần chạy cho một tập hệ số còn mang tính thủ công mà chưa tích hợp được trong ứng dụng.
- **Định hướng phát triển ứng dụng:**

- Tích hợp chức năng tính kết quả trung bình trong n lần thử nghiệm cho một tập hệ số.
- Tích hợp chức năng các hệ số với khoảng điều chỉnh phù hợp cho từng hệ số.
- Tích hợp chức năng ánh xạ các thuộc tính quan tâm của người dùng và thuộc tính vốn có của các hạng mục để có thể cho ra kết quả tư vấn cụ thể hơn, nâng tính thực tiễn của ứng dụng.

TÀI LIỆU THAM KHẢO

- [1] Sebastian Riedel, Limin Yao, Andrew McCallum, Benjamin M Marlin . (2014), *Relation extraction with matrix factorization and universal schemas* [online], viewed 17 July 2015, from < <https://people.cs.umass.edu/~lmyao/papers/univ-schema-tacl.pdf>>
- [2] Nicolas Gillis, Stephen A Vavasis. (2014), *Fast and robust recursive algorithms for separable nonnegative matrix factorization* [online], viewed 4 July 2015, from < <http://arxiv.org/pdf/1208.1237.pdf>>
- [3] Hoàng Kiếm. (2012). *Các hệ thống thông tin tri thức hỗ trợ sáng tạo*. Đại Học Quốc Gia TP.HCM
- [4] F. Ricci. (2012). *Content-Based Filtering and Hybrid Systems* [online], viewed 3 February 2015, from <<http://www.inf.unibz.it/~ricci/ISR/slides-2015/14-ContentBased-Hybrid.pdf>>.
- [5] F. Ricci. (2012). *Knowledge-Based Recommender Systems (Part 15)* [online], viewed 3 February 2015, from <http://www.ics.uci.edu/~welling/teaching/CS77Bwinter12/presentations/course_Ricci/15-KnowledgeBased.pdf>.
- [6] K. Nykjaer (2012). *How does the Amazon recommendation system work? – Analyze the algorithm and make a prototype that visualizes the algorithm* [online], viewed 15 February 2015, from <<https://kunuk.wordpress.com/2012/03/04/how-does-the-amazon-recommendation-system-work-analyze-the-algorithm-and-make-a-prototype-that-visualizes-the-algorithm/>>.

- [7] F. Ricci et al. (eds.) (2011). *Recommender Systems Handbook*. DOI 10.1007/978-0-387-85820-3_1, © Springer Science+Business Media [online], viewed 1 January 2015, from <http://www.cs.bme.hu/nagyadat/Recommender_systems_handbook.pdf>.
- [8] D. Asanov. (2010). *Algorithms and Methods in Recommender Systems* [online].viewed 11 March 2015, from <https://www.snet.tu-berlin.de/fileadmin/fg220/courses/SS11/snet-project/recommender-systems_asanov.pdf>.
- [9] G. Semeraro et al (2010). *Content-based Recommender Systems: problems, challenges and research directions*. [online]. Viewed 6 January 2015, from <http://ls13-www.cs.tu-dortmund.de/homepage/ITWP2010/slides/semeraro.pdf>
- [10] Y. Koren, R. Bell, C. Volinsky (2009). *Matrix factorization techniques for recommender systems*. Yahoo Research, AT&T Lab-Research [online]. Viewed 30 January 2015, from <http://www2.research.att.com/~volinsky/papers/ieeecomputer.pdf>
- [11] G. Adomavicius, A. Tuzhilin (2005). *Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions*. IEEE Transactions on Knowledge and Data Engineering, 17(6):734–749 (online), viewed 17 January 2015, from <<http://pages.stern.nyu.edu/~atuzhili/pdf/TKDE-Paper-as-Printed.pdf>>.
- [12] D. D. Lee, H. S. Seung (2001). *Algorithms for nonnegative matrix factorization*. Neural Information Processing Systems 13, pp 556-562 [online], viewed 1 March 2015, from <<http://www.dm.unibo.it/~simoncin/nmfconverge.pdf>>.

[13] R. Burke (2000). *Knowledge-based Recommender Systems*. In A. Kent (ed.), [online], viewed 9 January 2015, from <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.21.6029&rep=rep1&type=pdf>>.

[14] M. J. Pazzani (1999). *A framework for collaborative, content-based and demographic filtering*. *Artificial Intelligence Review*, 13(5-6):393–408, [online], viewed 27 February 2015, from <<http://www.ics.uci.edu/~pazzani/Publications/AIREVIEW.pdf>>