

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

BÙI AN LỘC

**CÁ NHÂN HÓA ỨNG DỤNG VÀ DỊCH VỤ DI
ĐỘNG HƯỚNG NGỮ CẢNH NGƯỜI DÙNG**

Ngành: Công nghệ thông tin

Chuyên ngành: Hệ thống thông tin

Mã số: 60.48.01.04

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC:

PGS.TS. NGUYỄN NGỌC HÓA

Hà Nội - 2016

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu do tôi tìm hiểu, nghiên cứu, tham khảo và tổng hợp từ các tài liệu nghiên cứu trước đây và làm theo hướng dẫn của người hướng dẫn khoa học. Phần nội dung đóng góp của luận văn do tôi thực hiện.

Các số liệu, kết quả nêu trong luận văn là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác, các nội dung được trích dẫn đã có tham chiếu đầy đủ.

Tôi xin chịu hoàn toàn trách nhiệm về lời cam đoan của mình. Nếu có điều gì sai trái, tôi xin chịu mọi hình thức kỷ luật theo quy định của nhà trường.

Tác giả

Bùi An Lộc

LỜI CẢM ƠN

Đầu tiên tôi xin gửi lời cảm ơn sâu sắc tới thầy PGS.TS. Nguyễn Ngọc Hóa, Bộ môn Hệ thống thông tin, Khoa Công nghệ Thông tin, Trường Đại học Công nghệ, Đại học Quốc gia Hà Nội, người đã định hướng đề tài và tận tình hướng dẫn, chỉ bảo cho tôi trong suốt quá trình thực hiện luận văn tốt nghiệp này.

Tôi cũng xin trân trọng cảm ơn các thầy cô trong Khoa Công nghệ Thông tin, Trường Đại học Công nghệ, Đại học Quốc Gia Hà Nội đã tận tình giảng dạy, hướng dẫn nghiên cứu khoa học cho tôi trong suốt thời gian theo học tại trường cũng như trong quá trình làm luận văn này.

Xin cảm ơn các anh, chị, em và các bạn học viên bộ môn Hệ thống thông tin, những người đã giúp đỡ, động viên tinh thần và chia sẻ kinh nghiệm quý báu giúp tôi vượt qua các khó khăn, vướng mắc để có thể hoàn thành luận văn này.

Mặc dù đã cố gắng, nhưng tôi tin chắc luận văn của tôi còn nhiều thiếu sót và có rất nhiều nội dung có thể hoàn thiện tốt hơn. Tôi rất mong nhận được những ý kiến đánh giá, phê bình và góp ý của các thầy cô, anh chị và các bạn.

Trân trọng,

Tác giả

Bùi An Lộc

MỤC LỤC

LỜI CAM ĐOAN	2
LỜI CẢM ƠN.....	3
Danh mục các ký hiệu và chữ viết tắt.....	6
Danh mục các bảng.....	6
Danh mục các hình vẽ, đồ thị	7
MỞ ĐẦU	8
Chương 1. TỔNG QUAN VỀ CÁ NHÂN HOÁ DỊCH VỤ DI ĐỘNG HƯỚNG NGỮ CẢNH	10
1.1 Các nhân hoá dịch vụ hướng ngữ cảnh.....	10
1.1.1 Định nghĩa ngữ cảnh	12
1.1.2 Các đặc trưng của ngữ cảnh	15
1.1.3 Phân loại ngữ cảnh	15
1.1.4 Mô hình làm việc cho ngữ cảnh	17
1.2 Nhận biết ngữ cảnh (context-awareness).....	18
1.2.1 Xu thế nhận biết ngữ cảnh và lợi ích trong việc cá nhân hóa ứng dụng di động	18
1.2.2 Khái niệm nhận biết ngữ cảnh.....	19
1.2.3 Tính toán nhận biết ngữ cảnh	19
1.3 Hệ gợi ý.....	20
1.3.1 Định nghĩa hệ gợi ý	20
1.3.2 Dự đoán trong hệ gợi ý.....	20
1.3.3 Các phương pháp tiếp cận truyền thống trong hệ gợi ý	21
1.3.4 Đánh giá hệ gợi ý	22
1.3.5 Các thách thức của hệ gợi ý truyền thống	25
1.4 Kết luận chương.....	25
Chương 2. HỆ GỢI Ý DỰA TRÊN NHẬN BIẾT NGỮ CẢNH.....	27
2.1 Cấu trúc thông tin ngữ cảnh trong hệ gợi ý	28

2.1.1	Cấu trúc dữ liệu phân cấp.....	28
2.1.2	Cấu trúc dữ liệu đa chiều.....	28
2.2	Cơ chế tích hợp ngữ cảnh vào hệ gợi ý.....	29
2.2.1	Lọc trước theo ngữ cảnh (Contextual Pre-Filtering).....	32
2.2.2	Lọc sau theo ngữ cảnh (Contextual Pos-Filtering).....	34
2.2.3	Mô hình hóa hướng ngữ cảnh (Contextual Modeling).....	36
2.3	Mô hình hóa ngữ cảnh dựa trên phân rã ma trận (Context Aware Matrix Factorization - CAMF).....	37
2.3.1	Kỹ thuật phân rã ma trận (Matrix Factorization - MF)	37
2.3.2	Kỹ thuật phân rã ma trận thiên vị (Biased Matrix Factorization – BMF)	41
2.3.3	Phân rã ma trận hướng ngữ cảnh (Context Aware Matrix Factorization - CAMF)	42
2.4	Kết luận chương	46
Chương 3. XÂY DỰNG ỨNG DỤNG DU LỊCH THEO HƯỚNG CÁ NHÂN HÓA DỰA TRÊN NGỮ CẢNH NGƯỜI DÙNG		47
3.1	Đặt bài toán	47
3.2	Giải pháp đề xuất	47
3.2.1	Mô hình kiến trúc ứng dụng	47
3.2.2	Thiết kế ứng dụng.....	49
3.2.3	Môi trường xây dựng và thử nghiệm.....	50
3.3	Thử nghiệm ứng dụng và đánh giá kết quả tại Mobifone.....	50
3.3.1	Kết quả ứng dụng	52
3.3.2	Đánh giá thử nghiệm	55
Chương 4. KẾT LUẬN CHUNG		58
4.1	Đóng góp của luận văn.....	58
4.2	Hướng phát triển	58
TÀI LIỆU THAM KHẢO		59

Danh mục các ký hiệu và chữ viết tắt

STT	Thuật ngữ, chữ viết tắt	Diễn giải
1	RS	Recommender System Hệ gợi ý.
2	CF	Collaborative Filter Lọc cộng tác
3	FM	Matrix Factorization Phân rã ma trận
4	BMF	Biases Matrix Factorization Phân rã ma trận thiên vị
5	CAMF, CAMF- CI, CAMF-CU, CAMF-CC, CAMF-C	Context-aware Matrix Factorization Phân rã ma trận dựa trên ngữ cảnh
6	CRD	Contextual Rating Deviation Độ lệch xếp hạng ngữ cảnh
7	SGD	Stochastic Gradient Descent Kỹ thuật giảm gradient ngẫu nhiên.
8	OLAP	Online Analytical Processing Phân tích dữ liệu trực tuyến
9	CARS	Context-awareness Recommender System Hệ gợi ý theo hướng ngữ cảnh
10	MSE	Mean Square Error Sai số bình phương trung bình
11	RMSE	Root Mean Square Error Căn bậc hai sai số bình phương trung bình
12	MAE	Mean Absolute Error Sai số tuyệt đối trung bình
13	LBS	Location Base service Hệ thống cung cấp vị trí thuê bao di động
14	Telco	Telecommunication Corporation Nhà cung cấp dịch vụ viễn thông di động
15	CSDL	Cơ sở dữ liệu

Danh mục các bảng

Bảng 1: Các thuộc tính đặc trưng của ngữ cảnh.....	15
Bảng 2: Phân loại các chiều của ngữ cảnh	16
Bảng 3: Bảng ma trận trọng số (đánh giá) của hệ gợi ý	21
Bảng 4: Các phương pháp đánh giá.....	24
Bảng 5: Đánh giá thời gian thực hiện	56
Bảng 6: Đánh giá độ chính xác dự đoán.....	57

Danh mục các hình vẽ, đồ thị

Hình 1: Mô hình ngữ cảnh.....	14
Hình 2: Kiến trúc quản lý ngữ cảnh mức cao	14
Hình 3: Không gian đặc tính ngữ cảnh	18
Hình 4: Các thành phần cơ bản của tiến trình gợi ý truyền thông.....	21
Hình 5: Kiến trúc tổng quan của hệ thống lọc cộng tác	22
Hình 6: Cấu trúc phân cấp của ngữ cảnh trong hệ gợi ý	28
Hình 7: Cấu trúc OLAP 3 chiều User x Item x Time trong hệ gợi ý.....	29
Hình 8: Các cơ chế tích hợp thông tin ngữ cảnh vào hệ gợi ý	32
Hình 9: Bước hiệu chỉnh danh sách gợi ý trong lọc sau theo ngữ cảnh	35
Hình 10: Minh họa kỹ thuật phân rã ma trận.....	38
Hình 11: Kiến trúc nền tảng mã nguồn mở CARSKIT	48
Hình 12: Kiến trúc thiết kế hệ thống	48
Hình 13: Thiết kế user case.....	49
Hình 14: Mô hình thực thể quan hệ quản lý user	49
Hình 15: Mô hình thực thể quan hệ quản lý địa điểm và gợi ý	50
Hình 16: Dữ liệu rating lưu trong CSDL.....	51
Hình 17: Dữ liệu địa điểm được lưu trong CSDL	52
Hình 18: Deploy hệ thống trên môi trường Linux.....	52
Hình 19: Giao diện xác thực người dùng.....	53
Hình 20: Giao diện home của client	54
Hình 21: Giao diện nhập thông tin ngữ cảnh tĩnh và gợi ý	55
Hình 22: Giao diện thông tin người dùng và đăng xuất hệ thống	55
Hình 23: So sánh độ chính xác của CAMF & BPMF theo RMSE.....	57

MỞ ĐẦU

Theo đánh giá của tổ chức Gartner, vào năm 2017 ứng dụng di động sẽ được tải về hơn 268 tỷ lần, tạo ra doanh thu hơn 77 tỷ USD và ứng dụng di động sẽ trở thành một trong những công cụ tính toán phổ biến nhất cho người dùng trên toàn cầu. Gartner cũng dự đoán rằng người dùng di động sẽ cung cấp dữ liệu cá nhân thông qua hơn 100 ứng dụng và dịch vụ mỗi ngày.

Ứng dụng và dịch vụ di động đã và đang trở thành phương tiện giao tiếp thông tin chính giữa người dùng – người dùng, người dùng – nhà cung cấp dịch vụ. Bài toán làm thế nào để các ứng dụng dịch vụ di động ngày càng trở nên thông minh hơn, linh hoạt hơn, hiệu quả hơn trong việc trao đổi thông tin, đáp ứng các nhu cầu và sở thích cá nhân của người dùng ngày càng trở nên cần thiết và khẩn cấp. Giải pháp cá nhân hóa các ứng dụng và dịch vụ di động theo hướng tiếp cận ngữ cảnh người dùng dựa trên các thông tin được thu thập qua việc trao đổi dữ liệu như vị trí, thời gian, thiết bị, thói quen, sở thích, ... của người dùng đã được áp dụng để giải quyết bài toán này và sẽ trở thành xu hướng phát triển công nghệ tất yếu nhằm mang lại cho người sử dụng các thiết bị di động các tiện ích tốt nhất, đáp ứng đầy đủ các nhu cầu cá nhân trong quá trình sử dụng thông tin thông qua các ứng dụng dịch vụ di động.

Theo số liệu của Tổng công ty viễn thông MobiFone, hiện tại MobiFone có hơn 15 triệu thuê bao di động, trong đó có 12 triệu thuê bao là đang sử dụng các ứng dụng và dịch vụ giá trị gia tăng. MobiFone cũng đang cung cấp cho khách hàng hơn 200 dịch vụ và ứng dụng giá trị gia tăng, bao gồm đầy đủ các lĩnh vực như nhóm dịch vụ thông tin tổng hợp, nhóm dịch vụ âm nhạc, phim ảnh clip, nhóm dịch vụ thể thao, nhóm dịch vụ game, nhóm dịch vụ tiện ích, ... Với số lượng dịch vụ và các lĩnh vực cung cấp phong phú như trên, tuy nhiên, thực tế số lượng thuê bao không tương tác với dịch vụ đang chiếm tỷ trọng lớn (hơn 80% thuê bao không thực hiện tương tác với dịch vụ qua kênh SMS hồi đáp). Việc không tương tác với ứng dụng dịch vụ cũng thể hiện khách hàng ít quan tâm đến các dịch vụ nội dung mà MobiFone cung cấp, hay nói cách khác các dịch vụ và ứng dụng của MobiFone chưa đáp ứng được nhu cầu ngày càng cao của khách hàng. Đây là một thách thức lớn cần phải được giải quyết nhằm tăng độ hài lòng khách hàng, tăng uy tín và giá trị thương hiệu của MobiFone trên thị trường viễn thông trong nước.

Với thực trạng nêu trên, luận văn này hướng đến mục tiêu nghiên cứu xây dựng giải pháp cá nhân hoá các ứng dụng và dịch vụ di động theo mô hình hướng ngữ cảnh hiện thời người dùng, áp dụng kỹ thuật gợi ý dựa trên thông tin ngữ cảnh và xây dựng ứng dụng thử nghiệm tại Mobifone.

Với mục tiêu trên, luận văn tập trung vào nghiên cứu các đặc trưng, mô hình quản lý của ngữ cảnh người dùng, cũng như các phương thức nhận biết ngữ cảnh. Trên cơ sở các nội dung ngữ cảnh, luận văn sẽ làm rõ việc tích hợp, ứng dụng thông tin ngữ cảnh vào trong hệ gợi ý dựa trên nhận biết ngữ cảnh, xem các yếu tố ngữ cảnh như các chiều dữ liệu tác động trực tiếp vào tiến trình dự đoán xếp hạng của người dùng và gợi ý các sản phẩm phù hợp với sở thích, nhu cầu của người dùng. Hệ gợi ý dựa trên nhận biết ngữ cảnh được nghiên cứu chính trong đề tài là hệ gợi ý phân rã ma trận hướng ngữ cảnh, một hệ thống mô hình hóa mở rộng của kỹ thuật phân rã ma trận truyền thống vẫn được xem như state-of-the-art trong lý thuyết hệ gợi ý. Ứng dụng các lý thuyết trên, luận văn cũng trình bày một hệ thống gợi ý về du lịch trên mạng MobiFone theo hướng ngữ cảnh người dùng, sử dụng mô hình phân rã ma trận hướng ngữ cảnh để dự đoán xếp hạng và gợi ý cho người dùng các địa điểm du lịch phù hợp.

Tổ chức của luận văn bao gồm các nội dung chính sau:

Chương 1: Tổng quan về cá nhân hóa dịch vụ di động hướng ngữ cảnh

Chương này trình bày tổng quan về khái niệm ngữ cảnh, về hệ gợi ý truyền thống và các vấn đề liên quan trong hệ gợi ý truyền thống

Chương 2: Hệ gợi ý dựa trên nhận biết ngữ cảnh

Chương này trình bày về vai trò của ngữ cảnh trong hệ gợi ý, các phương thức tích hợp ngữ cảnh vào trong hệ gợi ý. Các nội dung nghiên cứu trọng tâm cũng được trình bày trong chương này như kỹ thuật phân rã ma trận truyền thống, kỹ thuật phân rã ma trận thiên vị và kỹ thuật phân rã ma trận dựa trên ngữ cảnh, một kỹ thuật áp dụng phương thức tiếp cận mô hình hóa, dựa trên học máy để xây dựng hàm dự đoán xếp hạng và gợi ý với tập dữ liệu huấn luyện đầu vào gồm nhiều chiều dữ liệu (user, item, context1, ..., contextN).

Chương 3: Xây dựng ứng dụng du lịch theo hướng cá nhân hóa dựa trên ngữ cảnh người dùng và ứng dụng tại MobiFone

Chương này trình bày về hệ thống gợi ý du lịch theo hướng cá nhân hóa dựa trên ngữ cảnh người dùng, đặc tả hệ thống, thiết kế kiến trúc cũng như thiết kế cơ sở dữ liệu của hệ thống. Minh họa sản phẩm và kết quả thử nghiệm tại MobiFone cũng sẽ được trình bày trong chương 3.

Chương 1. TỔNG QUAN VỀ CÁ NHÂN HOÁ DỊCH VỤ DI ĐỘNG HƯỚNG NGỮ CẢNH

1.1 Các nhân hoá dịch vụ hướng ngữ cảnh

Nhận biết ngữ cảnh thường cần một giải pháp có khả năng đáp ứng được các thách thức như giúp cho các ứng dụng đảm bảo tính linh hoạt và tính tự trị (học máy). Các ứng dụng nhận biết ngữ cảnh thường khai thác các thông tin về ngữ cảnh như: vị trí, nhiệm vụ và sở thích của người dùng để thích ứng với hành vi trong khả năng thay đổi môi trường thực thi và các yêu cầu người dùng. Thông tin này được tích hợp từ các cảm biến hoặc từ người dùng.

Nếu ngữ cảnh chỉ đơn giản là vị trí thì việc có thể hiểu và nhận biết không lấy gì làm khó khăn cho các hệ thống. Tuy nhiên, trong nhiều trường hợp việc nhận biết này còn dựa trên các thông tin khác, vượt xa cả vị trí, và do đó sự phức tạp bắt đầu nảy sinh. Các thách thức trong tính toán nhận biết ngữ cảnh thường bao gồm:

- Phải hiểu khái niệm ngữ cảnh

Ngữ cảnh đó là gì và nó liên quan tới các tình huống trong thế giới thực như thế nào? Trong khi đó chúng ta vẫn chưa có một hiểu biết thật rõ ràng và cơ bản về thuật ngữ "các ngữ cảnh liên quan tới các tình huống như thế nào" và thông tin ngữ cảnh chung được sử dụng để hỗ trợ nâng cao các ứng dụng ra sao. Vấn đề này cũng đi kèm câu hỏi biểu diễn ngữ cảnh theo một cách chung nhất như thế nào?

- Làm thế nào để sử dụng ngữ cảnh?
- Làm thế nào để thu thập ngữ cảnh?

Thu thập ngữ cảnh là yêu cầu đầu tiên cho bất kỳ hệ thống nhận biết ngữ cảnh nào. Nhìn chung, việc lấy ngữ cảnh có thể xem như là quá trình xử lý trong đó tình huống thực trong thế giới thực được nắm bắt, các đặc tính hữu ích được xem xét đánh giá và một biểu diễn trừu tượng được tạo, sau đó nó được cung cấp tới các thành phần trong hệ thống với những mục đích sử dụng cao hơn. Các cách tiếp cận thu thập ngữ cảnh thì rất đa dạng như: lần vết vị trí, các hệ thống cảm biến và cả các cách tiếp cận mang tính chất dự đoán như mô hình hóa người dùng và hành vi của họ, ...

- Làm thế nào để kết nối ngữ cảnh thu được với ngữ cảnh sử dụng

Trong một hệ thống nhận biết vị trí, mối quan hệ giữa thu thập ngữ cảnh và sử dụng ngữ cảnh là rất gần, hầu hết các cảm biến vị trí được nạp vào các thiết bị định vị. Trong trường hợp này, biểu diễn ngữ cảnh cũng là giữa các

thành phần. Trong môi trường chung hơn, ngữ cảnh sử dụng và ngữ cảnh thu thập được phân tán. Ở đây, khó khăn thể hiện ở hai điểm: vượt quá khả năng phân tán bởi các thành phần mạng và tích hợp để biểu diễn với đa thành phần.

- Hiểu tác động của ngữ cảnh trong tương tác người máy

Khi các hệ thống nhận biết ngữ cảnh thì hành vi của chúng là độc lập với ngữ cảnh được dùng hoặc tình huống chung được dùng. Mục tiêu chung là tạo các hệ thống theo cách có thể hành xử như được biết trước bởi người dùng. Tuy nhiên trong đời sống thực, điều này gây nên các vấn đề phức tạp, cụ thể như nếu hệ thống hành xử khác với mong đợi của người dùng. Hai tiêu chí đặt ra là "người dùng có thể hiểu hệ thống và hành vi của nó như thế nào" và "người dùng điều khiển hệ thống như thế nào?"

- Làm thế nào để xây dựng các hệ thống nhận biết ngữ cảnh mọi nơi/nhân rộng

Nhận biết ngữ cảnh là một kỹ thuật hữu ích cho các hệ thống tính toán nhân rộng và do đó đây là yêu cầu chung khi hiện thực các hệ thống như vậy. Để xây dựng các môi trường tính toán nhân rộng một cách hiệu quả thì chúng ta cần phải cung cấp hỗ trợ để xây dựng các ứng dụng nhận biết ngữ cảnh. Đó là việc cung cấp các kỹ thuật thu thập ngữ cảnh, cung cấp ngữ cảnh và sử dụng ngữ cảnh, ...

- Đánh giá hệ thống nhận biết ngữ cảnh

Vì các hệ thống nhận biết ngữ cảnh được sử dụng trong một ngữ cảnh nhất định nên việc đánh giá chính nó cũng đòi hỏi phải được thực hiện trong ngữ cảnh đó. Trong trường hợp này, chức năng không chỉ sẵn có và hữu ích trong một ngữ cảnh chắc chắn mà nó còn được yêu cầu tạo hay mô phỏng một tình huống cụ thể với các kết quả trong ngữ cảnh mong muốn để đánh giá hệ thống. Tuy nhiên, tình huống và ngữ cảnh cụ thể ấy cũng phải phù hợp và có hiệu quả để làm thước đo cho việc đánh giá.

Như vậy việc hiểu rõ ngữ cảnh là gì và các đặc trưng của ngữ cảnh ra sao là rất quan trọng khi xây dựng và phát triển các hệ thống nhận biết ngữ cảnh. Việc hiểu rõ ngữ cảnh hay loại ngữ cảnh cần dùng giúp người phát triển có những phương pháp đặc tả và thiết kế phù hợp từ việc cảm nhận tới việc xử lý hành vi sao cho phù hợp với tính chất của các hệ thống trong môi trường hay thay đổi. Đó cũng là lý do mà ngay từ khi thuật ngữ "ngữ cảnh" xuất hiện (1990), các nhà nghiên cứu đã bắt đầu đưa ra các định nghĩa về nó. Qua thời gian phát triển của lĩnh vực nghiên cứu mới này, ngữ cảnh đã nhận được khá nhiều định nghĩa từ đơn giản tới được bổ sung một cách đầy đủ hơn.

1.1.1 Định nghĩa ngữ cảnh

Theo từ điển của Webster (Noah Webster - Mỹ), ngữ cảnh là *"toàn bộ tình huống, nền tảng hay môi trường có liên quan tới một vài sự kiện xảy ra hoặc cá nhân nào đó"*. Định nghĩa này rất là chung khi sử dụng trong tính toán nhận biết ngữ cảnh.

Ngữ cảnh là một vấn đề chính trong tương tác giữa người và máy tính, miêu tả các nhân tố xung quanh với ngữ nghĩa biểu đạt [1]. Trong lĩnh vực nghiên cứu tính toán di động, tham số vị trí thường được dùng nhất để chỉ ngữ cảnh và cài đặt các ứng dụng nhận biết ngữ cảnh.

Ngữ cảnh và nhận biết ngữ cảnh đã bắt đầu được nghiên cứu trong tính toán phân tán với sức mạnh của các thành phần tính toán di động từ những năm 90. Những nghiên cứu từ rất sớm này đã nhận biết vị trí của người dùng và sử dụng vị trí như là trung tâm của tính toán nhận biết ngữ cảnh.

Theo Schilit [2], ngữ cảnh là vị trí, các định danh gần người và các đối tượng cùng những thay đổi của đối tượng (1994). Cũng trong một định nghĩa tương tự, Brown, Bovey và Chen xác định ngữ cảnh là vị trí, các định danh của những người xung quanh người dùng, thời gian trong ngày, mùa, nhiệt độ, ... (1997). Ryan, Pascoe và Morse xác định ngữ cảnh là vị trí của người dùng, môi trường, định danh và thời gian. Dey đã liệt kê ngữ cảnh là trạng thái cảm xúc của người dùng, tập trung vào ý tưởng, vị trí, ngày giờ, các đối tượng và con người trong môi trường của người dùng (1998). Các định nghĩa này xác định ngữ cảnh bằng ví dụ nên rất khó khăn trong việc ứng dụng. Khi xem xét tiềm năng của kiểu dữ liệu mới là thông tin ngữ cảnh thì việc khái niệm như trên không rõ ràng để chúng ta có thể quyết định liệu nên phân lớp thông tin này là ngữ cảnh hay không. Ví dụ như với sở thích và các mối quan tâm của người dùng.

Cũng theo các định nghĩa trên chúng ta có thể thấy rằng các khía cạnh quan trọng nhất của ngữ cảnh là: người dùng đang ở đâu, người dùng đang ở cùng ai và các tài nguyên gần đó. Và ngữ cảnh này là cố định với những thay đổi của môi trường thực thi. Môi trường ở đây gồm ba yếu tố:

- Môi trường tính toán: bộ xử lý có sẵn, các thiết bị truy cập cho người dùng với đầu vào và hiển thị, khả năng mạng, các kết nối, chi phí tính toán
- Môi trường người dùng: vị trí, tập những người gần kề, tình huống xã hội
- Môi trường vật lý: ánh sáng, mức độ ồn, mức độ nhiễu, ...

Dev, Abowd và Wood định nghĩa ngữ cảnh là trạng thái vật lý, xã hội, cảm xúc và thông tin của người dùng.

Khái niệm về ngữ cảnh vẫn là một vấn đề được bàn luận trong suốt những năm qua với nhiều định nghĩa khác nhau được đưa ra. Chúng được chia thành định nghĩa mở và định nghĩa đóng.

Các định nghĩa mở trình bày về ngữ cảnh thông qua một danh sách các chiều ngữ cảnh có thể có và các giá trị đi kèm của chúng. Ngữ cảnh được biểu diễn bởi vị trí người dùng, các đối tượng xung quanh. Brown [3] định nghĩa ngữ cảnh là vị trí, gần với người khác, nhiệt độ, thời gian ... Trong [4], khái niệm ngữ cảnh được chia theo 3 hạng mục: ngữ cảnh tính toán (mạng, hiển thị, ...), ngữ cảnh người dùng (đặc tả, gần người đó, tình huống xã hội, ...) và ngữ

cảnh vật lý (ánh sáng, tiếng ồn, ...). Chen [5] thêm 2 hạng mục: ngữ cảnh thời gian (ngày, tháng, ...) và lịch sử.

Các định nghĩa đóng biểu diễn ngữ cảnh theo cách thông thường, định nghĩa của Dey. Theo Brazie và Brezillion, "ngữ cảnh hoạt động giống như tập các ràng buộc ảnh hưởng đến hành vi của một hệ thống (một người dùng hay một máy tính) nhúng trong một nhiệm vụ nào đó".

Các định nghĩa mở đường như hữu ích trong các ứng dụng cụ thể hơn, vì ở đó khái niệm ngữ cảnh được làm rõ. Tuy nhiên, từ góc nhìn lý thuyết thì chúng không hoàn toàn chính xác vì ngữ cảnh không thể được vạch ra chỉ bởi vài khía cạnh. Mặt khác các định nghĩa đóng thì được sử dụng ít trong thực tế nhưng nó lại thỏa mãn về mặt lý thuyết.

Các ứng dụng nhận biết ngữ cảnh tìm kiếm ai, ở đâu, khi nào và làm gì (tức hành động này đang xảy ra) của các thực thể và sử dụng thông tin này để xác định tại sao một tình huống đang xảy ra. Một ứng dụng không xác định được thực sự tại sao một tình huống đang xảy ra nhưng người thiết kế ứng dụng thì có thể làm được điều đó. Người thiết kế sử dụng ngữ cảnh nắm bắt được để xác định tại sao lại có tình huống đó và sử dụng điều này để lập trình các hành động trong ứng dụng [6]. Và cho đến nay, với một lượng khá lớn các hệ thống được xây dựng trong môi trường tính toán nhân rộng, thì khái niệm ngữ cảnh của Dey vẫn được sử dụng nhiều nhất và có thể coi gần như là chuẩn.

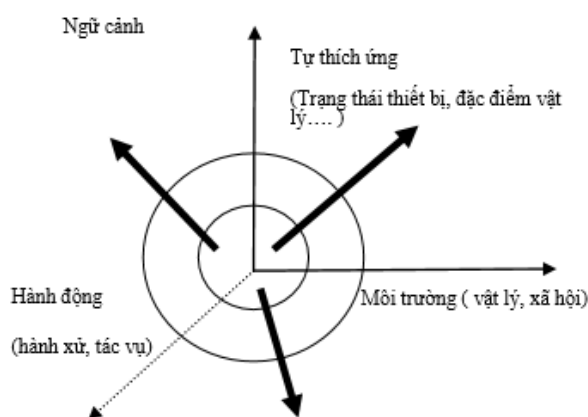
Dey [5] định nghĩa ngữ cảnh là "bất kỳ thông tin nào mà có thể sử dụng được để đặc tả một tình huống của một thực thể. Một thực thể là một người, một nơi hay một số đối tượng được xem là có liên quan đến tương tác giữa người dùng và ứng dụng, bao gồm cả chính người dùng và ứng dụng đó". Đồng thời, ông cũng cung cấp định nghĩa sau cho các hệ thống tính toán nhận biết ngữ cảnh: ***"Một hệ thống sử dụng ngữ cảnh để cung cấp các thông tin liên quan hoặc các dịch vụ cho người dùng trong đó mỗi liên quan phụ thuộc vào nhiệm vụ của người dùng".***

Albrecht Schmidt [3] xác định không gian ngữ cảnh C với định nghĩa là sự kết hợp của các tham số ngữ cảnh, các phân tử ontology miền và các miêu tả dịch vụ: $C = \{U, P, L, T, D, I, S\}$, trong đó:

- U: là tập các nhân tố người dùng và vai trò
- P: là tác vụ và xử lý
- L: là vị trí
- T: là thời gian
- D: là thiết bị
- I: là các đối tượng thông tin sẵn có
- S: là các dịch vụ sẵn có hoặc được miêu tả.

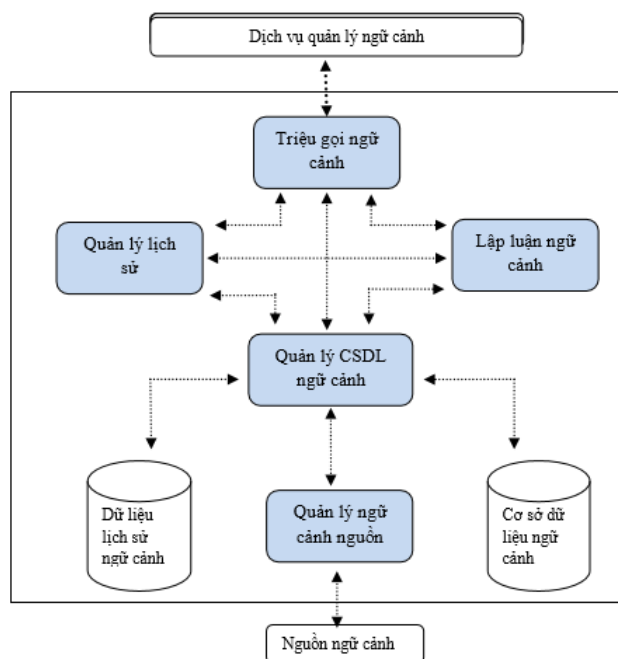
Một ngữ cảnh là một điểm cụ thể trong không gian ngữ cảnh.

Không gian liên quan R có thể được định nghĩa là sản phẩm của không gian ngữ cảnh với các nhân tố liên quan: $R = C * R$



Hình 1: Mô hình ngữ cảnh

Việc quản lý ngữ cảnh trong một hệ thống được thể hiện như trong hình 2. Trong đó mỗi thành phần có một chức năng nhiệm vụ liên quan tới ngữ cảnh riêng. Các thành phần như Bộ triệu gọi ngữ cảnh, lập luận ngữ cảnh, quản lý lịch sử ngữ cảnh và quản lý cơ sở dữ liệu của ngữ cảnh có mối quan hệ chặt chẽ với nhau. Cụ thể, việc lập luận ngữ cảnh cần phải lấy thông tin từ cơ sở dữ liệu và lịch sử ngữ cảnh. Bộ triệu gọi ngữ cảnh lấy thông tin từ cơ sở dữ liệu, bộ lập luận, lịch sử ngữ cảnh theo từng tình huống cụ thể của dịch vụ. Bộ quản lý cơ sở dữ liệu có chức năng lưu trữ mọi thay đổi của ngữ cảnh sau mỗi tác vụ. Ngoài ra còn có bộ quản lý ngữ cảnh nguồn, có nhiệm vụ truy xuất thông tin từ các nguồn phát sinh và lưu trữ dưới hai hình thức: ngữ cảnh lịch sử và ngữ cảnh hiện thời.



Hình 2: Kiến trúc quản lý ngữ cảnh mức cao

1.1.2 Các đặc trưng của ngữ cảnh

Trong phần này, theo Karen [7] thông tin ngữ cảnh có 4 đặc trưng được cho trong bảng sau. Theo đó, ta thấy đặc trưng ngữ cảnh phụ thuộc vào hai yếu tố như: *kiểu nhận biết ngữ cảnh*, *nguồn thu thập*. Điều đó ảnh hưởng tới giá trị thuộc tính của ngữ cảnh. Ví dụ nếu kiểu nhận biết ngữ cảnh là các cảm biến thì khả năng tồn tại của thông tin ngữ cảnh là không cao, chất lượng thông tin có thể chưa tốt tại một thời điểm nhất định khi gặp các sự cố như: thiết bị có lỗi, mất mạng, ...

Kiểu	Nguồn	Tính lâu dài	Chất lượng	Nguyên nhân lỗi
Cảm biến	Cảm biến vật lý và logic	Thấp	Có thể có lỗi	Lỗi thiết bị cảm biến, mất mạng, ngắt mạng, ...
Đặc tả (profiled)	Do người dùng đặc tả trực tiếp hoặc gián tiếp qua chương trình)	Mãi mãi	Có thể không chắc chắn	Lỗi con người
Được phát sinh	Các thông tin ngữ cảnh khác	Biến đổi	Dựa vào việc xử lý của nguồn phát	Đầu vào không đầy đủ, cơ chế nguồn, ...

Bảng 1: Các thuộc tính đặc trưng của ngữ cảnh

Ngoài ra khi nghiên cứu về các đặc trưng ngữ cảnh, còn một số vấn đề sau:

- **Chưa xác định** khi không có thông tin nào về vật chất là sẵn có
- **Mơ hồ (tối nghĩa)** khi có một số báo cáo khác nhau về vật chất, ví dụ 2 vị trí cùng được đọc cho một người được lấy từ các thiết bị định vị riêng
- **Không chính xác**: khi trạng thái được báo cáo không đúng với trạng thái đúng, ví dụ khi vị trí của 1 người được biết trong miền giới hạn, nhưng vị trí trong miền này không được chốt cho mức độ yêu cầu chuẩn xác
- **Sai**: khi có lỗi giữa trạng thái được báo cáo và trạng thái thực của vật chất

Tính chưa xác định thường là do các vấn đề về kết nối, cảm biến và các lỗi khác. Thông tin mơ hồ phát sinh khi giá trị của một vật chất có thể được lấy một cách độc lập từ nhiều nguồn. Thông tin ngữ cảnh được lấy từ cảm biến là thường xuyên thay đổi và chấp nhận các vấn đề như tính không chính xác và staleness (tính cũ, chưa cập nhật). Thông tin ngữ cảnh được cung cấp bởi người dùng thường chậm thay đổi, kiểu thông tin này được gọi là tĩnh (tức không bao giờ thay đổi nên độ chính xác cao). Kiểu thông tin ngữ cảnh profiled được lấy trực tiếp từ người dùng trong form về đặc tả thông tin của họ hoặc lấy gián tiếp qua ứng dụng của họ [7], ví dụ phần mềm lập lịch duy trì lịch sử hoạt động của người dùng. Thông tin profiled thường cũ và không đầy đủ. Cuối cùng, đặc điểm về thông tin mong muốn thường được xác định rộng bởi các thông tin cơ bản.

1.1.3 Phân loại ngữ cảnh

Shilit [8] xác định có 3 loại ngữ cảnh:

- **Ngữ cảnh thiết bị**: là các thông tin ngữ cảnh liên quan đến thiết bị như khả năng xử lý CPU, bộ nhớ, mạng

- Ngữ cảnh người dùng: gồm có thông tin người dùng, sở thích người dùng và thông tin về các ứng dụng của người dùng
- Ngữ cảnh vật lý: vị trí, thời tiết, ánh sáng, ...

Tất cả các thông tin ngữ cảnh này đến từ nhiều đối tượng khác nhau trong môi trường xung quanh như các cảm biến, các ứng dụng và các thiết bị. Và các đối tượng cung cấp ngữ cảnh này là không đồng nhất và được thể hiện trong một mô hình chung cần được định nghĩa khá tốt để người dùng ứng dụng có thể hiểu.

Pash [9] phân loại ngữ cảnh thành 4 chiều là: ngữ cảnh tĩnh người sử dụng, ngữ cảnh động của người dùng, kết nối mạng và ngữ cảnh môi trường. Mỗi chiều ngữ cảnh được miêu tả bởi tham số ngữ cảnh tương ứng, ví dụ tham số ngữ cảnh tĩnh của người dùng là profiled, các sở thích, mối quan tâm của anh ta, ...

Ngữ cảnh	Chiều ngữ cảnh	Các tham số ngữ cảnh
	Ngữ cảnh tĩnh của người dùng	Profiled, thói quen, sở thích
	Ngữ cảnh động của người dùng	Vị trí, nhiệm vụ hiện thời hoặc có liên quan tới người hay đối tượng khác
	Ngữ cảnh môi trường	Thời tiết, tiếng ồn, thời gian
	Kết nối mạng	Đặc tính mạng, các đặc tả thiết bị di động đầu cuối

Bảng 2: Phân loại các chiều của ngữ cảnh

Đối với những ngữ cảnh này, có 3 loại hành động được thể hiện. Bộ tích hợp ngữ cảnh thu thập dữ liệu ngữ cảnh thô từ các cảm biến để làm tăng dữ liệu. Bộ phân tích ngữ cảnh chuyển dữ liệu thô từ cảm biến thành các dữ liệu mức cao mà con người có thể hiểu. Các ngữ cảnh mức cao được tạo từ dữ liệu khác với các nguồn dữ liệu ngữ cảnh theo các chiều khác nhau (vị trí, nhiệt độ, ...). Bộ diễn dịch thực hiện bằng việc sử dụng các luật.

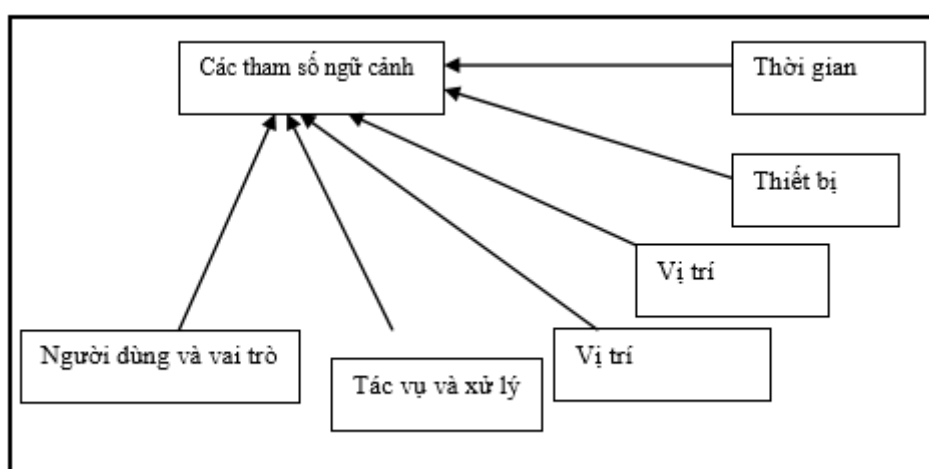
Câu hỏi cách tích hợp các phần tử ngữ cảnh, hay các tham số như thế nào trong một mô hình ứng dụng được tiếp cận theo 2 cách khác nhau. Đầu tiên, có thể hiểu là cho phép định nghĩa tùy ý các tham số ngữ cảnh và kết hợp tùy ý với các phần tử của ontology miền. Điều này sẽ cung cấp tính linh hoạt tối đa trong việc liên kết các tham số ngữ cảnh và các phần tử miền lĩnh vực, và đảm bảo mô hình có khả năng biểu diễn không giới hạn về mặt lý thuyết. Cách tiếp cận thứ 2 là xác định một hạng mục các tham số ngữ cảnh, và yêu cầu các giá trị đặc tả liên quan tới miền lĩnh vực để gán cho mỗi lớp. Điều này dường như là một hạn chế chính cho mô hình khái niệm, tuy nhiên việc sử dụng cụ thể chúng ta mặc định rằng việc nhóm các tham số là không thể tránh khỏi (xem hình 3), từ triển vọng mô hình hóa và việc sử dụng. Vì chúng ta giả thiết rằng các biểu mẫu tương tác với người được yêu cầu để hoàn thành và điều chỉnh mô hình, việc nhóm các tham số là rất cần thiết để người dùng duy trì một cái nhìn tổng quan về mô hình. Mặc khác, chúng ta giả thiết rằng không phải tất cả các phần tử ngữ cảnh đều sẽ được liên kết với ontology miền. Điều này có

nghĩa sẽ có một khái niệm "hàng xóm" hoặc cụ thể hơn là khoảng cách giữa các phần tử ngữ cảnh; các phần tử chắc chắn sẽ liên quan tới một ngữ cảnh được cho, thậm chí chúng không được liên kết trực tiếp tới miền lĩnh vực. Để đạt được một hàng xóm, một số biểu mẫu các hạng mục được yêu cầu. Hơn nữa Use & Role cung cấp một hạng mục về người dùng theo các nguyên tắc của họ như các kiểu khách hàng, hay các kiểu nhân viên khác nhau.

Tác vụ và xử lý biểu diễn một ngữ cảnh chức năng như các đối tượng công việc cho nhân viên.

Vị trí là một hạng mục của vị trí có liên quan đến ứng dụng, có thể là thành phố ...

Thời gian thể hiện kiểu khác của thông tin thời gian có thể liên quan như vùng thời gian của client, thời gian thực, thời gian ảo, ...



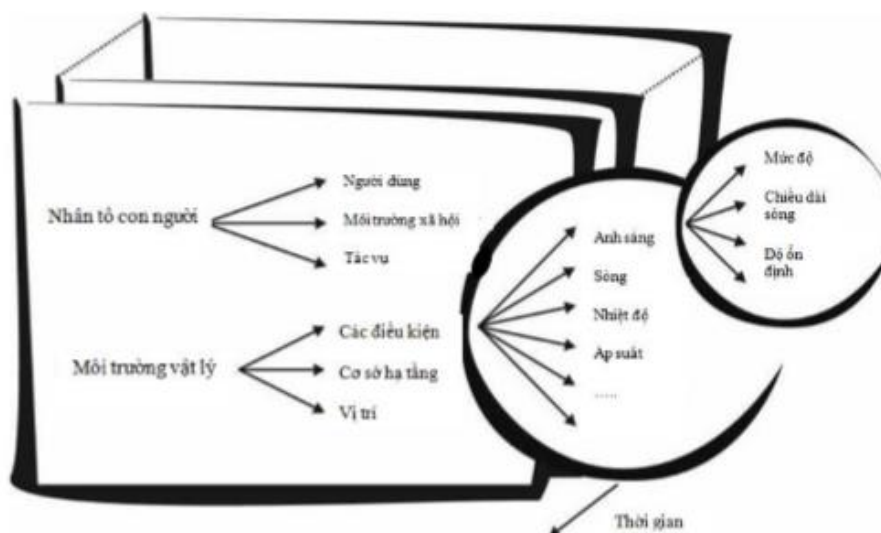
Hình 3: Tổng quan các hạng mục ngữ cảnh

1.1.4 Mô hình làm việc cho ngữ cảnh

Cấu trúc khái niệm theo [6] như sau:

- Một ngữ cảnh miêu tả một tình huống và môi trường mà một thiết bị hay người dùng ở trong đó
- Một ngữ cảnh được xác định bởi một tên duy nhất
- Một ngữ cảnh có một tập các đặc tính liên quan
- Mỗi đặc tính liên quan có một miền giá trị xác định (rõ ràng hoặc không rõ ràng) bởi ngữ cảnh

Theo như cách xác định này, ngữ cảnh liên quan tới nhân tố con người và ngữ cảnh liên quan tới một trường vật lý được phân biệt rõ ràng, và được phân theo từng hạng mục như hình 4 sau:



Hình 3: Không gian đặc tính ngữ cảnh

Ngữ cảnh liên quan tới nhân tố con người được cấu trúc theo ba hạng mục: thông tin về người dùng (tri thức về hành vi thói quen, trạng thái cảm xúc, điều kiện tâm sinh lý ...), môi trường xã hội của người dùng (cùng vị trí với những người nào khác, tương tác xã hội, nhóm tham gia, ...) và các tác vụ của người dùng (hoạt động tự nguyện, nhiệm vụ thực thi, mục tiêu chung ...). Cũng tương tự, ngữ cảnh liên quan tới môi trường vật lý được phân thành ba nhóm: vị trí (vị trí chắc chắn, vị trí liên quan, cùng vị trí, ...), cơ sở hạ tầng (các tài nguyên xung quanh dùng để thực hiện tính toán, sự giao tiếp, ...) và các điều kiện vật lý (nhiều, ồn, ánh sáng, áp suất ...).

1.2 Nhận biết ngữ cảnh (context-awareness)

1.2.1 Xu thế nhận biết ngữ cảnh và lợi ích trong việc cá nhân hóa ứng dụng di động

Nhận biết ngữ cảnh đang nổi lên như một cách tiếp cận phổ biến cho việc xây dựng các ứng dụng trong môi trường tính toán mọi nơi và di động, có khả năng thích ứng một cách tự trị với môi trường của chúng, làm giảm sự định hướng của người dùng. Ví dụ chung nhất về các ứng dụng nhận biết ngữ cảnh gồm có các ứng dụng hướng dẫn du lịch đưa thông tin chi tiết theo vị trí, sở thích và điện thoại di động để thích ứng với các hành vi tùy theo nơi mà người dùng đang đứng, người mà họ đứng cùng và điều mà họ thích làm. Hiện có một loạt các môi trường thông minh như: nhà, bệnh viện, phòng họp.

Nhận biết ngữ cảnh rất hữu ích cho môi trường tính toán với các thiết bị di động. Nó thích ứng theo thay đổi của môi trường và cải tiến theo giao diện người dùng. Với một thiết bị di động có kết nối internet thì người dùng trải qua nhiều tình huống: một mình trong văn phòng, công tác cùng đồng nghiệp, rời khỏi văn phòng, đi bộ trong công viên, xuống xe, ...

Nâng cấp tương tác giữa người và các thiết bị di động cũng là một động lực để phát triển các ứng dụng nhận biết ngữ cảnh. Về cơ bản, thông tin ngữ cảnh có thể đạt được một cách rõ ràng qua sự nhận biết bởi các cảm biến và không rõ ràng từ phía người dùng. Thông tin này được lọc theo từng luồng để ứng dụng sử dụng cho mục đích người dùng, đồng thời tránh được việc tràn thông

tin hay thông tin bị quá tải. Và với mỗi luồng mà ứng dụng cung cấp cho người dùng, thông tin ngữ cảnh có thể được thêm ngữ nghĩa bởi chính người dùng đó.

Phát triển cá nhân hóa các ứng dụng di động nhận được nhiều lợi ích từ việc nhận biết ngữ cảnh như:

- Giao diện đáp ứng người dùng: các kiểu tương tác và các chế độ hiển thị tùy thuộc rất nhiều vào môi trường xung quanh, nhận biết ngữ cảnh có thể thực thi các đáp ứng với các điều kiện môi trường này.
- Liên lạc nhận biết ngữ cảnh: các liên lạc chung là bắt buộc cho các thiết bị di động
- Chủ động lập lịch ứng dụng: việc lựa chọn trước để nhận biết ngữ cảnh của ứng dụng sẽ hỗ trợ tương tác kiểu adhoc

Vị trí, định danh, thời gian và hành động là những kiểu ngữ cảnh quan trọng đặc trưng cho một tình huống của một thực thể cụ thể. Các kiểu ngữ cảnh này trả lời cho câu hỏi ai, điều gì, khi nào và ở đâu. Ví dụ, cho định danh của một người, chúng ta có thể biết được nhiều thông tin liên quan khác như số điện thoại, địa chỉ nhà, địa chỉ mail, ngày sinh, danh sách bạn bè, các mối quan hệ với những người khác... Với vị trí của thực thể, chúng ta có thể xác định được đối tượng hay người nào gần thực thể và hành động gì đang xảy ra gần thực thể.

1.2.2 Khái niệm nhận biết ngữ cảnh

Tính toán nhận biết ngữ cảnh lần đầu tiên được xem xét bởi Schilit và Theimer năm 1994 với phần mềm thích ứng theo vị trí sử dụng để sưu tầm những đối tượng và người gần đó đồng thời thay đổi các đối tượng này theo thời gian.

Định nghĩa các ứng dụng nhận biết ngữ cảnh đầu tiên được đưa ra cũng bởi Schilit và Theimer đã giới hạn lại khái niệm của các ứng dụng một cách đơn giản là các ứng dụng thích ứng theo ngữ cảnh. Nhận biết ngữ cảnh trở thành một khái niệm gì đó mà gần với thuật ngữ là: “thích ứng” (adapt) (Brown 1996), “hành động lại” (“Cooperstock, Tanikoshi 1995”), “tương thích” (Elrod, Hall 1993), “theo tình huống” (Hull, Neaves 1997).

Tiếp đó, các định nghĩa dần trở nên rõ ràng và đặc tả hơn: thích ứng theo ngữ cảnh. Dey đã xác định khái niệm mới theo cách ngữ cảnh được dùng và các đặc tính nhận biết ngữ cảnh khác nhau. Theo ông, định nghĩa sự nhận biết ngữ cảnh như sau: **Một hệ thống là nhận biết ngữ cảnh nếu nó sử dụng ngữ cảnh để cung cấp thông tin liên quan hay các dịch vụ tới người dùng, trong đó, mức độ liên quan tùy thuộc vào tác vụ của người dùng. Và định nghĩa này được lựa chọn là định nghĩa dùng chung cho tính toán nhận biết ngữ cảnh.**

Định nghĩa này cho chúng ta biết cách để xác định xem liệu một ứng dụng có là nhận biết ngữ cảnh hay không. Điều này rất hữu ích khi xác định kiểu ứng dụng mà chúng ta muốn hỗ trợ. **Và theo Dey [10] nhận biết ngữ cảnh là một thuộc tính của một hệ thống có sử dụng ngữ cảnh để cung cấp các thông tin hay dịch vụ liên quan tới người dùng.**

1.2.3 Tính toán nhận biết ngữ cảnh

Tính toán nhận biết ngữ cảnh tức là nó giúp cho một ứng dụng hợp nhất tất cả thức về các chiều ngữ cảnh khác nhau như người dùng là ai, người dùng đang

làm gì, người dùng ở đâu và thiết bị tính toán nào người dùng đang sử dụng [11].

Nhận biết ngữ cảnh đang ngày càng nhận được nhiều quan tâm như một hướng tiếp cận thiết kế mới phù hợp cho tính toán mọi nơi. Các phần mềm nhận biết ngữ cảnh dựa trên đa dạng các kiểu thông tin ngữ cảnh để tạo các quyết định về cách thích ứng linh động đáp ứng các yêu cầu người dùng. Các thông tin này được lấy từ một tập các nguồn gồm profiled người dùng, các ứng dụng và cảm biến. Một vài kiểu thông tin ngữ cảnh được cảm nhận nội tại và phải được bảo vệ để đáp ứng yêu cầu riêng tư của người dùng.

1.3 Hệ gợi ý

1.3.1 Định nghĩa hệ gợi ý

Hệ gợi ý (Recommender system - RS) là một dạng của hệ thống lọc thông tin, nó được sử dụng để dự đoán sở thích (preference) hay xếp hạng (rating) mà người dùng có thể dành cho một mục thông tin (item) nào đó mà họ chưa xem xét tới trong quá khứ (item có thể là một sản phẩm, bộ phim, video clip, music, sách, ...) [12] nhằm gợi ý các mục thông tin “có thể quan tâm” bởi người dùng. Hệ gợi ý sẽ đưa ra các gợi ý dựa trên quá trình thu thập, xử lý và phân tích dữ liệu từ người dùng. Dữ liệu đó được chia làm 2 loại là tường minh (explicit) bằng cách yêu cầu người dùng phản hồi trực tiếp và tiềm ẩn (implicit) bằng cách tự động suy luận dựa trên những tương tác của người dùng với hệ thống như: vị trí thay đổi, số lần nhấp chuột, thời gian quan sát... Trong hầu hết các trường hợp, bài toán gợi ý được coi là bài toán dự đoán việc xếp hạng (rating) của các sản phẩm (phim, sản phẩm tiêu dùng, sách, nhạc...) chưa được người dùng biết đến. Việc dự đoán này thường dựa trên những đánh giá đã có của chính người dùng đó hoặc những người dùng khác. Ví dụ, những bộ phim được dự đoán là sẽ có xếp hạng cao nhất sẽ được dùng để gợi ý. Có khá nhiều ứng dụng nổi tiếng về hệ gợi ý như: gợi ý sản phẩm của Amazon và Ebay, hệ gợi ý phim của Netflix và Youtube,...

Hệ thống gợi ý đã chứng minh được ý nghĩa to lớn: giúp cho người sử dụng trực tuyến đối phó với tình trạng quá tải thông tin. Hệ thống gợi ý trở thành một trong những công cụ mạnh mẽ và phổ biến trong thương mại điện tử. Mục đích của hệ thống gợi ý là dựa vào hành vi từ thói quen, nhu cầu... trong quá khứ của người sử dụng để dự đoán sở thích trong tương lai của họ.

1.3.2 Dự đoán trong hệ gợi ý

Một cách hình thức, gọi U là tập người dùng, I là tập các sản phẩm có thể được gợi ý. Tập sản phẩm I có thể lên đến hàng trăm, hàng nghìn thậm chí là hàng triệu sản phẩm. Tương tự như vậy, tập người dùng U cũng có thể rất lớn lên đến hàng triệu trường hợp. Để dự đoán xếp hạng (hay tính tiện ích) của sản phẩm i đối với người dùng u thì người ta đưa ra hàm xếp hạng (rating) $r: U \times I \rightarrow R$, trong đó R là tập các giá trị xếp hạng được thứ tự toàn phần (ví dụ số nguyên dương hoặc số thực trong tập xác định). Mô hình này còn được gọi với tên gọi là mô hình dự đoán 2 chiều (two-dimensional recommendation framework).



Hình 4: Các thành phần cơ bản của tiến trình gợi ý truyền thống

Với mỗi người dùng $u \in U$, chúng ta có thể chọn được sản phẩm $i \in I$ sao cho hàm xếp hạng của người dùng u đối với item i là lớn nhất.

$$\forall u \in U, i'_u = \arg \max_{i \in I} \hat{r}(u, i)$$

Tập người dùng U ($u \in U$; $|U|=n$), tập sản phẩm I ($i \in I$; $|I|=m$), và $r_{ui} \in R$ là xếp hạng của người dùng u cho sản phẩm i . Trong hệ gợi ý, tính tiện ích của sản phẩm i thường biểu thị mức độ quan tâm của người dùng tới một mặt hàng cụ thể thông qua trọng số; ví dụ người dùng Alice đánh giá sản phẩm 3 có trọng số là 4 như trong bảng 3.

	Item1	Item2	Item3	Item4	Item5
Alice	5	3	4	4	?
Gil	3	1	2	3	3
Kai	4	3	4	3	5
Blue	1	5	5	2	1

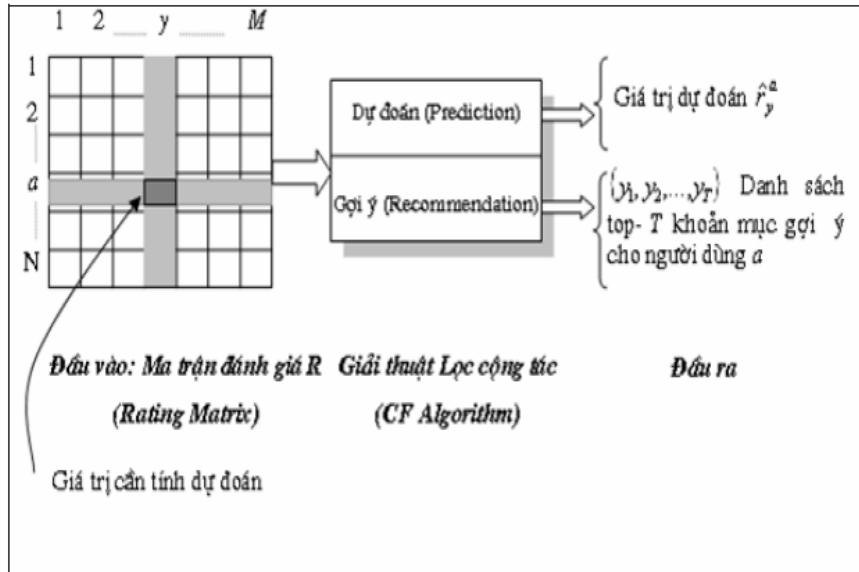
Bảng 3: Bảng ma trận trọng số (đánh giá) của hệ gợi ý

1.3.3 Các phương pháp tiếp cận truyền thống trong hệ gợi ý

Có rất nhiều phương pháp tiếp cận trong hệ gợi ý, tuy nhiên có thể chia thành 3 nhóm kỹ thuật chính như sau:

- Gợi ý dựa trên cộng tác (collaborative filtering - CF): người dùng sẽ được gợi ý những sản phẩm được ưa chuộng xuất phát từ những người dùng có cùng sở thích và thị hiếu với mình (có độ tương quan cao)
- Gợi ý dựa trên nội dung (content-based filtering): người dùng sẽ được gợi ý những sản phẩm tương tự với những sản phẩm đã được người dùng đó ưa thích trước đây
- Gợi ý dựa trên cách tiếp cận kết hợp (hybrid approach): kết hợp cả hai phương pháp lọc cộng tác và dựa trên nội dung.

Trong các phương pháp tiếp cận trên, phương pháp tiếp cận dựa trên lọc cộng tác thường được sử dụng nhiều nhất. Phương pháp này dựa trên những hành vi quá khứ của người dùng, ví dụ như: lịch sử giao dịch, đánh giá sản phẩm, xem một bộ phim, nghe một bài hát, ... và đặc biệt là nó không cần thiết phải tạo ra các hồ sơ tường minh (explicit feedback) cho người dùng. Để gợi ý các sản phẩm cho người dùng, hệ thống lọc cộng tác cần so sánh các đối tượng cơ bản khác nhau như các sản phẩm và người dùng. Một hệ thống lọc cộng tác truyền thống thường có kiến trúc như sau:



Hình 5: Kiến trúc tổng quan của hệ thống lọc cộng tác

1.3.4 Đánh giá hệ gợi ý

Theo cuốn sách ‘Recommender Systems Handbook’, chương 8 [13], việc đánh giá một hệ gợi ý trong nhiều trường hợp thì đó là việc so sánh những hướng tiếp cận nào tốt phù hợp để thiết kế quy trình hay hệ thống. Từ bước đầu tiên là chọn ra thuật toán phù hợp để quyết định những thuộc tính nào của ứng dụng được dùng để đưa ra quyết định, cụ thể là một hệ gợi ý có rất nhiều thuộc tính khác nhau có thể ảnh hưởng đến trải nghiệm người dùng như độ chính xác, chắc chắn, khả năng mở rộng... Các thuật toán có thể so sánh bằng những hệ số (metric). Có hai nhóm tiêu chí đánh giá: các tiêu chí định lượng và tiêu chí định tính. Các tiêu chí định lượng được giành riêng cho việc đánh giá số lượng các gợi ý liên quan, chúng tương ứng với độ chính xác. Các tiêu chí định tính được sử dụng để đánh giá chung về chất lượng của hệ gợi ý.

Các tiêu chí định lượng:

- *Đánh giá độ chính xác của hàm dự đoán xếp hạng (rating prediction)*: việc đánh giá chính xác các dự đoán có thể sử dụng sai số bình phương trung bình (Mean square error – MSE), căn của sai số bình phương trung bình (Root mean square error – RMSE), sai số trung bình tuyệt đối (Mean absolute error – MAE) [14]. Tính chính xác của các dự đoán được đo trên n quan sát, trong đó p_i là giá trị dự đoán xếp hạng của sản phẩm i và r_i là giá trị xếp hạng thực tế của sản phẩm i .

$$MSE = \frac{1}{n} \sum_{i=1}^n (p_i - r_i)^2$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (p_i - r_i)^2}$$

$$MAE = \frac{1}{n} \sum_{i=1}^n (p_i - r_i)$$

Các chỉ số này thích hợp cho một CSDL không phải nhị phân và cho một giá trị dự đoán là số. Nó giúp đo lường mức độ sai số của các dự đoán. Các giá trị này đo lường này bằng 0 khi hệ thống đạt hiệu quả tốt nhất. Giá trị này càng cao thì hiệu quả của hệ thống, hay nói cách khác là độ chính xác dự đoán của hệ thống càng thấp.

- *Đánh giá độ chính xác của hàm gợi ý (item recommendation)*: ngoài việc đánh giá tính chính xác của các dự đoán, một số chỉ số khác như *precision*, *recall* và *F_score*, *R_score* được dùng để đánh giá độ chính xác của hàm gợi ý (đánh giá việc sử dụng của các dự đoán) trong trường hợp CSDL nhị phân (Herlocker J.L *et al*, 2004).

- **Precision**: là tỷ lệ giữa số lượng các gợi ý phù hợp và tổng số các gợi ý đã cung cấp (đã tạo ra). Precision bằng 100% có nghĩa là tất cả các gợi ý cho người dùng đều phù hợp.

$$Precision = \frac{\text{Số lượng gợi ý phù hợp}}{\text{Số lượng gợi ý tạo ra}}$$

- **Recall**: được định nghĩa bởi tỷ lệ giữa số lượng các gợi ý phù hợp và số lượng các mục dữ liệu mà người dùng đã chọn (xem, nghe, mua, đọc, ...). Recall được sử dụng để đo khả năng hệ thống tìm được những mục dữ liệu phù hợp so với những gì mà người dùng cần.

$$Recall = \frac{\text{Số lượng gợi ý phù hợp}}{\text{Số lượng sản phẩm được chọn bởi người dùng}}$$

- **F-score**: Precision và Recall được xem là hữu ích trong việc đánh giá một hệ gợi ý. Tuy nhiên, trong một số trường hợp thì precision và recall lại có giá trị tỷ lệ nghịch với nhau. Ví dụ số lượng gợi ý mà hệ thống tạo ra là 10, số lượng gợi ý phù hợp là 3, số lượng sản phẩm chọn bởi người dùng là 3 thì độ chính xác thấp (30%), tuy nhiên giá trị recall lại cao (100%), nghĩa là độ chính xác thấp nhưng người dùng lại hài lòng bởi vì họ chọn đúng 3 sản phẩm mà hệ thống gợi ý đúng cả 3 sản phẩm đó. Trong tình huống đó, chỉ số **F-score** được sử dụng để đánh giá hiệu quả tổng thể của hệ thống bằng cách kết hợp hài hòa hai chỉ số Recall và Precision.

$$Fscore = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

- **R_{score} hay Bresse score [15]**: cũng là một trong những chỉ số đánh giá khả năng sử dụng dự đoán nhưng chỉ số này chính xác đến thứ tự của các gợi ý được xây dựng. R_{score} đánh giá vị trí của sản phẩm được chọn bởi người dùng trong danh sách sản phẩm gợi ý được tạo ra bởi hệ thống. Ví dụ, một hệ thống gợi ý cho người dùng 10 sản phẩm sắp xếp theo thứ tự ưu tiên từ cao đến thấp. Nếu người dùng chọn sản phẩm đầu tiên trong danh sách thì hệ thống gợi ý hiệu quả hơn khi người dùng chọn sản phẩm có thứ tự thứ 10. Chỉ số R_{score} được tính dựa vào tỷ lệ giữa thứ tự của mục gợi ý đúng (Rankscore_p) và thứ tự của mục gợi ý đúng tốt nhất (Rankscore_{max}) như công thức sau:

$$Rankscore = \frac{Rankscore_p}{Rankscore_{max}}$$

$$Rankscore_p = \sum_{i \in h} 2^{-\frac{Rank(i)-1}{\alpha}}$$

$$Rankscore_{max} = \sum_{i=1}^{|T|} 2^{-\frac{i-1}{\alpha}}$$

Trong đó:

- h là tập sản phẩm gợi ý người dùng
- Rank trả về thứ tự sắp xếp của một sản phẩm trong danh sách gợi ý
- T là tập tất cả các sản phẩm người dùng quan tâm
- α là chu kỳ nửa phân kỳ (xác suất mà mục dữ liệu trong danh sách gợi ý được chọn là 50%).

Các chỉ số Precision, Recall, và F_score, Rscore thường được sử dụng đối với các hệ gợi ý trong lĩnh vực thương mại điện tử. Các chỉ số đánh giá, công thức tương ứng và một số hệ gợi ý/nghiên cứu đã áp dụng các chỉ số tương ứng được trình bày trong bảng 4:

STT	Chỉ số	Công thức	Hệ thống đã áp dụng
1	MAE	$MAE = \frac{1}{n} \sum_{i=1}^n (p_i - r_i)$	MovieLens
2	MSE	$MSE = \frac{1}{n} \sum_{i=1}^n (p_i - r_i)^2$	Netflix
3	RMSE	$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (p_i - r_i)^2}$	BookCrossing
4	Precision	$\frac{\text{Số lượng gợi ý phù hợp}}{\text{Số lượng gợi ý tạo ra}}$	EachMovie
5	Recall	$\frac{\text{Số lượng gợi ý phù hợp}}{\text{Số lượng sản phẩm được chọn bởi người dùng}}$	Yeong et al, 2005
6	F-score	$Fscore = \frac{2 \times Precision \times Recall}{Precision + Recall}$	MovieLens
7	R_score	$Rankscore = \frac{Rankscore_p}{Rankscore_{max}}$ $Rankscore_p = \sum_{i \in h} 2^{-\frac{Rank(i)-1}{\alpha}}$ $Rankscore_{max} = \sum_{i=1}^{ T } 2^{-\frac{i-1}{\alpha}}$	TaFeng, B&Q (Breese, J.S and D.Heckerman, 1998, Hsu, C. and H.Chung. 2004)

Bảng 4: Các phương pháp đánh giá

Tiêu chí định tính:

Trong những giai đoạn đầu phát triển thì hệ gợi ý chỉ sử dụng các độ đo chính xác định lượng như đã trình bày. Tuy nhiên, người dùng ngày càng có yêu cầu cao hơn và nhiều hơn về chất lượng của các gợi ý. Nếu chỉ xét độ chính

xác thì không đủ để đánh giá hiệu quả của một hệ gợi ý nên cần đưa thêm các thuộc tính chất lượng của các gợi ý, bao gồm:

- *Tính mới của các gợi ý*
- *Tính đa dạng (Diversity) của các gợi ý*
- *Độ bao phủ của các gợi ý*
- *Sự hài lòng của người dùng*

1.3.5 Các thách thức của hệ gợi ý truyền thống

Những thách thức từ khi hệ gợi ý được quan tâm phát triển đến nay có thể kể đến như dữ liệu thưa, khả năng mở rộng phạm vi, biến thể từ (nhóm từ đồng nghĩa), nhập nhằng nghĩa, quyền riêng tư... Đặc tính những thách thức này cụ thể như sau:

Dữ liệu thưa (sparsity). Trường hợp thường thấy khi tập sản phẩm item thường lớn và ngày càng được mở rộng thêm, lượng người dùng mới cũng tăng theo thời gian, cộng thêm người dùng chỉ xem một phần rất nhỏ trong danh sách sản phẩm đó, dẫn đến ma trận user – item cực kỳ thưa. Người dùng đặc biệt là người dùng tại Việt Nam cũng không có thói quen đưa ra đánh giá hay nhận xét cho sản phẩm họ xem hoặc mua, rất khó để nhận định họ thích hay không. Một khởi đầu khó cho nhóm người dùng mới và sản phẩm mới, hay còn được biết đến là vấn đề cold start[16]

Khả năng mở rộng phạm vi. Khi số người dùng và sản phẩm ngày càng mở rộng, phạm vi xử lý sẽ trở thành một vấn đề lớn.

Sản phẩm mới, người dùng mới (cold-start). Những đối tượng này rất khó gợi ý nếu chưa hề hoặc rất ít thông tin lịch sử trước đó. Không chỉ những sản phẩm mới ngay cả những sản phẩm mô tả cụ thể, chung chung sẽ gây nhầm. Với người dùng mới hệ thống cần học để hiểu về sở thích của họ, có một vài phương pháp đã được đề xuất để giải quyết vấn đề này, là theo hướng lai, kết hợp các phương pháp, có thể là lọc cộng tác và nội dung, ngoài ra còn có một kỹ thuật để xác định những sản phẩm tốt nhất cho người dùng mới được đề cập trong [17], đây là những kỹ thuật áp dụng cho chiến lược lọc cộng tác.

Gian lận là khi những nhà cung cấp sản phẩm, hay dịch vụ có những thủ thuật giả thông tin khách hàng, giúp họ tăng lợi ích.

Để bị tấn công là vấn đề thường xảy ra như kiểu tấn công dịch vụ, làm trì trệ khả năng cung cấp dịch vụ tới người dùng thực

Bảo mật riêng tư: xu hướng cá nhân hóa là khi hệ gợi ý cần hiểu rõ về từng người dùng.

Một số thách thức về môi trường như: Lượng bán lẻ, người dùng và sản phẩm có thể rất lớn (lên tới hàng triệu); các ứng dụng thường yêu cầu kết quả trả về trong thời gian thực (không quá nửa giây) trong khi vẫn phải đáp ứng gợi ý chất lượng tốt; người dùng cũ có rất nhiều thông tin thừa, người dùng mới lại thiếu thông tin và dữ liệu người dùng thì dễ bị trôi mất.

Ngoài ra còn rất nhiều vấn đề khác cần quan tâm như: sự đa dạng và chính xác, dễ dàng bị tấn công, giá trị thời gian, đánh giá độ hiệu quả, giao diện người dùng,...

1.4 Kết luận chương

Qua chương 1, đề tài đã làm rõ các nội dung về khái niệm ngữ cảnh, các đặc trưng, mô hình quản lý cũng như các phương pháp nhận biết ngữ cảnh. Nội dung chương này cũng đã làm rõ các lý thuyết về hệ gợi ý, các phương pháp tiếp cận trong dự đoán xếp hạng cũng như các kỹ thuật chính áp dụng trong hệ gợi ý truyền thống dựa trên dữ liệu 2 chiều (users, items). Đây là những nội dung cơ bản, là tiền đề cho việc tiếp cận ứng dụng ngữ cảnh vào trong hệ gợi ý mà sẽ được đề tài làm rõ trong Chương 2 - Hệ gợi ý dựa trên nhận biết ngữ cảnh.

Chương 2. HỆ GỢI Ý DỰA TRÊN NHẬN BIẾT NGỮ CẢNH

Như đã trình bày trong phần 1.4 Chương 1, một hệ gợi ý truyền thống thường xuyên phải đối diện với các thách thức như **khởi động chậm (cold start)**, **dữ liệu rời rạc (sparsity)** có thể dẫn đến việc gợi ý và xếp hạng các sản phẩm cho các người dùng mất đi độ chính xác cần thiết. Để giải quyết các thách thức đó, rất nhiều các phương pháp tiếp cận hiện đại đã được nghiên cứu và áp dụng, trong đó phương pháp tiếp cận sử dụng các thông tin ngữ cảnh như là một trong các yếu tố tiềm ẩn để tích hợp vào các hệ gợi ý nhằm nâng cao tính chính xác cũng như hiệu quả gợi ý cho người dùng đang là một xu hướng của các hệ thống gợi ý hiện đại. Các thông tin ngữ cảnh người dùng sẽ được mô hình hóa và tích hợp vào các hệ gợi ý, khi đó một hệ gợi ý dựa trên nhận thức ngữ cảnh (Context-awareness Recommender System – CARS) [18] sẽ tính toán xếp hạng và gợi ý cho người dùng không chỉ dựa trên thông tin người dùng và sản phẩm như hệ gợi ý truyền thống mà còn sử dụng cả thông tin ngữ cảnh liên quan. Nếu gọi Contexts là tập ngữ cảnh thì hàm gợi ý khi đó được viết lại như sau:

r: Users x Items x Contexts -> Ratings

Ví dụ: Xem xét một ứng dụng xem phim có tích hợp tính năng gợi ý phim phù hợp cho người dùng, khi đó người dùng và các phim được mô tả như là các quan hệ bao gồm các thuộc tính sau:

- Phim (Movie): là tập hợp tất cả các bộ phim có thể được gợi ý, được định nghĩa bởi Movie(MovieID, Title, Length, ReleaseYear, Director, Genre).
- Người dùng (User): là tập hợp tất cả người dùng được gợi ý xem phim, được định nghĩa bởi User(UserID, Name, Address, Age, Gender, Profession).

Kế tiếp, thông tin ngữ cảnh bao gồm 3 loại thông tin sau cũng sẽ được định nghĩa tương ứng như là các quan hệ có các thuộc tính sau:

- Rạp chiếu (Theatre): là các rạp chiếu phim, được định nghĩa bởi Theatre(TheatreID, Name, Address, Capacity, State, Country).
- Thời gian (Time): là thời gian mà bộ phim được chiếu hoặc đã chiếu, được định nghĩa bởi Time(Date, DayOfWeek, TimeOfWeek, Month, Quarter, Year). Trong đó thuộc tính DayOfWeek sẽ có các giá trị Mon, Tue, Wed, Thu, Fri, Sat, Sun và thuộc tính TimeOfWeek sẽ có các giá trị “Weekday” and “Weekend”.
- Bạn cùng xem phim (Companion): đại diện cho một người hoặc một nhóm người có thể xem một bộ phim, được định nghĩa bởi Companion(companionType), trong đó thuộc tính companionType sẽ có các giá trị “alone”, “friends”, “girlfriend/boyfriend”, “family”, “co-worker”, và “others”.

Với các tham số đầu vào như trên thì giá trị xếp hạng của một bộ phim được đánh giá bởi một người dùng sẽ phụ thuộc vào các thông tin ngữ cảnh như vị trí chiếu phim, thời gian xem phim, xem phim với ai, và xem như thế nào. Khi đó hàm xếp hạng của ứng dụng sẽ được biểu diễn như sau:

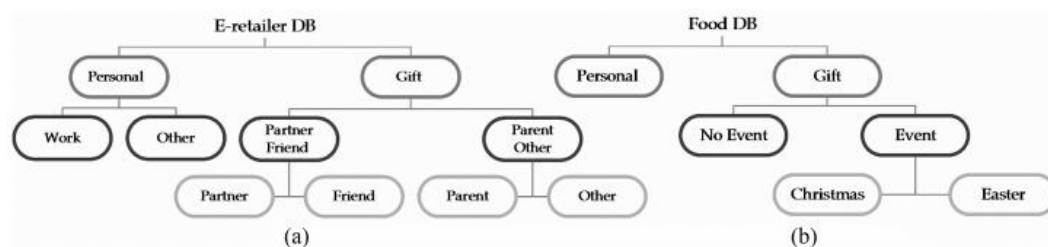
R: Users x Movies x Theatres x Times x Companions -> Rating

2.1 Cấu trúc thông tin ngữ cảnh trong hệ gợi ý

Trong một hệ gợi ý dựa trên ngữ cảnh, thông tin ngữ cảnh có thể biểu diễn dưới 2 dạng cấu trúc bao gồm cấu trúc dữ liệu cây phân cấp và cấu trúc dữ liệu đa chiều.

2.1.1 Cấu trúc dữ liệu phân cấp

Thông tin ngữ cảnh được định nghĩa bởi một tập K các chiều ngữ cảnh, mỗi chiều ngữ cảnh K trong tập K được định nghĩa bởi một tập q thuộc tính $K = (K_1, \dots, K_q)$, K có cấu trúc phân cấp và thể hiện cho một loại ngữ cảnh cụ thể. Ngữ cảnh được xác định bởi thuộc tính K_q sẽ có mức định nghĩa thấp hơn so với ngữ cảnh được xác định bởi thuộc tính K_1 . Ví dụ biểu diễn dữ liệu ngữ cảnh dưới dạng cấu trúc cây phân cấp 4 mức trong các ứng dụng e-retailer tại hình 7. như sau:



Hình 6: Cấu trúc phân cấp của ngữ cảnh trong hệ gợi ý

Tại đỉnh của cây phân cấp sẽ biểu diễn cho ngữ cảnh mua bất kỳ, tại mức kế tiếp ngữ cảnh được biểu diễn bởi thuộc tính $K_1 = (\text{Personal}, \text{Gift})$, thể hiện mục đích mua hàng có thể là cá nhân (personal) hoặc tặng quà (gift). Tại mức kế tiếp của cây phân cấp, ngữ cảnh Personal lại được biểu diễn theo ngữ cảnh chi tiết hơn: mua phục vụ công việc hay mục đích khác. Tương tự, với ngữ cảnh Gift, tại mức kế tiếp sẽ được biểu diễn chi tiết hơn bởi các ngữ cảnh mua quà cho bạn hoặc đồng nghiệp và mua quà cho gia đình hoặc khác. Với cách định nghĩa như trên, thuộc tính ngữ cảnh $K_2 = \{\text{PersonalWork}, \text{PersonalOther}, \text{GiftPartner/Friend}, \text{GiftParent/Other}\}$.

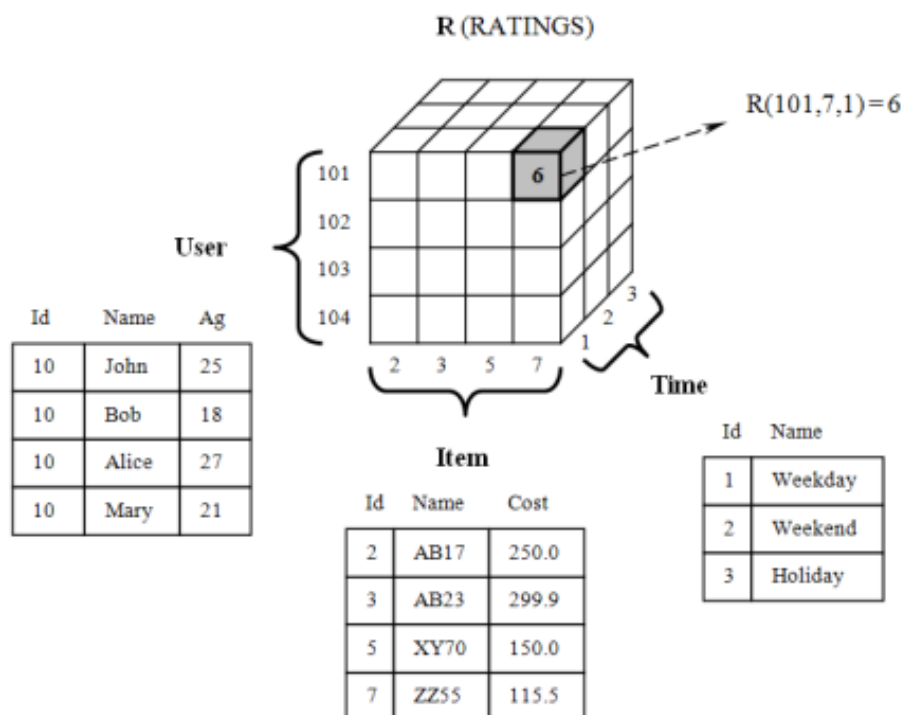
2.1.2 Cấu trúc dữ liệu đa chiều

Nếu xem ngữ cảnh như một chiều dữ liệu trong hệ gợi ý bên cạnh các chiều User, và Item, khi đó có thể biểu diễn dữ liệu của hệ gợi ý dựa trên ngữ cảnh dưới cấu trúc dữ liệu đa chiều OLAP (Online Analytical Processing, được sử dụng rộng rãi trong các hệ thống kho dữ liệu Datawarehouse). Cụ thể, gọi D_1, D_2, \dots, D_n là các chiều dữ liệu, khi đó hai chiều dữ liệu đầu tiên sẽ là User, Item, các chiều dữ liệu còn lại sẽ là ngữ cảnh. Di sẽ là tập con của tập tích Đề các (Cartesian) bao gồm các thuộc tính A_{ij} , ($j=1, \dots, k_i$), $D_i \subseteq A_{i1} \times A_{i2} \times \dots \times A_{ik_i}$. Ví dụ tại hình 8, với các dữ liệu có 3 chiều, User x Item x Time, chiều User sẽ được xác định bởi $User \subseteq \text{UName} \times \text{Address} \times \text{Income} \times \text{Age}$. Tương tự, chiều Item sẽ được xác định bởi $Item \subseteq \text{IName} \times \text{Type} \times \text{Price}$. Cuối cùng, chiều Time sẽ được xác định bởi $Time \subseteq \text{Year} \times \text{Month} \times \text{Day}$.

Khi đó gọi S là không gian gợi ý, S được xác định bởi tích đề các của các thuộc tính D_1, D_2, \dots, D_n , đồng thời hàm xếp hạng R cũng sẽ được xác định như sau:

$$R: D_1 \times D_2 \times \dots \times D_n \rightarrow Rating$$

Như trong hình 8, với User có Id là 101, Item với id là 7, thì đánh giá xếp hạng rating sẽ là 6 trong suốt tuần, $R(101,7,1) = 6$.



Hình 7: Cấu trúc OLAP 3 chiều User x Item x Time trong hệ gợi ý

2.2 Cơ chế tích hợp ngữ cảnh vào hệ gợi ý

Việc ứng dụng thông tin ngữ cảnh trong các hệ thống gợi ý có thể tìm thấy trong các nghiên cứu của Herlocker và Konstan [19], trong đó các tác giả cho rằng việc lồng ghép các tri thức về hành vi người dùng vào trong thuật toán gợi ý trong các ứng dụng chuyên biệt sẽ dẫn đến các kết quả gợi ý có chất lượng tốt hơn. Ví dụ, nếu chúng ta muốn mua tặng một đĩa trẻ nào đó một vài quyển sách, thì chúng ta muốn biết được các quyển sách trước đây đĩa trẻ đã từng đọc hoặc thích đọc và cung cấp các thông tin này cho hệ gợi ý để tính toán và đưa ra các gợi ý mới về các quyển sách phù hợp. Hướng tiếp cận này đã được áp dụng trong các hệ gợi ý truyền thống không gian 2 chiều User x Item, trong đó thông tin về hành vi của một người dùng cụ thể đối với các sản phẩm cụ thể được xem như là dữ liệu mẫu dùng để huấn luyện hệ thống. Hay nói cách khác, trong các hệ gợi ý truyền thống không đề cập đến việc áp dụng thông tin ngữ cảnh trong quá trình tính toán để đưa ra các gợi ý cho người dùng. Tuy nhiên, cách tiếp cận trên là minh họa rõ nét cho việc có thể tích hợp các thông tin ngữ cảnh liên quan vào trong các hệ gợi ý lọc cộng tác chuẩn hóa.

Có nhiều cách tiếp cận khác nhau trong việc tích hợp thông tin ngữ cảnh vào trong các hệ gợi ý, tuy nhiên chúng ta có thể phân nhóm các cách tiếp cận này theo hai nhóm chính, cụ thể:

Gợi ý dựa trên tìm kiếm và truy vấn hướng ngữ cảnh (context-driven querying and search):

Hướng tiếp cận này được ứng dụng rộng rãi trong các hệ gợi ý di động và du lịch [20,21,22]. Các hệ thống theo cách tiếp cận này sử dụng thông tin ngữ cảnh thu thập được (từ người dùng, từ thiết bị di động, ...) để thực hiện truy vấn và tìm kiếm các tập sản phẩm và cung cấp cho người dùng sản phẩm phù hợp nhất. Ví dụ, trong một hệ gợi ý về du lịch, dựa trên thông tin sở thích của người dùng (thu thập theo thông tin hồ sơ ban đầu của người dùng), thông tin vị trí hiện tại của người dùng (thu thập dựa trên tọa độ từ mạng di động hoặc tọa độ GPS của thiết bị di động) mà hệ thống có thể tìm kiếm và lựa chọn tập địa điểm du lịch xung quanh vị trí hiện tại của người dùng để cung cấp cho người dùng địa điểm du lịch phù hợp nhất. Một số thông tin ngữ cảnh hay được sử dụng trong các hệ gợi ý này thường bao gồm: sở thích, giờ địa phương, thời tiết, vị trí hiện tại, ... Một trong những hệ thống điển hình theo hướng tiếp cận trên có thể kể đến là Cyberguide [23], được phát triển như một hệ gợi ý du lịch trên nhiều nền tảng di động khác nhau. Một số hệ thống khác có thể kể tên ra như GUIDE [24], INTRIGUE [25], COMPASS[26] và MyMap [27].

Gợi ý dựa trên suy luận và tối ưu tham chiếu ngữ cảnh (contextual preference elicitation and estimation):

Một hướng tiếp cận khác trong việc ứng dụng thông tin ngữ cảnh vào trong hệ gợi ý là gợi ý dựa trên suy luận và tối ưu tham chiếu ngữ cảnh. Đây cũng là xu hướng tiếp cận mới nhất trong việc nghiên cứu các hệ gợi ý dựa trên ngữ cảnh [28,29,30,31]. Khác với hướng tiếp cận gợi ý dựa trên truy vấn và tìm kiếm hướng ngữ cảnh, chủ yếu dựa trên các thông tin ngữ cảnh hiện tại của người dùng, các kỹ thuật thuộc hướng tiếp cận này sẽ cố gắng mô hình hóa và học các sở thích của người dùng bằng cách theo dõi tương tác của người dùng cần xem xét và các người dùng khác đối với hệ thống, hoặc bằng cách thu thập thông tin phản hồi của người dùng đối với các sản phẩm đã được hệ thống gợi ý trước đây. Để mô hình hóa sở thích thay đổi theo ngữ cảnh của người dùng, các kỹ thuật này thường hoặc chấp thuận các phương thức gợi ý dựa trên lọc cộng tác (collaborative filtering), gợi ý dựa trên nội dung (content-based filtering) hoặc gợi ý hỗn hợp (hybrid) như là các thiết lập của hệ gợi ý dựa trên ngữ cảnh, hoặc áp dụng các phương pháp phân tích dữ liệu thông minh khác nhau trong khai phá dữ liệu, hoặc học máy để đạt được mục tiêu (kỹ thuật phân loại Bayes, ...).

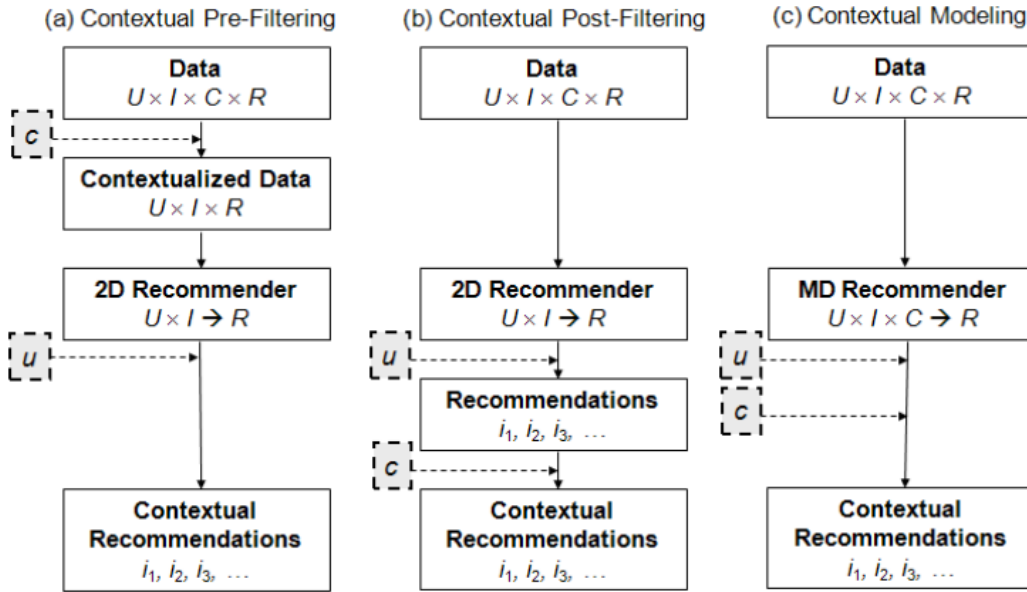
Cả hai hướng tiếp cận trên đều mở ra rất nhiều cơ hội và thách thức cho các nhà nghiên cứu để phân tích và xây dựng một hệ gợi ý dựa trên nhận biết ngữ cảnh hiệu quả. Tuy nhiên, hiện nay, hướng tiếp cận thứ hai đang trở thành xu thế chính để tiếp cận và xây dựng các hệ gợi ý theo phương pháp suy luận và tối ưu tham chiếu ngữ cảnh. Ngoài ra, vẫn có nhiều nhóm nghiên cứu lựa chọn giải pháp kết hợp cả hai hướng tiếp cận để xây dựng các hệ thống gợi ý. Các hệ thống điển hình trong nhóm tiếp cận kết hợp có thể kể đến như hệ thống UbiquiTO [21], được xây dựng để triển khai các ứng dụng hướng dẫn du lịch. Hệ thống này không chỉ dựa trên các thông tin ngữ cảnh cụ thể mà còn kết hợp các kỹ thuật khác như kỹ thuật gợi ý dựa trên tập luật (rule-based) và tập hợp mờ (fuzzy set) để đáp ứng các yêu cầu về nội dung của ứng dụng dựa trên sự thay đổi sở thích và sự quan tâm của người dùng.

Như đã đề cập trong phần 1.4.2, các hệ gợi ý truyền thống được xây dựng dựa trên tri thức về sở thích rời rạc của người dùng, ví dụ như sở thích người dùng đối với một vài (thông thường là có giới hạn) tập sản phẩm, khi đó đầu vào của các hệ gợi ý truyền thống thường là các bảng ghi có dạng $\langle \text{user}, \text{item}, \text{rating} \rangle$. Khác với các hệ gợi ý truyền thống, hệ gợi ý dựa trên nhận biết ngữ cảnh được xây dựng dựa trên tri thức về sở thích tùy biến theo ngữ cảnh rời rạc của người dùng, khi đó dữ liệu đầu vào của nó sẽ là các bảng ghi theo dạng $\langle \text{user}, \text{item}, \text{context}, \text{rating} \rangle$, trong đó mỗi bảng ghi không chỉ chứa thông tin về mức độ thích của người dùng đối với sản phẩm, mà còn chứa thông tin về ngữ cảnh cụ thể khi người dùng sử dụng sản phẩm. Khi đưa các thông tin ngữ cảnh vào hệ gợi ý như là các chiều dữ liệu đầu vào thì cũng làm phát sinh các vấn đề sau cần được giải quyết, cụ thể:

- Nên tích hợp ánh xạ dữ liệu ngữ cảnh vào hệ gợi ý như thế nào khi thực hiện mô hình hóa sở thích người dùng?
- Có thể sử dụng lại các thành tựu của hệ gợi ý truyền thống (không có thông tin ngữ cảnh) để xây dựng các hệ gợi ý dựa trên nhận biết ngữ cảnh hay không?

Để giải quyết các bài toán nêu trên, chúng ta bắt đầu xem xét với tập dữ liệu đầu vào có dạng $U \times I \times C \times R$, với C là chiều dữ liệu ngữ cảnh thêm vào và kết thúc là tập sản phẩm gợi ý dựa trên ngữ cảnh i_1, i_2, i_3, \dots cho từng người dùng u . Khi đó, các tiến trình gợi ý dựa trên suy luận và tối ưu tham chiếu ngữ cảnh người dùng có thể triển khai dựa trên 1 trong 3 cơ chế sau (hình 9):

- **Lọc trước theo ngữ cảnh** (Contextual Pre-Filtering, hay còn gọi là ngữ cảnh hóa dữ liệu đầu vào): theo cơ chế này, thông tin ngữ cảnh sẽ điều khiển việc lựa chọn dữ liệu hoặc xây dựng dữ liệu theo một ngữ cảnh cụ thể. Hay nói cách khác, thông tin về ngữ cảnh hiện tại c sẽ được sử dụng để lựa chọn hoặc xây dựng tập dữ liệu liên quan (ví dụ như bảng xếp hạng sản phẩm). Sau đó, các xếp hạng này sẽ được dự đoán bằng cách sử dụng các hệ gợi ý truyền thống trên các tập dữ liệu đã được lựa chọn trước đó.
- **Lọc sau theo ngữ cảnh** (Contextual Post-Filtering, hay còn gọi là ngữ cảnh hóa dữ liệu đầu ra): theo cơ chế này, thông tin ngữ cảnh sẽ được bỏ qua tại bước khởi tạo, và các xếp hạng sẽ được dự đoán dựa trên các hệ gợi ý truyền thống đối với toàn bộ dữ liệu. Sau đó, tập kết quả gợi ý sẽ được tinh chỉnh (ngữ cảnh hóa) cho từng người dùng phù hợp với thông tin ngữ cảnh.
- **Mô hình hóa hướng ngữ cảnh** (Contextual Modeling, hay còn gọi là mô hình hóa theo ngữ cảnh hàm gợi ý): theo cơ chế này, thông tin ngữ cảnh sẽ được sử dụng trực tiếp trong kỹ thuật mô hình hóa với vai trò như là một thành phần để tối ưu hàm xếp hạng.



Hình 8: Các cơ chế tích hợp thông tin ngữ cảnh vào hệ gợi ý

Sau đây, bài luận sẽ trình bày chi tiết về 3 cơ chế tích hợp trên.

2.2.1 Lọc trước theo ngữ cảnh (Contextual Pre-Filtering)

Như trình bày trong hình 9.a, hướng tiếp cận lọc trước theo ngữ cảnh sử dụng thông tin ngữ cảnh để chọn lọc dữ liệu 2D (User x Item) phù hợp nhất để tạo ra các gợi ý cho người dùng. Một trong những lợi ích quan trọng nhất của hướng tiếp cận này là việc có thể sử dụng lại các phương pháp gợi ý truyền thống 2 chiều đã được đề cập trong phần giới thiệu về hệ gợi ý truyền thống. Theo cách tiếp cận này, ngữ cảnh c sẽ được sử dụng như một truy vấn để lọc ra các dữ liệu xếp hạng liên quan. Một ví dụ sử dụng ngữ cảnh để lọc dữ liệu trong một hệ gợi ý về phim như sau: nếu 1 người dùng muốn xem một bộ phim vào thứ 7 thì chỉ những bộ phim được chiếu vào thứ 7 đã được xếp hạng mới được dùng để gợi ý cho người dùng. Trong ví dụ trên, bộ truy vấn lọc dữ liệu được xây dựng dựa trên ngữ cảnh cụ thể xác định về thời gian (thứ 7), hay nói cách khác bộ lọc trước dữ liệu trong trường hợp này được gọi là lọc trước dữ liệu chính xác.

Với cơ chế lọc trước dữ liệu theo ngữ cảnh, tác giả Adomavicius [28] đã đề xuất một hướng tiếp cận theo phương thức rút gọn dữ liệu (reduction-based) nhằm đưa một bài toán gợi ý trong không gian dữ liệu nhiều chiều về bài toán hệ gợi ý truyền thống 2 chiều. Như vậy, với bất kỳ phương pháp lọc trước theo ngữ cảnh nào, một lợi thế quan trọng của hướng tiếp cận rút gọn dữ liệu đó là tất cả các nghiên cứu trước đây về hệ gợi ý truyền thống 2 chiều đều có thể áp dụng trực tiếp trong các hệ gợi ý nhiều chiều sau khi bước rút gọn dữ liệu được thực hiện. Cụ thể, cho $\mathbf{R}^D_{\text{User} \times \text{Item}}: \mathbf{U} \times \mathbf{I} \rightarrow \text{Rating}$ là hàm xếp hạng 2 chiều bảy kỳ, D là tập dữ liệu xếp hạng mẫu bao gồm các bảng ghi $\langle \text{user}, \text{item}, \text{rating} \rangle$, thì hoàn toàn có thể dự đoán được cho bất kỳ xếp hạng nào của người dùng đối với sản phẩm, ví dụ: $\mathbf{R}^D_{\text{User} \times \text{Item}}(\text{John}, \text{StarWar})$. Khi đó, hàm dự đoán xếp hạng 3 chiều với ngữ cảnh là thời gian (Time) có thể được định nghĩa bởi công thức: $\mathbf{R}^D_{\text{User} \times \text{Item} \times \text{Time}}: \mathbf{U} \times \mathbf{I} \times \mathbf{T} \rightarrow \text{Rating}$, trong đó D là tập dữ liệu

chứa các bảng ghi (user, item, time, rating) chứa thông tin xếp hạng của người dùng. Hàm dự đoán xếp hạng 3 chiều trên có thể được mở rộng thông qua hàm dự đoán xếp hạng 2 chiều theo rất nhiều cách theo công thức sau:

$$\begin{aligned} \forall (u, i, t) \in U \times I \times T, R_{User \times Item \times Time}^D(u, i, t) \\ = R_{User \times Item}^{D_{[Time=t]}(User, Item, Rating)}(u, i) \end{aligned}$$

- [Time = t] ký hiệu cho bộ lọc trước theo ngữ cảnh thời gian, và
- $D_{[Time=t]}(user, item, rating)$ ký hiệu cho tập dữ liệu xếp hạng nhận được từ tập dữ liệu D ban đầu bằng cách chỉ chọn những bảng ghi thỏa mãn chiều dữ liệu Time có giá trị bằng t và chỉ giữ lại 2 chiều dữ liệu User, Item và giá trị xếp hạng rating tương ứng.

Từ phân tích trên ta thấy, nếu xem tập dữ liệu 3 chiều D là 1 quan hệ thì $D_{[Time=t]}(user, item, rating)$ đơn giản là 1 quan hệ khác nhận được từ D bằng cách thực hiện các phép toán quan hệ sau: phép chọn, phép trừ và phép chiếu.

Tuy nhiên, theo phương pháp bộ lọc chính xác nhiều lúc sẽ làm cho kết quả thu được sau khi rút gọn dữ liệu là rất hạn chế. Ví dụ, xem xét một ngữ cảnh cho việc một người dùng xem phim với bạn gái trong rạp chiếu phim vào thứ 7, khi đó $c = (Girlfriend, Theater, Saturday)$. Sử dụng ngữ cảnh chính xác này để thực hiện truy vấn lọc dữ liệu có thể gây ra nhiều vấn đề hạn chế dữ liệu do các nguyên nhân khác nhau. Đầu tiên, chắc chắn tồn tại một tỷ lệ không nhỏ các ngữ cảnh chính xác không đủ ý nghĩa. Ví dụ, sở thích xem phim với bạn gái vào thứ 7 có thể tương tự với xem phim với bạn gái vào chủ nhật, tuy nhiên sẽ không giống vào thứ 4. Do đó, nên xem xét dùng ngữ cảnh tổng quát hơn như Weekend (ngày cuối tuần) thay vì chỉ dùng chính xác ngữ cảnh thứ 7. Thứ hai, các ngữ cảnh chính xác có thể sẽ không đủ dữ liệu cho việc dự đoán xếp hạng, hay được biết đến với vấn đề dữ liệu thưa (sparsity) trong các hệ gợi ý truyền thống. Tóm lại, hệ gợi ý sẽ không có đủ các dữ liệu quá khứ về sở thích xem phim với bạn gái trong rạp chiếu phim vào thứ 7 đối với người dùng cần xem xét.

Tổng quát hóa ngữ cảnh (context generalization)

Hàm tổng quát lọc trước theo ngữ cảnh cho phép tổng quát hóa truy vấn lọc dữ liệu thu được dựa trên một ngữ cảnh cụ thể [3]. Cho $c' = (c_1', \dots, c_k')$ làm một ngữ cảnh tổng quát hóa của $c = (c_1, \dots, c_k)$ nếu và chỉ nếu $c_i \rightarrow c_i'$ với $\forall i = 1, \dots, k$ trong trật tự ngữ cảnh tương ứng. Khi đó, c' (thay vì c) có thể được dùng như 1 truy vấn để thu được dữ liệu xếp hạng theo ngữ cảnh.

Phát triển theo ý tưởng tổng quát hóa ngữ cảnh, Adomavicius [3] đề xuất không sử dụng ngữ cảnh chính xác [Time = t], mà thay vào đó sử dụng hàm lọc trước tổng quát hóa [Time $\in S_t$], với S_t là ký hiệu cho tập ngữ cảnh cha của t. Khi đó, S_t được gọi là miền ngữ cảnh hóa. Ví dụ, trong một hệ gợi ý về phim, nếu ta muốn dự đoán xếp hạng của người dùng John Doe với bộ phim Gladiator khi xem vào thứ 2, để tính $R_{User \times Item \times Time}^D(\text{JohnDoe}, \text{Gladiator}, \text{Monday})$, ta nên dùng ngữ cảnh xếp hạng theo Weekday hơn là chỉ dùng ngữ cảnh xếp hạng theo Monday khi thực hiện lọc dữ liệu dự đoán. Hay nói một cách khác, với mỗi (u,i,t) mà $t \in \text{Weekday}$, ta có thể dự đoán được xếp hạng R:

$$R_{User \times User \times Time}^D(u, i, t) = R_{User \times Item}^{D[Time \in Weekday]}(User, Item, ARR(G(Rating)))(u, i)$$

Tổng quát hơn, để dự đoán xếp hạng $R(u, i, t)$, ta có thể dùng miền ngữ cảnh xác định S_t như sau:

$$R_{User \times User \times Time}^D(u, i, t) = R_{User \times Item}^{D[Time \in S_t]}(User, Item, ARR(G(Rating)))(u, i)$$

Trong công thức trên, $ARR(G(Rating))$ là hàm đại số dùng để xác định giá trị xếp hạng chung cho cả tập trên miền S_t . Hàm đại số này có thể là hàm tính giá trị xếp hạng trung bình khi thực hiện rút gọn chiều dữ liệu của không gian miền gợi ý.

Hàm rút gọn 3 chiều ở trên có thể được mở rộng thành phương thức rút gọn tổng quát n chiều về không gian gợi ý m chiều ($m < n$). Thông thường các hệ thống hay áp dụng $m=2$ vì các hệ gợi ý truyền thống chỉ được xây dựng với 2 chiều dữ liệu $User \times Item$.

Trong thực tế sẽ có rất nhiều khả năng xảy ra khi tổng quát hóa ngữ cảnh, phụ thuộc vào phân loại ngữ cảnh và mức độ chi tiết của ngữ cảnh xem xét. Quay trở lại với ví dụ xem phim, ta sẽ có các loại ngữ cảnh nhận được do suy luận từ phân cấp ngữ cảnh sau:

- Company: GirlFriend -> Friends -> NotAlone -> AnyCompany;
- Place: Theater -> AnyPlace;
- Time: Saturday -> Weekend -> Anytime;

Khi đó ngữ cảnh tổng quát c' suy luận từ ngữ cảnh $c=(GirlFriend, Theater, Saturday)$ có thể sẽ có các giá trị sau:

- $c' = (GirlFriend, Anyplace, Saturday)$
- $c' = (Friends, Theater, Anytime)$
- $c' = (Notalone, Theater, Weekend)$

Từ ví dụ trên, có thể nhận thấy việc lựa chọn hàm lọc trước tổng quát phù hợp có vai trò rất quan trọng trong hướng tiếp cận rút gọn dữ liệu. Thông thường, có hai tùy chọn sau hay được áp dụng. Thứ nhất, chọn manual theo ý kiến chuyên gia, ví dụ luôn luôn tổng quát hóa các ngày trong tuần thành Weekday hoặc Weekend. Một tùy chọn khác là chọn theo hướng tự động, sau đó đánh giá hiệu quả dự đoán theo kinh nghiệm. Theo phương thức này, hệ thống sẽ thực hiện dự đoán theo các truy vấn lọc trước tổng quát khác nhau và chọn ra hàm lọc có hiệu quả nhất.

2.2.2 Lọc sau theo ngữ cảnh (Contextual Pos-Filtering)

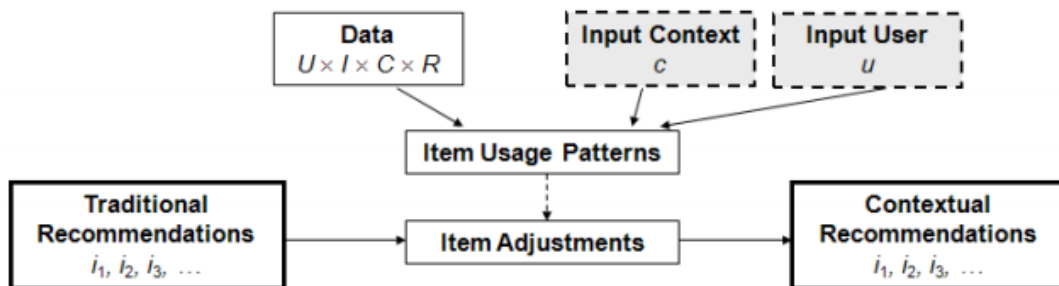
Như biểu diễn trong hình 9.b, hướng tiếp cận lọc sau bỏ qua yếu tố ngữ cảnh trong dữ liệu đầu vào khi thực hiện tính toán các gợi ý cho người dùng. Sau khi có kết quả gợi ý thu được với các kỹ thuật gợi ý 2 chiều truyền thống, hàm lọc sau theo ngữ cảnh sẽ được áp dụng trên danh sách các gợi ý thu được dựa trên các thông tin ngữ cảnh. Danh sách gợi ý tinh chỉnh có thể thu được theo các bước:

- Lọc bỏ các gợi ý không liên quan với ngữ cảnh áp dụng

- Hiệu chỉnh thứ tự xếp hạng các gợi ý trong danh sách

Ví dụ, trong các hệ gợi ý phim, nếu 1 người dùng muốn xem 1 bộ phim vào cuối tuần và trong cuối tuần người dùng ấy chỉ muốn xem thể loại phim hài. Sau khi thực hiện dự đoán danh sách phim gợi ý theo các kỹ thuật gợi ý 2 chiều (bỏ qua yếu tố thông tin ngữ cảnh), hệ thống có thể lọc loại bỏ các bộ phim không liên quan đến thể loại phim hài từ danh sách phim gợi ý.

Tổng quát, ý tưởng của hướng tiếp cận lọc sau theo ngữ cảnh là phân tích dữ liệu sở thích theo ngữ cảnh của một người dùng cụ thể trong một ngữ cảnh cụ thể để tìm ra các mẫu gợi ý điển hình, sau đó dùng các mẫu này để hiệu chỉnh danh sách gợi ý cho người dùng. Kết quả là các sản phẩm phù hợp với ngữ cảnh sẽ được gợi ý cho người dùng phù hợp.



Hình 9: Bước hiệu chỉnh danh sách gợi ý trong lọc sau theo ngữ cảnh

Tương tự như các kỹ thuật gợi ý khác, hướng tiếp cận lọc sau theo ngữ cảnh cũng được phân loại theo 2 nhóm: nhóm dựa trên kinh nghiệm (heuristic) và nhóm dựa trên mô hình (model-based).

- Nhóm dựa trên kinh nghiệm: tập trung vào tìm kiếm các thuộc tính đặc trưng của sản phẩm phù hợp với người dùng được cho theo ngữ cảnh được cho (ví dụ, diễn viên yêu thích của người dùng muốn được xem trong ngữ cảnh được cho), sau đó sử dụng các thuộc tính này để hiệu chỉnh danh sách gợi ý, bao gồm:
 - Lọc bỏ các sản phẩm gợi ý không có đủ số lượng các thuộc tính cần có (ví dụ, để được gợi ý, bộ phim phải có ít nhất 2 diễn viên yêu thích của người dùng trong ngữ cảnh xem xét)
 - Xếp hạng lại danh sách gợi ý dựa trên số lượng thuộc tính đặc trưng mà các sản phẩm có (ví dụ, bộ phim có càng nhiều diễn viên yêu thích thì càng được xếp hạng cao hơn các bộ phim khác)
- Nhóm dựa trên mô hình: xây dựng các mô hình dự đoán có thể dùng để tính ra được xác suất mà người dùng có thể lựa chọn một thể loại chắc chắn của sản phẩm theo ngữ cảnh xem xét, chẳng hạn như xác suất liên quan, sau đó sử dụng xác suất này để hiệu chỉnh danh sách gợi ý, bao gồm:
 - Lọc bỏ các sản phẩm gợi ý mà xác suất liên quan của nó nhỏ hơn một giá trị ngưỡng (threshold) tối thiểu do hệ thống định nghĩa sẵn
 - Xếp hạng danh sách gợi ý dựa trên giá trị trọng số xếp hạng dự đoán theo xác suất liên quan.

Tương tự như hướng tiếp cận lọc trước, hướng tiếp cận lọc sau cũng có một lợi thế quan trọng đó là có thể dùng bất kỳ kỹ thuật gợi ý 2 chiều truyền thống nào khi tính toán gợi ý cho người dùng. Hơn thế, do tính chất tương tự giữa 2 hướng tiếp cận mà việc kết hợp cả 2 trong bài toán xây dựng hệ gợi ý hướng cảnh sẽ là một hướng nghiên cứu thú vị trong tương lai.

2.2.3 Mô hình hóa hướng ngữ cảnh (Contextual Modeling)

Như biểu diễn trong hình 9.c, hướng tiếp cận mô hình hóa sử dụng thông tin ngữ cảnh trực tiếp trong hàm gợi ý như một dự đoán tường minh về xếp hạng của người dùng đối với sản phẩm. Trong 3 hướng tiếp cận, hướng tiếp cận mô hình hóa mới thực sự sử dụng các hàm gợi ý nhiều chiều (multi-dimension), biểu diễn bởi các mô hình dự đoán (được xây dựng dựa trên các kỹ thuật cây quyết định, đại số hồi quy, mô hình thống kê, ...) hoặc các tính toán dựa trên kinh nghiệm kết hợp với các thông tin ngữ cảnh thêm vào bên cạnh thông tin User và Item, theo công thức xếp hạng $Rating = R(User, Item, Context)$.

Dựa trên kinh nghiệm: ý tưởng của hướng tiếp cận này là mở rộng mô hình 2 chiều truyền thống có kết hợp thông tin ngữ cảnh như các chiều dữ liệu thêm vào bên cạnh User và Item. Một trong những mô hình 2 chiều hay được áp dụng để mở rộng là mô hình gợi ý người láng giềng gần nhất (nearest neighborhood) [15,32]. Khi đó, hệ thống sẽ thực hiện xây dựng hàm tính đơn vị khoảng cách n-chiều thay vì chỉ sử dụng các độ đo tương quan truyền thống là user-user, item-item. Để hiểu được cơ chế thực hiện của hướng tiếp cận này, ta hãy xem xét ví dụ về không gian gợi ý User x Item x Time. Áp dụng kỹ thuật người láng giềng gần nhất (dựa trên trọng số của tổng các xếp hạng dự đoán liên quan), hàm dự đoán xếp hạng $r_{u,i,t}$ sẽ được tính theo công thức sau [33]:

$$r_{u,i,t} = k \sum_{(u',i',t') \neq (u,i,t)} W((u,i,t), (u',i',t')) \times r_{u',i',t'}$$

với $W((u,i,t), (u',i',t'))$ là trọng số của xếp hạng $r_{u',i',t'}$ mang trong dự đoán của $r_{u,i,t}$ và k là yếu tố chuẩn hóa. Trọng số W là giá trị nghịch đảo với khoảng cách tương quan giữa 2 điểm (u',i',t') và (u,i,t) trong không gian nhiều chiều. Hay nói cách khác, khoảng cách tương quan giữa 2 điểm (u',i',t') và (u,i,t) càng nhỏ thì trọng số W càng lớn, $W((u,i,t), (u',i',t')) = 1/\text{dist}[(u',i',t'), (u,i,t)]$. Việc lựa chọn hàm tính khoảng cách tương quan dist phụ thuộc vào từng ứng dụng cụ thể.

Một trong những phương pháp đơn giản nhất để xây dựng hàm khoảng cách trong không gian nhiều chiều đó là sử dụng hướng tiếp cận gần giống với phương pháp rút gọn dữ liệu đã trình bày tại phần 2.2.1, cụ thể hàm tính khoảng cách cho các điểm trong không gian nhiều chiều có cùng ngữ cảnh sẽ là:

$$\text{dist}[(u,i,t), (u',i',t')] = \begin{cases} \text{dist}[(u,i), (u',i')], & \text{if } t = t' \\ +\infty & \text{otherwise} \end{cases}$$

Hàm khoảng cách này làm cho $r_{u,i,t}$ chỉ còn phụ thuộc vào các giá trị xếp hạng trong cùng miền ngữ cảnh có giá trị thời gian $\text{time}=t$. Do đó, trường hợp này sẽ được rút gọn về bài toán tối ưu hệ gợi ý 2 chiều truyền thống trên miền đánh giá có cùng giá trị ngữ cảnh t của điểm (u,i,t) . Hơn thế, nếu ta định nghĩa

chi tiết hơn hàm khoảng cách $\mathbf{dist}[(\mathbf{u}', \mathbf{i}', t'), (\mathbf{u}, \mathbf{i}, t)]$ sao cho nó chỉ còn phụ thuộc khoảng cách giữa 2 user khi $i = i'$, khi đó ta sẽ thu được 1 phương thức tương tự với hướng tiếp cận lọc trước đã được trình bày tại phần 2.2.1.

Hướng tiếp cận này dễ dàng mở rộng cho không gian n chiều bằng cách thiết lập khoảng cách giữa 2 điểm xếp hạng thành khoảng cách $\mathbf{dist}[(\mathbf{u}, \mathbf{i}), (\mathbf{u}', \mathbf{i}')] |$ khi và chỉ khi các ngữ cảnh của 2 điểm xếp hạng này giống nhau.

Một số phương thức khác hay được dùng để xây dựng hàm khoảng cách này như:

- Đo khoảng cách theo trọng số Mahattan:

$$\mathit{dist}[(u, i, t), (u', i', t')] = w_1 d_1(u, u') + w_2 d_2(i, i') + w_3 d_3(t, t')$$

- Đo khoảng cách theo trọng số Euclidean:

$$\begin{aligned} \mathit{dist}[(u, i, t), (u', i', t')] \\ = \sqrt{w_1 d_1^2(u, u') + w_2 d_2^2(i, i') + w_3 d_3^2(t, t')} \end{aligned}$$

với:

d_1, d_2, d_3 : khoảng cách theo user, item và time

w_1, w_2, w_3 : trọng số gán cho từng chiều dữ liệu tương ứng (tương quan với mức độ quan trọng của từng chiều dữ liệu)

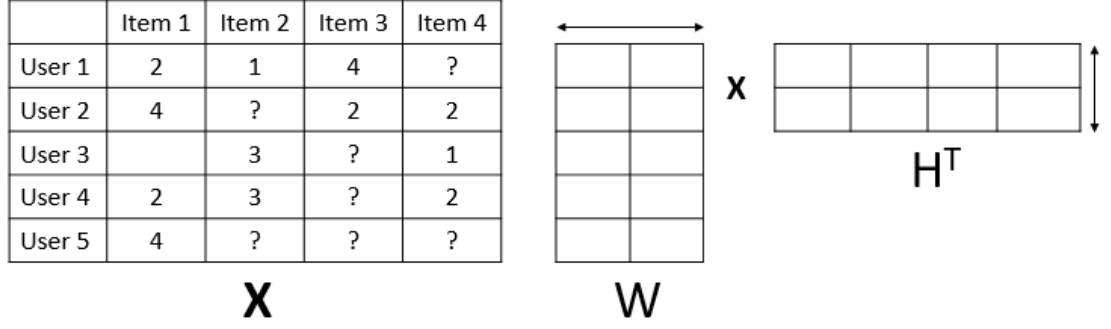
Tổng kết lại, khoảng cách dist có thể được tính theo nhiều cách khác nhau và trong nhiều hệ thống nó được tính giữa các giá trị xếp hạng của cùng user hoặc cùng item. Hướng tiếp cận dựa trên kinh nghiệm tiếp tục là một hướng nghiên cứu rất thú vị nhằm tìm ra các cách khác nhau để định nghĩa khoảng cách dist cũng như so sánh giữa các phương thức dựa trên hiệu năng dự đoán của các phương thức.

Dựa trên mô hình: ý tưởng của hướng tiếp cận này là mở rộng các hệ gợi ý 2 chiều truyền thống được xây dựng dựa trên việc mô hình hóa hàm dự đoán xếp hạng. Một trong những kỹ thuật hay được áp dụng để mở rộng đó là kỹ thuật phân rã ma trận (Matrix Factorization - MF). Khi đó ta gọi phương pháp này là kỹ thuật gợi ý hướng ngữ cảnh dựa trên phân rã ma trận, viết tắt là CAMF (Context-Aware Matrix Factorization - CAMF). Phần tiếp theo trong luận văn sẽ trình bày về kỹ thuật này.

2.3 Mô hình hóa ngữ cảnh dựa trên phân rã ma trận (Context Aware Matrix Factorization - CAMF)

2.3.1 Kỹ thuật phân rã ma trận (Matrix Factorization - MF)

Kỹ thuật MF hiện vẫn là state-of-the-art trong hệ gợi ý truyền thống [34]. Kỹ thuật phân rã ma trận là việc chia một ma trận lớn X thành hai ma trận có kích thước nhỏ hơn W và H , sao cho ta có thể xây dựng lại X từ hai ma trận nhỏ hơn này càng chính xác càng tốt, nghĩa là $X \sim WH^T$, như minh họa trong hình 11:



Hình 10: Minh họa kỹ thuật phân rã ma trận

Mục tiêu chính của kỹ thuật này là phân rã ma trận X thành 2 ma trận nhỏ hơn W và H sao cho ta có thể xây dựng lại X từ 2 ma trận con này:

$$X \sim WH^T \quad (1)$$

Với W và H là 2 ma trận con: $W = R^{|U| \times K}$ $H = R^{|I| \times K}$ (2)

W là một ma trận mà mỗi dòng u là một vector bao gồm K nhân tố tiềm ẩn mô tả người dùng u ; và H là một ma trận mà mỗi dòng i là một vectore bao gồm K nhân tố tiềm ẩn mô tả cho item I ; K : là số nhân tố tiềm ẩn (latent factors) $K \leq |U|$; $K \leq |I|$. Gọi w_{uk} và h_{ik} là các phần tử tương ứng của hai ma trận W và H , khi đó công thức dự đoán xếp hạng của người u trên sản phẩm I là:

$$\hat{r}_{ui} = \sum_{k=1}^K w_{uk} h_{ik} = w_u h_i^T \quad (3)$$

Chi tiết về giải thuật học, chúng ta thực hiện huấn luyện sao cho tìm được hai ma trận W và H được tối ưu theo một điều kiện nào đó (chẳng hạn như RMSE – chương tổng quan). Ví dụ, hàm mục tiêu cần tối ưu là:

$$O^{MF} = \sum_{(u,i,r) \in D^{train}} e_{ui}^2 \quad (4)$$

Với:

$$e_{ui}^2 = (r_{ui} - \hat{r}_{ui})^2 = \left(r_{ui} - \sum_{k=1}^K w_{uk} h_{ik} \right)^2 \quad (5)$$

Một trong những kỹ thuật có thể dùng để tối ưu hóa hàm mục tiêu là dùng SGD (Stochastic Gradient Descent) [34] – kỹ thuật giảm gradient ngẫu nhiên. Để tối ưu hóa hàm mục tiêu, trước tiên ta khởi tạo các giá trị ngẫu nhiên cho

W và H, sau đó từng bước cập nhật giá trị của chúng cho đến khi hàm mục tiêu hội tụ về giá trị nhỏ nhất. Để làm được điều đó, ta cần phải xác định là nên tăng hay nên giảm các giá trị của W và H qua mỗi lần cập nhật, do vậy cần phải tìm đạo hàm từng phần của chúng:

$$\frac{\partial}{\partial w_{uk}} O^{MF} = -2(r_{ui} - \hat{r}_{ui})h_{ik} \quad (6)$$

$$\frac{\partial}{\partial h_{ik}} O^{MF} = -2(r_{ui} - \hat{r}_{ui})w_{uk} \quad (7)$$

Tiếp theo, ta cần giảm thiểu tối đa độ lỗi này và cập nhật lại giá trị cho w_{uk} và h_{ik} được lặp đi lặp lại bằng kỹ thuật SGD:

$$w'_{uk} = w_{uk} - \beta \frac{\partial}{\partial w_{uk}} e_{ui}^2 = w_{uk} + 2\beta e_{ui} h_{ik} = w_{uk} + 2\beta(r_{ui} - \hat{r}_{ui})h_{ik} \quad (8)$$

$$h'_{ik} = h_{ik} - \beta \frac{\partial}{\partial h_{ik}} e_{ui}^2 = h_{ik} + 2\beta e_{ui} w_{uk} = h_{ik} + 2\beta(r_{ui} - \hat{r}_{ui})w_{uk} \quad (9)$$

β : tốc độ học (learning rate). Sau quá trình tối ưu, ta có được giá trị của W và H.

“Học vẹt” (overfitting) xảy ra khi mô hình dự đoán cho kết quả tốt trên dữ liệu huấn luyện (D_{train}) nhưng cho kết quả kém trên dữ liệu thử nghiệm. Để giảm bớt việc học vẹt, người ta cho thêm vào hàm mục tiêu một đại lượng gọi là hàm chính tắc (Regularization). Hàm mục tiêu cần tối ưu bây giờ là:

$$O^{MF} = \sum_{(u,i,r) \in D^{train}} \left(r_{ui} - \sum_{k=1}^K w_{uk} h_{ik} \right)^2 + \lambda (\|W\|_F^2 + \|H\|_F^2) \quad (10)$$

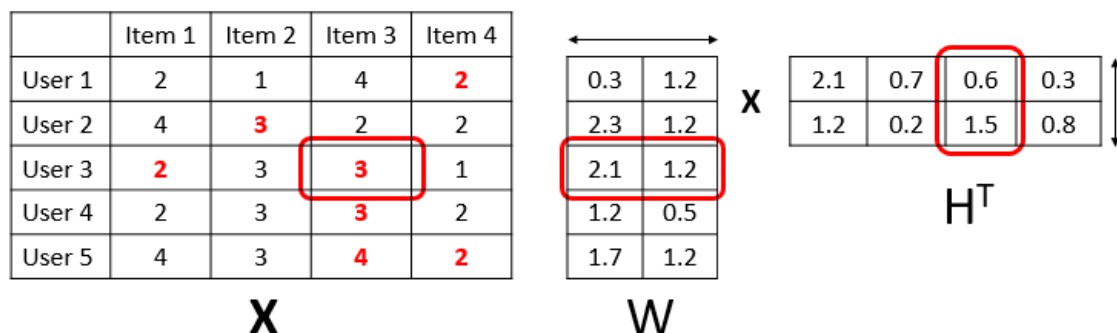
Và giá trị W và H sau tối ưu với hàm mục tiêu mới ta được:

$$w'_{uk} = w_{uk} + \beta(2e_{ui}h_{ik} - \lambda w_{uk}) \quad (11)$$

$$h'_{ik} = h_{ik} + \beta(2e_{ui}w_{uk} - \lambda h_{ik}) \quad (12)$$

Sau khi có được giá trị gradient, chúng ta cập nhật lại giá trị của w_{uk} và h_{ik} với tốc độ học là β . Lặp lại quá trình cập nhật giá trị của W và H cho đến khi đạt độ lỗi chấp nhận ϵ hoặc lặp lại đến số lần qui định trước.

Quá trình dự đoán: sau quá trình huấn luyện ta được 2 ma trận W và H đã tối ưu thì quá trình dự đoán (công thức 3) được thực hiện như minh họa trong Hình 12.



Hình 12. Minh họa cách tính dự đoán xếp hạng của người dùng 3 cho sản phẩm 3

Giá trị dự đoán của người dùng 3 cho sản phẩm 3: $P_{33} = 2.1 * 0.6 + 1.2 * 1.5 = 3.06$

Giải thuật phân rã ma trận MF sử dụng phương pháp SGD để huấn luyện mô hình MF. Đầu tiên các ma trận W và H được khởi tạo ngẫu nhiên bằng phân bố chuẩn $\mathcal{N}(0, \sigma^2)$, sau đó thực hiện giải thuật MF để tối ưu hàm mục tiêu nhằm thu được các giá trị ma trận W, H . Cụ thể như sau:

Procedure MATRIXFACTORIZATION($D^{\text{train}}, K, \beta, \lambda$, stopping condition)

//Cho $W[[U]][K]$ and $H[[I]][K]$ là các yếu tố tiềm ẩn của các người dùng và sản phẩm

$W \leftarrow \mathcal{N}(0, \sigma^2)$ //khởi tạo W bằng hàm phân bố chuẩn

$H \leftarrow \mathcal{N}(0, \sigma^2)$ //khởi tạo H bằng hàm phân bố chuẩn

While (stopping condition is NOT met) **do**

Draw randomly (u, i, r) from D^{train}

$\hat{r} \leftarrow 0$

For $k \leftarrow 1, \dots, K$ **do**

$\hat{r} \leftarrow \hat{r} + W[u][k].H[i][k]$

End for

$e_{ui} = r - \hat{r}$

For $k \leftarrow 1, \dots, K$ **do**

$W[u][k] \leftarrow W[u][k] + \beta.(e_{ui}.H[i][k] - \lambda.W[u][k])$

$H[i][k] \leftarrow H[i][k] + \beta.(e_{ui}.W[u][k] - \lambda.H[i][k])$

End for

End While

Return{ W, H }

End Procedure

2.3.2 Kỹ thuật phân rã ma trận thiên vị (Biased Matrix Factorization – BMF)

Dựa vào các khái niệm cơ sở của kỹ thuật MF và thêm giá trị lệch (bias) vào MF, ta sẽ thu được kỹ thuật BMF [34]. Khi đó hàm dự đoán xếp hạng của người dùng u cho sản phẩm i được biểu diễn bởi công thức sau:

$$\hat{r}_{ui} = \mu + b_u + b_i + \sum_{k=1}^K w_{uk} h_{ik} \quad (13)$$

Với giá trị μ là giá trị trung bình toàn cục, là giá trị xếp hạng trung bình của tất cả người dùng trên tất cả sản phẩm với tập dữ liệu huấn luyện:

$$\mu = \frac{\sum_{(\mathbf{u}, \mathbf{i}, \mathbf{r}) \in \mathbf{D}^{train}} \mathbf{r}}{|\mathbf{D}^{train}|} \quad (14)$$

Giá trị b_u là độ lệch người dùng (là giá trị lệch trung bình của các người dùng so với giá trị trung bình toàn cục):

$$b_u = \frac{\sum_{(\mathbf{u}', \mathbf{i}, \mathbf{r}) \in \mathbf{D}^{train} | \mathbf{u}' = \mathbf{u}} (\mathbf{r} - \mu)}{|\{(\mathbf{u}', \mathbf{i}, \mathbf{r}) \in \mathbf{D}^{train} | \mathbf{u}' = \mathbf{u}\}|} \quad (15)$$

Giá trị b_i là độ lệch của sản phẩm (là giá trị lệch trung bình của các sản phẩm so với giá trị trung bình toàn cục):

$$b_i = \frac{\sum_{(\mathbf{u}, \mathbf{i}', \mathbf{r}) \in \mathbf{D}^{train} | \mathbf{i}' = \mathbf{i}} (\mathbf{r} - \mu)}{|\{(\mathbf{u}, \mathbf{i}', \mathbf{r}) \in \mathbf{D}^{train} | \mathbf{i}' = \mathbf{i}\}|} \quad (16)$$

Quá trình dự đoán: sau quá trình huấn luyện ta được các giá trị lệch (biases) và 2 ma trận W và H đã tối ưu, quá trình dự đoán được thực hiện tương tự MF, dùng công thức (13). Giải thuật phân rã ma trận thiên vị BMF được mô tả như sau:

Procedure BiasMatrixFactorization(\mathbf{D}^{train} , K , β , λ , stopping condition)

Let $u \in U$ be a user, $i \in I$ be a item, $r \in R$ be a rate

Let $W[|U|][|K|]$, $H[|I|][|K|]$, $b_u[|U|]$, and $b_i[|I|]$

$$\mu \leftarrow \frac{\sum_{r \in \mathbf{D}^{train}} r}{|\mathbf{D}^{train}|}$$

For each user u **do**

$$b_u[u] \leftarrow \frac{\sum_u (r_{ui} - \mu)}{|\mathbf{D}_u^{train}|}$$

```

End for
For each item  $i$  do
     $b_i[I] \leftarrow \frac{\sum_i(r_{ui}-\mu)}{|D_i^{train}|}$ 
End for
 $W \leftarrow \mathcal{N}(0, \sigma^2)$ 
 $H \leftarrow \mathcal{N}(0, \sigma^2)$ 
While (stopping condition is NOT met) do
    Draw randomly  $(u,i,r)$  from  $D^{train}$ 
     $\hat{r}_{ui} \leftarrow \mu + b_u[u] + b_i[i] + \sum_k^K(W[u][k] * H[i][k]);$ 
     $e_{ui} = r_{ui} - \hat{r}_{ui};$ 
     $\mu \leftarrow \mu + \beta * e_{ui};$ 
     $b_u \leftarrow b_u + \beta * (e_{ui} - \lambda * b_u[u]);$ 
     $b_i \leftarrow b_i + \beta * (e_{ui} - \lambda * b_i[i]);$ 
    For  $k \leftarrow 1, \dots, K$  do
         $W[u][k] \leftarrow W[u][k] + \beta * (2 * e_{ui} * H[i][k] - \lambda * W[u][k]);$ 
         $H[i][k] \leftarrow H[i][k] + \beta * (2 * e_{ui} * W[u][k] - \lambda * H[i][k]);$ 
    End for
End While
Return  $\{W, H, b_u, b_i, \mu\}$ 
End Procedure

```

2.3.3 Phân rã ma trận hướng ngữ cảnh (Context Aware Matrix Factorization - CAMF)

CAMF là một kỹ thuật dự đoán xếp hạng dựa trên mô hình hóa ngữ cảnh mở rộng từ kỹ thuật gợi ý truyền thống phân rã ma trận. Kỹ thuật này thuộc nhóm các kỹ thuật mô hình hóa ngữ cảnh dựa trên độ lệch ngữ cảnh (deviation-based contextual modeling). Ta gọi CRD (Contextual Rating Deviation) là độ lệch xếp hạng theo ngữ cảnh, độ lệch này chỉ ra giá trị lệch giữa các xếp hạng của người dùng với các sản phẩm trong các ngữ cảnh khác nhau. Để hiểu rõ về CRD ta xét ví dụ tại bảng sau:

Context	D1: Time	D2: Location
c1	Weekend	Home
c2	Weekday	Cinema
CRD(Di)	0.5	-0.1

CRD(D1) = 0.5 \rightarrow xếp hạng của người dùng trong ngày thường (weekday) thường cao hơn so với trong ngày cuối tuần (weekend) là 0.5

$CRD(D2) = -0.1$ -> xếp hạng của người dùng tại rạp chiếu phim (cinema) thường thấp hơn so với tại nhà (home) là 0.1

Giả thiết rằng giá trị xếp hạng của người dùng đối với sản phẩm trong ngữ cảnh $c1$ (weekend, home) $Rating(U,I,c1) = 4$, khi đó giá trị dự đoán xếp hạng của người dùng đối với sản phẩm trong ngữ cảnh $c2$ (weekday, cinema) sẽ là:

$$\mathbf{Predict\ Rating(U,I,c2) = Rating(U,I,c1) + CRDs = 4 + 0.5 - 0.1 = 4.4}$$

Tiếp theo, ta xét trường hợp \emptyset là một tình huống đặc biệt mà không quan tâm đến các yếu tố ngữ cảnh, cụ thể:

Context	D1: Time	D2: Location
\emptyset	UnKnown	UnKnown
c2	Weekday	Cinema
CRD(Di)	0.5	-0.1

Ta cũng giả thiết rằng $Rating(U,I,\emptyset) = Rating(U,I) = 4$, khi đó giá trị dự đoán xếp hạng $Predict\ Rating(U,I,c2) = 4 + 0.5 - 0.1 = 4.4$.

Theo hướng tiếp cận này thì hàm dự đoán xếp hạng tổng quát của các kỹ thuật dựa trên độ lệch xếp hạng theo ngữ cảnh $Rating(U,I,C)$ sẽ được tính theo công thức sau:

$$Rating(U, I, C) = Rating(U, I) + \sum_{i=1}^N CRD(i)$$

(17)

Nếu xem xét góc độ mô hình xếp hạng phụ thuộc vào tương quan giữa người dùng và các yếu tố ngữ cảnh (user-personalized model) ta sẽ có công thức:

$$Rating(U, I, C) = Rating(U, I) + \sum_{i=1}^N CRD(i, U)$$

(18)

Nếu xem xét góc độ mô hình xếp hạng phụ thuộc vào tương quan giữa sản phẩm và các yếu tố ngữ cảnh (item-personalized) ta sẽ có công thức:

$$Rating(U, I, C) = Rating(U, I) + \sum_{i=1}^N CRD(i, I)$$

(19)

Trong đó, $Rating(U,I)$ là hàm gợi ý truyền thống và CRD chính là độ lệch xếp hạng phụ thuộc các yếu tố ngữ cảnh.

CAMF chính là sự mở rộng hệ gợi ý truyền thống MF, khi đó các yếu tố thông tin ngữ cảnh được tích hợp trực tiếp trong các bước thực hiện tính toán để đưa ra gợi ý cho người dùng. Kỹ thuật CAMF được giới thiệu lần đầu tiên trong [35] với mô hình mà các dự đoán xếp hạng phụ thuộc vào ngữ cảnh thời gian. Tuy nhiên, ngoài yếu tố thời gian, CAMF còn có thể được thực hiện với

số lượng lớn các yếu tố ngữ cảnh khác, theo các tương tác khác nhau giữa ngữ cảnh và người dùng, ngữ cảnh và sản phẩm có thể tác động đến giá trị xếp hạng.

Quay trở lại kỹ thuật phân rã ma trận thiên vị BMF đã trình bày trong 2.3.2, ta có công thức dự đoán xếp hạng của BMF như sau:

$$\hat{r}_{ui} = \mu + b_u + b_i + \vec{v}_u \vec{q}_i \quad (20)$$

Kỹ thuật CAMF thực hiện thay thế các giá trị lệch (bias) trong công thức của BMF bằng các tương tác giữa ngữ cảnh với người dùng hoặc giữa ngữ cảnh và sản phẩm [36]. Khi đó hàm dự đoán xếp hạng CAMF được tính theo công thức sau:

$$\hat{r}_{uic_{k,1}c_{k,2}\dots c_{k,L}} = \mu + b_u + \sum_{j=1}^L B_{ijc_{k,j}} + \vec{v}_u \vec{q}_i \quad (21)$$

Giả thiết L là tổng các chiều ngữ cảnh, $c_k = \{c_{k,1}c_{k,2}\dots c_{k,L}\}$ là một tình huống ngữ cảnh, khi đó $c_{k,j}$ sẽ biểu diễn cho điều kiện ngữ cảnh trong chiều ngữ cảnh thứ j. Do đó, $B_{ijc_{k,j}}$ sẽ biểu diễn cho độ lệch xếp hạng ngữ cảnh kết hợp giữa sản phẩm i và điều kiện ngữ cảnh trong chiều ngữ cảnh thứ j.

So sánh giữa 2 công thức (20) và (21), ta thấy rằng kỹ thuật CAMF đơn giản thay thế độ lệch bi bởi tổng các độ lệch xếp hạng ngữ cảnh $\sum_{j=1}^L B_{ijc_{k,j}}$ với giả thiết rằng độ lệch xếp hạng ngữ cảnh phụ thuộc vào các sản phẩm. Cách tiếp cận này được gọi là CAMF_CI. Ngoài ra, độ lệch xếp hạng ngữ cảnh cũng có thể được xét chỉ phụ thuộc vào người dùng, thành phần được thay thế là b_u , khi đó cách tiếp cận này được gọi là CAMF_CU và có công thức tính tương ứng là:

$$\hat{r}_{uic_{k,1}c_{k,2}\dots c_{k,L}} = \mu + \sum_{j=1}^L B_{ujc_{k,j}} + b_i + \vec{v}_u \vec{q}_i \quad (22)$$

Ngoài 2 kỹ thuật CAMF_CI, CAMF_CU đã trình bày ở trên, còn có thêm kỹ thuật CAMF_C, trong đó giả thiết rằng độ lệch xếp hạng ngữ cảnh hoàn toàn độc lập với cả người dùng và sản phẩm.

Áp dụng phương pháp tối ưu RMSE, SGD tương tự như đã thực hiện trong kỹ thuật MF và BMF, sau quá trình huấn luyện dựa trên tập dữ liệu huấn luyện, ta sẽ thu được các giá trị tham số vector người dùng, vector sản phẩm, độ lệch người dùng/độ lệch sản phẩm và độ lệch xếp hạng ngữ cảnh tương ứng.

Cho R là tập dữ liệu huấn luyện, $R = \{(u, i, c_1, \dots, c_k) | r_{uic_1\dots c_k} \text{ đã biết}\}$, $r_{uic_1\dots c_k}$ cho biết giá trị xếp hạng người dùng u đối với sản phẩm i trong ngữ cảnh c_1, \dots, c_k , trong đó $c_j = 0, 1, \dots, z_j$. Chọn kỹ thuật CAMF_CI để triển khai tối ưu như sau:

- Hàm dự đoán xếp hạng:

$$\hat{r}_{uic_1\dots c_k} = \vec{v}_u \vec{q}_i + \bar{r} + b_u + \sum_{j=1}^k B_{ijc_j} \quad (23)$$

Trong đó \vec{v}_u , \vec{q}_i là các vector người dùng; \bar{r} là giá trị xếp hạng trung bình của sản phẩm i trong tập dữ liệu R ; b_u là giá trị lệch cơ sở của người dùng u , tính theo công thức (15); B_{ijc_j} là các tham số mô hình hóa sự tương tác giữa ngữ cảnh và sản phẩm, hay còn gọi là độ lệch xếp hạng theo ngữ cảnh. Cụ thể:

$$\bar{r} = \frac{\sum r(u, i', c_1, \dots, c_k) \in R | i' = i}{|\{(u, i', c_1, \dots, c_k) \in R | i' = i\}|} \quad (24)$$

$$B_{ijc_j} = \frac{\sum [r(u, i', c') - \bar{r}] | i' = i, c' = c_j}{|\{r(u, i', c') \in R | i' = i, c' = c_j\}|} \quad (25)$$

- Tối ưu hàm mục tiêu:

$$\min_{v^*, q^*, b^*, B^*} \sum_{r \in R} \left[\begin{aligned} & \left(r_{uic_1\dots c_k} - \vec{v}_u \vec{q}_i - \bar{r} - b_u - \sum_{j=1}^k B_{ijc_j} \right)^2 \\ & + \lambda \left(b_u^2 + \|\vec{v}_u\|^2 + \|\vec{q}_i\|^2 + \sum_{j=1}^k \sum_{c_j=1}^{z_j} B_{ijc_j}^2 \right) \end{aligned} \right] \quad (26)$$

- Cập nhật giá trị tham số trong quá trình tối ưu:

$$\begin{aligned} b_u &\leftarrow b_u + \beta(err - \lambda b_u) \\ b_{ic_j} &\leftarrow b_{ic_j} + \beta(err - \lambda b_{ic_j}), \forall c_j \neq 0, j = 1, \dots, k \\ v_u &\leftarrow v_u + \beta(err \cdot q_i - \lambda v_u) \\ q_i &\leftarrow q_i + \beta(err \cdot v_u - \lambda q_i) \end{aligned} \quad (27)$$

Qua các công thức (26), (27), ta thấy mô hình CAMF có thể thực hiện huấn luyện trong thời gian là một hàm tuyến tính với một số lượng lớn dữ liệu kỳ vọng cần xử lý cũng như các yếu tố ngữ cảnh tác động lên hệ thống. Đây là một thuận lợi lớn khi phải xử lý các bài toán với số lượng lớn dữ liệu và rất

nhiều yếu tố thông tin ngữ cảnh. Một nghiên cứu tại [36] đã thực hiện thử nghiệm với một tập dữ liệu bao gồm 14 yếu tố ngữ cảnh, với tổng cộng 52 điều kiện ngữ cảnh khác nhau, và cho kết quả áp dụng CAMF đáp ứng được mong đợi.

2.4 Kết luận chương

Trong chương 2, luận văn đã làm rõ giải pháp tích hợp ngữ cảnh vào trong quá trình tính toán dự đoán xếp hạng và gợi ý cho người dùng. Từ 2 cách thức quản lý dữ liệu ngữ cảnh (phân cấp, OLAP) trong hệ gợi ý nhiều chiều, đến 3 phương thức tiếp cận chính khi tích hợp ngữ cảnh bao gồm tích hợp lọc trước, lọc sau và mô hình hóa cũng đã được trình bày trong chương này. Quan trọng nhất trong chương này, luận văn đã trình bày chi tiết về các kỹ thuật phân rã ma trận, phân rã ma trận thiên vị để từ đó áp dụng mở rộng để xây dựng kỹ thuật mô hình hóa phân rã ma trận hướng ngữ cảnh, một kỹ thuật được phân nhóm trong các hệ gợi ý hướng ngữ cảnh dựa trên độ lệch xếp hạng ngữ cảnh hiệu quả hiện nay. Đây cũng chính là nền tảng để luận văn xây dựng hệ thống gợi ý địa điểm du lịch theo hướng cá nhân hóa dựa trên ngữ cảnh người dùng để thử nghiệm tại MobiFone. Nội dung này sẽ được luận văn trình bày trong Chương 3 tiếp theo.

Chương 3. XÂY DỰNG ỨNG DỤNG DU LỊCH THEO HƯỚNG CÁ NHÂN HÓA DỰA TRÊN NGỮ CẢNH NGƯỜI DÙNG

3.1 Đặt bài toán

Dựa trên kỹ thuật phân rã ma trận hướng ngữ cảnh đã trình bày trong Chương 2, sử dụng Framework mã nguồn mở CARSKIT [37], trong phần này luận văn sẽ trình bày về giải pháp xây dựng một hệ thống gợi ý địa điểm du lịch cho người dùng, sử dụng các ngữ cảnh của người dùng để cá nhân hóa ứng dụng cũng như các gợi ý về địa điểm du lịch phù hợp với người dùng.

Hệ thống quản lý, lưu trữ các địa điểm du lịch hấp dẫn ở Việt Nam cho phép người dùng có thể tìm kiếm các địa điểm du lịch. Hệ thống hiển thị các địa điểm du lịch theo đúng nhu cầu người dùng quan tâm. Hệ thống có các chức năng cho người dùng đánh giá, like và comment cho mỗi địa điểm du lịch. Đặc biệt hệ thống tự động thu thập các dữ liệu của người dùng để đưa ra các gợi ý địa điểm du lịch hợp lý với ngữ cảnh của người dùng.

Người dùng muốn thực hiện các chức năng thì phải đăng ký thành viên và đăng nhập vào hệ thống. Sau khi người dùng đăng nhập vào hệ thống, hệ thống sẽ thu thập các thông tin về ngữ cảnh của người dùng. Về vị trí, thời tiết, thời gian để đưa ra các gợi ý phù hợp với người dùng ngoài ra hệ thống còn dựa vào các địa điểm mà người dùng đánh giá, quan tâm để đưa ra các gợi ý đến người dùng.

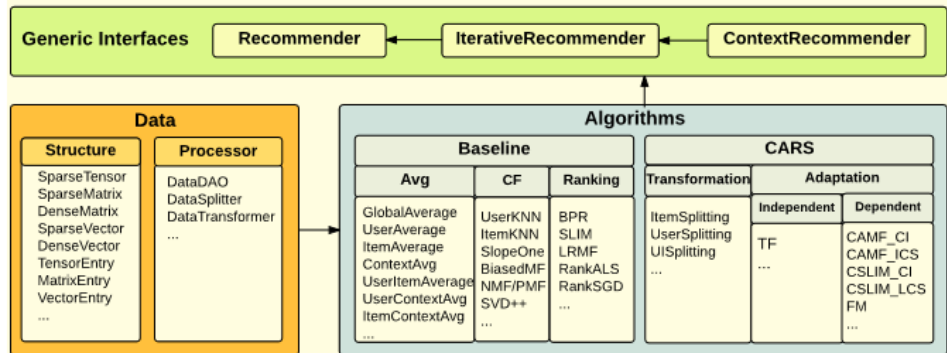
3.2 Giải pháp đề xuất

3.2.1 Mô hình kiến trúc ứng dụng

Hệ thống được xây dựng gồm các thành phần chính sau đây:

- Ứng dụng client:
 - o Phần client được xây dựng trên nền tảng Android sử dụng ngôn ngữ Java và Android SDK. Client giao tiếp với hệ thống server bằng qua giao thức http restful web service. Phương thức xác thực dựa trên các phiên làm việc (session) được sinh ra từ hệ thống xác thực (authenticate)
 - o Thu thập thông tin ngữ cảnh của người dùng: vị trí, thời gian, thời tiết và các ngữ cảnh tĩnh do người dùng nhập vào ứng dụng.
 - o Gửi thông tin ngữ cảnh về server, yêu cầu phía server dự đoán xếp hạng và nhận danh sách gợi ý từ server.
- Phần server:
 - o Phần server quản lý địa điểm du lịch và các tác động của người dùng cũng như quản lý user và xác thực hệ thống. Server được xây dựng trên ngôn ngữ Java, phần cơ sở dữ liệu xây dựng trên hệ quản trị cơ sở dữ liệu mysql.
 - o Nhận biết và chuẩn hóa ngữ cảnh người dùng, xây dựng dữ liệu ngữ cảnh theo định dạng chuẩn trước khi gửi sang hệ gợi ý để tiến hành quá trình học và xây dựng hàm dự đoán xếp hạng

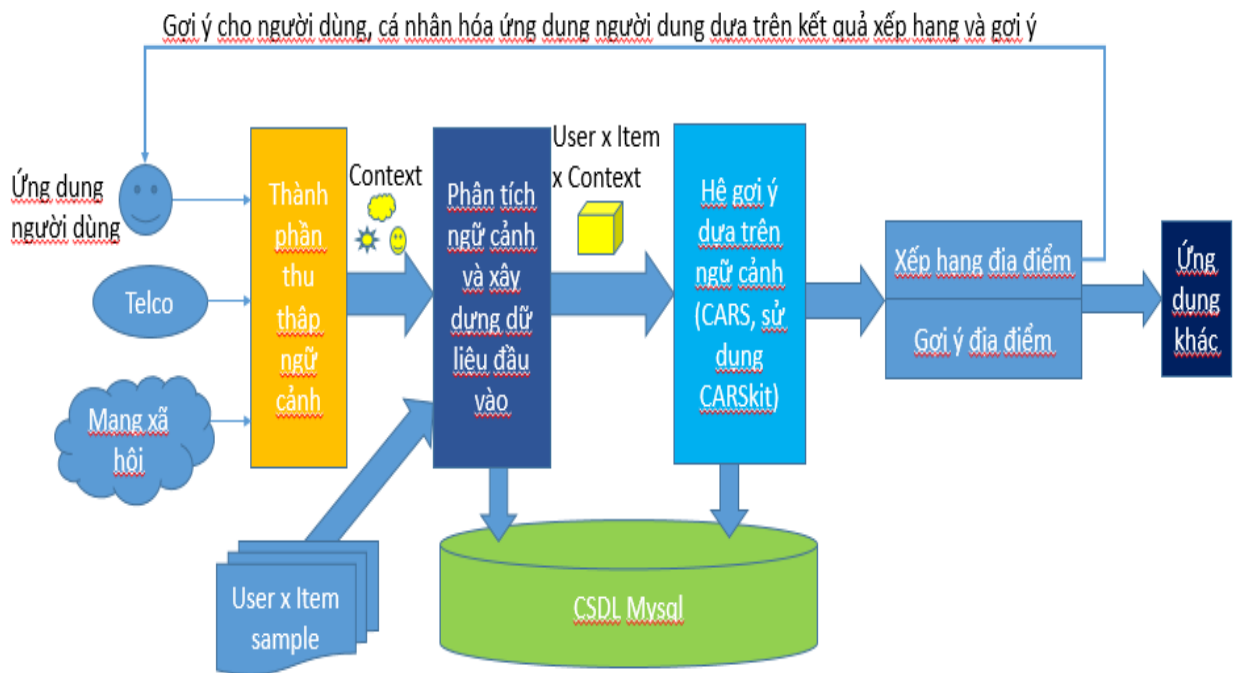
- Thành phần gợi ý: dựa trên framework mã nguồn mở CARSKit Framework (CARSKit - A java based context aware recommendation engine), sử dụng kỹ thuật CAMF-CU để học và xây dựng hàm dự đoán xếp hạng, có kiến trúc như sau:



Hình 11: Kiến trúc nền tảng mã nguồn mở CARSKIT

- Thành phần giao tiếp với client, gửi danh sách gợi ý cho client cũng như cho các ứng dụng bên thứ 3 (third-party).

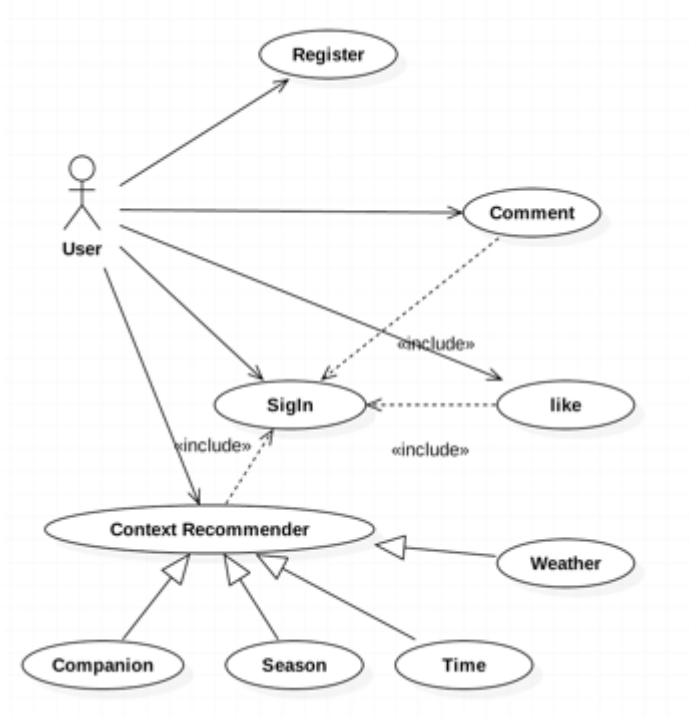
Mô hình thiết kế hệ thống:



Hình 12: Kiến trúc thiết kế hệ thống

3.2.2 Thiết kế ứng dụng

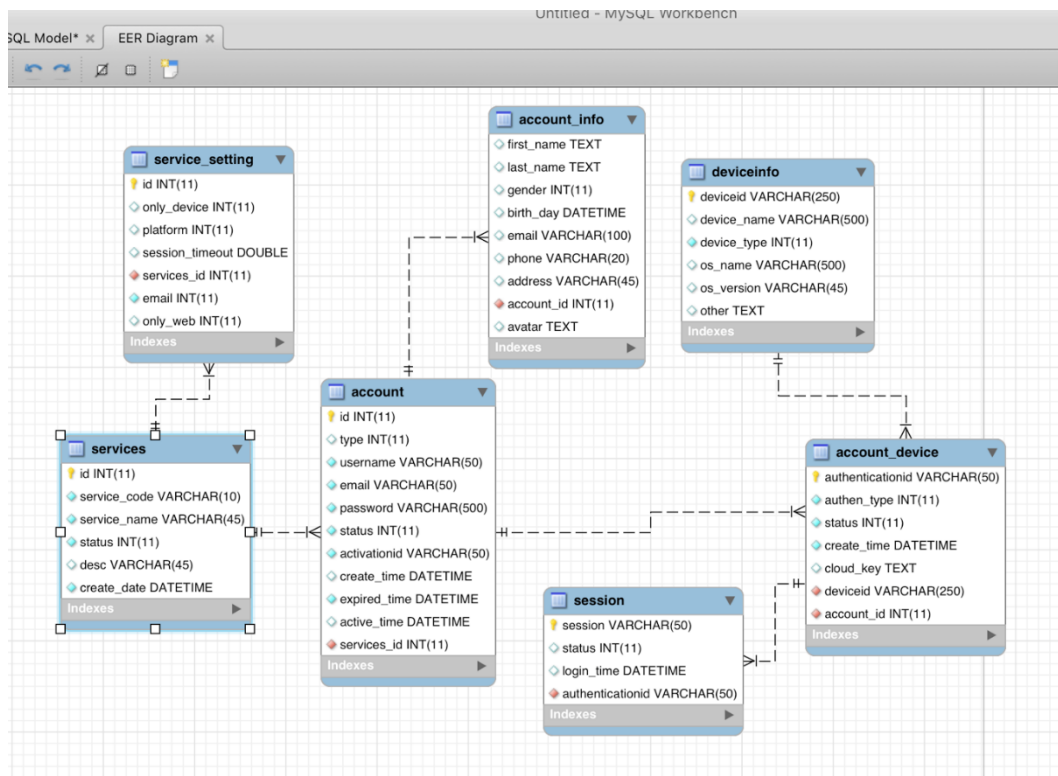
3.2.2.1 Sơ đồ ca người dùng



Hình 13: Thiết kế user case

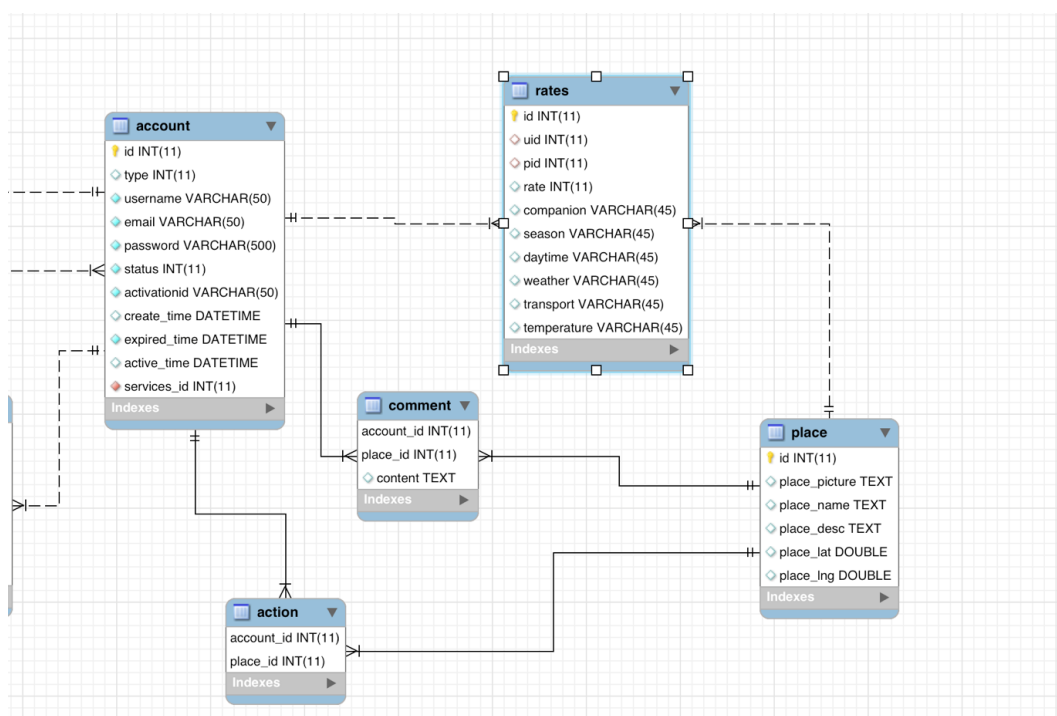
3.2.2.2 Thiết kế mô hình cơ sở dữ liệu

Mô hình thực thể quan hệ quản lý người dùng:



Hình 14: Mô hình thực thể quan hệ quản lý user

Mô hình thực thể quan hệ quản lý địa điểm du lịch và gợi ý người dùng:



Hình 15: Mô hình thực thể quan hệ quản lý địa điểm và gợi ý

3.2.3 Môi trường xây dựng và thử nghiệm

Phần server xử lý dữ liệu người dùng gồm 2 module: Xác thực và Recommender, có cấu hình như sau:

- Hardware:
 - o CPU: Dual-Core AMD Opteron(tm) Processor 1222 (3.0 Hz)
2 core 4 Thread
 - o RAM: 4G
- OS:
 - o Red Hat Enterprise Linux Server release 5.7
 - o 64 bit
- MYSQL
 - o Version: 5.1.73
- Java:
 - o Version "1.8.0_51"
- Tomcat
 - o Version 7.0.70

Client:

- Android SDK
- Min version: 4.0.0

3.3 Thử nghiệm ứng dụng và đánh giá kết quả tại Mobifone

Tập dữ liệu địa điểm du lịch đầu vào được xây dựng dựa trên thông tin địa điểm du lịch Việt Nam từ các nguồn trên internet, các dữ liệu ngữ cảnh cũng như giá trị xếp hạng ban đầu được hệ thống xây dựng tự động theo phương pháp ngẫu nhiên. Các dữ liệu này sau đó được chuẩn hóa và đưa vào lưu

trong CSDL với cấu trúc dữ liệu nhiều chiều là dữ liệu xếp hạng của người dùng theo ngữ cảnh:

UID / PID / RATE / COMPANION / SEASON / DAYTIME / WEATHER / TEMPERATURE

Trong đó

- UID: id của người dùng
- PID: id của địa điểm du lịch
- RATE: Điểm đánh giá của người dùng, có giá trị từ 0 - 5
- SEASON: Ngữ cảnh mùa, bao gồm:
 - o Mùa xuân
 - o Mùa hạ
 - o Mùa thu
 - o Mùa đông
- COMPANION: Ngữ cảnh người đi du lịch cùng, bao gồm:
 - o Một mình (alone)
 - o Bạn bè
 - o Gia đình
 - o Người yêu
- DAYTIME: Ngữ cảnh thời gian, là số ngày đi du lịch:
 - o Nửa ngày (đi trong ngày)
 - o Một ngày (đi qua đêm)
 - o Nhiều ngày
- WEATHER: Ngữ cảnh thời tiết, là thời tiết tại địa điểm du lịch:
 - o Trong xanh
 - o Có nắng
 - o Râm mát
 - o Có mưa
- TEMPERATURE: Ngữ cảnh nhiệt độ

Các dữ liệu này được lưu thành bảng dữ liệu nhiều chiều OLAP trong CSDL (hình 16, 17), tuy nhiên trước khi đưa vào hàm CAMF thì sẽ được chuyển sang dữ liệu dạng mảng theo định dạng của CARSKIT.

id	uid	pid	rate	companion	season	daytime	weather	transport	temperature
1	25	29	4	with lovely	autumn	half day	clear sky	[Null]	na
2	40	16	2	na	summer	one day	sunny	[Null]	burning
3	44	9	1	with family	na	na	sunny	[Null]	na
4	29	33	2	na	winter	more than one day	sunny	[Null]	warn
5	44	1	2	with lovely	na	more than one day	clear sky	[Null]	burning
6	39	41	1	with lovely	na	na	cloudy	[Null]	hot
7	29	26	3	with family	na	one day	rainy	[Null]	warn
8	8	40	1	alone	summer	one day	rainy	[Null]	warn
9	21	10	1	alone	winter	half day	rainy	[Null]	burning
10	35	26	4	alone	spring	one day	rainy	[Null]	cool
11	19	10	1	na	autumn	half day	rainy	[Null]	hot
12	13	4	3	with family	spring	more than one day	sunny	[Null]	warn
13	29	6	2	with family	summer	one day	rainy	[Null]	cool
14	14	36	3	na	autumn	one day	na	[Null]	freezing
15	34	33	4	na	summer	half day	cloudy	[Null]	hot
16	7	4	4	alone	autumn	more than one day	na	[Null]	na
17	35	36	2	alone	na	half day	clear sky	[Null]	na
18	33	38	4	na	summer	one day	cloudy	[Null]	burning
19	33	8	4	na	autumn	more than one day	clear sky	[Null]	na
20	28	37	4	with lovely	spring	half day	sunny	[Null]	hot
21	1	16	2	with friends	winter	na	clear sky	[Null]	hot

Hình 16: Dữ liệu rating lưu trong CSDL

id	place_picture	place_name	place_desc	place_lat	place_lng
1	https://upload.wikimedia.org/wikipedia/commons/thu	Cố đô Hoa Lư	Cố đô Hoa Lư là quần thể di tích quốc gia đặc biệt quan trọng của	20.270304	105.917309
2	https://upload.wikimedia.org/wikipedia/vi/thumb/2/2t	Khu di tích Pác Bó	Khu di tích Pác Bó là một khu di tích lịch sử cách mạng quốc gia -	22.673954	106.258662
3	https://upload.wikimedia.org/wikipedia/commons/thu	Dinh Độc Lập	Dinh Độc Lập (tên gọi trước đây là dinh Norodom, ngày nay còn g	10.777021	106.6932252
4	https://upload.wikimedia.org/wikipedia/commons/thu	Hoàng thành Thăng Long	Hoàng thành Thăng Long (chữ Hán: 昇龍皇城 / Thăng Long hoàng t	21.0364204	105.8382185
5	https://upload.wikimedia.org/wikipedia/commons/thu	ATK	ATK, viết tắt của an toàn khu, là khu vực mà Quân đội Nhân dân V	20.8423897	105.457061
6	https://upload.wikimedia.org/wikipedia/commons/thu	Khu di tích chiến t	Khu di tích chiến trường Điện Biên Phủ là một quần thể các di tích	21.3806714	103.0062053
7	https://upload.wikimedia.org/wikipedia/vi/thumb/a/a2	Khu di tích Côn Sơn	Khu di tích Côn Sơn - Kiếp Bạc là một trong 62 di tích quốc gia đ	21.130519	106.383883
8	https://upload.wikimedia.org/wikipedia/commons/thu	Khu di tích khởi ng	Khu di tích khởi nghĩa Yên Thế gồm nhiều di tích liên quan đến cư	21.481055	106.11442
9	https://upload.wikimedia.org/wikipedia/vi/thumb/b/bc	Khu di tích Phủ Chu	Khu di tích Chủ tịch Hồ Chí Minh tại Phủ Chủ tịch, gọi tắt là Khu di	21.039697	105.834572
10	https://upload.wikimedia.org/wikipedia/commons/thu	Văn Miếu - Quốc T	Văn Miếu - Quốc Tử Giám (chữ Hán: 文廟 - 國子監) là quần thể di t	21.028888	105.836294
11	https://upload.wikimedia.org/wikipedia/commons/thu	Vườn quốc gia Phư	Vườn quốc gia Phong Nha-Kẻ Bàng là một vườn quốc gia tại huyệ	17.612112	106.306096
12	https://upload.wikimedia.org/wikipedia/commons/thu	Vịnh Hạ Long	Vịnh Hạ Long (vịnh nơi rồng đáp xuống) là một vịnh nhỏ thuộc ph	20.945846	107.153026
13	https://upload.wikimedia.org/wikipedia/commons/thu	Phố cổ Hội An	Phố cổ Hội An là một đô thị cổ nằm ở hạ lưu sông Thu Bồn, thuộc	15.881142	108.335487
14	https://upload.wikimedia.org/wikipedia/commons/thu	Đền Hùng	Đền Hùng là tên gọi khái quát của Khu di tích lịch sử Đền Hùng - c	21.361898	105.32451
15	https://upload.wikimedia.org/wikipedia/commons/thu	Vườn quốc gia Ba	Vườn quốc gia Ba Bể là một vườn quốc gia, rừng đặc dụng, khu d	22.415953	105.606742
16	https://upload.wikimedia.org/wikipedia/commons/thu	Vườn quốc gia Bái	Vườn quốc gia Bái Tử Long là một khu bảo tồn sinh quyển cấp qu	21.081293	107.437134
17	https://upload.wikimedia.org/wikipedia/commons/thu	Vườn quốc gia Ho	Vườn quốc gia Hoàng Liên là một vườn quốc gia Việt Nam được th	22.352472	103.77024
18	https://upload.wikimedia.org/wikipedia/commons/thu	Vườn quốc gia Tan	Vườn quốc gia Tam Đảo là một vườn quốc gia của Việt Nam, nằm	21.484037	105.611095
19	https://upload.wikimedia.org/wikipedia/commons/thu	Vườn quốc gia Xu	Vườn quốc gia Xuân Sơn là một vườn quốc gia nằm trên địa bàn h	21.133446	104.985859
20	https://upload.wikimedia.org/wikipedia/vi/thumb/5/55	Vườn quốc gia Ba	Vườn quốc gia Ba Vì là một vườn quốc gia ở Việt Nam, được thành	21.096522	105.400879
21	https://upload.wikimedia.org/wikipedia/commons/thu	Vườn quốc gia Cát	Vườn quốc gia Cát Bà là khu rừng đặc dụng của Việt Nam, khu d	20.735397	107.043007

Hình 17: Dữ liệu địa điểm được lưu trong CSDL

Dữ liệu training bao gồm:

- Số lượng địa điểm: 42 địa điểm
- Số lượng user: 50
- Số lượng rating: 1000
- Số lượng yếu tố tiềm ẩn: $K = 10$
- Chỉ số learning rate alpha: $0 < \alpha \leq 1$
- Chỉ số chuẩn hóa beta: $0 < \beta \leq 1$
- Chỉ số Interval: 100

3.3.1 Kết quả ứng dụng

- Hình ảnh deploy server:

```

|-- nginx
|-- bin
|-- META-INF
|-- WEB-INF
|-- config
|-- log

|-- server
|-- apache-tomcat-7.0.70
|-- bin
|-- conf
|-- lib
|-- logs
|-- temp
|-- webapps
|-- work

|-- tom
|-- bin
|-- META-INF
|-- WEB-INF
|-- config
|-- log

|-- test

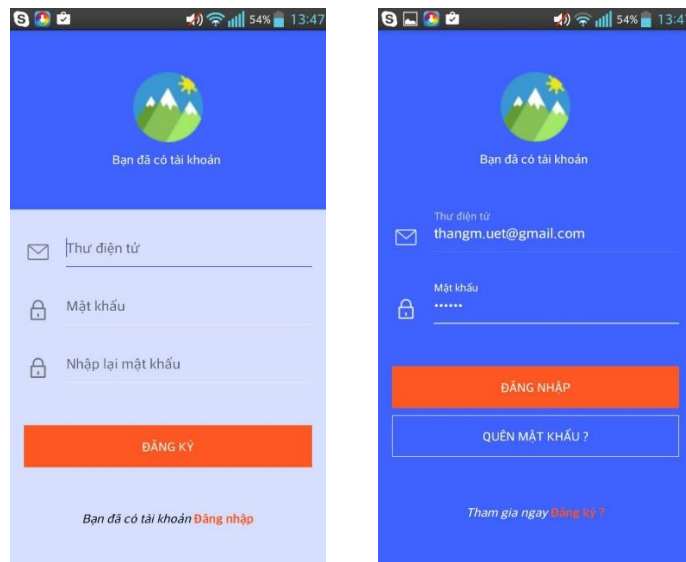
|-- travel
|-- bin
|-- META-INF
|-- WEB-INF
|-- config
|-- log
  
```

Hình 18: Deploy hệ thống trên môi trường Linux

Các module deploy đều có 3 thư mục chính gồm:

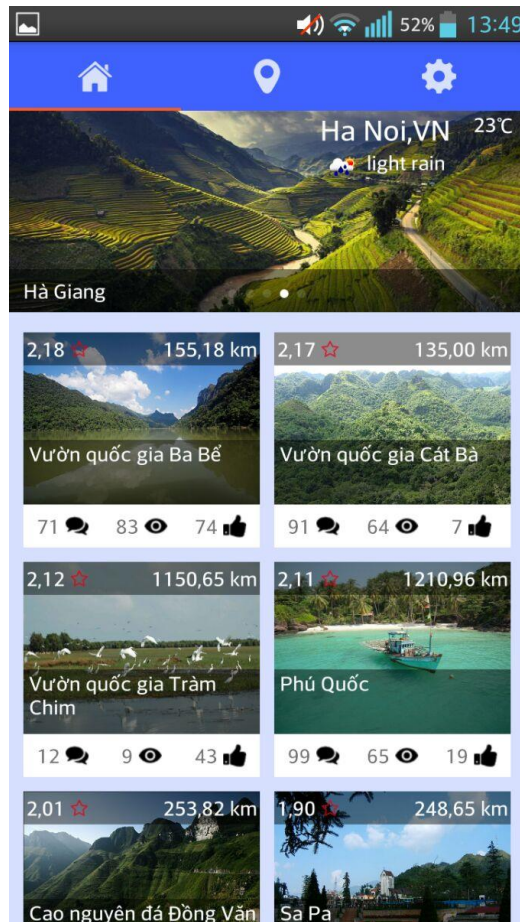
- BIN: Chứa các file biên dịch class

- CONFIG: Chứa các file config của hệ thống
 - LOG: Chứa file log trong quá trình module chạy runtime.
- Phần client thu thập và hiển thị thông tin cho người dùng đầu cuối.
- Màn hình đăng ký đăng nhập:



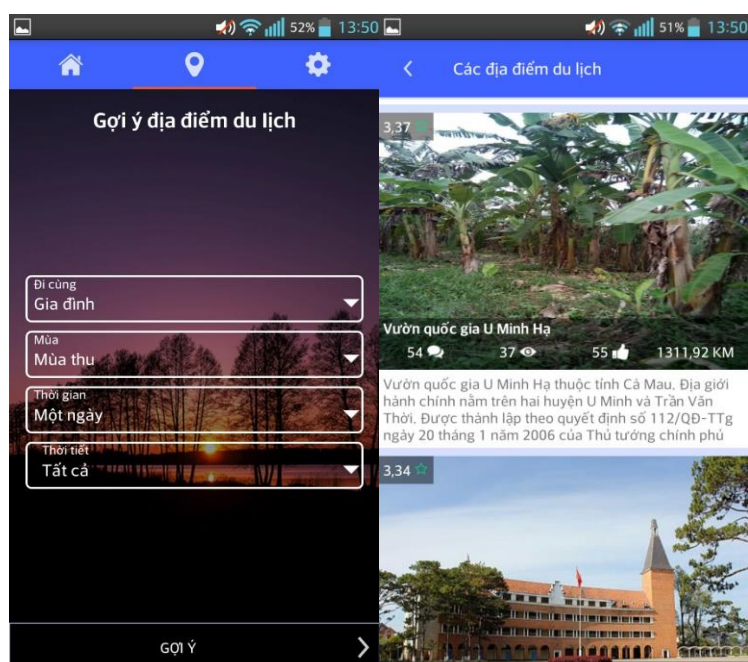
Hình 19: Giao diện xác thực người dùng

- Màn hình home (màn hình này client sẽ thu thập thông tin ngữ cảnh gồm vị trí, thời gian, thời tiết) hiển thị danh sách địa điểm du lịch gợi ý cho người dùng dựa trên ngữ cảnh thu thập được (client sẽ gửi dữ liệu về server để tính toán dự đoán xếp hạng và nhận các gợi ý của hệ thống trả lại cho người dùng):



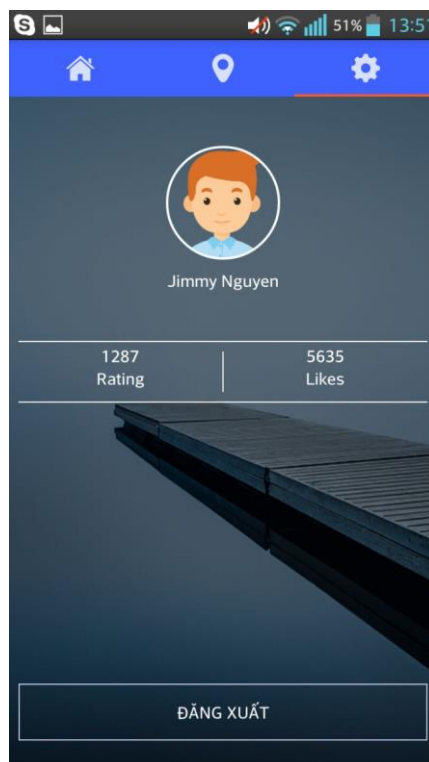
Hình 20: Giao diện home của client

- Màn hình gợi ý (dựa trên các ngữ cảnh tĩnh của người dùng lựa chọn để đưa ra các địa điểm du lịch phù hợp)



Hình 21: Giao diện nhập thông tin ngữ cảnh tĩnh và gợi ý

- Màn hình thông tin người dùng và đăng xuất khỏi ứng dụng



Hình 22: Giao diện thông tin người dùng và đăng xuất hệ thống

3.3.2 Đánh giá thử nghiệm

Đánh giá về tốc độ tính toán của hệ thống: Như đã trình bày trong phần 2.3.3, kỹ thuật CAMF được thực hiện tính toán trong thời gian tuyến tính, đây là một lợi thế rất lớn của CAMF khi đưa vào áp dụng trong thực tế. Trong ứng dụng thực nghiệm, khi ngữ cảnh của người dùng thay đổi (vị trí thay đổi, thời tiết thay đổi, ...) thì màn hình HOME của ứng dụng sẽ tự động thay đổi và đưa ra các gợi ý địa điểm được đánh giá cao, phù hợp với người dùng. Tương tự như thế, khi người dùng chọn tính năng lựa chọn các ngữ cảnh đầu vào manual để yêu cầu hệ thống đưa ra các gợi ý, thì hệ thống sẽ căn cứ trên dữ liệu ngữ cảnh đầu vào để tính toán và đưa ra gợi ý địa điểm phù hợp cho người dùng. Để mang lại cảm giác tốt nhất cho người dùng thì thời gian mà hệ thống thực hiện phải đáp ứng realtime. Với môi trường được dùng để xây dựng hệ thống thực nghiệm trong luận án (sử dụng chỉ 1 server cấu hình tương đối thấp, chưa áp dụng tính toán song song, ...), tập dữ liệu training gồm 1000 ratings cho 50 user và 42 địa điểm mẫu thì thời gian đáp ứng trung bình cho một lần tính toán là xấp xỉ 2s. So với các phương pháp gợi ý truyền thống, đây là thời gian thực hiện hiệu quả và cơ bản đáp ứng được yêu cầu của người dùng. Đây cũng là tiền đề để triển khai và phát triển các hệ thống sử dụng CAMF trên các tập dữ liệu lớn, tất nhiên khi đó chúng ta phải sử dụng nhiều hơn các tài nguyên về phần cứng, đồng thời áp dụng các kỹ thuật tính toán song song thì khả năng đáp ứng về thời

gian của hệ thống sẽ hoàn toàn thỏa mãn được nhu cầu của người dùng.
Đánh giá về thời gian thực hiện:

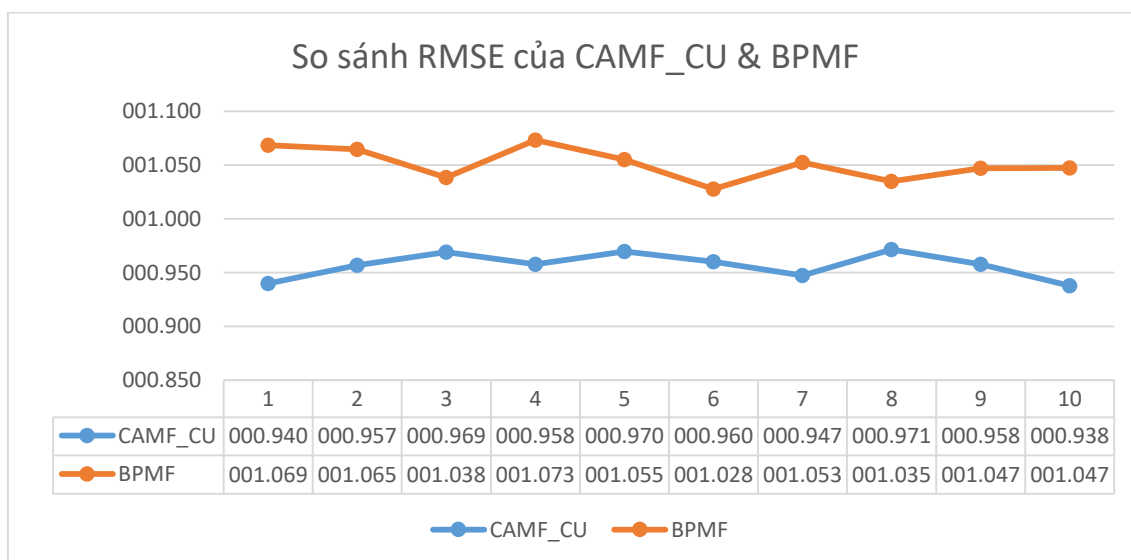
Lần thực hiện	Thời gian học (ms) (T1)	Thời gian học và đưa ra gợi ý (ms) (T2)	Thời gian thu thập dữ liệu từ CSDL (ms) (T3)	Tổng thời gian (ms) (T1)+(T2)+(T3)
Lần 1	781	1936	97	2814
Lần 2		2900	16	3697
Lần 3		1724	20	2525
Lần 4		2438	16	3235
Lần 5		2042	26	2849
Trung bình				2377

Bảng 5: Đánh giá thời gian thực hiện

Đánh giá về độ chính xác dự đoán theo RMSE so sánh giữa CAMF_CU và BPMF (Bayesian Probabilistic Matrix Factorization): một trong những tiêu chí quan trọng để đánh giá một hệ thống gợi ý chính là độ chính xác dự đoán. Trong phần này, luận văn lựa chọn độ đo RMSE đã trình bày tại phần 1.3.4 với các tham số tốc độ học α cũng như tham số chuẩn hóa β thay đổi khác nhau. Do tập dữ liệu đầu vào được xây dựng dựa trên nguồn thu thập trên internet và các giá trị rating được xác định ngẫu nhiên nhằm mục đích thử nghiệm kỹ thuật CAMF, do đó đề tài chọn độ đo RMSE để đánh giá độ chính xác thay vì chọn các độ đo khác như Precision, Recall, ... (cần phải đánh giá offline, khảo sát thực tế đánh giá của người dùng so với kết quả tính toán). Qua các lần tính toán khác nhau, kỹ thuật CAMF đều cho ra các kết quả đánh giá RMSE ~ 0.9 , so với phương pháp phân rã ma trận 2 chiều BPMF (RMSE > 1) thì độ chính xác dự đoán của CAMF rất ấn tượng. Giá trị RMSE ~ 0.9 sau tính toán của kỹ thuật CAMF cũng chứng minh các giá trị dự đoán xếp hạng sau quá trình học máy trên tập dữ liệu đầu vào có độ chính xác cao, đồng nghĩa với các gợi ý địa điểm đưa ra phù hợp với đánh giá của người dùng. Bảng dưới đây là số liệu tính toán RMSE qua 10 lần thực hiện với α, β thay đổi, tương ứng với 2 kỹ thuật CAMF và MF truyền thống, cụ thể:

Lần thực hiện	Tham số	CAMF_CU	BPMF
1	$\alpha = 0.1 \beta = 0.5$	0.94008	1.06854
2	$\alpha = 0.2 \beta = 0.5$	0.95678	1.06468
3	$\alpha = 0.5 \beta = 0.5$	0.96924	1.03847
4	$\alpha = 0.5 \beta = 0.1$	0.95791	1.07317
5	$\alpha = 0.5 \beta = 0.2$	0.96967	1.05508
6	$\alpha = 1 \beta = 0.5$	0.96027	1.02787
7	$\alpha = 0.5 \beta = 1$	0.94744	1.05255
8	$\alpha = 0.1 \beta = 0.9$	0.97139	1.03496
9	$\alpha = 0.5 \beta = 0.9$	0.95778	1.04715
10	$\alpha = 0.9 \beta = 0.9$	0.93789	1.04727

Bảng 6: Đánh giá độ chính xác dự đoán



Hình 23: So sánh độ chính xác của CAMF & BPMF theo RMSE

Chương 4. KẾT LUẬN CHUNG

4.1 Đóng góp của luận văn

Trong quá trình tìm hiểu để đưa ra cách giải quyết cho bài toán cá nhân hóa ứng dụng và dịch vụ di động theo hướng ngữ cảnh người dùng, luận văn nêu lại những mảng kiến thức tổng quan về ngữ cảnh, khái niệm, đặc trưng, phương thức quản lý cũng như các hướng tiếp cận nhận biết ngữ cảnh. Phần tổng quan trong luận văn cũng nêu lại các nội dung về bài toán lớn hệ gợi ý, các hướng tiếp cận, phương pháp giải quyết, ứng dụng và đánh giá... Với hướng tiếp cận sử dụng kết hợp thông tin ngữ cảnh trong bài toán gợi ý, luận văn đã làm rõ về hệ gợi ý dựa trên nhận biết ngữ cảnh, cách thức quản lý ngữ cảnh trong hệ gợi ý, các phương thức tích hợp ngữ cảnh vào trong tiến trình tính toán dự đoán gợi ý.

Đặc biệt, luận văn đã trình bày chi tiết về hệ gợi ý phân rã ma trận dựa trên nhận biết ngữ cảnh (CAMF). Một hệ thống gợi ý mà phương thức xây dựng áp dụng theo đúng cách tiếp cận mô hình hóa, thực hiện tính toán, học máy, mở rộng từ kỹ thuật phân rã ma trận thiên vị trên tập dữ liệu tập luyện nhiều chiều ($R: \text{USER} \times \text{ITEM} \times \text{CONTEXT} \Rightarrow \text{RATING}$) để đưa ra các gợi ý cho người dùng. Hệ thống CAMF sẽ giải quyết được các thách thức đối với các hệ gợi ý truyền thống đang gặp phải như vấn đề dữ liệu thừa, vấn đề khởi động chậm cũng như vấn đề mở rộng hệ thống, với thời gian tính toán và học máy được thực hiện theo hàm tuyến tính. Từ những cơ sở đó, luận văn đưa ra phương pháp và lý do áp dụng cho bài toán thực nghiệm của luận văn: Xây dựng ứng dụng du lịch theo hướng cá nhân hóa dựa trên ngữ cảnh của người dùng.

Kết quả sản phẩm của luận văn là đã xây dựng thành công một hệ thống client – server cho các thiết bị di động. Chức năng chính của hệ thống là tích hợp ngữ cảnh người dùng như vị trí, thời gian và thời tiết hiện tại để hệ thống sẽ đưa ra gợi ý cho riêng người dùng đó với tùy biến cá nhân về các địa điểm du lịch phù hợp với ngữ cảnh động và ngữ cảnh tĩnh của người dùng. Hướng gợi ý của hệ thống gồm hai hướng: 1) gợi ý danh sách các địa điểm được dự đoán xếp hạng cao phù hợp với người dùng tương ứng với ngữ cảnh hiện tại và 2) gợi ý danh sách các địa điểm được dự đoán xếp hạng cao dựa vào ngữ cảnh tĩnh mà người dùng lựa chọn tìm kiếm. Phương pháp áp dụng để tính toán chính là dựa trên hệ gợi ý phân rã ma trận theo hướng nhận biết ngữ cảnh người dùng.

Kết quả chạy thực nghiệm đã chứng minh được nhận định trước đó về độ chính xác dự đoán, về thời gian thực hiện từ bước thu thập, nhận biết và chuẩn hóa dữ liệu ngữ cảnh đầu vào đến bước hiển thị kết quả xếp hạng địa điểm gợi ý cho người dùng. Kết quả cũng là minh chứng tính khoa học khi lựa chọn và sử dụng phương pháp CAMF cho các hệ gợi ý dựa trên nhận biết ngữ cảnh. Tuy rằng thực nghiệm mới dừng lại ở bước ngoại tuyến offline, chưa có điều kiện áp dụng ở những bước khảo sát nhóm người dùng và đánh giá ở môi trường thực tế, nhưng kết quả mang lại cũng có những ý nghĩa nhất định về tính khả thi khi áp dụng triển khai phương pháp dùng một hệ gợi ý dựa trên nhận biết ngữ cảnh để giải quyết các bài toán cá nhân hóa ứng dụng di động hiện nay.

4.2 Hướng phát triển

Một trong những thách thức của hệ gợi ý dựa trên nhận biết ngữ cảnh vị trí chính là độ chính xác trong quá trình thu thập, nhận biết và chuẩn hóa dữ liệu

ngữ cảnh đầu vào. Hệ thống phải thực hiện chọn lọc thông tin ngữ cảnh vừa đủ để đảm bảo thông tin có hàm lượng ảnh hưởng hiệu quả lên quá trình tính toán nhưng cũng phải đảm bảo không dư thừa dữ liệu vì khi đó sẽ phát sinh các vấn đề về thời gian tính toán và hệ thống bị mở rộng không mong muốn.

Một hướng giải quyết của luận văn trong tương lai với thách thức này là thu thập thông tin về vị trí của người dùng dựa trực tiếp từ các nhà cung cấp dịch vụ viễn thông (Telco), dựa trên các nền tảng như hệ thống LBS (Location Based Service). Khi đó, các hệ thống gợi ý sẽ không còn phụ thuộc vào thông tin ngữ cảnh vị trí truyền nhận từ phía client phần mềm (bắt buộc phải được người dùng cài đặt trước khi thu thập được thông tin) mà có thể nhận thông tin về vị trí của người dùng trực tiếp từ các hệ thống LBS của Telco với độ chính xác cao về thông tin vị trí.

Ngoài ra đề gợi ý tốt hơn cho nhóm người dùng mới hay dữ liệu mới, có thể áp dụng mở rộng các chiều ngữ cảnh tích hợp vào hệ thống, như các ngữ cảnh về cảm nhận người dùng sau khi sử dụng, ngữ cảnh quan hệ xã hội thu thập từ các mạng xã hội đang thịnh hành hiện nay như Facebook, Zalo, ... Khi đó, có thể mở rộng chức năng của hệ thống, không chỉ là thực hiện gợi ý cho người dùng các địa điểm du lịch mà có thể là các tính năng gợi ý trong thương mại điện tử, trong quảng cáo trực tuyến trên các nền tảng di động theo hướng cá nhân hóa người dùng, đang và sẽ là xu hướng của ngành công nghệ thông tin hiện đại với hệ gợi ý, học máy, tính toán song song là các nền tảng công nghệ.

Hy vọng những vấn đề được đề cập trong luận văn từ lớn đến nhỏ, từ cách tiếp cận, đến những vướng mắc khi giải quyết các bước thực hiện, sẽ giúp phần nào đưa ra cách giải quyết, cách tư duy trong những bài toán thực tế nói chung và bài toán gợi ý nói riêng, góp phần nào đó chứng minh được tính khả thi của hướng áp dụng các hệ gợi ý dựa trên nhận biết ngữ cảnh theo mô hình hóa trong việc xây dựng các hệ thống ứng dụng cá nhân hóa hướng ngữ cảnh người dùng.

TÀI LIỆU THAM KHẢO

1. Albrecht Schmidt, Michael Beigl, and Hans-W. Gellersen, “There is more to Context than Location”, University of Karlsruhe
2. Schmidt, A., Aidoo, K.A., Takaluoma, A, Tuomela, U., Van Laerhoven, K., Van de Velde, W. (1999), “Advanced Interaction In Context”. Lecture Notes in Computer Science, Hand Held and Ubiquitous Computing .
3. P.J. Brown and J.D. Bovey and C. Xian (1997), “Context-aware applications: from the laboratory to the marketplace”, IEEE Personal Communications, 4(5).
4. A.Newell and H.A. Simon (1965), “Computer Augmentation of Human Reasoning” Spartan Books, Washington DC, USA.
5. G. Chen and D. Kotz (2000), “A Survey of Context-Aware Mobile Computing Research”, Techreport TR2000-381, Dept. of Computer Science, Dartmouth College.
6. Albrecht Schmidt (2002), “Ubiquitous Computing –Computing in Context”, Computing Department, Lancaster University, U.K.

7. Karen Henricksen (2005), "Modelling and Using Imperfect Context Information", School of Information Technology and Electrical Engineering, The University of Queensland.
8. A.Schilit, N. Adams, and R. Want (1994), "Context-aware computing applications", In IEEE Workshop on Mobile Computing Systems and Applications.
9. Pashtan A., Blattler R., Heusser A. and Scheurmann P. (2003), "CATIS: A Context-Aware Tourist Information System", Proceedings of the 4th International Workshop of Mobile Computing, Rostock.
10. Anind K. Dey (2000), "Providing Architectural Support for Building Context-Aware Applications", Georgia Institute of Technology, November.
11. Yun Her, Su-Kyoung Kim, YoungTaek Jin (2010), "A Context-Aware Framework using Ontology for Smart Phone Platform", International Journal of Digital Content Technology.
12. Nguyen Thai-Nghe. 2013. An introduction to factorization technique for building recommendation systems. Vol.6/2013, pp. 44-53, Journal of Science - University of Da Lat, ISSN 0866-787X.
13. Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor. 2010. Recommender Systems Handbook (1st ed.). Springer-Verlag New York, Inc., New York, NY, USA.
14. Herlocker, J.L., Konstan, J.A., Terveen, K., et al. (2004) Evaluating Collaborative Filtering Recommender Systems. ACM Transactions on Information and Systems, 22, 5-53.
15. Empirical analysis of predictive algorithms for collaborative filtering, JS Breese, D Heckerman, C Kadie - Proceedings of the Fourteenth conference on ..., 1998.
16. G. Adomavicius and a Tuzhilin, "Toward the Next Generation of Recommender Systems: a Survey of the State of the Art and Possible Extensions," IEEE Trans. Knowl. Data Eng., vol. 17, no. 6, pp. 734-749, 2005.
17. A.M. Rashid, I. Albert, D. Cosley, S. K. Lam, S. M. McNee, J. a Konstan, and J. Riedl, "Getting to Know You: Learning New User Preferences in Recommender Systems," Int. Conf. Intell. User Interfaces, IUI 2002, pp. 127-134, 2002.
18. Gediminas Adomavicius, Alexander Tuzhilin. Context-Aware Recommender Systems, Recommender Systems Handbook. 2011, pp 217-253. Springer
19. J. L. Herlocker and J. A. Konstan. Content-independent task-focused recommendation. IEEE Internet Computing, pages 40-47, 2001.
20. S. Abbar, M. Bouzeghoub, and S. Lopez. Context-aware recommender systems: A service-oriented approach. VLDB PersDB Workshop, 2009.
21. F. Cena, L. Console, C. Gena, A. Goy, G. Levi, S. Modeo, and I. Torre. Integrating heterogeneous adaptation techniques to build a flexible and usable mobile tourist guide. AI Communications, 19(4):369-384, 2006.
22. S. Chaudhuri and U. Dayal. An overview of data warehousing and olap technology. ACM Sigmod record, 26(1):65-74, 1997.
23. G. D. Abowd, C. G. Atkeson, J. Hong, S. Long, R. Kooper, and M. Pinkerton. Cyberguide: A mobile context-aware tour guide. Wireless Networks, 3(5):421-433, 1997.

24. K. Cheverst, N. Davies, K. Mitchell, A. Friday, and C. Efstratiou. Developing a context-aware electronic tourist guide: some issues and experiences. In Proceedings of the SIGCHI conference on Human factors in computing systems, pages 17–24. ACM, 2000.
25. L. Ardissono, A. Goy, G. Petrone, M. Segnan, and P. Torasso. Intrigue: personalized recommendation of tourist attractions for desktop and hand held devices. *Applied Artificial Intelligence*, 17(8):687–714, 2003.
26. M. Van Setten, S. Pokraev, and J. Koolwaaij. Context-aware recommendations in the mobile tourist application compass. In W. Nejdl and P. De Bra, editors, *Adaptive Hypermedia*, pages 235–244. Springer Verlag, 2004.
27. B. De Carolis, I. Mazzotta, N. Novielli, and V. Silvestri. Using common sense in providing personalized recommendations in the tourism domain. In *Workshop on Context-Aware Recommender Systems (CARS 2009)*. New York, 2009.
28. G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)*, 23(1):103–145, 2005.
29. K. Oku, S. Nakajima, J. Miyazaki, and S. Uemura. Context-aware SVM for context-dependent information recommendation. In *Proceedings of the 7th International Conference on Mobile Data Management*, page 109, 2006.
30. U. Panniello, A. Tuzhilin, M. Gorgoglione, C. Palmisano, and A. Pedone. Experimental comparison of pre-vs. post-filtering approaches in context-aware recommender systems. In *Proceedings of the 3rd ACM conference on Recommender systems*, pages 265–268. ACM, 2009.
31. Z. Yu, X. Zhou, D. Zhang, C. Y. Chin, X. Wang, and J. Men. Supporting context-aware media recommendations for smart phones. *IEEE Pervasive Computing*, 5(3):68–75, 2006.
32. B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.
33. G. Adomavicius and A. Tuzhilin. Incorporating context into recommender systems using multidimensional rating estimation methods. In *Proceedings of the 1st International Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces (WPRSIUI 2005)*, 2005.
34. Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (August 2009), 30-37. DOI=10.1109/MC.2009.263
35. Y. Koren. Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge Discovery and Data Mining, KDD '09*, pages 447–456, New York, NY, USA, 2009. ACM.
36. Baltrunas, Ludwig, Ricci. *Matrix Factorization Techniques for Context Aware Recommendation*, RecSys 2011
37. Yong Zheng, Bamshad Mobasher, Robin Burke. *CARSKit: A Java-Based Context-aware Recommendation Engine*