ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

NGUYỄN THÚY NGỌC

Tên đề tài:

TĂNG CƯỜNG SỰ PHONG PHÚ VỀ CHẤT LƯỢNG CỦA HỆ THỐNG TƯ VẤN

LUẬN VĂN THẠC SĨ
NGÀNH HỆ THỐNG THÔNG TIN

THÀNH PHỐ HỒ CHÍ MINH – 2013

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

NGUYỄN THÚY NGỌC

Tên đề tài:

TĂNG CƯỜNG SỰ PHONG PHÚ VỀ CHẤT LƯỢNG CỦA HỆ THỐNG TƯ VẤN

LUẬN VĂN THẠC SĨ NGÀNH HỆ THỐNG THÔNG TIN

Mã số: 60 48 05

NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. NGUYỄN AN TẾ

THÀNH PHỐ HỒ CHÍ MINH – 2013

LỜI CẢM ƠN

Đầu tiên, tôi xin bày tỏ lòng biết ơn chân thành và sâu sắc nhất đến Thầy Nguyễn An Tế – giáo viên hướng dẫn luận văn. Trong quá trình tìm hiểu và nghiên cứu, tôi đã gặp rất nhiều khó khăn, nhưng nhờ Thầy đã luôn động viên, hết lòng hướng dẫn và giúp đỡ nên tôi đã hoàn thành luận văn này.

Tôi xin gửi lời cảm ơn chân thành đến quý Thầy/Cô – Trường Đại học Khoa học tự nhiên TP. HCM đã truyền đạt những kiến thức quý báu cho tôi trong quá trình học tập. Đồng thời, tôi cũng xin gửi lời cảm ơn đến Ban chủ nhiệm khoa Công nghệ thông tin – Trường Đại học Sư phạm TP. HCM đã hỗ trợ và tạo điều kiện cho tôi trong thời gian qua.

Cuối cùng, tôi xin bày tỏ lòng biết ơn sâu sắc đối với gia đình đã luôn động viên và giúp đỡ tôi trong suốt quá trình học tập cũng như thực hiện luận văn.

Học viên thực hiện

Nguyễn Thúy Ngọc

TP. HCM tháng 09 năm 2013

MỤC LỤC

DANH MỤC THUẬT NGỮ VÀ VIẾT TẮT

α-Community Spaces Model	αCSM
Collaborative Filtering	CF
Content-based Filtering	CbF
Context-aware Collaborative Filtering	CACF
Context-aware Matrix Factorization	CAMF
Context-aware Recommendation based on Enrichment from Ordered Multi-criteria Communities	CREOC
Context-aware Recommendation based on Enrichment from Similar Multi-criteria Communities	CRESC
Context-aware Recommendation based on Multi-criteria Communities	CRMC
Context-aware Recommender System	CARS
Matrix Factorization	MF
Recommender System	RS

DANH MỤC CÁC BẢNG

DANH MỤC HÌNH VẼ

Chương 1. **Giới thiệu**

Chương 1 sẽ trình bày một số vấn đề đã thúc đẩy luận văn đi tìm hiểu và tiến hành nghiên cứu về các hệ thống tư vấn theo ngữ cảnh. Tiếp theo đó, chương mở đầu này cũng sẽ giới thiệu những mục tiêu, nội dung nghiên cứu và tóm tắt những kết quả đạt được của luận văn.

1.1 Đặt vấn đề

Từ nhiều năm nay, sự ra đời của những search engines như Google đã giúp chúng ta giải quyết nhu cầu về thông tin trong nhiều lãnh vực của cuộc sống hằng ngày. Thông thường, sau khi cung cấp một vài từ khóa thể hiện nhu cầu thông tin của mình, người sử dụng (NSD) sẽ nhận được một danh sách những thông tin có liên quan. Lúc này NSD phải đối mặt với một vấn đề mới nảy sinh: danh sách kết quả trả về chứa quá nhiều thông tin (có khi lên đến hàng triệu thông tin) và họ phải tốn nhiều thời gian, công sức để loại bỏ những thông tin không phù hợp và chọn lọc lại những gì thật sự có ích. NSD cũng có thể tinh chế lại tập từ khóa để thu hẹp danh sách thông tin kết quả, nhưng vấn đề mấu chốt ở đây là các hệ thống đã *đồng nhất nhu cầu của mọi cá nhân*. Nhìn chung, danh sách kết quả chứa những thông tin có liên quan đến tập từ khóa nhưng không ít trong số đó là không phù hợp với NSD. Ví dụ, nếu cùng thời điểm mà một giảng viên và một sinh viên đều nhập vào từ khóa “Information Retrieval” thì cả hai sẽ nhận được danh sách kết quả giống như nhau, trong đó có một phần tài liệu cơ bản chỉ phù hợp với sinh viên và một phần khác gồm những tài liệu nâng cao chỉ phù hợp với trình độ của giảng viên.

Hiện nay, các hệ thống tư vấn (*Recommender Systems* – RSs) đang phát triển rất mạnh nhằm cung cấp cho NSD những thông tin, hàng hóa, hay dịch vụ, … (gọi chung là *items*) một cách phù hợp với những đặc trưng riêng của từng cá nhân [14], [19]. Ví dụ, các hệ thống như Amazon, MovieLens có chức năng cung cấp những thông tin về sách, phim ảnh theo nhu cầu, mối quan tâm hay sở thích của từng NSD. Đa số các RS hiện nay được xây dựng theo ba cách tiếp cận chính: dựa trên nội dung (*Content-based Filtering – CbF*), dựa trên sự cộng tác (*Collaborative Filtering – CF*) và dựa trên việc lai ghép các phương pháp với nhau (*Hybrid Approach*).

Trong cách tiếp cận dựa trên nội dung CbF, mỗi NSD sở hữu một hồ sơ đặc trưng (*profile*) mà tùy theo lãnh vực ứng dụng sẽ bao gồm những thông tin khác nhau dùng để mô tả về mình như: tuổi, nghề nghiệp, mối quan tâm, nhu cầu, sở thích, … Sau đó, NSD sẽ thường xuyên nhận được những items phù hợp với đặc trưng của cá nhân nhờ vào việc hệ thống đã thực hiện sự so khớp giữa profile của NSD và các items. Ngược lại, NSD phải cung cấp cho hệ thống những ý kiến phản hồi (*feedback*) hay những đánh giá (*ratings*) trên các items đã nhận được để hệ thống có thể cập nhật profile một cách đúng đắn [7], [14].

Cách tiếp cận CbF có ưu điểm là không còn đánh đồng nhu cầu của mọi cá nhân như trong lãnh vực *Information Retrieval*, nhưng lại có khuyết điểm về lối mòn trong khai thác, nghĩa là một khi profile đã "ổn định" thì NSD chỉ nhận được những items có liên quan đến những gì được mô tả trong profile và không có cơ hội khám phá những lãnh vực mới mà có thể cũng rất đáng quan tâm.

Hiện nay, cách tiếp cận dựa trên sự cộng tác CF đang phát triển rất mạnh và chiếm ưu thế so với CbF [6], [7], [14]. Cách tiếp cận này dựa trên một ma trận đánh giá R (*ratings matrix*), gồm các r_{ui} chính là điểm số của NSD u đánh giá trên item i. Với mỗi NSD, hệ thống sẽ xác định một cộng đồng cho người này dựa trên độ tương đồng giữa vectơ (dòng) điểm số của NSD đó với các vectơ điểm số của

những người khác trong ma trận *R*. Sau khi xác định cộng đồng cho NSD *u*, người này sẽ được hệ thống tư vấn những items mà cộng đồng của mình cho điểm cao.

Ưu điểm của CF chính là sự chia sẻ thông tin giữa những NSD, đồng thời hệ thống có thể bỏ qua công đoạn phân tích nội dung, không cần phải so khớp giữa profile và items mà chỉ cần dựa trên ma trận đánh giá. Từ đó, CF đã giải quyết được vấn đề lối mòn trong khai thác, bởi vì thông qua cộng đồng của mình, NSD có thể khám phá thêm những chủ đề mới, lãnh vực mới, chưa được thể hiện trong profile mà có thể chúng rất thú vị.

Nguyên tắc của cách tiếp cận lai ghép là sự kết hợp của nhiều phương pháp tư vấn khác nhau để tạo ra sự hỗ trợ qua lại giữa những ưu và khuyết điểm của từng phương pháp riêng lẻ. Ví dụ, với ưu điểm là khả năng chia sẻ thông tin, phương pháp CF có thể khắc phục được khuyết điểm của phương pháp CbF về lối mòn trong khai thác [7].

Tuy nhiên, phần lớn những cách tiếp cận của các RS hiện nay như CbF, CF hay cách tiếp cận lai ghép chỉ quan tâm chủ yếu đến mối quan hệ giữa hai đối tượng là NSD và items mà không xét đến yếu tố ngữ cảnh của những mối quan hệ đó như: thời gian, không gian, mục tiêu, tâm trạng, sự kiện, … Thực tế cho thấy những hành vi, quyết định hay sở thích của NSD có thể bị tác động bởi các yếu tố ngữ cảnh. Ví dụ, mặc dù một NSD có sở thích đi du lịch ở biển nhưng nếu đang là mùa đông thì hệ thống cần tư vấn cho NSD một địa điểm du lịch khác như lên núi trượt tuyết [1], [4], [20], hoặc khi đang làm việc thì một NSD rất quan tâm đến các sách, tạp chí thuộc những chủ đề về chuyên môn nhưng khi cần thư giãn thì NSD đó lại quan tâm đến các tài liệu về chủ đề thể thao, âm nhạc, …

Trong những năm gần đây, các hệ thống tư vấn theo ngữ cảnh (*Context-Aware Recommender Systems* - CARSs) đã bắt đầu được nghiên cứu nhiều với mục đích để giải quyết mặt hạn chế của các RS, đó là tạo ra danh sách tư vấn mà không quan tâm đến yếu tố ngữ cảnh [5]. Việc ***tích hợp yếu tố ngữ cảnh*** vào quá trình tư vấn sẽ

giúp các hệ thống có thể đưa ra những gợi ý phù hợp hơn cũng như thỏa mãn nhu cầu của NSD trong những điều kiện hay những hoàn cảnh khác nhau.

Bên cạnh đó, một vấn đề cũng rất đáng quan tâm là phần lớn các phương pháp tư vấn áp dụng trong CARSs dựa trên cách tiếp cận CF chỉ xây dựng cộng đồng theo một tiêu chí duy nhất là ratings của NSD. Điều đó có nghĩa là mỗi NSD chỉ thuộc về một cộng đồng duy nhất được thành lập từ việc so khớp các ratings nhưng trên thực tế, NSD có thể thuộc rất nhiều cộng đồng khác nhau liên quan đến nghề nghiệp, nhóm tuổi, sở thích, … Việc xây dựng cộng đồng chỉ dựa trên một tiêu chí duy nhất là ratings sẽ dẫn đến tình trạng chất lượng của tư vấn bị ảnh hưởng xấu nếu như số lượng ratings của NSD quá ít, không đủ để hệ thống thành lập cộng đồng. Nói cụ thể hơn, thông thường một NSD thường cung cấp rất ít (vài chục) ratings so với số lượng hàng ngàn items và do đó, ma trận đánh giá *R* nhìn chung là rất thưa (***sparsity problem***). Khi đó, hai NSD có thể không có ratings trên những items chung và như vậy, hệ thống không có cơ sở để đánh giá họ có cùng cộng đồng hay không. Vấn đề này càng trở nên trầm trọng hơn khi việc bổ sung thêm yếu tố ngữ cảnh sẽ biến ma trận đánh giá thành một ma trận 3D với tổng số "ô" bằng tích số của số lượng NSD, số lượng items và số lượng ngữ cảnh trong khi số lượng ratings của NSD nhìn chung là không thay đổi (xem phần 2.2) vì NSD thường hay có thói quen chỉ đánh giá items một lần (ngữ cảnh đầu tiên gặp phải). Trong những trường hợp như trên, luận văn cho rằng việc xây dựng và khai thác những cộng đồng theo nhiều tiêu chí khác nhau (cộng đồng đa tiêu chí) trong CARSs có thể sẽ giúp cải thiện chất lượng tư vấn.

1.2 Mục tiêu của luận văn

Dựa trên những hiện trạng tóm tắt ở trên, mục tiêu nghiên cứu của luận văn là tăng cường sự phong phú về chất lượng của các RS. Sự phong phú ở đây được thể hiện trước hết qua việc xem xét đến yếu tố ngữ cảnh trong quá trình tư vấn thay vì độc lập với ngữ cảnh như các RS truyền thống. Ví dụ, một thông tin tư vấn có thể

được NSD đánh giá cao trong một ngữ cảnh này nhưng rất có thể không phù hợp với NSD trong một ngữ cảnh khác, hoặc ngược lại. Bên cạnh đó, sự phong phú của RS còn được tăng cường bằng việc thay vì chỉ dựa trên các cộng đồng được xây dựng từ ma trận đánh giá, luận văn sẽ hướng đến việc khai thác mối quan hệ giữa các cộng đồng đa tiêu chí và ngữ cảnh cho thuật toán tư vấn vì vai trò và mức độ ảnh hưởng của các cộng đồng trong từng ngữ cảnh cụ thể không phải lúc nào cũng giống như nhau. Ví dụ trong ngữ cảnh này thì cộng đồng gia đình, họ hàng có thể có ảnh hưởng lớn nhất đến quyết định hay đánh giá của NSD nhưng trong ngữ cảnh khác thì cộng đồng nghề nghiệp lại có vai trò quan trọng hơn.

1.3 Nội dung thực hiện

Nhằm đạt được những mục tiêu đã nêu, về mặt lý thuyết, luận văn sẽ tiến hành nghiên cứu những công trình, những thuật giải có liên quan đến các yếu tố ngữ cảnh trong quá trình tư vấn. Những công việc cụ thể bao gồm:

- tìm hiểu hiện trạng về các RS, phân tích ưu và khuyết điểm của những phương pháp được áp dụng phổ biến.
- tìm hiểu về các CARS cùng với các thuật toán tư vấn theo ngữ cảnh đạt hiệu quả cao như: item splitting, matrix factorization, contextualizing users' latent features, …
- nghiên cứu khả năng khai thác các cộng đồng đa tiêu chí thông qua việc xem xét thêm mối quan hệ giữa yếu tố ngữ cảnh và các cộng đồng.
- xây dựng những thuật giải tư vấn theo ngữ cảnh dựa trên các cộng đồng đa tiêu chí với mối quan tâm đặc biệt đến vấn đề dữ liệu thưa.

Về mặt thực nghiệm, những thuật giải tư vấn do luận văn đề xuất sẽ được thử nghiệm và đánh giá theo phương pháp *offline* [22], nghĩa là thuật giải sẽ được thử nghiệm trên một bộ dữ liệu mẫu trong lãnh vực hệ thống tư vấn, ví dụ như MovieLens [15] được xem là một bộ dữ liệu "chuẩn" để tiến hành những thực

nghiệm trong lãnh vực của luận văn. Quá trình thử nghiệm của luận văn sẽ được tiến hành theo những bước chính như sau:

- chuẩn bị dữ liệu
- xây dựng quy trình thử nghiệm
- tiến hành thử nghiệm theo phương pháp offline để phân tích và đánh giá.

1.4 Tóm tắt những đóng góp của luận văn

Luận văn đã sử dụng những ký hiệu và khái niệm cơ bản về đại số của mô hình không gian cộng đồng đa tiêu chí (α-*Community Spaces Model* – αCSM) [16] và từ đó đề xuất mở rộng bằng cách bổ sung yếu tố ngữ cảnh và định nghĩa một thứ tự ưu tiên cho các tiêu chí được khai thác trong từng ngữ cảnh cụ thể. Trên cơ sở đó, luận văn đã đề xuất ba thuật toán có thể được áp dụng trong tư vấn theo ngữ cảnh là: CRMC, CRESC và CREOC. Các thuật toán CRESC và CREOC có khả năng làm giảm sự tác động xấu của vấn đề dữ liệu thưa để có thể tạo ra những danh sách tư vấn đạt kết quả tốt hơn so với một số phương pháp tư vấn theo ngữ cảnh.

Luận văn đã tiến hành thử nghiệm các phương pháp đề xuất cũng như phân tích kết quả ở nhiều khía cạnh khác nhau. Thông qua kết quả thực nghiệm, luận văn đã biện minh được giả định là vai trò hay mức độ ảnh hưởng của các cộng đồng được xây dựng theo những tiêu chí khác nhau trong từng ngữ cảnh cụ thể sẽ bị thay đổi. Đồng thời luận văn cũng phân tích, làm rõ ưu và khuyết điểm của các thuật toán đề xuất để từ đó gợi ý cách chọn áp dụng các thuật toán CRMC, CRESC và CREOC sao cho có hiệu quả nhất.

Luận văn được thực hiện trong khoảng thời gian từ tháng 10/2012 đến tháng 8/2013. Những kết quả chính của luận văn đã được chấp nhận đăng tải trong kỷ yếu và sẽ được trình bày tại 3 hội nghị khoa học quốc tế ở Mỹ vào tháng 10 và tháng 12 năm 2013 (xem trang 77).

Báo cáo của luận văn được trình bày thành năm chương như sau:

Chương 1 giới thiệu tổng quan về những vấn đề của RS, nêu lên mục tiêu, nội dung nghiên cứu và những kết quả đạt được của luận văn.

Chương 2 trình bày hiện trạng của các RS truyền thống với những cách tiếp cận như: lọc theo nội dung, lọc cộng tác và lai ghép các phương pháp cũng như trình bày ba mô hình thường được áp dụng trong các CARS gồm: pre-filtering, post-filtering và contextual modeling.

Chương 3 giới thiệu ba thuật toán đề xuất CRMC, CRESC và CREOC được xây dựng từ việc mở rộng mô hình αCSM.

Chương 4 trình bày quá trình tiến hành cũng như kết quả thử nghiệm của các thuật toán đề xuất trên bộ dữ liệu MovieLens. Những đánh giá và phân tích cũng được trình bày nhằm giúp cho việc lựa chọn áp dụng các thuật toán đề xuất.

Chương 5 là phần kết luận và nêu lên một số hướng phát triển trong tương lai của luận văn.

Chương 2. **Hiện trạng về hệ thống tư vấn**

Chương 2 sẽ trình bày hiện trạng nghiên cứu có liên quan đến luận văn, bao gồm phần tổng quan về các hệ thống tư vấn và những cách tiếp cận chính thường được áp dụng trong những hệ thống này. Trong số đó, phương pháp Matrix Factorization được đánh giá là có hiệu quả cao nhất. Đặc biệt, chương này sẽ giới thiệu những cách tiếp cận về hình thức tư vấn theo ngữ cảnh.

2.1 Các hệ thống tư vấn

Sự phát triển của Internet đã đem lại rất nhiều tiện nghi cho cuộc sống của con người nhưng bên cạnh đó, nó cũng là một trong những nguyên nhân chính gây ra vấn đề quá tải về thông tin (information overload problem). Ví dụ, một website thương mại điện tử có thể giới thiệu và gợi ý cho NSD hàng trăm mặt hàng thuộc nhiều chủng loại khác nhau. Kết quả là NSD thường xuyên phải đối mặt với tình trạng lúng túng, không biết nên mua mặt hàng nào, xem phim gì, tham khảo tài liệu nào, … là phù hợp nhất với nhu cầu của mình. Thực tế là với quá nhiều thông tin được cung cấp, NSD sẽ cảm thấy rất khó khăn, phải mất nhiều thời gian và công sức để loại bỏ những thông tin không thật sự cần thiết và chọn ra những gì phù hợp nhất cho bản thân mình. Xuất phát từ hiện trạng như trên mà các hệ thống tư vấn (RS) đã được xây dựng để gợi ý cho NSD những mặt hàng, dịch vụ hay thông tin (gọi chung là *items*) phù hợp nhất với nhu cầu hay sở thích của từng cá nhân [19].

Trong số các RS hiện nay, có hai phương pháp được áp dụng phổ biến nhất, đó là: lọc theo nội dung và lọc cộng tác. Bên cạnh đó, hướng tiếp cận lai ghép [7] cũng được áp dụng dựa vào sự kết hợp của hai phương pháp vừa nêu.

2.1.1 *Phương pháp lọc theo nội dung*

Phương pháp *lọc theo nội dung* (CbF) gợi ý cho NSD những items tương đồng về mặt nội dung với những items mà NSD đó đã ưa thích, đã mua, đã xem hay đã cho đánh giá (rating) cao trong quá khứ [14]. Trong cách tiếp cận này, mỗi NSD sẽ sở hữu một hồ sơ đặc trưng (*profile*) mô tả chủ yếu những sở thích hay sự quan tâm của NSD tùy theo phạm vi của ứng dụng như danh sách các thể loại phim, thể loại nhạc yêu thích hay những lĩnh vực nghiên cứu đang quan tâm. Sau đó, hệ thống sẽ so khớp thông tin mô tả trong profile và phần biểu diễn nội dung của item để dự đoán mức độ ưa thích của NSD dành cho item đó.

Có rất nhiều mô hình có thể được áp dụng cho CbF và trong phần này, luận văn sẽ trình bày một phương pháp khá đơn giản và phổ biến nhất, đó là cách tiếp cận dựa theo mô hình không gian vectơ, kế thừa từ lĩnh vực tìm kiếm thông tin (*Information Retrieval*). Trong cách tiếp cận này, mỗi item và mỗi profile sẽ lần lượt được biểu diễn bằng các vectơ. Để ước lượng độ tương đồng giữa hai vectơ biểu diễn cho item và profile, phương pháp này thường sử dụng công thức cosine của góc tạo bởi hai vectơ [14]. Công thức tính cosine của góc tạo bởi vectơ item $\vec{i}$ và vectơ profile $\vec{p}$ như sau:

$$\text{cosine}(\vec{i}, \vec{p}) = \frac{\vec{i}.\vec{p}}{\| \vec{i} \| . \| \vec{p} \|} \tag{2.1}$$

Nếu giá trị cosine tiến về 1 thì hai vectơ này rất khớp với nhau. Điều đó có nghĩa là item rất phù hợp với NSD, và item này sẽ được tư vấn cho NSD. Nếu giá trị cosine tiến về 0 thì vectơ nội dung của item vuông góc với vectơ profile của NSD. Điều này có nghĩa là item và sở thích của NSD không có mối liên hệ với

nhau. Cuối cùng, nếu giá trị cosine tiến về -1 thì item *i* trái ngược hoàn toàn với nhu cầu hay sở thích của NSD.

Ví dụ, ta có vectơ profile biểu diễn mức độ yêu thích của NSD *u* dành cho các thể loại phim như sau (giá trị dương là yêu thích và giá trị âm là không thích):

	Action	*Adventure*	*Children*	*Comedy*	*Crime*	*Drama*
u	0.5	0.4	-0.2	0.7	-0.4	-0.6

Tương tự, vectơ biểu diễn nội dung của phim Home Alone được biểu diễn như bên dưới. Đây là vectơ nhị phân với hai giá trị 0 và 1.

	Action	*Adventure*	*Children*	*Comedy*	*Crime*	*Drama*
Home Alone	1	0	1	1	0	0

Giá trị của ô (Home Alone, Action) = 1 có nghĩa là phim Home Alone thuộc thể loại phim hành động, ngược lại (Home Alone, Drama) = 0 có nghĩa là phim này không thuộc thể loại phim tình cảm. Từ đó, để xác định phim Home Alone có phù hợp để tư vấn cho NSD *u* hay không thì hệ thống CbF sẽ tính độ tương đồng của hai vectơ thông qua độ đo cosine như sau:

$$\text{cosine}\,(u, \text{Home Alone}) = \frac{0.5 - 0.2 + 0.7}{\sqrt{1.46}\sqrt{3}} = 0.48$$

Các giá trị trong vectơ profile của NSD có thể được cập nhật dựa trên những ratings của NSD đối với những items đã được hệ thống từng tư vấn [14].

Phương pháp CbF thường đạt hiệu quả cao khi việc trích chọn và biểu diễn các đặc trưng của items thành vectơ được thực hiện một cách hợp lý. Tuy nhiên, thách thức lớn của kỹ thuật này là phải làm thế nào để trích chọn được các đặc trưng của item, trong khi đối với một số lĩnh vực ứng dụng như âm nhạc hay tranh ảnh thì việc phân tích và biểu diễn nội dung của item không phải lúc nào cũng được thực hiện một cách dễ dàng.

Bên cạnh đó, các items được tư vấn dựa theo CbF rất ít khi thể hiện được tính đa dạng và bất ngờ [22] vì NSD có thể dễ dàng hình dung trước hệ thống sẽ tư vấn những items có nội dung tương tự như những items mà mình đã thích trước đó. Ví dụ, nếu NSD thích phim tình cảm thì hệ thống RS áp dụng CbF sẽ chỉ tư vấn những phim thuộc thể loại tình cảm và điều này có thể gây nên sự nhàm chán cho NSD vì không có khả năng khám phá những thể loại mới có thể rất thú vị mà trước đây chưa từng biết đến.

2.1.2 *Phương pháp lọc cộng tác*

Trong phương pháp *lọc cộng tác* (CF), việc tư vấn những items cho NSD được dựa theo các ý kiến phản hồi (*feedback*) hay đánh giá (*rating*) từ cộng đồng của NSD này. Cộng đồng của một NSD bao gồm những người cùng chia sẻ sở thích hay mối quan tâm với NSD đó [14].

Để xây dựng các cộng đồng, CF khai thác một ma trận đánh giá (*rating matrix*) trong đó mỗi NSD và mỗi item được lần lượt biểu diễn ở các dòng và các cột của ma trận (xem Bảng 2-1). Mỗi ô của ma trận tương ứng với một giá trị rating thể hiện mức độ ưa thích của NSD (trên dòng) dành cho item (trên cột).

Bảng 2-1. Minh họa ma trận đánh giá.

NSD \ Phim	Titanic	Mummy	Apollo 13	Spider Man
u_1	5	3		1
u_2	4			1
u_3	1	1		5
u_4	1			4
u_5		1	5	4

Về nguyên tắc, phương pháp CF sử dụng tập các ratings đã được biết trước của NSD để ước lượng hàm dự đoán *f* dành cho những items chưa được đánh giá:

$$f: User \times Item \rightarrow Rating$$

trong đó *Rating* là thang điểm đánh giá. Thang điểm này sẽ phụ thuộc vào từng hệ thống, ví dụ như giá trị rating nằm trong khoảng từ 1 (tương ứng với không thích) đến 5 (tương ứng với rất thích) hoặc chỉ hai giá trị thích/không thích. Công việc dự đoán trong CF là ước lượng các *f*(*user*, *item*) chưa được đánh giá bởi NSD. Ví dụ, theo như ma trận trên thì NSD u_1 đã đánh giá ba phim Titanic, Mummy và Spider Man lần lượt với các số điểm là 5, 3, 1 và đánh giá của NSD u_1 dành cho phim Apollo 13 là chưa biết, do đó RS cần phải dự đoán số điểm mà u_1 sẽ dành cho bộ phim này để quyết định có nên tư vấn phim Apollo 13 cho NSD hay không.

Trên nguyên tắc, việc quyết định hai NSD có phải là láng giềng trong một cộng đồng hay không sẽ tùy thuộc vào độ tương đồng giữa các ratings của họ. Hiện nay, các thuật toán trong CF được chia thành hai nhóm chính, dựa vào cách thức xây dựng và khai thác cộng đồng, đó là: nhóm thuật toán dựa trên bộ nhớ (*memory-based*) và nhóm thuật toán dựa trên mô hình (*model-based*) [2], [14].

2.1.2.1 *Cách tiếp cận dựa trên bộ nhớ*

Cách tiếp cận dựa trên bộ nhớ (*Memory-based Approach*) hay còn được gọi là cách tiếp cận dựa vào láng giềng (*Neighborhood-based*) tập trung chủ yếu vào việc khai thác mối quan hệ giữa những NSD hoặc giữa những items với nhau. Cách tiếp cận này không cần quá trình huấn luyện (training phase) bởi vì việc tư vấn được dựa vào ratings của những láng giềng gần nhất với NSD đang được xét đến. Để tìm các láng giềng gần nhất của NSD, cách tiếp cận này cần sử dụng một độ đo về mức độ tương đồng (hay khoảng cách) giữa những NSD. Các độ đo phổ biến nhất hiện nay là tương quan Pearson (Pearson correlation), cosine hay khoảng cách Euclidean, ... [6]. Công thức tính toán mức độ tương đồng giữa hai NSD bằng tương quan Pearson được định nghĩa như sau:

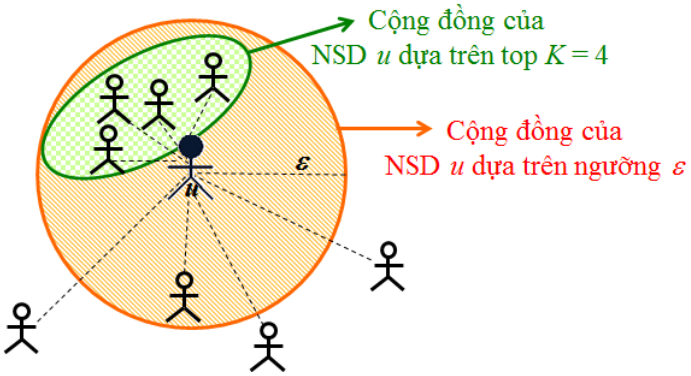
$$sim(u_a, u_b) = \frac{\sum_{i_t \in I_{a,b}} (r_{a,t} - \overline{r}_a).(r_{b,t} - \overline{r}_b)}{\sqrt{\sum_{i_t \in I_{a,b}} (r_{a,t} - \overline{r}_a)^2} \sqrt{\sum_{i_t \in I_{a,b}} (r_{b,t} - \overline{r}_b)^2}} \qquad (2.2)$$

Trong công thức (2.2), $I_{a,b}$ là tập những items mà cả hai NSD u_a và u_b cùng đánh giá, $r_{a,t}$ là rating của NSD u_a dành cho item i_t, $\overline{r}_a$ là rating trung bình của NSD u_a trên tất cả các items. Giá trị của công thức tính tương quan Pearson sẽ nằm trong đoạn [-1, 1]. Nếu độ tương đồng của hai NSD là 1 có nghĩa là các ratings của họ dành cho những items chung là hoàn toàn giống nhau. Ngược lại, nếu độ tương đồng có giá trị là -1 thì hai NSD có sự khác biệt lớn trong cách đánh giá các item chung.

Xét ma trận đánh giá trong Bảng 2-1, giả sử hệ thống muốn tính độ tương đồng của hai NSD u_3 và u_4 thì đầu tiên hệ thống cần xác định danh sách các phim mà cả hai NSD đã cùng đánh giá đó là {Titanic, Spider Man}. Trung bình đánh giá của u_3 và u_4 lần lượt là 3.3 và 2.5 điểm. Áp dụng công thức (2.2), chúng ta có độ tương đồng của u_3 và u_4 được tính như sau:

$$sim(u_3,u_4)=\frac{(1-3.3)(1-2.5)+(5-3.3)(4-2.5)}{\sqrt{(1-3.3)^2+(5-3.3)^2}\sqrt{(1-2.5)^2+(4-2.5)^2}}=0.97$$

Sau khi hệ thống đã tính toán mức độ tương đồng giữa NSD u với những NSD còn lại, hệ thống có thể chọn lấy K láng giềng gần nhất (*K-Nearest Neighborhood* – KNN) với NSD u hoặc sẽ lựa chọn những NSD u' có khoảng cách với NSD u nhỏ hơn một ngưỡng nào đó: $sim(u, u') < \varepsilon$ như trong Hình 2-1 để thành lập cộng đồng $G(u)$ cho NSD u.



Hình 2-1. Minh họa cách thành lập cộng đồng $G(u)$ của NSD u.

Cuối cùng, để dự đoán rating của NSD *u* dành cho item *i*, hệ thống sẽ có nhiều cách thực hiện. Cách đầu tiên đơn giản nhất là lấy trung bình cộng ratings của các thành viên trong cộng đồng *G*(*u*) dành cho item *i*.

$$\hat{r}_{u,i} = \frac{1}{N} \sum_{u' \in G(u)} r_{u',i} \tag{2.3}$$

Tuy nhiên, với cách lấy trung bình theo như công thức trên thì mức độ ảnh hưởng của các thành viên trong cộng đồng *G*(*u*) đối với NSD *u* đã được cho là hoàn toàn giống như nhau, nhưng trên thực tế, những láng giềng càng gần gũi với NSD *u* thì mức độ tin cậy hay mức độ ảnh hưởng của họ đến rating của NSD *u* càng cao. Ngoài ra, một số NSD có xu hướng khắt khe (chấm điểm thấp) hơn những người khác và ngược lại, cũng có những NSD rộng rãi (cho điểm cao) hơn so với các những người khác. Do đó, Breese và các cộng sự đã đề xuất một công thức tính dự đoán cho NSD phù hợp hơn và công thức này hiện nay vẫn còn đang được sử dụng rất rộng rãi trong các thuật toán CF [6]:

$$\hat{r}_{u,i} = \overline{r}_u + \frac{\sum_{u' \in G(u)} sim(u,u').(r_{u',i} - \overline{r}_{u'})}{\sum_{u' \in G(u)} |sim(u,u')|} \tag{2.4}$$

Trong công thức trên, *G*(*u*) là cộng đồng của NSD *u*, $\overline{r}_u$ là rating trung bình của NSD *u* trên tất cả các items, *sim*(*u*, *u'*) thể hiện mức độ tương đồng của hai NSD *u* và *u'*, phần mẫu số trong công thức được dùng để chuẩn hóa giá trị dự đoán.

Ưu điểm chính của cách tiếp cận dựa trên bộ nhớ là khá đơn giản trong việc cài đặt, không cần tốn chi phí cho quá trình huấn luyện, giúp NSD có thể dễ dàng hiểu được tại sao họ lại nhận được những items tư vấn như vậy cũng như mối quan hệ giữa họ với cộng đồng, và cuối cùng là cách tiếp cận này có được sự ổn định do không cần phải huấn luyện lại cho hệ thống [8]. Tuy nhiên, bên cạnh những ưu điểm thì mặt hạn chế của hướng tiếp cận này là nó bị ảnh hưởng nhiều bởi vấn đề dữ liệu thưa (đã được đề cập ở đoạn cuối của phần 1.1). Ngoài ra, cách tiếp cận này

không thể đáp ứng cho những RS cần thời gian tư vấn trực tuyến (online) do các ratings dự đoán không được tính toán trước hay nói cách khác là các ratings dự đoán chỉ được tính toán tại thời điểm yêu cầu tư vấn cho nên tốc độ đưa ra danh sách tư vấn khá chậm.

2.1.2.2 *Cách tiếp cận dựa trên mô hình*

Mặt hạn chế của cách tiếp cận dựa trên bộ nhớ là điểm khởi đầu cho cách tiếp cận dựa trên mô hình (*Model-based Approach*). Ý tưởng chính sẽ là hệ thống sử dụng các ratings trước đó của NSD để xây dựng một mô hình dự đoán lưu trữ sẵn (offline), sau đó mô hình này sẽ được áp dụng để dự đoán rating và đưa ra danh sách tư vấn trực tuyến một cách nhanh nhất có thể được [6], [14]. Ví dụ, hệ thống có thể xây dựng mô hình gom nhóm (cluster model) để chia tập NSD thành *k* nhóm theo mức độ tương đồng. Tiếp theo, việc xác định cộng đồng cho NSD mới trở thành bài toán phân lớp (classification).

Các thuật toán dựa trên mô hình có thể sử dụng rất nhiều phương pháp khác nhau để xây dựng mô hình dự đoán như bằng các phương pháp xác suất thống kê hay các phương pháp machine learning. Trong mô hình dự đoán được xây dựng theo hướng tiếp cận xác suất do Breese và các cộng sự đề xuất [6], mỗi rating chưa biết giá trị sẽ được dự đoán dựa vào công thức sau:

$$\hat{r}_{u,i} = \sum_{j=0}^{n} j.\Pr(r_{u,i} = j \mid r_{u,i'}, i' \in I_u) \qquad (2.5)$$

Trong công thức trên, $\hat{r}_{u,i}$ là rating chưa biết và cần được dự đoán của NSD *u* đối với item *i*. Phương pháp giả định thang điểm đánh giá là từ 0 đến *n*, và như vậy, *j* là rating sẽ thuộc đoạn [0, n], I_u là danh sách các items mà NSD *u* đã đánh giá. Việc dự đoán giá trị của $\hat{r}_{u,i}$ được chuyển thành bài toán tính xác suất có điều kiện như sau: tìm xác suất để NSD *u* sẽ đánh giá điểm *j* cho item *i* khi biết các ratings trước đó của NSD *u*.

Để giải quyết bài toán xác suất trên, các tác giả đã đề xuất sử dụng mô hình gom nhóm (cluster model) và mạng Bayesian (Bayesian networks). Trong mô hình đầu tiên, những NSD có sự tương đồng trên quan điểm đánh giá sẽ được gom vào các nhóm tương ứng. Số lượng nhóm đã được xác định trước dựa vào việc học các dữ liệu có sẵn. Như vậy, việc xác định nhóm cho NSD sẽ trở thành bài toán tính xác suất để NSD đó có thể thuộc vào một nhóm và việc dự đoán rating của NSD trên một item sẽ được dựa trên các ratings của những người thuộc về cùng nhóm. Trong mô hình mạng Bayesian mỗi item sẽ được biểu diễn thành một node trong mạng và mô hình thể hiện sự phụ thuộc, mối liên hệ giữa các node (items) [6]. Đây là hình thức phân nhóm items thay vì phân nhóm những NSD.

Như vậy, thông qua việc xây dựng mô hình dự đoán offline, các thuật toán dựa trên mô hình thường đưa ra kết quả dự đoán rất nhanh tại thời điểm được yêu cầu mặc dù chúng phải tốn chi phí cho quá trình huấn luyện để xây dựng mô hình dự đoán. Do đó, các phương pháp dựa trên mô hình thường được lựa chọn trong trường hợp tập dữ liệu lớn và tốc độ tư vấn là một yếu tố quan trọng.

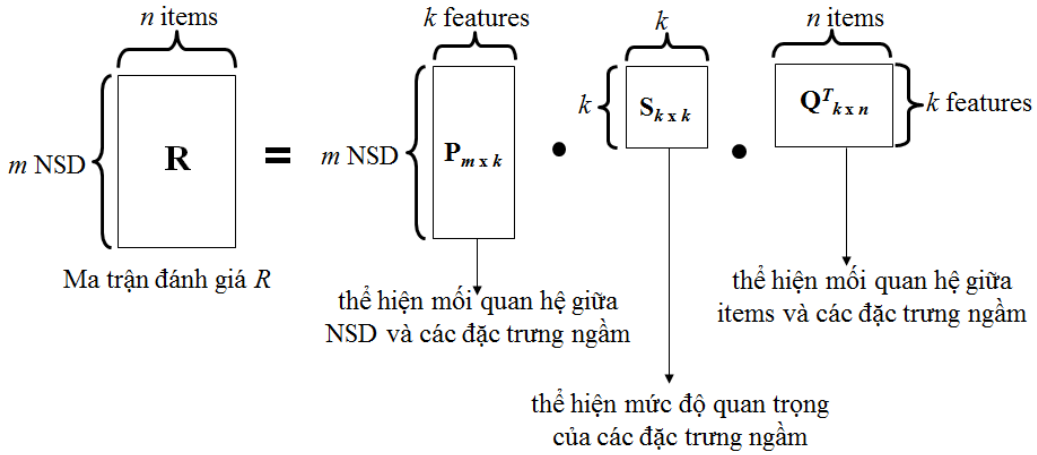
2.1.2.3 *Cách tiếp cận dựa trên mô hình đặc trưng ngầm*

Trong những năm gần đây, các phương pháp machine learning như thuật toán SVM, mô hình học tăng cường (boosting), … đã được các nhà nghiên cứu trong lĩnh vực RS đưa vào khai thác để giải quyết bài toán tư vấn. Trong đó, chúng ta không thể không kể đến một mô hình đang được áp dụng rất phổ biến trong RS vì tính hiệu quả và tính chính xác của nó, đó là *latent factor models* với phương pháp điển hình là *Matrix Factorization* (MF). MF được xem là một trong những phương pháp tốt nhất của cách tiếp cận mô hình đặc trưng ngầm, điều này đã được minh chứng bằng việc Koren sử dụng phương pháp MF để giành chiến thắng tại giải thưởng Netflix [12], [13]. Thông qua thực nghiệm, Koren đã chứng minh rằng MF đạt được độ chính xác cao hơn so với các phương pháp khác dựa trên bộ nhớ.

Cơ sở lý thuyết của phương pháp MF là mọi ma trận R_{mxn} bất kỳ đều có thể được biến đổi thành tích của các ma trận thừa số như trong công thức (2.6).

$$R_{m \times n} = P_{m \times k}.S_{k \times k}.Q^{T}_{k \times n} \quad (2.6)$$

Trên cơ sở đó, ma trận đánh giá R trong một RS cũng sẽ được phân tích thành ba ma trận thừa số như trong Hình 2-2 với ma trận P_{mxk} thể hiện mối quan hệ giữa m NSD với k đặc trưng ngầm (latent features) và ma trận Q_{kxn} thể hiện mối quan hệ giữa n items và k đặc trưng ngầm. Ma trận S_{kxk} biểu diễn cho vai trò của các đặc trưng ngầm. Sở dĩ người ta gọi là những đặc trưng ngầm là vì không thể xác định được ngữ nghĩa của đặc trưng một cách rõ ràng, chúng không được thể hiện một cách tường minh trong biểu diễn của item mà chỉ biết rằng chúng có những mối liên hệ với NSD và items. Ví dụ, kết thúc có hậu, nhân vật chính hay bị ung thư, có nhiều cảnh thiên nhiên, … là những đặc trưng ngầm của phim.



Hình 2-2. Hình minh họa công thức MF.

Các ma trận thừa số sau khi được phân tích phải thỏa mãn những điều kiện sau:

- P là ma trận trực giao ($P^TP = PP^T = I$ với I là ma trận đơn vị).
- Q là ma trận trực giao ($Q^TQ = QQ^T = I$ với I là ma trận đơn vị).
- S là ma trận đối xứng, chỉ có các giá trị không âm trên đường chéo chính, được sắp xếp thứ tự giảm dần từ lớn đến nhỏ, và các giá trị khác đều bằng 0.

Như vậy, ma trận R_{mxn} ban đầu đã được biến đổi thành tích của các ma trận thừa số với số chiều nhỏ hơn, vì số lượng các đặc trưng ngầm rất nhỏ so với số lượng NSD ($k << m$) và so với số lượng items ($k << n$).

Dựa trên cơ sở toán học đã trình bày ở trên, mục tiêu của phương pháp MF là tìm hai ma trận thừa số P và Q sao cho tích của hai ma trận P và Q^T xấp xỉ với ma trận R ban đầu. Như vậy, ma trận S đã được bỏ qua vì điều này sẽ giúp đơn giản hóa việc tính toán (giảm chi phí). Bên cạnh đó, nguyên do là những đặc trưng ngầm cho nên không thể xác định rõ ràng được ngữ nghĩa của các đặc trưng và MF không cần thể hiện tường minh vai trò của các đặc trưng thông qua ma trận S.

$$R_{m \text{ x } n} \approx P_{m \text{ x } k}.Q^T_{k \text{ x } n} \tag{2.7}$$

Trong công thức (2.7) R là ma trận đánh giá *User* x *Item.* Mỗi dòng trong ma trận P biểu diễn mối quan hệ giữa một NSD và các đặc trưng ngầm. Tương tự, mỗi cột của Q thể hiện mối liên hệ giữa một item và các đặc trưng ngầm. Việc thừa số hóa ma trận đánh giá đã giúp cho MF có thể phát hiện ra những mối quan hệ giữa các đặc trưng ngầm với từng NSD và từng item trong dữ liệu [13].

Như đã đề cập ở trên, đặc điểm của mô hình đặc trưng ngầm là có thể suy luận ra mối liên hệ giữa đặc trưng với NSD và với item mà không cần biết chính xác ngữ nghĩa của đặc trưng đó. Với ý tưởng này, để dự đoán rating của NSD u_a dành cho item i_t, MF sử dụng công thức sau:

$$\hat{r}(u_a, i_t) = P_a.Q_t^T = \sum_{x=1}^{k} p_{ax}.q_{xt}^T \tag{2.8}$$

Ví dụ minh họa dự đoán rating mà NSD u_2 sẽ dành cho item *Apollo 13* bằng phương pháp MF như sau:

	Titanic	Mummy	Apollo 13	Spider Man
u_1	5	3		1
u_2	4		**?**	1
u_3	1	1		5
u_4	1			4
u_5		1	5	4

Ma trận đánh giá R_{mxn}, với $m = 5$, $n = 4$

$\approx$

	f_1	f_2
u_1	2.44	0.23
u_2	1.96	0.27
u_3	0.46	2.16
u_4	0.46	1.75
u_5	0.66	1.77

Ma trận thừa số P_{mxk} , với $k = 2$

$\bullet$

	Titanic	Mummy	Apollo 13	Spider Man
f_1	1.95	1.14	1.54	0.22
f_2	0.06	0.18	2.17	2.13

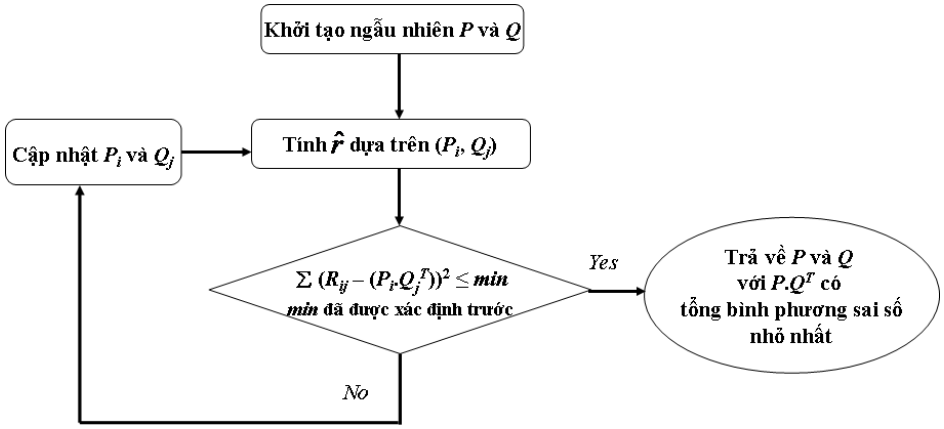
Ma trận thừa số Q^T_{kxn}, với $k = 2$

$$\hat{r}(u_2, \text{Apollo 13}) = 1.96 \text{ x } 1.54 + 0.27 \text{ x } 2.17 = 3.6$$

Như vậy, mục tiêu của phương pháp MF là tìm hai ma trận P và Q sao cho $P.Q^T$ hội tụ về một giá trị có bình phương sai số so với ma trận R ban đầu là nhỏ nhất. Hàm mục tiêu được biểu diễn trong công thức bên dưới:

$$\min_{P_i, Q_j} \sum_{x=1}^{k} (R_{ij} - P_i.Q_j^T)^2 \tag{2.9}$$

Các bước tìm hai ma trận P và Q được mô tả dưới dạng lưu đồ sau:



Hình 2-3. Lưu đồ thực hiện thuật toán MF.

Trong quá trình huấn luyện, để tránh tình trạng quá khớp (overfitting) khi mà các bộ tham số sau khi học chỉ đạt được độ chính xác cao trên tập huấn luyện nhưng lại có độ chính xác thấp trên tập thử nghiệm, hàm mục tiêu ban đầu của MF trong công thức (2.9) được điều chỉnh lại thành công thức (2.10), trong đó λ giữ vai trò là tham số điều hòa (regularization parameter).

$$\min_{P_i,Q_j} \sum_{x=1}^{k} [(R_{ij} - P_i.Q_j^T)^2 + \lambda(\| P_i \|^2 + \| Q_j \|^2)] \tag{2.10}$$

Các bước cập nhật hai ma trận P và Q được thực hiện bằng cách tiếp cận stochastic gradient descent như trong công thức (2.11), trong đó e_{ij} là độ lệch giữa rating thật sự so với dự đoán và γ là hệ số học.

$$\begin{aligned} e_{ij} &= R_{ij} - P_i.Q_j^T \\ Q_j &= Q_j + \gamma(e_{ij}P_i - \lambda Q_j) \\ P_i &= P_i + \gamma(e_{ij}Q_j - \lambda P_i) \end{aligned} \tag{2.11}$$

Thực tế cho thấy một số NSD có xu hướng luôn đánh giá cao (rộng rãi) hơn hay thấp (khắt khe) hơn những NSD khác; hoặc tương tự, một số items sẽ có xu hướng nhận được đánh giá cao hơn hoặc thấp hơn các item khác. Ví dụ, phim Titanic thường nhận được ratings cao hơn 0.5 điểm so với trung bình của các phim khác. Để giải quyết vấn đề trên, công thức dự đoán trong (2.8) sẽ được bổ sung thêm thành phần thể hiện xu hướng đánh giá của NSD và của item (adding biases) như sau:

$$\hat{r}(u_a, i_t) = \mu + b_a + b_t + P_a.Q_t^T \tag{2.12}$$

Trong công thức (2.12), μ là rating trung bình của tất cả items; b_a là độ lệch giữa rating cho bởi NSD u_a và rating trung bình của tất cả NSD; b_t là độ lệch giữa rating của item i_t và rating trung bình của tất cả items. Ví dụ, rating trung bình của tất cả các phim là 2.8, NSD u_2 có xu hướng đánh giá thấp hơn 0.3 so với trung bình

các NSD khác, phim Apollo 13 có xu hướng nhận được cao hơn 0.5 so với trung bình các phim khác. Như vậy, để ước lượng rating của NSD u_2 dành cho phim Apollo 13, phương pháp MF sẽ tính như sau:

$$\hat{r}(u_2, \text{Apollo 13}) = 2.8 - 0.3 + 0.5 + P_{u_2} Q^T_{Apollo13}$$

Hiện nay, mô hình đặc trưng ngầm đang trở thành một trong những cách tiếp cận được ưa thích trong CF, đặc biệt là khi nó được kết hợp với cách tiếp cận dựa trên láng giềng (dựa trên bộ nhớ) [12].

Nói tóm lại, phương pháp CF được áp dụng rộng rãi trong phần lớn các RS bởi vì phương pháp này không đòi hỏi tri thức của lãnh vực ứng dụng cũng như không cần phải phân tích nội dung items. Hơn nữa, việc tạo danh sách tư vấn với phương pháp CF giúp NSD có khả năng khám phá các items thuộc những lĩnh vực mới mà họ chưa từng biết đến [12], [14]. Tuy nhiên, một trong những hạn chế của phương pháp CF là chỉ đạt kết quả tốt hơn những cách tiếp cận khác khi hệ thống có đủ số lượng các ratings của NSD để thực hiện việc tư vấn. Nói một cách cụ thể hơn, chất lượng của những RS dựa vào CF thường bị giảm sút đáng kể nếu gặp phải vấn đề dữ liệu thưa, khi một NSD đánh giá quá ít items hay một item nhận rất ít ratings từ những NSD. Hiện tượng này làm cho hệ thống không có đủ số lượng các ratings để tính toán độ tương đồng giữa những NSD với nhau, từ đó không thể gom nhóm và xây dựng cộng đồng theo tiêu chí ratings.

2.1.3 *Lai ghép các phương pháp*

Cách tiếp cận lai ghép (*Hybrid Approach*) được xây dựng dựa trên việc kết hợp hai hay nhiều kỹ thuật của RS để giải quyết các mặt hạn chế của một kỹ thuật đơn lẻ cũng như khai thác các ưu điểm của từng cách tiếp cận [7]. Burke giới thiệu nhiều chiến thuật lai ghép có thể được áp dụng để tạo ra danh sách tư vấn như weighted, switching, mixed hybrid,…

- Weighted hybrid: mỗi phương pháp P_j sẽ gán cho item i một giá trị dự đoán $prediction_j(i)$. Như vậy, giá trị $prediction(i)$ cuối cùng sẽ được tính bằng công thức sau:

$$prediction(i) = \sum_j w_j . prediction_j(i)$$

Cách lai ghép weighted cần phải xác định các trọng số w_j của từng phương pháp và ta có thể thấy là giá trị dự đoán cuối cùng không phụ thuộc vào thứ tự thực hiện các phương pháp. Đặc điểm của cách lai ghép này là đơn giản, khá hiệu quả, và dễ dàng hiệu chỉnh vai trò của các phương pháp thông qua việc điều chỉnh trọng số w_j. Khuyết điểm của cách lai ghép này là vai trò của các phương pháp độc lập với ngữ cảnh khai thác tức là trong mọi trường hợp, các phương pháp đều kết hợp với nhau thông qua giá trị trọng số đã được định sẵn.

- Switching hybrid: việc lựa chọn áp dụng một phương pháp P_j phụ thuộc vào các quy luật đã được định trước. Như vậy, giá trị $prediction(i)$ cuối cùng được tính bởi phương pháp P_j được chọn. Đặc điểm của phương pháp này là cần phải xác định các luật hay các tiêu chí để lựa chọn các phương pháp, và giá trị dự đoán cuối cùng không phụ thuộc vào thứ tự thực hiện các phương pháp. Ưu điểm của cách lai ghép switching là có quan tâm đến ngữ cảnh khai thác như loại thông tin, loại NSD. Ví dụ, trong một trường hợp như thế này thì sử dụng phương pháp P_1 và trong trường hợp như thế kia thì sử dụng phương pháp P_2. Khuyết điểm của cách lai ghép này là độ phức tạp và chi phí để xác định các luật lựa chọn phương pháp.

- Mixed hybrid: khai thác toàn bộ kết quả của tất cả các phương pháp P_j. Kết quả cuối cùng có thể được trình bày chung cho tất cả phương pháp, hoặc trình bày riêng kết quả của từng phương pháp. Ưu điểm của phương pháp mixed là nó thỏa mãn những NSD có nhu cầu nhận được càng nhiều thông tin càng tốt và góp phần giải quyết được vấn đề khi có những items mới xuất hiện trong hệ thống thì chúng chưa nhận được bất kỳ đánh giá nào từ NSD và do đó hệ

thống sẽ không biết tư vấn các items mới đó cho NSD nào. Vấn đề này được gọi là vấn đề cold start problem (new items). Khuyết điểm của mixed hybrid là các kết quả từ các phương pháp có thể mâu thuẫn với nhau.

Nhìn chung, đa số các RS có áp dụng phương pháp CF thường chỉ dựa trên những ratings có sẵn của một nhóm người trong cộng đồng để dự đoán mức độ ưa thích của NSD dành cho các items và cho rằng sự ưa thích đó là cố định, không thay đổi theo ngữ cảnh. Trong thực tế, mặc dù sở thích của NSD có thể khá bền vững nhưng mức độ ưa thích của NSD dành cho một item có thể bị thay đổi rất lớn bởi nhiều yếu tố tác động như ngữ cảnh. Ví dụ, trong hệ thống tư vấn du lịch, yếu tố thời tiết có thể đóng một vai trò quan trọng trong việc lựa chọn nên đi tắm biển hay đi xem triển lãm tranh trong nhà. Đây là một vấn đề quan trọng chưa được quan tâm, xem xét trong cách tiếp cận CF truyền thống. Do đó, hiện nay đã có rất nhiều nghiên cứu đề xuất sự mở rộng CF để tích hợp thông tin ngữ cảnh vào trong quá trình tư vấn.

2.2 Các hệ thống tư vấn theo ngữ cảnh

Có rất nhiều định nghĩa khác nhau về ngữ cảnh được đưa ra tùy thuộc vào lĩnh vực đang xem xét [1]. Đối với lĩnh vực tư vấn thông tin, định nghĩa về ngữ cảnh của Dey được sử dụng khá phổ biến [4], [9], [21]:

"*Ngữ cảnh là bất cứ thông tin gì có thể đặc trưng cho trạng thái của một thực thể*".

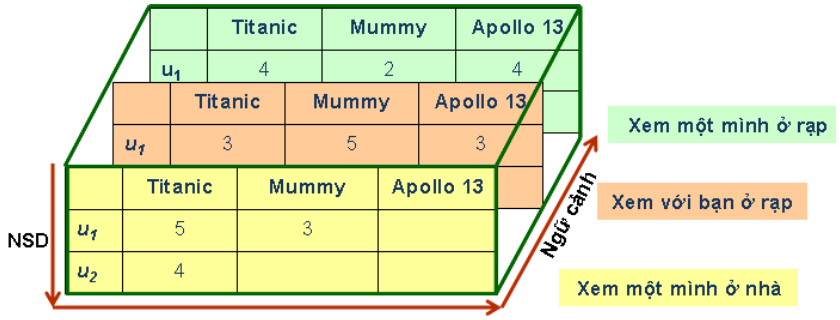
Các thực thể trong RS thường là NSD, item và những điều kiện mà NSD thực hiện việc đánh giá các items. Như vậy, một thuộc tính ngữ cảnh (*contextual feature*) có thể phụ thuộc vào NSD, vào item hay phụ thuộc vào cả hai. Ví dụ, đối với thuộc tính ngữ cảnh là thời tiết thì những ngữ cảnh có thể là nắng, mưa, có mây, có tuyết,… hay khi xét thuộc tính ngữ cảnh là tâm trạng của NSD thì các ngữ cảnh có thể có là vui, buồn, lo lắng, …

Trong thời gian gần đây, các hệ thống CARS đã đưa yếu tố ngữ cảnh vào ma trận đánh giá bằng cách mở rộng hàm ước lượng dự đoán như sau:

$$f: User \times Item \times Context \rightarrow Rating$$

Context là tập các giá trị của các thuộc tính ngữ cảnh liên quan đến ứng dụng.

Trong hàm ước lượng dự đoán ở trên, rating phụ thuộc vào ba yếu tố NSD, item và ngữ cảnh. Ví dụ, trong Hình 2-4, chúng ta có thể thấy là rating của NSD u_1 dành cho phim Titanic thay đổi theo ngữ cảnh, cụ thể là khi xem phim một mình ở nhà thì NSD u_1 đánh giá phim Titanic 5 điểm nhưng với ngữ cảnh là xem phim với bạn ở rạp thì NSD u_1 chỉ đánh giá 3 điểm cho bộ phim này.



Hình 2-4. Mô hình lưu trữ các đánh giá trong hệ thống CARS.

	Titanic	Mummy	Apollo 13
u_1	5	3	
u_2	4		

	Titanic	Mummy	Apollo 13
u_1	3	5	3
u_2		4	4

	Titanic	Mummy	Apollo 13
u_1	4	2	4
u_2		4	

Hình 2-5. Ma trận đánh giá tương ứng với các ngữ cảnh.

Các giá trị ngữ cảnh trong tập *Context* có thể được ghi nhận bằng nhiều cách: tường minh (explicit) hoặc không tường minh (implicit) [1]. Đối với cách đầu tiên, thông tin ngữ cảnh sẽ được cung cấp trực tiếp từ NSD hoặc từ các thiết bị có khả năng ghi nhận một số thông tin về vị trí địa lý. Ví dụ, một ứng dụng tư vấn âm nhạc có thể hỏi NSD muốn nghe những bài nhạc trong tâm trạng buồn, vui hay chỉ với mục đích thư giãn. Nếu là ứng dụng tư vấn trên thiết bị di động thì các thông tin

ngữ cảnh có thể thu được thông qua các thiết bị GPS để xác định địa điểm, thời gian, khí hậu, …

Đối với trường hợp thu thập thông tin ngữ cảnh một cách không tường minh thì thông thường hệ thống cần xây dựng một mô hình học những hành vi của NSD dựa trên các phương pháp suy luận. Ví dụ, hệ thống cần phân biệt được sự khác nhau khi NSD mua hàng cho mục đích cá nhân và cho mục đích tặng bạn bè, …

Ưu điểm của cách tiếp cận tường minh là không tốn chi phí cho việc học hành vi của NSD. Tuy nhiên, khuyết điểm lớn của cách tiếp cận này là NSD phải tốn công sức trả lời để xác định ngữ cảnh của mình hay trong một số trường hợp mà NSD không thể diễn tả ngữ cảnh một cách chính xác, ví dụ như hệ thống yêu cầu NSD xác định tâm trạng của mình. Ngược lại, với cách tiếp cận không tường minh thì quá trình lấy thông tin ngữ cảnh không ảnh hưởng đến NSD, nhưng hệ thống cần phải tốn chi phí cho việc xây dựng một mô hình suy diễn từ hành động của NSD và kết quả suy diễn từ mô hình có thể bị ảnh hưởng nhiều bởi thông tin nhiễu.

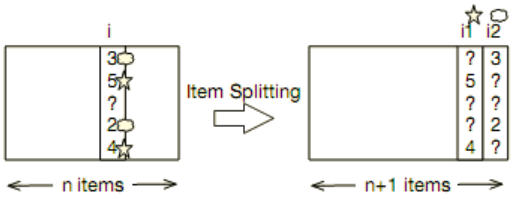
Hiện nay, dựa vào cách thức hay thời điểm mà ngữ cảnh được sử dụng trong quá trình tư vấn, các thuật toán tư vấn theo ngữ cảnh được chia thành ba mô hình phổ biến: pre-filtering, post-filtering và contextual modeling [1].

2.2.1 *Pre-filtering*

Pre-filtering sử dụng yếu tố ngữ cảnh để xử lý hay loại bỏ các ratings không có liên quan (đến ngữ cảnh) trước khi áp dụng phương pháp tư vấn CF truyền thống trên ma trận đánh giá *User* x *Item* (thường được gọi là phương pháp 2D). Một ví dụ của hệ thống tư vấn phim có xét đến yếu tố ngữ cảnh sử dụng mô hình pre-filtering là nếu NSD muốn xem phim vào ngày thứ bảy thì chỉ có những dữ liệu, những ratings trong các ngày thứ bảy mới được sử dụng để tư vấn phim. Ưu điểm lớn nhất của cách tiếp cận này là cho phép áp dụng nhiều kỹ thuật tư vấn truyền thống sau khi đã loại bỏ những ratings không liên quan.

Điển hình cho mô hình pre-filtering là phương pháp item-splitting do Baltrunas và Ricci đề xuất dựa trên ý tưởng chính là một item ban đầu sẽ được tách thành hai items ảo (artificial items) tương ứng với hai điều kiện ngữ cảnh nếu như ratings của hai items ảo này có sự khác biệt đáng kể về mặt thống kê [4].

Phương pháp này giả định rằng mỗi rating r_{ui} trong ma trận đánh giá R_{mxn} (m NSD và n items) sẽ gắn với một điều kiện ngữ cảnh $c(u, i)$ mà NSD u đã đánh giá item i. Ta có: $c(u, i) = (c_1, \ldots, c_k)$, $c_j \in C_j$ trong đó C_j là thuộc tính ngữ cảnh như thời tiết, tâm trạng, địa điểm, … Ví dụ, nếu C_j là thời tiết thì giá trị của c_j sẽ thuộc tập {nắng, mưa, mây, tuyết}. Với mỗi item, thuật toán sẽ tách các ratings của item đó thành hai tập con. Việc phân tách item phụ thuộc vào giá trị cụ thể của ngữ cảnh c_j. Ví dụ, xét item i sau khi được tách ra, tập con thứ nhất sẽ chứa các ratings gắn với item i_1 trong ngữ cảnh $c_j = v$ và tập con thứ hai sẽ chứa các ratings gắn với item i_2 trong ngữ cảnh $c_j \neq v$. Cách thức phân tách một item được minh họa trong Hình 2-6. Như vậy ta có thể thấy tổng số ratings của ma trận đánh giá sau khi thực hiện việc phân tách là không đổi, nhưng do số cột tăng lên cho nên ma trận sẽ trở nên thưa thớt hơn so với ban đầu.



Hình 2-6. Ví dụ minh họa item-splitting.

Với mỗi item, thuật toán sẽ phải tìm kiếm giá trị ngữ cảnh c_j mà có thể sử dụng để phân chia item. Sau đó, thuật toán sẽ kiểm tra xem ratings của hai tập con sau khi phân tách có sự khác biệt đáng kể về mặt thống kê hay không. Nếu có sự khác biệt đáng kể thì việc phân chia hoàn tất và item ban đầu (original item) sẽ được thay thế bởi hai items mới. Cuối cùng, rating dự đoán sẽ được tính toán dựa theo một trong số các items mới phát sinh. Giả sử item i được chia thành hai items

i_1, i_2 trong đó i_1 chứa rating của i trong ngữ cảnh c_j = trời nắng, ngược lại i_2 sẽ chứa rating của i trong trường hợp $c_j \neq$ trời nắng. Do đó, nếu hệ thống yêu cầu dự đoán rating của user u dành cho item i trong ngữ cảnh c_j = trời nắng thì hệ thống chỉ cần dự đoán đánh giá của user u dành cho item i_1: $\hat{r}(u,i,c_j) = \hat{r}(u,i_1)$. Ngược lại, nếu trong ngữ cảnh $c_j \neq$ trời nắng thì hệ thống sẽ dự đoán đánh giá của user u dành cho item i_2: $\hat{r}(u,i,c_j) = \hat{r}(u,i_2)$.

Các tác giả cho rằng việc phân tách sẽ có lợi nếu như các ratings của hai items mới sau khi tách đã có được sự đồng nhất hoặc sự khác biệt đáng kể về mặt thống kê. Như vậy, phương pháp này cần một tiêu chuẩn để đánh giá sự đồng nhất cũng như sự khác biệt giữa các ratings trong hai tập con. Trong bài báo, các tác giả đã đề xuất sử dụng các tiêu chuẩn t_{mean}, t_{prop}, t_{size}, t_{IG}, t_{random}. Do với mỗi tập con chúng ta sẽ có nhiều cách chia nên các tác giả đã chọn cách chia s sao cho giá trị kết quả của việc phân tách item i theo tiêu chí t, ký hiệu là $t(i, s)$, đạt giá trị lớn nhất.

- $t_{mean}(i, s)$ sử dụng t-test để ước lượng sự khác biệt về giá trị ratings trung bình trong hai tập ratings con khi sử dụng s.

- $t_{prop}(i, s)$ sử dụng hai thành phần z-test (*two-proportion z-test*) để xác định có sự khác biệt đáng kể giữa phần đánh giá đạt điểm số cao (rating > 3) và phần đánh giá đạt điểm số thấp (rating < 4) trong hai tập ratings mới được tạo ra hay không.

- $t_{size}(i, s)$ đo số lượng ratings của item i mà không phụ thuộc vào cách chia s. Theo các tác giả, tiêu chuẩn này được sử dụng để xác định item nào sẽ được chọn để phân tách trước vì item có đủ hay nhiều ratings sẽ được ưu tiên tách trước.

- $t_{IG}(i, s)$ là tiêu chuẩn đo mức độ có lợi của thông tin (*information gain*) được tạo ra từ cách chia s đối với các đánh giá của item i.

- $t_{random}(i, s)$ được dùng như là hình thức tham khảo để so sánh với các phương pháp khác. Tiêu chuẩn trả về giá trị ngẫu nhiên cho mỗi cách chia s.

Phương pháp item-splitting được kiểm thử trên hai tập dữ liệu thật Yahoo! Webscope movies dataset và MovieLens, và trên một tập dữ liệu bán nhân tạo (*semi-syntheic data*). Thuộc tính ngữ cảnh mà bài báo chọn là nhóm tuổi và giới tính của NSD.

Tập dữ liệu semi-syntheic dataset được tạo ra trên cơ sở mở rộng bộ dữ liệu Yahoo! ban đầu bằng cách chèn thêm vào thuộc tính tuổi tác và giới tính một giá trị ảo ngẫu nhiên $c \in \{0, 1\}$. Giá trị c này đóng vai trò thể hiện sự tác động của yếu tố ngữ cảnh lên các ratings của NSD. Các tác giả chọn ra ngẫu nhiên một số lượng ratings là (α * 100%) để thay đổi chúng. Cụ thể là sẽ tăng giá trị rating lên một điểm nếu như $c = 1$ và rating đó vẫn còn nhỏ hơn 5. Tương tự, các tác giả sẽ giảm giá trị rating xuống một điểm nếu như $c = 0$ và rating đó vẫn còn lớn hơn 1. Như vậy, nếu $\alpha = 0.5$ thì sẽ có 50% ratings bị thay đổi tăng hoặc giảm tùy thuộc vào giá trị của c.

Trong quá trình thử nghiệm, các kỹ thuật KNN (K-Nearest Neighbor), FACT (Matrix Factorization) và AVG (Item Average) được sử dụng để đánh giá hiệu quả (độ chính xác) của việc tách và không tách items. Thông qua độ đo MAE (Mean Absolute Error) cùng với tiêu chuẩn t_{prop} và t_{IG}, kết quả thực nghiệm cho thấy đối với hai bộ dữ liệu thật Yahoo! và MovieLens thì việc sử dụng phương pháp item-splitting đem lại hiệu quả không nhiều so với việc không sử dụng item-splitting. Các tác giả cũng giải thích rằng do việc lựa chọn thuộc tính ngữ cảnh là giới tính và nhóm tuổi không ảnh hưởng nhiều đến ratings của NSD. Thực chất giới tính và nhóm tuổi là đặc trưng của NSD, không phải là điều kiện ngữ cảnh.

Tuy nhiên, với bộ dữ liệu semi-syntheic dataset, kết quả thực nghiệm cho thấy việc tách các items đã cải thiện đáng kể độ chính xác so với việc không tách items. Đặc biệt, phương pháp item-splitting sẽ đem lại hiệu quả cao khi tham số $\alpha > 0.4$. Điều đó cho thấy khi mà tác động của yếu tố ngữ cảnh c lên ratings của NSD càng lớn thì việc sử dụng phương pháp item-splitting càng có lợi vì khi giá trị α tăng có nghĩa là số lượng các ratings bị thay đổi (tăng hoặc giảm) càng nhiều.

Một cách tiếp khác cũng tương tự như ý tưởng của item-splitting đó là "micro-profiles", trong đó mỗi profile của NSD được chia thành những hồ sơ đặc trưng con (*sub-profiles*) có mối liên hệ ngữ cảnh. Mỗi sub-profiles biểu diễn cho sở thích của NSD trong từng ngữ cảnh [3]. Theo phương pháp này, quá trình tư vấn sẽ sử dụng những micro-profiles phù hợp cho từng ngữ cảnh thay vì sử dụng một profile duy nhất cho mọi ngữ cảnh.

2.2.2 *Post-filtering*

Trong cách tiếp cận post-filtering, các ngữ cảnh sẽ được khai thác sau khi hệ thống đã áp dụng việc tính toán tư vấn bằng các phương pháp truyền thống 2D. Hay nói một cách cụ thể, cách tiếp cận post-filtering sẽ sử dụng các thông tin ngữ cảnh để tinh chế lại danh sách tư vấn từ các phương pháp truyền thống sao cho phù hợp với NSD trong một ngữ cảnh xác định. Việc tinh chế danh sách tư vấn có thể được thực hiện bằng một trong hai cách sau. Cách thứ nhất là loại bỏ những items tư vấn nào không có liên quan đến ngữ cảnh đang xét và cách thứ hai là xếp hạng lại danh sách items tư vấn theo thứ tự ưu tiên từ cao xuống thấp theo mức độ liên quan đến ngữ cảnh. Đây cũng là những cách thức mà hai thuật toán post-filtering: Weight và Filter đã áp dụng [18]. Chúng ta xem xét một ví dụ, nếu NSD muốn xem phim vào cuối tuần và bằng cách thức nào đó, hệ thống nhận thấy được vào cuối tuần thì NSD đó có thói quen chỉ xem phim thuộc thể loại hài kịch thì theo thuật toán Filter, những bộ phim nào thuộc thể loại khác với hài kịch sẽ bị loại bỏ, còn theo thuật toán Weight thì trong danh sách tư vấn, những phim thuộc thể loại hài kịch sẽ được xếp phía trên và những phim thuộc thể loại khác sẽ được xếp phía dưới. Nhìn chung, tương tự như pre-filtering, ưu điểm lớn của cách tiếp cận này là cho phép hệ thống CARS sử dụng nhiều kỹ thuật tư vấn truyền thống 2D đã được đề xuất trong các công trình [2].

Paniello và các cộng sự đã xem xét hai thuật toán post-filtering vừa nêu trên cũng như tìm hiểu và phân tích xem những trường hợp nào nên sử dụng pre-

filtering hay post-filtering để đạt được hiệu quả tốt hơn [18]. Đối với thuật toán Weight, các items tư vấn sẽ được xếp hạng dựa vào giá trị của rating dự đoán và xác suất để một item có mối liên hệ với ngữ cảnh cho trước. Trong khi với thuật toán Filter, các items sẽ bị loại bỏ khỏi danh sách tư vấn nếu như xác suất để chúng được tư vấn trong ngữ cảnh là quá thấp. Điểm tương đồng của hai thuật toán trên là ứng với một NSD u và một ngữ cảnh c, hệ thống cần tính toán xác suất để NSD u sẽ ưa thích item i trong ngữ cảnh c, ký hiệu là $P_c(u, i)$.

Bên cạnh đó, bằng việc so sánh kết quả thực nghiệm của hai mô hình pre-filtering và post-filtering, các tác giả đã kết luận rằng việc lựa chọn áp dụng cách tiếp cận pre- hay post-filtering để đem lại kết quả tốt nhất là thật sự phụ thuộc vào từng ứng dụng cụ thể.

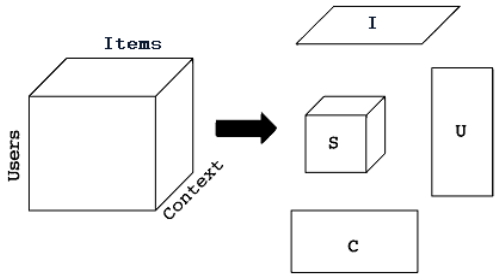
2.2.3 *Contextual modeling*

Đối với cách tiếp cận contextual modeling, các ngữ cảnh sẽ được sử dụng một cách trực tiếp trong thuật toán tư vấn, ví dụ ngữ cảnh sẽ là một thành phần trong công thức để dự đoán mức độ ưa thích của NSD đối với một item trong ngữ cảnh đó. Dựa vào tính hiệu quả của MF trong các thuật toán tư vấn truyền thống 2D, nhiều nhà nghiên cứu trong lĩnh vực RS cũng đã tìm cách áp dụng MF cho việc tư vấn theo ngữ cảnh.

Công trình có thể kể đến đầu tiên là của Shi và cộng sự [23] đã mở rộng phương pháp MF truyền thống để tích hợp thêm các thông tin về tâm trạng khi xem phim (*movie mood tag*). Đầu tiên, nhóm tác giả tính toán độ tương đồng giữa các phim với nhau dựa trên các thẻ cảm xúc (*mood tags*) và sau đó tích hợp các thông tin về độ tương đồng này vào mô hình dự đoán thông qua việc sử dụng trọng số kết hợp. Với cách thức này, các tác giả đã bổ sung được thông tin về ngữ cảnh chính là tâm trạng của NSD cho mô hình dự đoán.

Một cách tiếp cận khác cũng dựa trên nền tảng của kỹ thuật MF là Tensor Factorization (TF) được đề xuất bởi Karatzoglou và cộng sự [11]. Cách tiếp cận này

được xem như là phương pháp điển hình cho mô hình contextual modeling. Trước đây, kỹ thuật MF truyền thống chỉ khai thác ma trận đánh giá hai chiều *R*(*User* x *Item*) thì bây giờ các tác giả mở rộng MF cho không gian N-chiều. Mô hình TF được minh họa trong Hình 2-7.



Hình 2-7. Minh họa mô hình TF.

Dựa vào mô hình trên, các tác giả đã đề xuất công thức dự đoán rating của NSD *u* dành cho item *i* trong ngữ cảnh *c* như sau:

$$\hat{r}_{uic} = S \times_U U_u \times_I I_i \times_C C_c \tag{2.13}$$

Trong công thức trên $U \in \mathbb{R}^{m \times d_U}$, $I \in \mathbb{R}^{n \times d_I}$, $C \in \mathbb{R}^{c \times d_C}$ với *m*, *n* và *c* lần lượt là số lượng NSD, items và ngữ cảnh. d_U thể hiện số lượng các đặc trưng ngầm được rút trích có mối quan hệ với NSD; tương tự, d_I và d_C là số lượng các đặc trưng ngầm có mối liên hệ với items và ngữ cảnh. Lưu ý, các tham số d_U, d_I và d_C sẽ được học và điều chỉnh theo từng bộ dữ liệu. Các tác giả ký hiệu $\times_U$, $\times_I$, $\times_C$ là phép toán nhân ma trận giữa các mặt tương ứng trong khối *S* với ma trận *U*, *I* và *C*. Như vậy, chúng ta có thể thấy yếu tố ngữ cảnh đã được tích hợp trực tiếp trong công thức dự đoán của mô hình TF và việc sử dụng mô hình này có thể giúp hệ thống khai thác thêm được mối quan hệ giữa đặc trưng ngầm và ngữ cảnh bên cạnh mối quan hệ giữa đặc trưng ngầm với NSD và items như đã biết trong phương pháp MF truyền thống.

Thông qua thực nghiệm, các tác giả cũng so sánh cách tiếp cận TF với cách tiếp cận item-splitting trong mô hình pre-filtering và kết quả cho thấy TF cải thiện được độ chính xác một cách đáng kể. Tuy nhiên, một hạn chế của TF là phương

pháp này yêu cầu một số lượng tham số rất lớn để có thể học trên tập huấn luyện và đưa ra mô hình dự đoán.

Thực tế cho thấy, để có được mô hình dự đoán N-chiều thì số lượng tham số cần cho việc huấn luyện sẽ tăng theo bậc lũy thừa của số lượng các thuộc tính về ngữ cảnh. Bên cạnh đó, việc sử dụng nhiều tham số huấn luyện cho các thông tin ngữ cảnh sẽ có ảnh hưởng xấu đến độ chính xác của dự đoán nếu những thông tin ngữ cảnh đó là thông tin nhiễu. Do đó, Baltrunas và các cộng sự đã đề xuất phương pháp CAMF (*Context-aware Matrix Factorization*) và cho rằng trong những trường hợp mà bộ dữ liệu nhỏ thì một mô hình dự đoán đơn giản ít tham số huấn luyện hơn TF như CAMF sẽ có thể đạt được kết quả ngang bằng hay thậm chí tốt hơn so với phương pháp TF [5].

Phương pháp CAMF là sự mở rộng của cách tiếp cận MF truyền thống bằng cách tích hợp thêm yếu tố ngữ cảnh vào quá trình dự đoán của RS. Các tác giả đã đề xuất ba mô hình tư vấn có xét đến yếu tố ngữ cảnh:

- **CAMF-C** (**C**ontext): trong mô hình này, các tác giả cho rằng mỗi ngữ cảnh sẽ có ảnh hưởng toàn cục lên tất cả các ratings và độc lập với item. Ví dụ, trong hệ thống tư vấn địa điểm du lịch, khi thuộc tính ngữ cảnh thời tiết có giá trị là trời nắng thì ratings của tất cả địa điểm du lịch đều sẽ nhận được đánh giá cao cho dù là địa điểm du lịch đó ở trong nhà hay ở ngoài trời. Do đó, đối với mô hình này, chúng ta chỉ cần một tham số cho mỗi giá trị của thuộc tính ngữ cảnh. Tham số này biểu diễn độ lệch trong đánh giá của NSD dành cho item khi có ngữ cảnh và khi không có ngữ cảnh.

- **CAMF-CI** (**C**ontext & **I**tem): trong mô hình thứ hai này, các tác giả cần một tham số cho mỗi cặp ngữ cảnh và item (c, i). Như vậy, sự tác động của yếu tố ngữ cảnh lên các ratings đã được chia mịn hơn. Hay nói cách khác, ratings sẽ phụ thuộc vào cả hai yếu tố là ngữ cảnh và item thay vì chỉ phụ thuộc vào một yếu tố ngữ cảnh như trong mô hình CAMF-C. Do đó, mô hình CAMF-CI sẽ cần một số lượng lớn các tham số để học cho mỗi cặp ngữ cảnh

và item. Mô hình này sẽ dự đoán ratings tốt hơn trong trường hợp sự ảnh hưởng của ngữ cảnh lên các items là khác nhau. Ví dụ, ngữ cảnh thời tiết là trời nắng có thể làm cho đánh giá đối với những địa điểm du lịch ngoài trời tăng lên nhưng sẽ không ảnh hưởng đến những đánh giá cho các địa điểm du lịch trong nhà như viện bảo tàng.

- **CAMF-CC** (**C**ontext & Item **C**ategory): số lượng tham số cần cho mô hình này sẽ ít hơn trong mô hình thứ hai vì mỗi tham số sẽ tương ứng với một cặp ngữ cảnh và chủng loại item (số lượng chủng loại sẽ ít hơn rất nhiều so với số lượng items). Ở đây, mô hình giả định rằng các items có thể được nhóm lại theo các chủng loại (dựa vào kiến thức của chuyên gia trong lĩnh vực). Ví dụ, các địa điểm du lịch như nhà bảo tàng, nhà hát, trung tâm mua sắm, … được gom vào một loại là địa điểm du lịch trong nhà. Như vậy, với ngữ cảnh là trời nắng thì sự ảnh hưởng của ngữ cảnh này đối với nhóm địa điểm du lịch trong nhà sẽ khác với nhóm địa điểm du lịch ở ngoài trời.

Về mặt hình thức, các tác giả ký hiệu r_{uic} là rating của NSD u dành cho item i trong một ngữ cảnh c. Ngữ cảnh c được định nghĩa từ các giá trị trong tập thuộc tính ngữ cảnh. Ví dụ, tập thuộc tính ngữ cảnh có thể là thời gian, địa điểm, mùa trong năm, … Nếu thuộc tính ngữ cảnh là mùa trong năm thì các giá trị có thể có là {xuân, hạ, thu, đông}. Một cách tổng quát, ký hiệu $r_{uic1\ldots ch}$ là rating của NSD u dành cho item i trong ngữ cảnh $c = (c_1, \ldots, c_h)$. Giá trị của $c_j = 0, 1, \ldots, z_j$ trong đó nếu $c_j = 0$ có nghĩa là giá trị của thuộc tính ngữ cảnh thứ j không xác định, ngược lại, nếu $c_j \neq 0$ ngữ cảnh sẽ nhận giá trị của thuộc tính ngữ cảnh thứ j tại chỉ số tương ứng. Ví dụ, nếu hệ thống có hai thuộc tính ngữ cảnh là địa điểm *location* = {ở nhà, ở rạp} và mùa trong năm *season* = {xuân, hạ, thu, đông} thì location [0] có nghĩa là không xác định được địa điểm, location [1] = ở nhà, location [2] = ở rạp, tương tự cho thuộc tính ngữ cảnh season. Như vậy ký hiệu r_{ui20} có nghĩa là đánh giá của NSD u dành cho item i trong ngữ cảnh là ở rạp, tương tự r_{ui13} có nghĩa là NSD u đánh giá item i trong ngữ cảnh đang ở nhà vào mùa thu.

Mục tiêu của phương pháp MF truyền thống là phân tích ma trận đánh giá (*m* NSD x *n* items) ban đầu thành hai ma trận thừa số $P_{m \times k}$ và $Q_{n \times k}$. Mối quan hệ giữa một NSD *u* với các đặc trưng ngầm cũng như giữa một item *i* và các đặc trưng ngầm lần lượt được biểu diễn thành vectơ dòng $\vec{p}_u$ và vectơ cột $\vec{q}_i$ trong ma trận *P* và *Q*. Dựa trên cơ sở đó, Baltrunas và cộng sự đã đề xuất biểu thức dự đoán rating có xét đến yếu tố ngữ cảnh như sau:

$$\hat{r}_{uic_1 \ldots c_h} = \vec{p}_u . \vec{q}_i^T + \overline{i} + b_u + \sum_{j=1}^{h} B_{ijc_j} \tag{2.14}$$

Trong công thức trên, $\overline{i}$ là giá trị rating trung bình của item *i*, b_u là tham số thể hiện xu hướng đánh giá của NSD (user's biases). B_{ijcj} là tham số mô hình cho sự tương tác giữa yếu tố ngữ cảnh và items. Các tham số này dựa vào ba mô hình CAMF-C, CAMF-CI và CAMF-CC đã được trình bày ở phần trên. Đối với mô hình CAMF-C thì tham số B_{ijcj} trở thành B_{jcj} vì ngữ cảnh tác động lên ratings một cách độc lập với item. Đối với mô hình CAMF-CI, nếu như *H* là số lượng giá trị của tất cả các thuộc tính ngữ cảnh thì để dự đoán cho *n* items, B_{ijcj} cần có *H.n* tham số. Trong mô hình còn lại CAMF-CC, số lượng tham số cần xem xét sẽ ít hơn trong mô hình CAMF-CI vì số lượng các chủng loại của item ký hiệu là *t* sẽ nhỏ hơn rất nhiều so với số lượng item ($t << n$). Cụ thể là trong mô hình này, B_{ijcj} cần *H.t* tham số.

Hàm mục tiêu được các tác giả đề xuất cho phương pháp CAMF được trình bày trong công thức như sau:

$$\min_{p_*, q_*, b_*, B_*} \sum_{r \in R} \left[(r_{uic_1 \ldots c_h} - \vec{p}_u . \vec{q}_i^T - \overline{i} - b_u - \sum_{j=1}^{h} B_{ijc_j})^2 + \lambda (b_u^2 + \| \vec{p}_u \|^2 + \| \vec{q}_i \|^2 + \sum_{j=1}^{h} \sum_{c_j=1}^{z_j} B_{ijc_j}^2) \right] \tag{2.15}$$

Tương tự như phương pháp MF truyền thống, tham số λ trong (2.15) dùng để kiểm soát tình trạng overfitting. Các tác giả cũng sử dụng phương pháp stochastic gradient descent để cập nhật lại các giá trị tham số trong quá trình tìm kiếm bộ giá trị thỏa mãn hàm mục tiêu (xem 2.1.2.3).

Các tác giả đã sử dụng bộ dữ liệu thật MovieAT và ba bộ dữ liệu bán nhân tạo (*semi-artificial*) cho phần thực nghiệm. Bộ dữ liệu semi-artificial được dựa trên bộ dữ liệu gốc là Yahoo! Webscope và cách thực hiện cũng tương tự như đã trình bày trong phần item-splitting (xem 2.2.1). Độ đo dùng để đánh giá chất lượng của các thuật toán là MAE (Mean Absolute Error).

Trong thực nghiệm đầu tiên, các tác giả so sánh CAMF với thuật toán TF [11] trên các bộ dữ liệu thử nghiệm và nhận thấy CAMF đạt kết quả tốt hơn trên bộ dữ liệu thật MovieAT và trên bộ dữ liệu semi-artificial có hệ số $\alpha = 0.1$. Kết quả thực nghiệm cũng cho thấy khi mức độ ảnh hưởng của ngữ cảnh lên ratings càng nhiều như trong hai bộ dữ liệu semi-artificial (có hệ số $\alpha = 0.5$ và $\alpha = 0.9$) và khi dữ liệu đủ lớn để huấn luyện cho các tham số thì TF đạt kết quả tốt hơn CAMF.

Trong thực nghiệm thứ hai, các tác giả so sánh độ phức tạp và độ chính xác của ba mô hình đã đề xuất. Độ phức tạp được đánh giá thông qua số lượng tham số cần sử dụng cho mô hình, do đó CAMF-CI có độ phức tạp cao nhất và đơn giản nhất là mô hình CAMF-C. Về độ chính xác, mô hình CAMF-CC luôn có kết quả tốt hơn các mô hình khác. Bên cạnh đó, các tác giả cũng kết luận rằng do mô hình CAMF-CI cần quá nhiều tham số nên sẽ làm giảm độ chính xác của việc dự đoán.

Gần đây, Odic và cộng sự đề xuất hai cách để tích hợp ngữ cảnh vào MF [17]: contextualizing users' biases như trong (2.16) và users' latent features trong (2.17). Hai phương pháp này thực chất là tích hợp yếu tố ngữ cảnh vào mô hình dự đoán theo những cách khác nhau và cùng trên cơ sở là phương pháp MF.

$$\hat{r}(u,i,c) = \mu + b_i + b_u(c) + \vec{q}_i^T \vec{p}_u \tag{2.16}$$

$$\hat{r}(u,i,c) = \mu + b_i + b_u + \vec{q}_i^T \vec{p}_u(c) \tag{2.17}$$

Trong các công thức ở trên, $\hat{r}(u,i,c)$ là dự đoán đánh giá của NSD u dành cho item i trong ngữ cảnh c. Các tham số μ, b_u và b_i lần lượt là đánh giá toàn cục trên toàn ngữ cảnh, xu hướng đánh giá (user's biases) của NSD và của item (item's

biases). Xu hướng đánh giá của NSD trong ngữ cảnh c được ký hiệu $b_u(c)$. Hai vectơ $\vec{q}_i$ và $\vec{p}_u$ lần lượt thể hiện mối quan hệ giữa các đặc trưng ngầm với item và với NSD. Vectơ $\vec{p}_u(c)$ biểu diễn đặc trưng ngầm của NSD trong ngữ cảnh c.

Cả hai phương pháp trên đều sở hữu những đặc điểm của cách tiếp cận dựa trên đặc trưng ngầm, do đó ưu điểm của chúng vẫn là phát hiện tốt những mối quan hệ của sở thích NSD và các đặc trưng ngầm trong từng ngữ cảnh. Hơn nữa, chúng mang lại hiệu quả cao trong việc ước lượng các đánh giá có liên quan đến hầu hết hay toàn bộ NSD và item trong một ngữ cảnh cho trước. Ngược lại, một trong những mặt hạn chế của hai phương pháp này là trong một số ngữ cảnh, chúng thường không để ý đến mối quan hệ giữa những NSD có liên quan với nhau cũng như mối quan hệ giữa các items với nhau, trong khi điều này lại được những cách tiếp cận dựa trên láng giềng thực hiện tốt hơn.

Tóm lại, thông qua phần khảo sát và trình bày trong Chương 2 về một số công trình trong lĩnh vực RS và CARS, luận văn nhận thấy rằng tuyệt đại đa số các phương pháp chủ yếu chỉ dựa vào ratings như là tiêu chí duy nhất để xây dựng cộng đồng hay để dự đoán rating trong quá trình tư vấn mà quên rằng trong thực tế, sở thích của NSD có khả năng bị ảnh hưởng bởi các cộng đồng được xây dựng trên những tiêu chí khác nhau như cộng đồng nghề nghiệp, cộng đồng về nhóm tuổi, hay cộng đồng gồm những người thích xem phim tình cảm lãng mạn, … Ngoài ra, vấn đề dữ liệu thưa của ma trận đánh giá có tác động lớn trong việc làm giảm độ chính xác của thuật toán tư vấn. Do đó, mục tiêu của luận văn sẽ là khai thác những cộng đồng đa tiêu chí trong các hệ thống CARS để tăng cường sự phong phú, nâng cao chất lượng tư vấn theo ngữ cảnh.

Chương 3. **Tư vấn thông tin theo ngữ cảnh**

Chương 3 sẽ trình bày những đóng góp về mặt lý thuyết của luận văn. Trước hết, luận văn sử dụng những ký hiệu cơ bản về đại số của mô hình không gian các cộng đồng đa tiêu chí và sau đó, luận văn sẽ định nghĩa bổ sung một số khái niệm cùng với một quan hệ tiền thứ tự trên tập hợp các tiêu chí thành lập cộng đồng để làm cơ sở cho việc đề xuất ba thuật toán CRMC, CRESC và CREOC có thể được ứng dụng trong các hệ thống tư vấn theo ngữ cảnh.

3.1 Cách tiếp cận theo các cộng đồng đa tiêu chí

Ý tưởng chính của cách tiếp cận được đề xuất trong luận văn xuất phát từ việc một NSD có thể đồng thời thuộc nhiều loại cộng đồng khác nhau bên cạnh cộng đồng được xây dựng từ tiêu chí ratings đã rất quen thuộc trong các hệ thống tư vấn truyền thống. Điều này giúp cho NSD có thể nhận được nhiều sự tư vấn phong phú và hấp dẫn hơn. Ví dụ, khi một NSD có nhu cầu xem phim thì có thể nhận được rất nhiều sự gợi ý khác nhau từ các cộng đồng như những người thân trong gia đình, các đồng nghiệp hay cộng đồng những người thích xem phim tình cảm cổ điển mà người này đang là một thành viên.

Tuy nhiên, điều cần lưu ý là mức độ ảnh hưởng của các cộng đồng đối với việc tư vấn cho NSD trong từng ngữ cảnh cụ thể sẽ khác nhau. Ví dụ, trong ngữ cảnh xem phim một mình ở nhà vào cuối tuần, một giảng viên có thể thích phim "Cuốn theo chiều gió" dựa theo ý kiến của các thành viên trong hội những người

thích thể loại phim tình cảm cổ điển. Nhưng trong ngữ cảnh muốn đi xem phim với bạn ở rạp chiếu phim, giảng viên đó có thể chọn phim "Chiến tranh giữa các vì sao" thuộc thể loại phim hành động giả tưởng theo sự giới thiệu của đồng nghiệp.

Thông qua ví dụ trên, chúng ta có thể thấy trong ngữ cảnh xem phim một mình ở nhà vào cuối tuần thì cộng đồng theo sở thích cá nhân (thể loại) có ảnh hưởng mạnh đến sự lựa chọn của NSD trong khi với ngữ cảnh xem phim với bạn ở rạp chiếu phim thì cộng đồng theo nghề nghiệp lại có sự tác động nhiều hơn. Như vậy, khi ngữ cảnh thay đổi, mức độ ảnh hưởng của các cộng đồng đối với việc tư vấn cho NSD nhìn chung cũng sẽ thay đổi.

Dựa vào ý tưởng trên, một hệ thống tư vấn theo ngữ cảnh (CARS) trong cách tiếp cận đề xuất của luận văn có thể sử dụng các thông tin trong hồ sơ đặc trưng của NSD (user profile) như là các tiêu chí để phân hoạch những NSD thành nhiều cộng đồng khác nhau. Ví dụ, hệ thống có thể phân nhóm những NSD theo các tiêu chí như nghề nghiệp, nhóm tuổi, địa bàn cư trú hay thể loại phim yêu thích,… Bên cạnh đó, vì mức độ ảnh hưởng của các tiêu chí đối với sự tư vấn trong từng ngữ cảnh nhìn chung sẽ khác nhau cho nên trước khi áp dụng một thuật toán nào đó thì luận văn cần xác định tiêu chí nào là phù hợp nhất và trên cơ sở đó sẽ đưa ra danh sách tư vấn cho NSD trong từng ngữ cảnh cụ thể. Như vậy, trong mô hình đề xuất, luận văn sẽ định nghĩa một quan hệ tiền thứ tự (*pre-order*) trên tập hợp các tiêu chí cho từng ngữ cảnh và đồng thời đề xuất ba thuật toán tư vấn theo ngữ cảnh có khai thác các quan hệ tiền thứ tự này.

Trong phần tiếp theo, dựa vào việc sử dụng những ký hiệu cơ bản về đại số của mô hình không gian các cộng đồng đa tiêu chí (α-*Community Spaces Model-*αCSM) [16], luận văn đề xuất một sự mở rộng cho mô hình này bằng cách tích hợp yếu tố ngữ cảnh vào mô hình và định nghĩa một quan hệ tiền thứ tự trên tập hợp các tiêu chí tương ứng với từng ngữ cảnh cụ thể. Quan hệ tiền thứ tự này được xây dựng nhằm mục đích thể hiện mức độ phù hợp giữa các tiêu chí với một ngữ cảnh cụ thể đang được xét đến. Cuối cùng, luận văn sẽ trình bày ba thuật toán tư vấn CRMC,

CRESC và CREOC, có khai thác mô hình mở rộng trên vào quá trình tư vấn theo ngữ cảnh.

3.1.1 *Các khái niệm cơ bản*

Trong mô hình αCSM, tác giả sử dụng các định nghĩa U là tập hợp những NSD và A là tập hợp các tiêu chí có thể dùng để phân hoạch những NSD. Ví dụ, trong bộ dữ liệu MovieLens [15] thì các tiêu chí có thể khai thác bao gồm: nhóm tuổi (age), nghề nghiệp (occupation), thể loại yêu thích (favorite genre) và ratings. Với một tiêu chí $\alpha \in A$, chúng ta có thể định nghĩa một quan hệ tương đương $\mathfrak{R}^{(\alpha)}$ trên tập hợp U như sau:

$$\forall u, u' \in U, (u \; \mathfrak{R}^{(\alpha)} \; u') \Leftrightarrow \alpha(u) = \alpha(u') \tag{3.1}$$

trong đó $\alpha(u)$ là giá trị của tiêu chí α gắn với NSD u.

Ví dụ, $(u \; \mathfrak{R}^{(\text{occupation})} \; u') \Leftrightarrow \text{occupation}(u) = \text{occupation}(u')$, nghĩa là u và u' có cùng nghề nghiệp. Một lớp tương đương của quan hệ $\mathfrak{R}^{(\alpha)}$, ký hiệu $G_k^{(\alpha)}$, sẽ được gọi là một cộng đồng theo tiêu chí α (α-community).

Trong mô hình này, mỗi NSD u sẽ sở hữu một vectơ định vị (*postion vector*) ký hiệu là $P(u)$ như trong (3.2). Vectơ định vị được định nghĩa gồm nhiều thành phần $G^{(\alpha i)}(u)$ được ký hiệu là cộng đồng của NSD u theo tiêu chí α_i.

$$P(u) = (G^{(\alpha_1)}(u), ..., G^{(\alpha_n)}(u)), \forall \alpha_i \in A \tag{3.2}$$

Tất cả các vectơ định vị của NSD có thể được gom lại thành một bảng như minh họa trong Bảng 3-1. Bảng này thể hiện một NSD sẽ có những cộng đồng khác nhau tùy thuộc vào tiêu chí α. Ví dụ, NSD u_2 trong Bảng 3-1 có vectơ định vị $P(u_2)$ = (25–34, Kỹ sư, Hài kịch, Nhóm#2). Do đó, nếu gom nhóm theo tiêu chí nhóm tuổi thì u_2 sẽ chung cộng đồng với u_3, theo tiêu chí nghề nghiệp thì u_2 sẽ chung cộng

đồng với u_4, và nếu gom nhóm theo tiêu chí thể loại yêu thích hay theo ratings thì u_2 sẽ chung cộng đồng với u_1.

Bảng 3-1. Minh họa NSD có thể thuộc nhiều cộng đồng.

NSD	α_1 = Nhóm tuổi	α_2 = Nghề nghiệp	α_3 = Thể loại	α_4 = Ratings
u_1	18–24	Sinh viên	Hài kịch	Nhóm#2
u_2	25–34	Kỹ sư	Hài kịch	Nhóm#2
u_3	25–34	Giáo viên	Phiêu lưu	Nhóm#3
u_4	45–49	Kỹ sư	Tình cảm	Nhóm#5

Các cộng đồng có thể được tính toán theo nhiều cách khác nhau phụ thuộc vào bản chất của tiêu chí α [6], [8], [14], [16].

3.1.2 *Sự mở rộng của mô hình αCSM*

Đầu tiên, luận văn định nghĩa $G^{(\alpha)}(u, i)$ bao gồm những láng giềng của NSD u đã từng đánh giá item i. Kế đến, trong mô hình αCSM, tác giả không xét đến yếu tố ngữ cảnh, nhưng trong cách tiếp cận của luận văn, một ngữ cảnh sẽ được hình thành từ tập hợp các thuộc tính ngữ cảnh (*contextual feature*): $C = \{C_1, \ldots, C_t\}$. Khi đó, một ngữ cảnh sẽ được định nghĩa như một vectơ $c = (c_1, \ldots, c_t)$, trong đó, c_i là giá trị của thuộc tính ngữ cảnh C_i. Ví dụ, trong một hệ thống tư vấn có xét đến tập hợp gồm hai thuộc tính ngữ cảnh C = { *location*, *day_type* } thì những ngữ cảnh có thể xảy ra là (*at cinema*, *weekend*), (*at home*, *workday*), ...

Với phương pháp CF truyền thống, các hệ thống tư vấn chỉ khai thác một cộng đồng duy nhất của NSD u theo tiêu chí ratings, $G^{(ratings)}(u)$, để dự đoán đánh giá của NSD này đối với một item i ***trong tất cả các ngữ cảnh***. Trái lại, luận văn sẽ dựa trên giả thiết rằng mức độ quan trọng (hay mức độ phù hợp) của từng tiêu chí $\alpha_i \in$ A đối với một ngữ cảnh cụ thể nhìn chung là khác nhau.

Yếu tố ngữ cảnh sẽ được tích hợp vào mô hình dựa trên định nghĩa: $G^{(\alpha)}(u, i, c)$ là cộng đồng chứa tất cả những người "tương đồng" với NSD u dựa trên

tiêu chí α, và những người này đã từng đánh giá item *i* trong ngữ cảnh *c*. Tương tự, cộng đồng $G^{(\alpha)}(u, c)$ được định nghĩa gồm tất cả các láng giềng của NSD *u* theo tiêu chí α mà đã có đánh giá ít nhất một item trong ngữ cảnh *c*. Chúng ta có thể dễ dàng thấy rằng: $G^{(\alpha)}(u, i, c) \subseteq G^{(\alpha)}(u, c) \subseteq G^{(\alpha)}(u)$.

Giả sử chúng ta có một thuật toán tư vấn $\mathcal{T}$ có thể áp dụng cho hệ thống CARS và vấn đề ở đây là $\mathcal{T}$ sẽ khai thác cộng đồng theo tiêu chí α nào của NSD *u* để tạo danh sách tư vấn? Dựa trên giả thiết vai trò của từng tiêu chí $\alpha_i \in$ A trong một ngữ cảnh *c* cụ thể là không giống nhau, luận văn đề xuất khai thác một thứ tự ưu tiên trên tập hợp các tiêu chí *A* ứng với mỗi ngữ cảnh *c*. Thứ tự ưu tiên này có thể được thể hiện bằng một quan hệ tiền thứ tự đơn giản trên tập *A* như sau:

$$\forall \alpha_i, \alpha_j \in A, (\alpha_i \prec_c \alpha_j) \Leftrightarrow error\_rate(\alpha_i, c) \leq error\_rate(\alpha_j, c) \qquad (3.3)$$

trong đó $error\_rate(\alpha_i, c)$ là tỷ lệ sai số của thuật toán $\mathcal{T}$ đã được chọn áp dụng trên cộng đồng $G^{(\alpha i)}(u, i, c)$ để phát sinh các tư vấn trong ngữ cảnh *c*.

Lưu ý:

- Giá trị $error\_rate(\alpha_i, c)$ có thể được tính toán trong bước huấn luyện (training phase) của thuật toán $\mathcal{T}$.
- $(\alpha_i \prec_c \alpha_j)$ có nghĩa là tiêu chí α_i *ưu tiên* (hay phù hợp) hơn so với tiêu chí α_j trong việc xây dựng cộng đồng để đưa ra những tư vấn trong ngữ cảnh *c*. Khi đó, ta cũng nói rằng cộng đồng $G^{(\alpha i)}(u)$ ưu tiên hơn cộng đồng $G^{(\alpha j)}(u)$ trong ngữ cảnh *c*, hay cộng đồng $G^{(\alpha i)}(u, i, c)$ ưu tiên hơn cộng đồng $G^{(\alpha j)}(u, i, c)$ trong ngữ cảnh *c*.

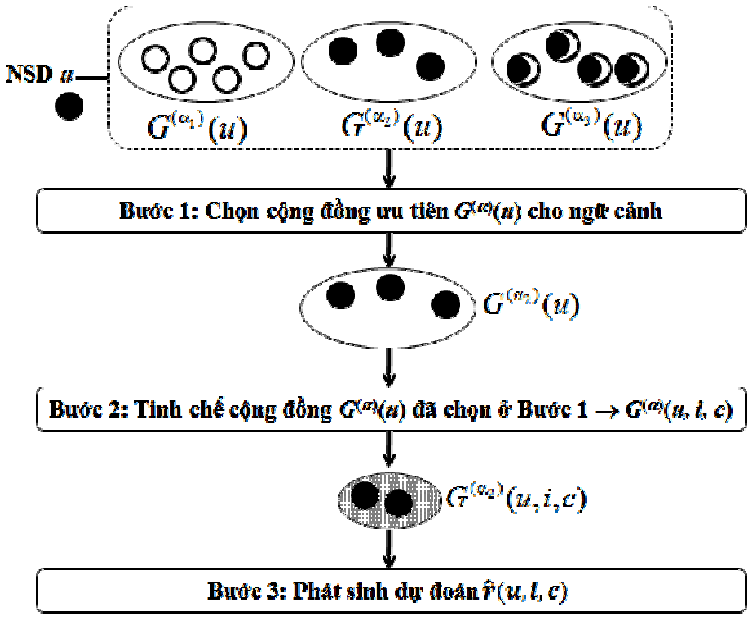
3.2 Các thuật toán đề xuất cho hệ thống CARS

Dựa trên mô hình mở rộng đã được trình bày ở phần trên, luận văn đề xuất ba thuật toán để tạo ra các tư vấn có xét đến yếu tố ngữ cảnh trong các hệ thống CARS. Thứ nhất, thuật toán CRMC được xây dựng dựa trên ý tưởng cơ bản là đi tìm cộng đồng $G^{(\alpha)}(u)$ của NSD u phù hợp nhất cho một ngữ cảnh cụ thể để áp dụng các công thức dự đoán mức độ ưa thích item i của NSD u trong ngữ cảnh c, ký hiệu $\hat{r}(u,i,c)$. Tiếp đến, do CRMC cũng được dựa trên phương pháp CF cho nên thuật toán này vẫn sẽ gặp phải vấn đề dữ liệu thưa (xem phần 2.1.2.1) và do đó, nhằm mục đích cải thiện chất lượng của tư vấn, luận văn đề xuất hai thuật toán CRESC và CREOC được xem như là sự mở rộng của CRMC bằng cách áp dụng các quy trình bổ sung thêm láng giềng của NSD u cho cộng đồng ưu tiên $G^{(\alpha)}(u)$ trong trường hợp cộng đồng này có quá ít NSD đánh giá item i trong ngữ cảnh c. Sự khác biệt chính giữa hai thuật toán CRESC và CREOC là các phương pháp được áp dụng trong việc bổ sung thêm láng giềng từ những cộng đồng khác của NSD u vào cộng đồng ưu tiên $G^{(\alpha)}(u)$.

3.2.1 *Thuật toán CRMC*

Nhìn chung, một hệ thống CARS khi áp dụng kỹ thuật CF thường chỉ dựa vào một cộng đồng duy nhất được xây dựng trên tiêu chí ratings để dự đoán mức độ ưa thích của NSD đối với các items trong tất cả các ngữ cảnh. Khác với những hệ thống trên, thuật toán CRMC (*Context-aware Recommendation based on Multi-criteria Communities*) dự đoán ratings của NSD trong một ngữ cảnh cho trước bằng cách dựa vào các cộng đồng được xây dựng từ nhiều tiêu chí khác nhau. Hay nói cụ thể hơn là thuật toán sẽ dựa vào cộng đồng được xây dựng từ tiêu chí có độ ưu tiên cao nhất trong ngữ cảnh đó. Trên nguyên tắc, với những ngữ cảnh khác nhau thì tiêu chí ưu tiên nhất cũng sẽ khác nhau. Ví dụ, khi NSD u muốn xem phim ở rạp vào những ngày làm việc trong tuần thì cộng đồng được xây dựng theo tiêu chí nhóm tuổi được xác định là phù hợp nhất để đưa ra tư vấn, trong khi đối với ngữ

cảnh ở nhà vào cuối tuần thì cộng đồng được hình thành từ những người có cùng sở thích về thể loại phim trở nên phù hợp hơn để phát sinh tư vấn. Thuật toán CRMC dùng để tính toán giá trị của $\hat{r}(u,i,c)$, là rating dự đoán của NSD u đối với item i trong ngữ cảnh c, sẽ có ba bước như minh họa trong Hình 3-1. Lưu ý, hệ thống CARS sẽ quyết định có tư vấn item i cho NSD u hay không là tùy thuộc vào giá trị của $\hat{r}(u,i,c)$: nếu giá trị này vượt qua một ngưỡng đã chọn thì item i sẽ được hệ thống tư vấn cho NSD u.



Hình 3-1. Các bước trong thuật toán CRMC.

Bước 1: Chọn cộng đồng ưu tiên cho ngữ cảnh. Mục tiêu của bước đầu tiên này là xác định tiêu chí α phù hợp nhất để dựa vào đó mà đưa ra tư vấn cho ngữ cảnh đang xét. Giả sử sau khi thuật toán CRMC đã được xây dựng xong thì nó có thể được áp dụng trên một tập dữ liệu huấn luyện để đưa ra một thứ tự ưu tiên cho các tiêu chí trong từng ngữ cảnh cụ thể dựa vào quan hệ tiền thứ tự đã trình bày trong (3.3). Nói một cách cụ thể, hệ thống CARS sẽ lần lượt xem mỗi $\alpha_j \in A$ là tiêu chí ưu tiên để áp dụng các bước 2 và 3 của CRMC (xem phần tiếp theo) để tính toán $\hat{r}(u,i,c)$ và ước lượng sai số so với rating thực sự $r(u, i, c)$ trong tập dữ liệu thử. Từ đó, hệ thống sẽ tích lũy giá trị cho $error\_rate(\alpha_j, c)$ và có thể sắp xếp các tiêu chí α_j

$\in A$ theo quan hệ tiền thứ tự (3.3). Như vậy, cộng đồng $G^{(\alpha)}(u)$ được xây dựng theo tiêu chí α có độ ưu tiên cao nhất (so với các tiêu chí còn lại) sẽ là kết quả của bước đầu tiên này.

Lưu ý rằng những thứ tự ưu tiên trên tập hợp tiêu chí A cho từng ngữ cảnh có thể được tính toán trước ở bước tiền xử lý và được lưu trữ trên bộ nhớ thứ cấp. Bên cạnh đó, các cộng đồng $G^{(\alpha)}(u)$ cũng có thể được xây dựng offline theo các phương pháp khác nhau tùy thuộc vào bản chất của tiêu chí α. Nếu α là một tiêu chí đơn giản trong việc gom nhóm NSD như các thuộc tính về nhân thân thì cộng đồng $G^{(\alpha)}(u)$ của NSD u có thể được xác định thông qua quan hệ $\mathcal{R}^{(\alpha)}$ được định nghĩa trong (3.1). Ngược lại, đối với các tiêu chí α phức tạp như thể loại yêu thích hay ratings, nghĩa là nếu muốn gom nhóm NSD theo tiêu chí α thì cần phải trải qua một số bước thực hiện, và một số độ đo như cosine, khoảng cách Euclidean hay tương quan Pearson có thể được sử dụng để tính toán sự tương đồng giữa những NSD với nhau [8].

Bước 2: Tinh chế cộng đồng ưu tiên $G^{(\alpha)}(u)$ đã chọn ở Bước 1. Sau khi đã lựa chọn được cộng đồng ưu tiên $G^{(\alpha)}(u)$, tức là cộng đồng phù hợp nhất cho ngữ cảnh c, thì ở bước này, $G^{(\alpha)}(u)$ sẽ được tinh chế, chọn lọc lại bằng cách xét thêm yếu tố ngữ cảnh. Điều đó có nghĩa là thay vì sử dụng tất cả các láng giềng của NSD u thì bước này chỉ giữ lại những láng giềng nào đã đánh giá item i trong ngữ cảnh c. Lưu ý, $G^{(\alpha)}(u, i, c)$ chính là cộng đồng sau khi đã được tinh chế từ $G^{(\alpha)}(u)$, chỉ còn lại những láng giềng của NSD u dựa trên tiêu chí α và đã đánh giá item i trong ngữ cảnh c. Việc dự đoán rating $\hat{r}(u,i,c)$ của NSD u ở bước tiếp theo sẽ được dựa trên ratings của những NSD thuộc cộng đồng đã được tinh chế này.

Bước tinh chế này là không thể thiếu trong quy trình tư vấn vì trong thực tế chúng ta khó có thể xây dựng sẵn các cộng đồng theo từng ngữ cảnh. Có hai lý do chính của sự khó khăn này. Thứ nhất, thông thường các cộng đồng của NSD sẽ gắn với sở thích dài hạn của NSD đó, tức dựa trên những yếu tố khá bền vững, ít bị tác

động bởi ngữ cảnh. Ngoài ra, nếu số lượng ngữ cảnh càng nhiều thì việc xây dựng trước những cộng đồng theo ngữ cảnh sẽ đòi hỏi một lượng bộ nhớ rất lớn để có thể lưu trữ và duy trì danh sách các cộng đồng theo từng ngữ cảnh trong khi việc tinh chế (online) thì đơn giản và không tốn nhiều chi phí.

Bước 3: Phát sinh dự đoán. Xuất phát từ quan điểm là trong một số trường hợp, mức độ ưa thích của NSD *u* đối với một số items sẽ thay đổi theo ngữ cảnh, và ngược lại, cũng sẽ có những trường hợp mà mức độ ưa thích của NSD *u* đối với những items lại không bị ảnh hưởng bởi ngữ cảnh. Ví dụ, một NSD có thể luôn cảm thấy thích thú khi xem bộ phim Forest Gump vào bất cứ hoàn cảnh (tâm trạng) nào, nhưng lại chỉ thích xem phim Thời đại tân kỳ của danh hài Charlot khi đang vui mà không thích xem khi đang buồn. Theo giả định này, luận văn xem những sở thích bị thay đổi nhất thời do tác động của yếu tố ngữ cảnh là sở thích ngắn hạn (short-term preference), và ngược lại, những sở thích mang tính bền vững, không bị tác động nhiều theo ngữ cảnh chính là sở thích dài hạn (long-term preference). Do đó, luận văn đề xuất công thức dự đoán mức độ ưa thích của NSD *u* dành cho item *i* trong ngữ cảnh *c* sẽ là sự kết hợp của hai thành phần: phần dự đoán chung mang tính độc lập ngữ cảnh (hay nói cách khác là không xét đến yếu tố ngữ cảnh) thể hiện cho sở thích dài hạn và phần dự đoán có xét đến yếu tố ngữ cảnh, thể hiện cho sở thích ngắn hạn. Việc kết hợp hai thành phần này được thực hiện thông qua tham số γ như trong công thức (3.4) bên dưới.

$$\hat{r}(u,i,c) = \gamma.\hat{r}_{Ctx}^{(\alpha)}(u,i,c) + (1-\gamma).\hat{r}_{nonCtx}(u,i) \tag{3.4}$$

Đối với thành phần dự đoán không xét đến yếu tố ngữ cảnh $\hat{r}_{nonCtx}(u,i)$, thuật toán CRMC áp dụng MF (xem 2.1.2.3) là một trong những phương pháp thành công nhất của cách tiếp cận latent factor models [13]. Ý tưởng chính để thực hiện phần này là ma trận đánh giá ban đầu *R* sẽ được thừa số hóa thành hai ma trận thành phần mà không cần xét đến yếu tố ngữ cảnh để dự đoán mức độ ưa thích chung của NSD *u* cho item *i* trong mọi ngữ cảnh. Các thành phần của vectơ $\vec{q}_i$ trong (3.5) thể hiện

mức độ quan trọng của thành phần đặc trưng ngầm đối với item *i*, và các thành phần của $\vec{p}_u$ thể hiện mức độ ảnh hưởng của thành phần đặc trưng ngầm đối với sở thích của NSD.

$$\hat{r}_{nonCtx}(u,i) = \vec{p}_u.\vec{q}_i^T \tag{3.5}$$

Thành phần $\hat{r}_{Ctx}^{(\alpha)}(u,i,c)$ có xét đến yếu tố ngữ cảnh trong công thức (3.4) sẽ phụ thuộc vào các đánh giá của cộng đồng $G^{(\alpha)}(u, i, c)$ đã được tinh chế ở Bước 2. Công thức tính toán cho phần dự đoán có xét đến yếu tố ngữ cảnh này sẽ tùy thuộc vào đặc điểm của cộng đồng ưu tiên $G^{(\alpha)}(u)$ được chọn ở Bước 1. Nếu như cộng đồng ưu tiên này được xây dựng ban đầu từ một trong các tiêu chí đơn giản, ký hiệu α+, như các thuộc tính về nhân thân thì giá trị của $\hat{r}_{Ctx}^{(\alpha+)}(u,i,c)$ sẽ được ước lượng bằng cách lấy trung bình của tất cả các đánh giá từ những láng giềng của NSD *u* thuộc cộng đồng đã được tinh chế ở Bước 2 dành cho item *i* trong ngữ cảnh *c* như trong công thức (3.6), với *r*(*u*', *i*, *c*) là rating của láng giềng *u'* dành cho item *i* trong ngữ cảnh *c*.

$$\hat{r}_{Ctx}^{(\alpha+)}(u,i,c) = \frac{\sum_{u' \in G^{(\alpha)}(u,i,c)} r(u',i,c)}{|G^{(\alpha)}(u,i,c)|} \tag{3.6}$$

Trong trường hợp cộng đồng được chọn ở Bước 1 đã được xây dựng từ một trong những tiêu chí phức tạp, ký hiệu α*, như favorite genre hay ratings thì khi đó, thành phần có xét đến yếu tố ngữ cảnh được tính toán dựa theo công thức tư vấn của phương pháp CF truyền thống:

$$\hat{r}_{Ctx}^{(\alpha*)}(u,i,c) = \frac{\sum_{u' \in G^{(\alpha)}(u,i,c)} sim(u,u').r(u',i,c)}{\sum_{u' \in G^{(\alpha)}(u,i,c)} |sim(u,u')|} \tag{3.7}$$

trong đó, *sim*(*u*, *u*') chính là mức độ tương đồng giữa hai NSD *u* và *u'*. Đoạn mã giả của thuật toán CRMC được trình bày trong Hình 3-2.

Thuật toán CRMC

Input: NSD u, item i, ngữ cảnh c

Output: giá trị dự đoán $\hat{r}(u,i,c)$

$\alpha = \arg\min_{\alpha' \in A}[error\_rate(\alpha', c)]$

Tinh chế cộng đồng $G^{(\alpha)}(u)$ để tạo $G^{(\alpha)}(u, i, c)$

Tính $\hat{r}_{nonCtx}(u,i)$ như trong (3.5)

if (α là tiêu chí đơn giản)

then Tính $\hat{r}_{Ctx}^{(\alpha+)}(u,i,c)$ như trong (3.6)

else Tính $\hat{r}_{Ctx}^{(\alpha*)}(u,i,c)$ như trong (3.7)

Tính $\hat{r}(u,i,c)$ như trong (3.4)

Return $\hat{r}(u,i,c)$

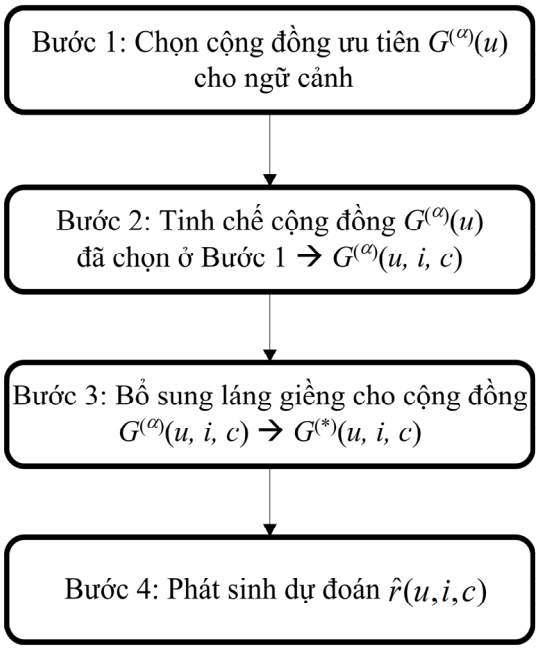
Hình 3-2. Mô tả thuật toán CRMC.

Xét mối tương quan giữa hai thành phần dự đoán có yếu tố ngữ cảnh và độc lập ngữ cảnh, việc sử dụng tham số γ trong (3.4) hướng theo giả thuyết nếu kích thước của cộng đồng $G^{(\alpha)}(u, i, c)$ càng lớn thì mức độ ảnh hưởng của ngữ cảnh c đến mức độ ưa thích của NSD u cho item i càng lớn và ngược lại. Nói một cách chi tiết, nếu số lượng NSD có trong cộng đồng $G^{(\alpha)}(u, i, c)$ vượt qua một ngưỡng λ_{max} thì giá trị của tham số kết hợp γ có thể được điều chỉnh tiến về 1, và ngược lại, nếu $|G^{(\alpha)}(u, i, c)|$ nhỏ hơn một ngưỡng λ_{min} thì giá trị γ có thể được thay đổi để tiến về 0. Trong trường hợp $|G^{(\alpha)}(u, i, c)|$ nằm trong đoạn $[\lambda_{min}, \lambda_{max}]$ thì giá trị của γ có thể được gán trong khoảng lân cận của 0.5.

3.2.2 *Thuật toán CRESC*

Trong thuật toán CRMC, khi mà số lượng NSD thuộc cộng đồng sau khi được tinh chế $G^{(\alpha)}(u, i, c)$ quá ít thì có thể làm cho độ chính xác của việc tính toán

$\hat{r}(u,i,c)$ bị giảm đi đáng kể. Do đó, luận văn đề xuất thuật toán CRESC (*Context-aware Recommendation based on Enrichment from Similar multi-criteria Communities*) như là một sự mở rộng của CRMC với việc tích hợp quy trình bổ sung láng giềng cho cộng đồng $G^{(\alpha)}(u, i, c)$ trong những trường hợp cần thiết trước khi phát sinh ra dự đoán. Thuật toán CRESC có bốn bước được minh họa như trong Hình 3-3 với Bước 1 và Bước 2 giống với hai bước đầu tiên trong thuật toán CRMC.



Hình 3-3. Các bước trong thuật toán CRESC.

Trong Bước 3, khi kích thước của cộng đồng đã được tinh chế $G^{(\alpha)}(u, i, c)$ nhỏ hơn ngưỡng λ_{max}, thuật toán CRESC sẽ bổ sung vào cộng đồng $G^{(\alpha)}(u, i, c)$ những láng giềng thuộc các cộng đồng khác của NSD *u*. Việc bổ sung này sẽ dựa trên những cộng đồng có sự tương đồng nhiều nhất với $G^{(\alpha)}(u, i, c)$. Như vậy, thuật toán CRESC cần phải xác định mức độ tương đồng giữa cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ với các cộng đồng khác $G^{(\alpha')}(u, i, c)$, ($\alpha' \neq \alpha$). Mức độ tương đồng giữa hai cộng đồng sẽ được tính toán dựa trên độ lệch giữa rating trung bình trong ngữ cảnh *c* của

hai cộng đồng đang xét như trong công thức (3.8) với $\overline{r(u,\alpha,c)}$ là rating trung bình của cộng đồng $G^{(\alpha)}(u, c)$ gồm tất cả các láng giềng của NSD u theo tiêu chí α mà đã có đánh giá các items trong ngữ cảnh c.

$$sim(G^{(\alpha)}(u,i,c),G^{(\alpha')}(u,i,c)) = -|\overline{r(u,\alpha,c)} - \overline{r(u,\alpha',c)}| \quad (3.8)$$

Lưu ý, giá trị tuyệt đối trong công thức (3.8) cho độ lệch càng nhỏ thì độ tương đồng giữa hai cộng đồng càng lớn.

Luận văn định nghĩa $G^{(*)}(u, i, c)$ là cộng đồng đã được bổ sung những NSD từ các cộng đồng khác "gần gũi" với $G^{(\alpha)}(u, i, c)$. Đầu tiên, $G^{(*)}(u, i, c)$ được khởi tạo bằng những NSD thuộc cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$. Sau đó, thuật toán CRESC sẽ đi tìm một cộng đồng $G^{(\alpha')}(u, i, c)$ mà có độ tương đồng gần nhất với $G^{(\alpha)}(u, i, c)$ và gộp những NSD trong cộng đồng $G^{(\alpha')}(u, i, c)$ vào $G^{(*)}(u, i, c)$. Quy trình bổ sung này sẽ được lặp lại cho đến khi $G^{(*)}(u, i, c)$ tập hợp đủ số lượng NSD cần thiết (xem Hình 3-4).

Bổ sung láng giềng trong thuật toán CRESC

Input: cộng đồng $G^{(\alpha)}(u, i, c)$, tập các tiêu chí A, ngưỡng λ_{max}

Output: cộng đồng sau khi đã được bổ sung $G^{(*)}(u, i, c)$

$\alpha = \arg\min_{\alpha' \in A}[error\_rate(\alpha', c)]$

Tinh chế cộng đồng $G^{(\alpha)}(u)$ để tạo $G^{(\alpha)}(u, i, c)$

$G^{(*)}(u, i, c) = G^{(\alpha)}(u, i, c)$

while $|G^{(*)}(u, i, c)| < \lambda_{max}$ **and** $A \neq \varnothing$

$\alpha' = \arg\max_{\alpha_j \in A \setminus \{\alpha\}}[sim(G^{(\alpha)}(u,i,c), G^{(\alpha_j)}(u,i,c))]$

$G^{(*)}(u, i, c) = G^{(*)}(u, i, c) \cup G^{(\alpha')}(u, i, c)$

$A = A \setminus \{\alpha'\}$

end while

Return $G^{(*)}(u, i, c)$

Hình 3-4. Bước bổ sung láng giềng trong thuật toán CRESC.

Cuối cùng, bước 4 trong thuật toán CRESC cũng tương tự như Bước 3 của thuật CRMC, ngoại trừ việc cộng đồng $G^{(*)}(u, i, c)$ sẽ được sử dụng thay cho $G^{(\alpha)}(u, i, c)$ để tính toán giá trị của $\hat{r}(u,i,c)$.

3.2.3 *Thuật toán CREOC*

Thuật toán thứ ba mà luận văn đề xuất là thuật toán CREOC (*Context-aware Recommendation based on Enrichment from Ordered multi-criteria Communities*) với ý tưởng xây dựng thuật toán tương tự như CRESC. Thuật toán CREOC cũng có bốn bước, trong đó Bước 1, Bước 2, và Bước 4 giống như CRESC. Điểm khác biệt chủ yếu của hai thuật toán này chính là cách thức mà chúng bổ sung thêm láng giềng vào cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ ở Bước 3. Trong thuật toán CREOC, thay vì phải tính toán độ tương đồng giữa hai cộng đồng như trong công thức (3.8), thuật toán này sẽ thực hiện việc bổ sung NSD nhiều lần vào cộng đồng $G^{(*)}(u, i, c)$ bằng cách khai thác hoàn toàn thứ tự ưu tiên theo quan hệ tiền thứ tự $\prec_c$ trên tập các tiêu chí A. Lưu ý, thứ tự ưu tiên nói trên đã được xác định trước, cụ thể là trong bước huấn luyện của thuật toán.

Thuật toán CREOC cũng dựa trên giả định là nhằm tăng độ tin cậy cho các dự đoán có xét đến yếu tố ngữ cảnh thì số lượng NSD trong cộng đồng $G^{(*)}(u, i, c)$ ít nhất phải đạt đến ngưỡng λ_{max}. Do đó, khi xảy ra sự thiếu hụt NSD trong cộng đồng $G^{(*)}(u, i, c)$ thì thuật toán CREOC sẽ thực hiện việc gộp thêm những NSD trong $G^{(\alpha')}(u, i, c)$ vào $G^{(*)}(u, i, c)$ theo thứ tự độ ưu tiên từ cao đến thấp theo $\prec_c$ cho đến khi $G^{(*)}(u, i, c)$ đã tập hợp đủ số lượng NSD cần thiết.

Ví dụ, giả sử sau bước huấn luyện, với ngữ cảnh c = (at cinema, with friends) thì ta có được độ ưu tiên của các tiêu chí đối với ngữ cảnh c như sau:

$$\text{age} \prec_c \text{favorite genre} \prec_c \text{ratings} \prec_c \text{occupation}$$

Đầu tiên, $G^{(*)}(u, i, c)$ sẽ được khởi tạo bằng cộng đồng có độ ưu tiên cao nhất trong ngữ cảnh c là $G^{(age)}(u, i, c)$. Tiếp đến, nếu số lượng NSD trong $G^{(*)}(u, i, c)$ nhỏ hơn λ_{max} thì cộng đồng này sẽ được bổ sung thêm những NSD trong cộng đồng $G^{(\text{favorite genre})}(u, i, c)$. Quy trình bổ sung NSD cho $G^{(*)}(u, i, c)$ có thể được tiếp tục thực hiện với $G^{(\text{ratings})}(u, i, c)$ và $G^{(\text{occupation})}(u, i, c)$ nếu cần thiết.

Ưu điểm chính của thuật toán CREOC so với CRESC chính là CREOC tránh được việc phải tính toán độ tương đồng giữa cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ và những cộng đồng khác $G^{(\alpha')}(u, i, c)$.

Tóm lại, với việc áp dụng quy trình bổ sung NSD từ các cộng đồng đa tiêu chí thì các thuật toán CRESC và CREOC sẽ góp phần làm giảm bớt ảnh hưởng xấu của vấn đề dữ liệu thưa trong các hệ thống CARS.

Chương 4. **Thực nghiệm**

Chương 4 sẽ trình bày chi tiết những thực nghiệm mà luận văn đã tiến hành nhằm mục đích trước tiên là kiểm chứng giả thiết về vai trò của các cộng đồng đa tiêu chí trong các hệ thống CARS và đồng thời để đánh giá hiệu quả của ba thuật toán CRMC, CRESC và CREOC đã được đề xuất trong Chương 3.

Luận văn đã tiến hành hai thực nghiệm để kiểm chứng sự lựa chọn các cộng đồng đa tiêu chí trong từng ngữ cảnh có ảnh hưởng hay không đến kết quả tư vấn, và đồng thời để đánh giá chất lượng của các thuật toán đề xuất. Nói một cách cụ thể hơn là các thực nghiệm nhằm mục đích trả lời cho ba câu hỏi sau:

Q1. Các cộng đồng được xây dựng từ một tiêu chí ratings duy nhất có phải luôn luôn tạo ra những tư vấn tốt nhất trong mọi ngữ cảnh?

Q2. Nếu ratings không thể chiến thắng ở mọi ngữ cảnh thì có tiêu chí nào khác có thể thống trị đối với mọi ngữ cảnh không? hay là các tiêu chí sẽ có những mức độ ảnh hưởng khác nhau theo từng ngữ cảnh?

Q3. Hiệu quả (độ chính xác) của các thuật toán được đề xuất so với một số thuật toán cùng loại hiện có là như thế nào?

4.1 Qui trình thực nghiệm

Quá trình chuẩn bị cho các thực nghiệm trong luận văn bao gồm: tiền xử lý bộ dữ liệu, thiết lập giá trị các tham số, lựa chọn các thuật toán so sánh cùng với độ đo để đánh giá hiệu quả của các thuật toán.

4.1.1 *Dữ liệu*

Các thực nghiệm trong luận văn được thực hiện trên bộ dữ liệu MovieLens [15]. Đây được xem là bộ dữ liệu chuẩn để thử nghiệm trong lãnh vực RS. Bộ dữ liệu này bao gồm 100 000 ratings được cung cấp bởi 943 NSD trên 1682 bộ phim thuộc 19 thể loại khác nhau. Để có thể thu nhận được những kết quả tin cậy, luận văn đã sử dụng bộ dữ liệu huấn luyện (training set) và bộ dữ liệu kiểm thử (test set) *đã được phân chia sẵn* của MovieLens, trong đó 80% dữ liệu dành cho huấn luyện và 20% dành cho kiểm thử chất lượng dự đoán.

4.1.2 *Phương pháp thực hiện*

Bộ dữ liệu MovieLens ban đầu không chứa các thông tin ngữ cảnh cụ thể, do đó để thử nghiệm những thuật toán tư vấn có ngữ cảnh, luận văn đã áp dụng phương pháp suy luận ngữ cảnh của Said và cộng sự đã được sử dụng trong [21]. Trong phương pháp này, thuộc tính ngữ cảnh *location* về địa điểm mà NSD đã xem phim (at cinema hay at home) được suy luận thông qua việc kết hợp giữa ngày khởi chiếu bộ phim và thời điểm mà NSD đánh giá bộ phim đó. Việc suy luận dựa vào giả định là nếu bộ phim được đánh giá trong vòng hai tháng kể từ ngày khởi chiếu bộ phim thì xem như bộ phim đã được NSD xem ở rạp; còn nếu như sau hai tháng thì cho rằng bộ phim đã được NSD xem ở nhà. Hai thuộc tính ngữ cảnh *day type* (workday hay weekend), *season* thể hiện thời điểm mà NSD xem phim, và giá trị sẽ được suy luận dựa vào thời điểm phát sinh (timestamp) của ratings. Sau khi thực hiện quá trình suy luận ngữ cảnh như trên, qui trình thực nghiệm có được tập hợp các thuộc

tính về ngữ cảnh là {location, day type, season} và tập các ngữ cảnh C = {(workday, at home, spring), (weekend, at cinema, winter), …}.

Sau khi áp dụng phương pháp suy diễn ngữ cảnh của Said và cộng sự, luận văn cũng đã xem xét sự phân bố ratings của NSD theo từng ngữ cảnh và đã loại bỏ một số ngữ cảnh mà có quá ít ratings được NSD cung cấp trong những ngữ cảnh đó. Theo quan sát thực tế cho thấy có rất ít ratings của NSD trong ngữ cảnh "at cinema" và đại đa số là "at home". Do đó, thuộc tính location không còn đóng vai trò quan trọng và sẽ không được xét trong thực nghiệm. Việc loại bỏ thuộc tính location sẽ không ảnh hưởng đến kết quả thực nghiệm bởi vì luận văn chỉ quan tâm đến *số lượng ngữ cảnh* hơn là quan tâm đến số lượng các thuộc tính về ngữ cảnh cũng như không quan tâm đến ngữ nghĩa của các ngữ cảnh. Như vậy, sau bước suy luận theo phương pháp vừa nêu, bộ dữ liệu MovieLens dùng trong các thực nghiệm của luận văn có liên quan đến 8 ngữ cảnh, gồm: c_1 = (workday, spring), c_2 = (workday, summer), c_3 = (workday, fall), c_4 = (workday, winter), c_5 = (weekend, spring), c_6 = (weekend, summer), c_7 = (weekend, fall) và c_8 = (weekend, winter).

Tiếp theo, luận văn sẽ tóm tắt quá trình phân hoạch các cộng đồng đa tiêu chí được sử dụng trong thực nghiệm, với bốn tiêu chí được định nghĩa từ bộ dữ liệu MovieLens là: age, occupation, genre và ratings. Đối với tiêu chí *age*, tập hợp tất cả 943 NSD được chia thành 7 cộng đồng theo *sự phân nhóm từ trước* của nhà cung cấp MovieLens: dưới 18 tuổi, 18-24, 25-34, 35-44, 45-49, 50-55, và trên 55 tuổi. Đối với tiêu chí *occupation*, NSD được phân nhóm vào 21 loại nghề nghiệp dựa vào thông tin của NSD có sẵn trong bộ dữ liệu.

Còn lại hai tiêu chí phức tạp là *genre* và *ratings*, các cộng đồng được xây dựng thông qua hai bước gom nhóm, được mô tả chi tiết trong [16]. Đối với tiêu chí genre, mỗi NSD sở hữu một vectơ biểu diễn cho mức độ yêu thích của người này đối với 19 thể loại phim. Như vậy, chúng ta có một vectơ 19 chiều với mỗi chiều là giá trị trọng số $w(u, g_i)$ thể hiện mức độ yêu thích của NSD u đối với thể loại phim g_i. Trọng số $w(u, g_i)$ này được tính toán tùy thuộc vào hai thành phần chính đó là số

lượng phim thuộc về thể loại g_i mà NSD u đã đánh giá và giá trị trung bình của tất cả các đánh giá đó. Sau đó, độ đo cosine được sử dụng để tính toán sự tương đồng giữa các vectơ trên để xác định những NSD có phải là chung cộng đồng hay không. Đối với tiêu chí ratings, mỗi vectơ đặc trưng cho NSD là một dòng tương ứng trong ma trận đánh giá R. Độ tương quan Pearson được sử dụng để ước lượng độ tương đồng của những NSD trong việc thành lập cộng đồng theo tiêu chí ratings.

Liên quan đến những tham số của các thuật toán, trước hết luận văn đã chọn 20 và 50 lần lượt là các giá trị của hai ngưỡng λ_{min} và λ_{max} vì kích thước của một cộng đồng từ 20 đến 50 người được sử dụng một cách phổ biến trong các công trình [8]. Nếu kích thước của $G^{(\alpha)}(u, i, c)$ trong thuật toán CRMC hay của $G^{(*)}(u, i, c)$ trong hai thuật toán CRESC và CREOC lớn hơn ngưỡng λ_{max} thì giá trị của γ sẽ được chọn trong khoảng (0.5, 1] để thể hiện mức độ ảnh hưởng ít hay nhiều của thành phần ngữ cảnh trong công thức phát sinh dự đoán. Để đảm bảo sự khách quan và hợp lý trong khi so sánh các thuật toán, tham số γ được gán bằng 0.75 như là giá trị trung bình nằm ở giữa đoạn. Tương tự, nếu kích thước của các cộng đồng $G^{(\alpha)}(u, i, c)$ hay $G^{(*)}(u, i, c)$ nhỏ hơn ngưỡng λ_{min} thì giá trị của γ sẽ nằm trong khoảng [0, 0.5) để thể hiện mức độ ảnh hưởng của thành phần độc lập với ngữ cảnh tức là sở thích dài hạn trong việc dự đoán rating của NSD. Luận văn cũng đã chọn giá trị trung bình của đoạn là 0.25 để gán cho tham số γ trong trường hợp này. Ngoài ra, khi kích thước của cộng đồng nằm trong đoạn [λ_{min}, λ_{max}], luận văn giả định tác động của sở thích dài hạn và ngắn hạn là bằng nhau, và do đó luận văn sử dụng giá trị tham số γ là 0.5 để giữ cân bằng cho hai thành phần trên. Đối với các tham số của phương pháp MF được sử dụng cho thành phần dự đoán không ngữ cảnh, tham số điều hòa (regularization), hệ số học (learning rate) và số chiều của các đặc trưng ngầm (latent features) được gán lần lượt 0.015, 0.01 và 10. Lưu ý, đây cũng là các giá trị được sử dụng trong thực nghiệm của các thuật toán so sánh khác.

Về phần các thuật toán so sánh, đầu tiên luận văn sẽ chọn thuật toán context baseline predictor (UIC Baseline) [13]. Thuật toán tư vấn theo ngữ cảnh này là sự

kết hợp đơn giản giữa cách tiếp cận pre-filtering và baseline predictor với công thức dự đoán được thể hiện trong (4.1). Trong công thức này, μ là rating trung bình trên toàn bộ ma trận (overall average rating), $b_u(c)$ và $b_i(c)$ lần lượt thể hiện xu hướng đánh giá của NSD và của item trong ngữ cảnh c.

$$\hat{r}(u,i,c) = \mu + b_i(c) + b_u(c) \tag{4.1}$$

Dựa vào công thức (4.1), chúng ta có thể thấy rằng cho trước một ngữ cảnh *c,* phương pháp này sẽ chỉ chọn ra từ tập ratings ban đầu những ratings nào được đánh giá trong ngữ cảnh *c*, sau đó phát sinh ra ma trận *User* x *Items* chỉ chứa những ratings gắn với ngữ cảnh *c* và cuối cùng là áp dụng công thức 2D baseline predictor cho ma trận trên.

Cuối cùng, hai thuật toán do Odic và cộng sự đề xuất, đã được trình bày chi tiết trong phần 2.2.3, cũng được lựa chọn để so sánh với các thuật toán đề xuất. Sở dĩ, luận văn đã chọn hai thuật toán này là do chúng cùng áp dụng cách tiếp cận MF giống như các thuật toán được đề xuất trong luận văn. Đặc biệt, Odic và cộng sự đã tiến hành thử nghiệm hai thuật toán này và cho thấy kết quả dự đoán khá tốt.

Các thực nghiệm đã được tiến hành trên máy PC với cấu hình chính như sau: CPU Core i3 - 2330M 2.20 GHz, 4 GB RAM.

4.1.3 *Độ đo sử dụng để so sánh hiệu quả của các thuật toán*

Để đánh giá hiệu quả của các thuật toán tư vấn, luận văn sử dụng độ đo *Normalized Root Mean Square Error* (NRMSE) như là tiêu chuẩn đánh giá độ chính xác cho các thuật toán [22]. Trong công thức (4.2), T là bộ dữ liệu kiểm thử, *range_ratings* là phạm vi của thang điểm đánh giá trong bộ dữ liệu (ví dụ như là $r_{max} - r_{min}$), $\hat{r}(u,i,c)$ là rating được dự đoán bởi các thuật toán và $r(u, i, c)$ là rating thật sự mà NSD u đã đánh giá cho item i trong ngữ cảnh c. Trong các thực nghiệm của luận văn, độ chính xác được tính toán trên *top 5*, *top 10*, *top 15*, *top 20*, *top 25* và *top 30* của danh sách tư vấn.

$$NRMSE = \sqrt{\frac{1}{|T|}\sum_{(u,i,c)\in T}\frac{[\hat{r}(u,i,c) - r(u,i,c)]^2}{range\_ratings}} \quad (4.2)$$

NRMSE thực chất là một phiên bản của RMSE, đã được chuẩn hóa bằng phạm vi của thang điểm đánh giá cho nên kết quả xếp hạng các thuật toán dựa theo NRMSE cũng sẽ giống với kết quả mà RMSE xếp hạng các thuật toán.

4.2 Kết quả thực nghiệm

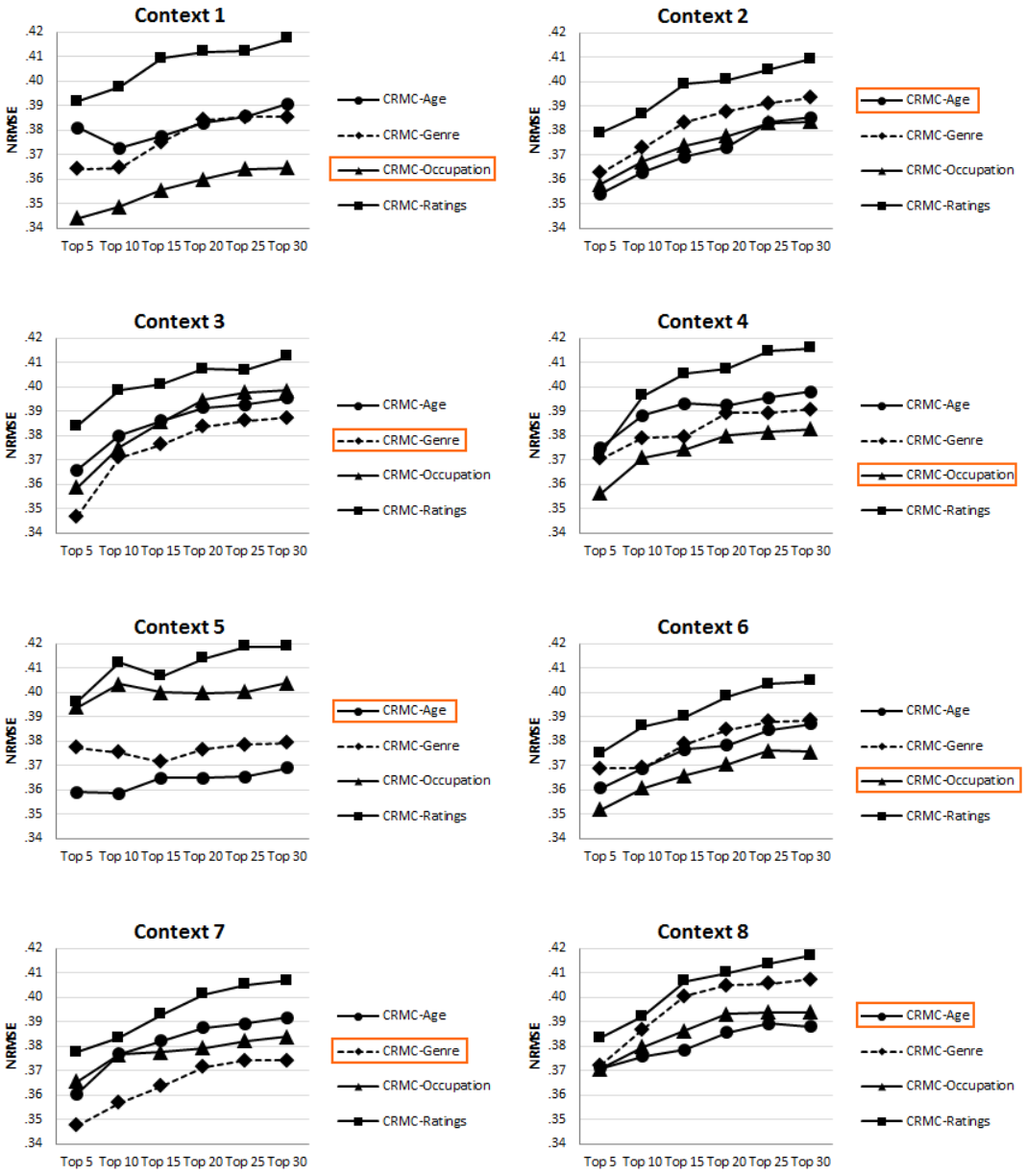
Như đã trình bày ở phần trên, luận văn đã tiến hành hai thực nghiệm để kiểm chứng giả định rằng việc lựa chọn cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ trong từng ngữ cảnh cụ thể sẽ tác động đến chất lượng tư vấn và cũng nhằm để đánh giá sự hiệu quả của ba thuật toán đề xuất trong luận văn. Bảng 4-1 minh họa tốc độ xử lý của các thuật toán CRMC, CRESC và CREOC. Theo các thực nghiệm được tiến hành thì kết quả đạt được là rất khả quan và đáng khích lệ.

Bảng 4-1. Tốc độ xử lý của các thuật toán được đề xuất.

	Training (80 000 ratings)		Testing (20 000 ratings)	
	∑	Trung bình	∑	Trung bình
CRMC	340 phút	0.25 giây / rating	22 phút	0.06 giây / rating
CRESC	340 phút	0.25 giây / rating	27 phút	0.08 giây / rating
CREOC	340 phút	0.25 giây / rating	25 phút	0.07 giây / rating

4.2.1 *Thực nghiệm 1: Phân tích vai trò của các tiêu chí*

Đầu tiên, luận văn áp dụng thuật toán CRMC lần lượt trên các cộng đồng xây dựng từ các tiêu chí age, genre, occupation và ratings để quan sát sự tác động của các tiêu chí lên trên từng ngữ cảnh. Kết quả thực nghiệm trong Hình 4-1 cho thấy rằng trong các ngữ cảnh, cộng đồng xây dựng từ tiêu chí ratings không đạt được kết quả tư vấn tốt hơn những cộng đồng của tiêu chí khác.



Hình 4-1. Kết quả so sánh các tiêu chí theo thuật toán CRMC.

Lưu ý, tiêu chí nào cho giá trị NRMSE càng nhỏ thì có nghĩa là tiêu chí đó đạt chất lượng tư vấn càng tốt. Kết quả trong Hình 4-1 cho thấy sự thống trị của các tiêu chí thay đổi trong từng ngữ cảnh. Cụ thể là cộng đồng xây dựng từ tiêu chí occupation đạt kết quả tốt nhất ở những ngữ cảnh 1, 4 và 6; cộng đồng của tiêu chí

genre thống trị ở hai ngữ cảnh 3 và 7, trong khi với những ngữ cảnh còn lại như 2, 5 và 8 thì cộng đồng theo tiêu chí age đạt được kết quả tốt hơn những tiêu chí khác.

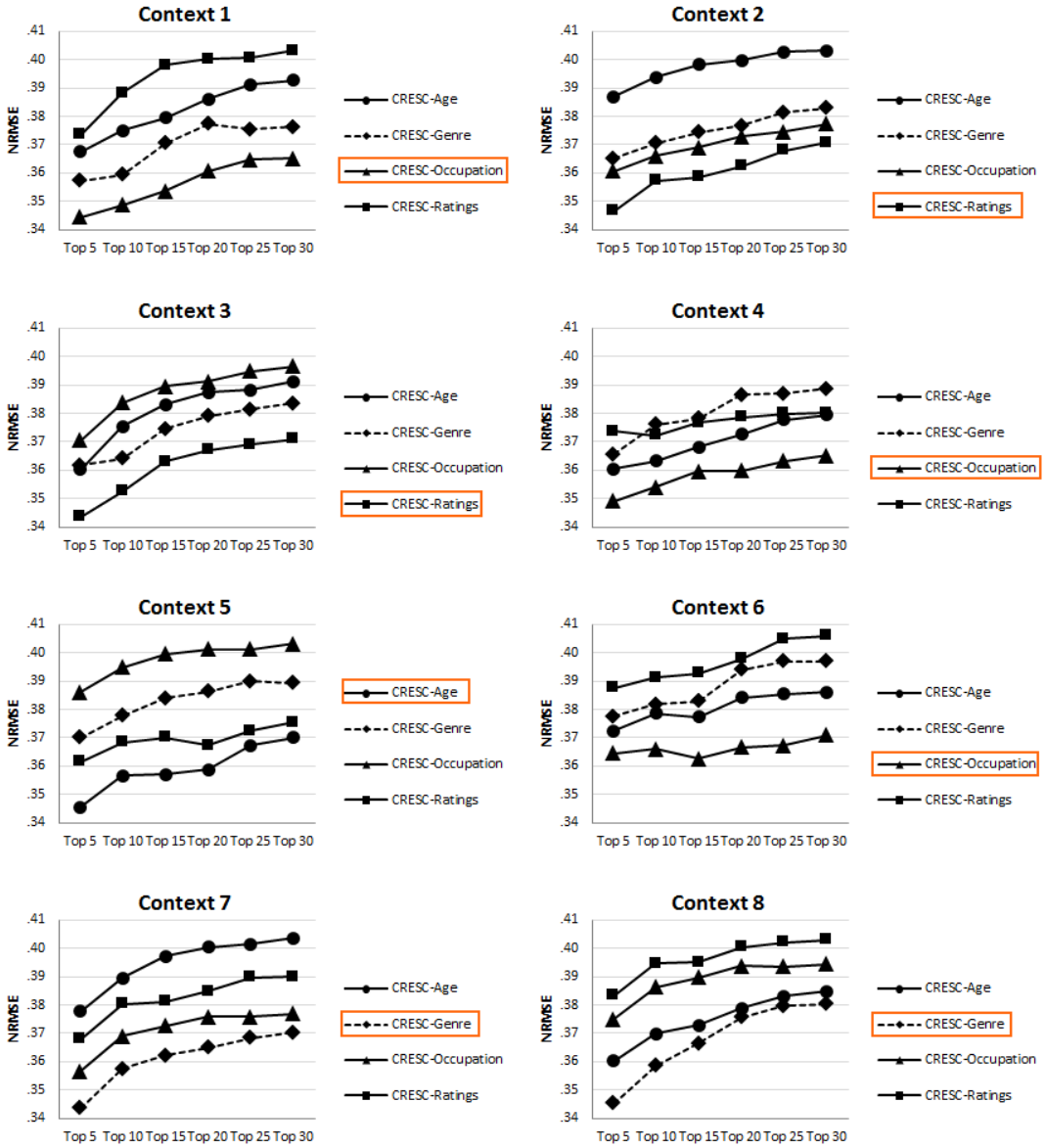
Điều đáng chú ý mà chúng ta có thể ghi nhận là cộng đồng được xây dựng theo tiêu chí ratings không thể chiến thắng ở bất kỳ ngữ cảnh nào trong khi những cộng đồng xây dựng theo các tiêu chí khác thì luân phiên đạt được kết quả tốt nhất ở những ngữ cảnh. Nguyên nhân của hiện tượng thất thế này của tiêu chí ratings là do sự thiếu hụt NSD trong các cộng đồng $G^{(ratings)}(u, i, c)$ là khá nghiêm trọng, vì vậy nếu việc tư vấn chỉ dựa vào duy nhất tiêu chí ratings thì hiệu quả tư vấn sẽ bị giảm sút một cách đáng kể.

Luận văn cũng đã tiến hành thực nghiệm 1 với thuật toán CRESC trên các tiêu chí age, genre, occupation và ratings để quan sát sự tác động của từng tiêu chí đối với sở thích của NSD trong từng ngữ cảnh theo thuật toán CRESC. Kết quả thực nghiệm ở Hình 4-2 cho thấy, cộng đồng khởi tạo từ tiêu chí ratings sau khi được bổ sung thêm láng giềng đã đạt kết quả tốt nhất ở hai ngữ cảnh 2 và 3. Kết quả thực nghiệm mới này đã giúp cho luận văn tin tưởng rằng trong trường hợp cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ có quá ít NSD thì việc áp dụng kỹ thuật bổ sung láng giềng vào cộng đồng như trong thuật toán CRESC hay CREOC sẽ rất có lợi.

Tuy nhiên, cộng đồng khởi tạo từ tiêu chí ratings $G^{(ratings)}(u, i, c)$ không thể đạt được sự thống trị trong những ngữ cảnh còn lại. Một cách chi tiết, trong ba ngữ cảnh 1, 4 và 6 thì cộng đồng ưu tiên được xây dựng từ tiêu chí occupation có kết quả tốt hơn các cộng đồng theo tiêu chí khác. Các cộng đồng theo tiêu chí age thống trị ở ngữ cảnh 5, và theo tiêu chí genre chiến thắng ở hai ngữ cảnh 7 và 8. Như vậy, việc lựa chọn tiêu chí phù hợp nhất để áp dụng trong từng ngữ cảnh cụ thể sẽ giúp cho độ chính xác của thuật toán CRESC được cải thiện hơn là chỉ dựa vào duy nhất tiêu chí ratings.

Để có thể hiểu đầy đủ hơn tại sao các tiêu chí thống trị lại bị luân phiên thay đổi trong từng ngữ cảnh, luận văn đã tiến hành xem xét số liệu chi tiết về các cộng đồng đa tiêu chí. Kết quả cho thấy rằng một tiêu chí α có khả năng trở thành phù

hợp và ưu tiên nhất trong một ngữ cảnh cụ thể khi mà sự thiếu hụt NSD trong cộng đồng $G^{(\alpha)}(u, i, c)$ được xây dựng theo tiêu chí này là không đáng kể. Hay nói theo cách khác, tiêu chí α nên được chọn ưu tiên khi mà bản thân cộng đồng $G^{(\alpha)}(u, i, c)$ đã có đủ số lượng NSD cần thiết cho việc đảm bảo độ tin cậy của dự đoán, hay nếu là khi hệ thống chỉ cần được bổ sung một hoặc hai cộng đồng tương đồng với nó nhất mà thôi.



Hình 4-2. Kết quả so sánh các tiêu chí theo thuật toán CRESC.

Cũng theo sự quan sát số liệu kết quả, luận văn còn nhận thấy rằng chất lượng tư vấn sẽ giảm đi một cách đáng kể nếu như số lượng NSD thuộc cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ thật sự quá ít. Trong một số ngữ cảnh, việc trộn quá nhiều NSD từ những cộng đồng thuộc các tiêu chí khác nhau vào cộng đồng $G^{(\alpha)}(u, i, c)$ sẽ làm giảm độ chính xác của dự đoán do sự ảnh hưởng của thông tin nhiễu.

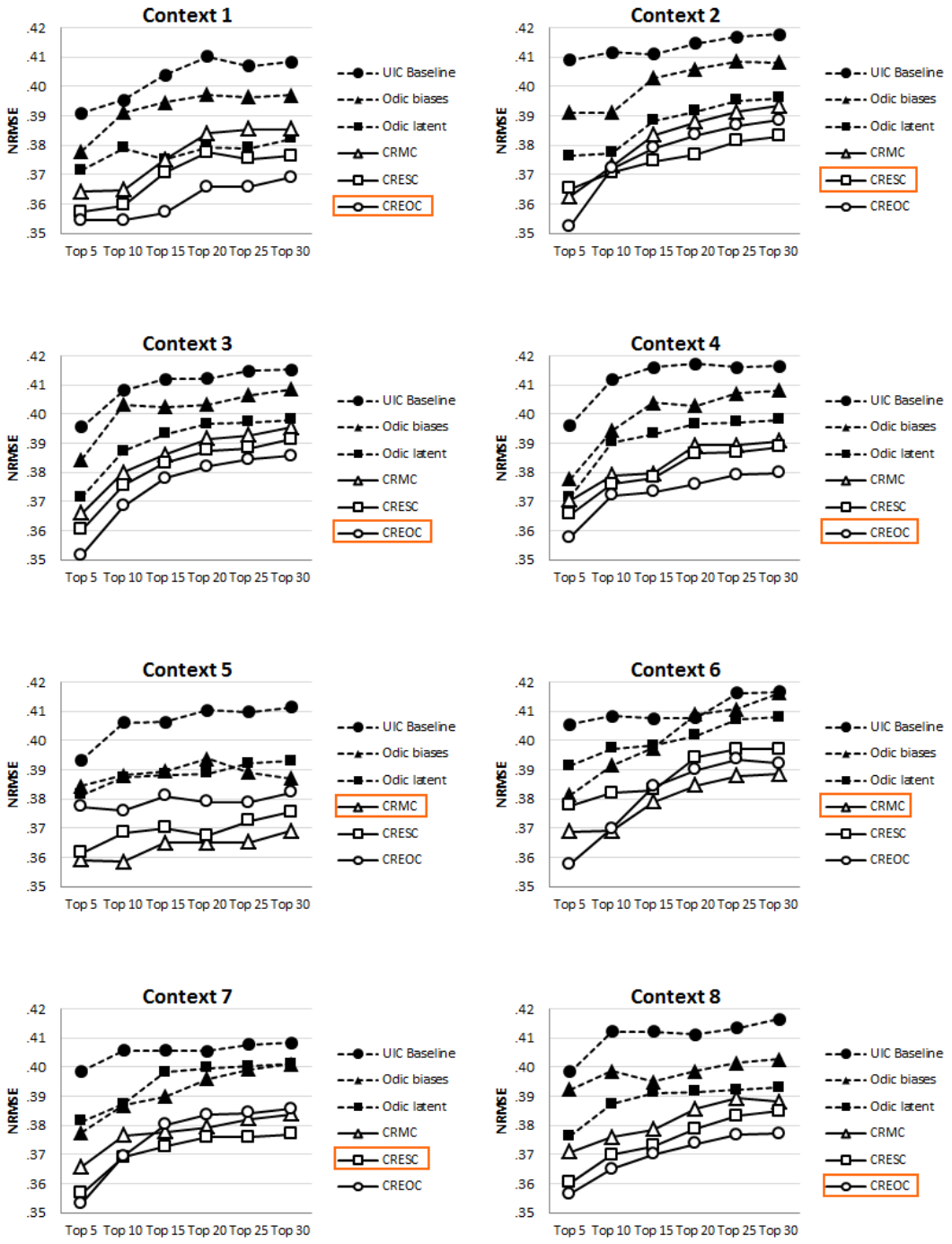
Ngoài ra, luận văn cũng đã áp dụng thuật toán CREOC trong thực nghiệm 1 và nhận được kết quả có xu hướng tương tự như trong thuật toán CRESC, do đó luận văn không trình bày chi tiết kết quả của thuật toán CREOC.

Tóm lại, kết quả của thực nghiệm 1 với cả ba thuật toán CRMC, CRESC và CREOC đã chứng minh cho thấy nếu chỉ khai thác những cộng đồng xây dựng theo một tiêu chí duy nhất là ratings thì sẽ không thể đạt kết quả tốt nhất trong mọi ngữ cảnh do tác động của vấn đề dữ liệu thưa. Đây là câu trả lời quan trọng cho câu hỏi Q1 ở phần đầu của Chương 4.

Bên cạnh đó, hiện tượng các tiêu chí luân phiên giành chiến thắng trong các ngữ cảnh cũng đã giúp luận văn có được một câu trả lời tích cực cho câu hỏi Q2 và đồng thời củng cố cho giả định của luận văn là mức độ ảnh hưởng của các cộng đồng đa tiêu chí đối với chất lượng tư vấn sẽ thay đổi tùy thuộc vào từng ngữ cảnh cụ thể. Điều đó có nghĩa là nếu chọn được một tiêu chí α phù hợp cho việc xây dựng cộng đồng để tư vấn trong một ngữ cảnh thì độ chính xác của tư vấn sẽ tăng lên một cách đáng kể, còn nếu không thì hiệu quả tư vấn sẽ trở nên tệ hơn.

4.2.2 *Thực nghiệm 2: So sánh các thuật toán*

Khi xem xét câu hỏi cuối cùng Q3 liên quan đến tính hiệu quả của các thuật toán đề xuất, kết quả so sánh giữa các thuật toán trong Hình 4-3 cho thấy trong mọi ngữ cảnh thì các thuật toán đã đề xuất trong luận văn đều luân phiên đạt kết quả tốt hơn so với những thuật toán tư vấn theo ngữ cảnh khác.



Hình 4-3. So sánh các thuật toán.

Sự thắng thế của các thuật toán đề xuất trong tất cả các ngữ cảnh của thực nghiệm cho thấy việc khai thác những cộng đồng đa tiêu chí sẽ đem đến những lợi

ích cho việc tư vấn theo ngữ cảnh. Cụ thể, thuật toán CRMC đạt được kết quả tốt nhất trong hai ngữ cảnh 5 và 6, thuật toán CREOC vượt trội hơn so với các thuật toán khác trong số bốn trên tám ngữ cảnh (1, 3, 4 và 8), trong khi thuật toán CRESC chiếm vị trí dẫn đầu toàn cục trong hai ngữ cảnh 2 và 7.

Nguyên nhân của sự vượt trội của các thuật toán CRMC, CRESC và CREOC là do trong một số trường hợp mà thông tin ngữ cảnh có tác động như là những dữ liệu nhiễu thì phương pháp contextualizing latent models của Odic và cộng sự sẽ không có khả năng phân biệt được thông tin nhiễu và mối quan hệ giữa ngữ nghĩa ngầm trong ngữ cảnh. Trái lại, lợi ích của các phương pháp đề xuất là có sự phối hợp giữa phương pháp latent factor model trong thành phần tư vấn độc lập với ngữ cảnh và phương pháp láng giềng trong các cộng đồng đa tiêu chí trong thành phần dự đoán có xét đến yếu tố ngữ cảnh thông qua việc điều chỉnh giá trị của tham số γ theo kích thước của $G^{(\alpha)}(u, i, c)$ trong thuật toán CRMC hay theo kích thước của $G^{(*)}(u, i, c)$ trong thuật toán CRESC và CREOC, thể hiện gián tiếp yếu tố ngữ cảnh. Như vậy, các thuật toán đề xuất trong luận văn có thể phát hiện ra các mối quan hệ cục bộ giữa các cộng đồng đa tiêu chí và ngữ cảnh cũng như khai thác các đánh giá toàn cục trên toàn bộ ngữ cảnh. Bằng cách này, các thuật toán đề xuất có thể làm giảm bớt tác động của thông tin ngữ cảnh bị nhiễu trong một số trường hợp, và từ đó có thể nâng cao mức độ chính xác của tư vấn.

Bên cạnh đó, luận văn cũng đã tìm hiểu việc lựa chọn áp dụng các thuật toán CRMC, CRESC hay CREOC trong những trường hợp nào sẽ đem lại hiệu quả tốt nhất. Trước hết, thông qua những kết quả thực nghiệm, luận văn nhận thấy khi mà số lượng NSD trong cộng đồng $G^{(\alpha)}(u, i, c)$ đã xấp xỉ ngưỡng λ_{max} thì thuật toán CRMC nên được áp dụng vì sự lựa chọn này giúp cho chúng ta có thể tránh được chi phí phải tính toán, tìm kiếm những NSD tương đồng trong các cộng đồng khác mà vẫn bảo đảm được một mức độ chính xác của việc dự đoán. Điều đó có nghĩa là trong những trường hợp mà sự thiếu hụt NSD không thật sự nhiều thì sự đóng góp của quy trình bổ sung láng giềng trong thuật toán CRESC hay trong thuật toán

CREOC là không đáng kể. Trong trường hợp ngược lại, nếu kích thước $G^{(\alpha)}(u, i, c)$ khá nhỏ thì hai thuật toán CRESC và CREOC có thể được áp dụng để nâng cao chất lượng tư vấn.

Ngoài ra, khi so sánh với CREOC thì thuật toán CRESC sẽ kém hiệu quả hơn trong những trường hợp mà sự khác biệt giữa rating trung bình trong một ngữ cảnh cụ thể của cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ và các cộng đồng khác là khá lớn. Như vậy, về mặt hình thức, thuật toán CRESC sẽ không đạt kết quả tốt nếu như $\max_{\alpha' \neq \alpha}(sim(G^{(\alpha)}(u,i,c), G^{(\alpha')}(u,i,c))) < \delta$ tức là khi độ tương đồng trong ngữ cảnh c giữa cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ và cộng đồng gần gũi với nó nhất $G^{(\alpha')}(u, i, c)$ nhỏ hơn ngưỡng δ cho trước. Khi sự tương đồng càng nhỏ thì khoảng cách càng lớn, điều này có nghĩa là việc lựa chọn các láng giềng trong những cộng đồng khác để bổ sung vào cộng đồng ưu tiên như trong công thức (3.8) của thuật toán CRESC có khả năng sẽ bị tác động bởi thông tin nhiễu và khi đó việc lựa chọn này mang đến những tác động xấu đối với chất lượng tư vấn. Khi đó, thuật toán CREOC nên được chọn để áp dụng trong những trường hợp trên bởi vì ưu điểm chính của nó là khai thác hoàn toàn thứ tự ưu tiên của các tiêu chí trong từng ngữ cảnh.

Trái lại, khi mà sự khác biệt giữa rating trung bình của $G^{(\alpha)}(u, c)$ và của các cộng đồng khác không quá lớn thì việc áp dụng thuật toán CRESC sẽ là một sự lựa chọn tốt hơn so với thuật toán CREOC. Nguyên nhân của điều này là do khi sự khác biệt giữa các rating trung bình trong một ngữ cảnh cụ thể vừa đủ nhỏ, có nghĩa là sự tương đồng giữa các cộng đồng sẽ càng lớn. Do đó, việc lựa chọn cộng đồng để bổ sung như phương pháp trong thuật toán CRESC sẽ mang lại hiệu quả. Thêm vào đó, do CRESC không phụ thuộc hoàn toàn vào thứ tự ưu tiên của các tiêu chí, ngoại trừ tiêu chí ưu tiên được chọn trong Bước 1, cho nên nó có lợi trong trường hợp mà thứ tự ưu tiên này bị lạc hậu.

Chương 5. **Tổng kết**

Chương 5 sẽ trình bày kết luận và một số hướng phát triển của luận văn trong tương lai.

5.1 Kết luận

Về mặt lý thuyết, luận văn đã đề xuất một cách tiếp cận mới trong việc khai thác các cộng đồng đa tiêu chí trong một hệ thống tư vấn theo ngữ cảnh (CARS). Ý tưởng nền tảng của cách tiếp cận đề xuất trong luận văn là vai trò hay mức độ ảnh hưởng của các cộng đồng được xây dựng từ những tiêu chí khác nhau sẽ thay đổi tùy thuộc vào từng ngữ cảnh cụ thể trong quá trình tư vấn. Như vậy, ứng với một ngữ cảnh cho trước, cách tiếp cận của luận văn cần phải xác lập một thứ tự ưu tiên trên tập hợp các tiêu chí thành lập cộng đồng để thuật toán tư vấn có thể áp dụng trên tiêu chí được xem là phù hợp nhất. Những tiêu chí này bao gồm ratings và những đặc trưng (hay thuộc tính) trong profile của NSD.

Để hiện thực ý tưởng của cách tiếp cận nêu trên, luận văn có sử dụng một số ký hiệu cơ bản về đại số của mô hình αCSM, và sau đó luận văn đã mở rộng mô hình này bằng cách tích hợp yếu tố ngữ cảnh và định nghĩa một quan hệ tiền thứ tự trên tập hợp các tiêu chí cho mỗi ngữ cảnh. Trên cơ sở đó, luận văn đã đề xuất ba thuật toán tư vấn theo ngữ cảnh là CRMC, CRESC và CREOC với sự kết hợp giữa phương pháp MF với việc khai thác các cộng đồng đa tiêu chí thay vì chỉ dựa trên các cộng đồng được xây dựng theo một tiêu chí duy nhất là ratings. Đặc biệt, hai

thuật toán CRESC và CREOC được xây dựng nhằm hướng đến việc góp phần giải quyết vấn đề dữ liệu thưa trong các hệ thống tư vấn theo ngữ cảnh.

Về mặt thực nghiệm, luận văn đã sử dụng bộ dữ liệu mẫu MovieLens cho ba thuật toán đề xuất cùng với ba thuật toán đối sánh. Kết quả thực nghiệm cho thấy việc khai thác mối quan hệ giữa các cộng đồng đa tiêu chí và ngữ cảnh trong ba thuật toán đề xuất CRMC, CRESC và CREOC sẽ mang lại kết quả tốt hơn là khi chỉ dựa trên một tiêu chí ratings để thành lập cộng đồng như trong các thuật toán đối sánh.

Luận văn cũng đã phân tích ưu và khuyết điểm của ba thuật toán đề xuất nhằm giúp định hướng cho việc lựa chọn áp dụng các thuật toán này sao cho đạt hiệu quả cao nhất. Ưu điểm của thuật toán CRMC là khá đơn giản và hiệu quả khi kích thước của cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ là đủ lớn để có thể đảm bảo cho độ tin cậy của thuật toán tư vấn.

Đối với trường hợp $G^{(\alpha)}(u, i, c)$ có sự thiếu hụt lớn về số lượng NSD thì các phương pháp bổ sung láng giềng trong hai thuật toán CRESC và CREOC sẽ có khả năng làm giảm bớt sự tác động của vấn đề dữ liệu thưa, từ đó chất lượng tư vấn có thể được cải thiện. Thuật toán CRESC có khuynh hướng đạt kết quả tốt khi rating trung bình trong từng ngữ cảnh của cộng đồng ưu tiên $G^{(\alpha)}(u, i, c)$ và của các cộng đồng khác không có sự khác biệt đáng kể. Trong khi đó, do thuật toán CREOC hoàn toàn khai thác thứ tự ưu tiên của các tiêu chí trong một ngữ cảnh nên việc áp dụng thuật toán này sẽ có lợi khi mà việc tính toán độ tương đồng giữa các cộng đồng như trong thuật toán CRESC không có nhiều tác dụng.

5.2 Hướng phát triển

Trong tương lai, luận văn dự kiến sẽ tiếp tục nghiên cứu một số hướng phát triển chính như sau:

1. *Nghiên cứu thêm những phương pháp hay những độ đo khác có thể được sử dụng để tính toán mức độ tương đồng giữa hai cộng đồng trong thuật toán CRESC.*

Việc có thêm những phương pháp hay những độ đo khác để ước lượng sự tương đồng giữa các cộng đồng, ngoài công thức (3.8), sẽ giúp cho việc bổ sung láng giềng vào cộng đồng $G^{(\alpha)}(u, i, c)$ trong Bước 3 của phương pháp CRESC trở nên linh hoạt hơn, và đặc biệt là hệ thống có thể chọn lựa phương pháp hay độ đo để áp dụng tùy theo ngữ cảnh khác nhau.

Luận văn dự kiến khai thác ý tưởng về đặc trưng cộng đồng, nghĩa là mỗi cộng đồng sẽ có một người đại diện mà có thể là một thành viên "tiêu biểu" (NSD thật) hay là một thành viên "ảo" thể hiện những đặc trưng chính của cộng đồng đó (ví dụ, nếu xem mỗi thành viên là một điểm trong không gian thì hệ thống sẽ chọn "trọng tâm" làm đại diện cho cộng đồng). Từ đó, bài toán ước lượng mức độ tương đồng giữa hai cộng đồng sẽ được quy về bài toán về độ tương đồng giữa hai người đại diện. Điều này sẽ giúp làm giảm chi phí tính toán một cách đáng kể.

2. *Nghiên cứu thử nghiệm mở rộng các thuật toán được đề xuất vào trong những lãnh vực mà sự tác động của các yếu tố ngữ cảnh lên sở thích của NSD là rõ ràng và đáng kể.*

Chúng ta có thể hình dung lãnh vực tư vấn âm nhạc, trong đó, một bài nhạc có thể được tư vấn cho NSD nghe đi nghe lại nhiều lần trong nhiều ngữ cảnh khác nhau, nhưng không phải ở lần nghe nào thì NSD cũng có những đánh giá giống như nhau. Ví dụ, bài hát Endless Love có thể được đánh giá cao khi NSD cảm thấy vui nhưng lại không được ưa thích khi NSD đó đang có tâm trạng căng thẳng, lo lắng; hay như bài hát Happy New Year thường được lựa chọn và ưa thích vào những dịp đầu năm nhưng ở những thời điểm khác trong năm thì bài hát này không phải là sự tư vấn phù hợp.

3. *Nghiên cứu mở rộng tiêu chuẩn về chất lượng của tư vấn.*

Hiện nay, phần lớn các hệ thống tư vấn theo ngữ cảnh đều chỉ chú trọng đến độ chính xác (*precision*) như là một tiêu chuẩn về chất lượng của thông tin tư vấn, nghĩa là hệ thống mong muốn cung cấp cho NSD một danh sách tư vấn gồm những items với độ chính xác cao nhất. Điều này là chưa đáp ứng hoàn toàn nhu cầu phong phú của NSD vì bên cạnh độ chính xác thì còn có rất nhiều tiêu chuẩn khác thể hiện chất lượng của thông tin tư vấn như tính đa dạng (*diversity*), tính mới lạ (*novelty*), tính bất ngờ (*serendipity*), … [10], [22], [24], [25].

Luận văn cho rằng hệ thống cũng có thể lựa chọn tiêu chuẩn chất lượng tùy theo ngữ cảnh để tạo ra danh sách items tư vấn. Ví dụ, khi NSD muốn xem phim một mình ở nhà thì hệ thống sẽ tư vấn phim theo tiêu chuẩn chất lượng là độ chính xác (gợi ý những phim phù hợp nhất với sở thích cá nhân) nhưng trong ngữ cảnh mà NSD muốn mời nhóm bạn xem phim trong buổi liên hoan tại nhà thì hệ thống nên tư vấn phim theo tiêu chuẩn về tính mới lạ và đa dạng để có thể đáp ứng sở thích của nhiều người, đồng thời mọi người cũng có dịp khám phá những thể loại phim mới.

Vì vậy, luận văn mong muốn sẽ tiếp tục nghiên cứu vấn đề tích hợp thông tin ngữ cảnh vào quá trình tư vấn đa mục tiêu nhằm thỏa mãn nhu cầu rất đa dạng và phong phú của NSD.

Những công trình tại các hội nghị khoa học quốc tế

Nguyen Thuy Ngoc & Nguyen An Te (2013). Context-aware Recommendation Based On Multi-Criteria Communities. *The IADIS International Conference Applied Computing 2013*, Fort Worth, Texas, USA (*accepted*).

Nguyen Thuy Ngoc & Nguyen An Te (2013). Hybridization of Matrix Factorization with Merging Multi-criteria Communities for Context-aware Recommendations. *The 12th IEEE International Conference on Machine Learning and Applications* (ICMLA 2013), Session Machine Learning for Predictive Models, Miami, Florida, USA (*accepted*).

Nguyen Thuy Ngoc & Nguyen An Te (2013). Towards Context-aware Recommendations: Strategies for Exploiting Multi-criteria Communities. *The 9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing* (CollaborateCom 2013), Austin, Texas, USA (*accepted*).

CollaborateCom 2013 notification for paper 23

IADIS Applied Computing 2013 – Invitation letter

ICMLA 2013 Acceptance Notification for paper 320

Tài liệu tham khảo

[1] G. Adomavicius, L. Baltrunas, T. Hussein, F. Ricci, and A. Tuzhilin (2011). Context-aware Recommender Systems. *AI Magazine*, Vol. 32, No. 3, pp. 67-80

[2] G. Adomavicius and A. Tuzhilin (2005). Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-art and Possible Extensions. *IEEE Trans: Knowledge and Data Engineer*, Vol. 17, No. 6, pp. 734-749.

[3] L. Baltrunas and X. Amatriain (2009). Towards Time-dependant Recommendation Based on Implicit Feedback. *Proceedings of the 3rd Workshop on Context-Aware Recommender Systems* (CARS'09), New York, USA.

[4] L. Baltrunas and F. Ricci (2009). Context-based Splitting of Item Ratings in Collaborative Filtering. *Proceedings of the 3rd ACM Conference on Recommender Systems* (RecSys'09), NY, USA.

[5] L. Baltrunas, B. Ludwig, and F. Ricci (2011). Matrix Factorization Techniques for Context Aware Recommendation. *Proceedings of the 5th ACM International Conference on Recommender Systems* (RecSys'11), Illinois, USA.

[6] J. S. Breese, D. Heckerman and C. Kadie (1998). Empirical Analysis of Predictive Algorithms for Collaborative Filtering. *Proceedings of the 14th Conference on Uncertainty In Artificial Intelligence* (UAI'98), Wisconsin, USA.

[7] R. Burke (2007). Hybrid Web Recommender Systems. *The Adaptive Web, Methods and Strategies of Web Personalization*, LNCS 4321.

[8] C. Desrosiers and G. Karypis (2011). A Comprehensive Survey of Neighborhood-based Recommendation Methods, In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, 1st ed., Springer, pp. 107-144.

[9] A. K. Dey (2001). Understanding and using context. *Personal and Ubiquitous Computing*, Vol. 5, No. 1, pp. 4-7.

[10] J. L. Herlocker, J. A. Konstan, L. G. Terveen and J. Riedl (2004). Evaluating Collaborative Filtering Recommender Systems. *ACM Transaction on Information Systems*, Vol. 22, No. 1, pp. 5-53.

[11] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver (2010). Multiverse Recommendation: N-dimensional Tensor Factorization for Context-aware Collaborative Filtering. *Proceedings of the 4th ACM Conference on Recommender Systems* (RecSys'10), Barcelona, SPAIN.

[12] Y. Koren (2008). Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model. *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Nevada, USA, 2008.

[13] Y. Koren and R. Bell (2011). Advances in Collaborative Filtering. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, 1st ed., Springer, pp. 145-186.

[14] M. Montaner, B. López and J. L. De La Rosa (2003). A Taxonomy of Recommender Agents on the Internet. *Artificial Intelligence Review*, Vol. 19.

[15] MovieLens datasets. http://www.grouplens.org/node/73/

[16] A. T. Nguyen, N. Denos, and C. Berrut (2007). Improving New NSD Recommendations with Rule-based Induction on Cold NSD Data. *Proceedings of the 1st ACM International Conference on Recommender Systems* (RecSys'07), Minnesota, USA.

[17] A. Odic, M. Tkalcic, J. Tasic, and A. Kosir (2012). Relevant Context in a Movie Recommender System: NSD' Opinion vs. Statistical Detection. *Proceedings of the 4th Workshop on Context-Aware Recommender Systems* (CARS'12), Dublin, IRELAND.

[18] U. Panniello, A. Tuzhilin, M. Gorgoglione, C. Palmisano, and A. Pedone (2009). Experimental Comparison of Pre- vs. Post-filtering Approaches in Context-aware Recommender Systems. *Proceedings of the 3rd ACM International Conference on Recommender Systems* (RecSys'09), New York, USA.

[19] F. Ricci, L. Rokach, and B. Shapira (2011). Introduction to Recommender Systems Handbook. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, 1st ed., Springer, pp. 1-35.

[20] F. Ricci (2012). Context-Aware Music Recommender Systems: Workshop Keynote Abstract. *Proceedings of the 21st international conference companion on World Wide Web* (WWW '12 Companion), New York, USA.

[21] A. Said, E. W. De Lucca, and S. Albayrak (2011). Inferring Contextual NSD Profiles – Improving Recommender Performance", *Proceedings of the 5th ACM International Conference on Recommender Systems* (RecSys'11), Illinois, USA.

[22] G. Shani and A. Gunawardana (2011). Evaluating Recommendation Systems. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, 1st ed., Springer, pp. 257-297.

[23] Y. Shi, M. Larson, and A. Hanjalic (2010). Mining Mood-specific Movie Similarity with Matrix Factorization for Context-aware Recommendation. *Proceedings of the Workshop on Context-Aware Movie Recommendation* (CAMRa'10), Barcelona, SPAIN.

[24] M. Zhang (2009). Enhancing Diversity in Top-N Recommendation. *Proceedings of the 3rd ACM conference on Recommender systems* (RecSys'09), NY, USA.

[25] M. Zhang and N. Hurley (2009). Novel Item Recommendation by NSD Profile Partitioning. *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology* (WI-IAT'09), Milan, ITALY.