**ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

**TRẦN NGUYÊN LỘC**

**KHẢO SÁT VỀ HỆ THỐNG KHUYẾN NGHỊ**

**TRÊN MỘT SỐ TRANG WEB XEM PHIM**

**ĐỒ ÁN CHUYÊN NGÀNH**

NGÀNH: CÔNG NGHỆ THÔNG TIN

**Thành phố Hồ Chí Minh, tháng 12 năm 2023**

**ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

**TRẦN NGUYÊN LỘC**

**KHẢO SÁT VỀ HỆ THỐNG KHUYẾN NGHỊ**

**TRÊN MỘT SỐ TRANG WEB XEM PHIM**

**ĐỒ ÁN CHUYÊN NGÀNH**

ngành: CÔNG NGHỆ THÔNG TIN

**Giảng viên phụ trách**

**TS. PHAN TẤN QUỐC**

**Thành phố Hồ Chí Minh, tháng 12 năm 2023**

**LỜI CẢM ƠN**

Trước hết em xin gửi đến lời cảm ơn chân thành và sâu sắc nhất đến thầy TS. Phan Tấn Quốc, người trực tiếp hướng dẫn và tận tình chỉ bảo cho em cho tới khi em hoàn thành đồ án của mình.

Tiếp đến em dành lời cảm ơn đến quý thầy cô khoa Công nghệ thông tin – trường Đại học Sài Gòn đã truyền đạt cho em những kiến thức vô cùng quý báu và bổ ích trong suốt quá trình nghiên cứu và học tập tại trường.

Xin chân thành cảm ơn tới những người bạn đã luôn sát cánh cùng em, những lời động viên, những lần hỗ trợ những lúc cần thiết đã phần nào giúp em hoàn thành đồ án này.

Cuối cùng, em xin cảm ơn đến ba mẹ và người thân trong gia đình đã hỗ trợ và tạo điều kiện thuận lợi cho em trong suốt thời gian học tập và nghiên cứu tại Đại học Sài Gòn.

Mục lục

**DANH MỤC HÌNH ẢNH** iv

**LỜI MỞ ĐẦU** 1

**CHƯƠNG 1. TỔNG QUAN VỀ HỆ THỐNG KHUYẾN NGHỊ** 4

**1.1. Giới thiệu chung** 4

**1.2. Phát biểu về bài toán khuyến nghị** 5

**1.3. Các hướng tiếp cận của bài toán khuyến nghị** 8

**1.3.1. Phương pháp lọc dựa trên nội dung** 8

**1.3.1.1. Định nghĩa** 8

**1.3.1.2. Khái quát bài toán** 8

**1.3.1.3. Phân loại các cách tiếp cận lọc dựa trên nội dung** 9

**1.3.1.4. Ưu điểm và khuyết điểm của lọc dựa trên nội dung** 11

**1.3.2. Phương pháp lọc cộng tác** 12

**1.3.2.1. Định nghĩa** 12

**1.3.2.2. Khái quát bài toán** 13

**1.3.2.3. Phân loại các cách tiếp cận lọc cộng tác** 14

**1.3.2.4. Ưu điểm và khuyết điểm của lọc cộng tác** 15

**1.3.3. Phương pháp tiếp cận lai** 16

**1.4. Khảo sát một số công trình nghiên cứu khoa học có liên quan** 22

**1.4.1. Nghiên cứu và xây dựng hệ thống khuyến nghị cho bài toán dịch vụ giá trị gia tăng trong ngành viễn thông** 22

**1.4.1.1. Tổng quan về công trình nghiên cứu** 22

**1.4.1.2. Đề xuất thuật toán** 24

**1.4.1.3. Kết quả thực nghiệm của công trình** 26

**1.4.1.4. Tóm tắt công trình nghiên cứu và hướng đi cuối** 27

**1.4.2. Giải quyết vấn đề phân phối trong hệ thống khuyến nghị dựa trên đặc trưng nội dung của đối tượng** 27

**1.4.2.1. Tổng quan về công trình nghiên cứu** 27

**1.4.2.2. Đề xuất thuật toán** 28

**1.4.2.3. Kết quả thực nghiệm của công trình** 32

**1.4.2.4. Tóm tắt công trình nghiên cứu và hướng đi cuối** 33

**1.4.3. Kết luận rút ra từ khảo sát các công trình nghiên cứu có liên quan** 34

**1.5. Giới thiệu bài toán khuyến nghị phim trên các trang web xem phim** 34

**1.6. Tóm tắt chương 1** 35

**CHƯƠNG 2. MỘT SỐ PHƯƠNG PHÁP GIẢI QUYẾT BÀI TOÁN KHUYẾN NGHỊ PHIM** 36

**2.1. PMF** 36

**2.1.1. Tổng quan** 36

**2.1.2. Mô hình bài toán** 37

**2.1.2.1. PMF không ràng buộc** 37

**2.1.2.2. PMF ràng buộc** 39

**2.1.3. Thực nghiệm và đánh giá** 40

**2.1.3.1. Mô tả tập dữ liệu thực nghiệm** 40

**2.1.3.2. Kết quả thực nghiệm của PMF không ràng buộc** 41

**2.1.3.3. Kết quả thực nghiệm của PMF ràng buộc** 43

**2.2. BPMF** 45

**2.2.1. Tổng quan** 45

**2.2.2. Mô hình bài toán** 46

**2.2.2.1. Lý thuyết** 46

**2.2.2.2. Giải thuật và mã giả** 48

**2.2.3. Quá trình và đánh giá thực nghiệm** 50

**2.2.3.1. Mô tả tập dữ liệu thực nghiệm** 50

**2.2.3.2. Mô tả quá trình huấn luyện mô hình PMF** 51

**2.2.3.3. Mô tả quá trình huấn luyện mô hình BPMF** 51

**2.2.3.4. Kết quả thực nghiệm thu được** 52

**2.3. ALS** 54

**2.3.1. Tổng quan và mô hình bài toán** 54

**2.3.2. Thực nghiệm và đánh giá** 56

**2.4. Hệ thống khuyến nghị phim MOVREC** 59

**2.4.1. Giới thiệu** 59

**2.4.2. Mô tả cấu trúc hệ thống** 59

**2.4.2.1. Tổng quan về thuật toán K-Means** 59

**2.4.2.2. Mô tả dữ liệu mô phỏng** 60

**2.4.2.3. Mô phỏng quá trình hoạt động của MOVREC** 61

**2.4.2.4. Mã giả thuật toán** 62

**2.4.2.6. Thành công đạt được** 64

**2.5. Tóm tắt chương 2** 65

**TÀI LIỆU THAM KHẢO** 66

**Tiếng anh** 66

**Tiếng việt** 67

# **DANH MỤC HÌNH ẢNH**

[Hình 1. 1. Ví dụ về hệ thống khuyến nghị của một trang web xem phim 4](#_Toc148874333)

[Hình 1. 2. Ví dụ ma trận đánh giá tổng quát Rij 6](#_Toc148874334)

[Hình 1. 3. Ví dụ mô hình kỹ thuật lọc dựa trên nội dung 8](#_Toc148874335)

[Hình 1. 4. Ví dụ mô hình kỹ thuật lọc cộng tác 12](#_Toc148874336)

[Hình 1. 5. Dấu ? là những giá trị cần tiên đoán trong ma trận đánh giá 13](#_Toc148874337)

[Hình 1. 6. Ví dụ minh họa cho phương pháp tiếp cận lai 17](#_Toc148874338)

[Hình 1. 7. Minh họa cho dịch vụ MCA của nhà mạng Viettel 23](#_Toc148874339)

[Hình 1. 8. Hình ảnh được lấy từ công trình nghiên cứu minh họa cho thông tin người dùng viễn thông 24](#_Toc148874340)

[Hình 1. 9. Hình ảnh được lấy từ công trình minh họa cho việc dữ liệu được phân cụm 30](#_Toc148874341)

[Hình 1. 10. Mô hình giả mã của tiến trình xử lý thuật toán của tác giả Nguyễn Văn Đạt 31](#_Toc148874342)

[Hình 1. 11. Kết quả thực nghiệm từ công trình của tác giả Nguyễn Văn Đạt (MSE áp dụng BOW+GFF, W2V+GFF) 32](#_Toc148874343)

[Hình 1. 12. Kết quả thực nghiệm từ công trình của tác giả Nguyễn Văn Đạt (MSE áp dụng GMM+GFF, GMM+ED) 33](#_Toc148874344)

[Hình 1. 13. Thời gian thực hiện truy vấn từ công trình của tác giả Nguyễn Văn Đạt 33](#_Toc148874345)

[Hình 2. 1. Kết quả thực nghiệm thu được bằng cách sử dụng PMF (Bản số 1) 40](#_Toc149441969)

[Hình 2. 2. Kết quả thực nghiệm thu được bằng cách sử dụng PMF (Bản số 2) 44](#_Toc149441970)

[Hình 2. 3. Kết quả thực nghiệm thu được bằng cách sử dụng PMF (Bản số 3) 44](#_Toc149441971)

[Hình 2. 4. Hình bên trái là mộ hình đồ thị PMF, Hình bên phải là mô hình đồ thị BPMF. 46](#_Toc149441972)

[Hình 2. 5. Kết quả thực nghiệm thu được bằng cách sử dụng phương pháp BPMF 52](#_Toc149441973)

[Hình 2. 6. Bảng dữ liệu thực nghiệm so sánh PMF và BPMF 53](#_Toc149441974)

[Hình 2. 7. Bảng cài đặt tham số tham khảo cho thực nghiệm 56](#_Toc149441975)

[Hình 2. 8. Tập khuyến nghị sử dụng thuật toán ALS 56](#_Toc149441976)

[Hình 2. 9. Giá trị RMSE tương ứng với số lần lặp trong thực nghiệm 57](#_Toc149441977)

[Hình 2. 10. Mô hình biễu diễn giá trị RMSE 57](#_Toc149441978)

[Hình 2. 11. So sánh giá trị RMSE giữa các phương pháp ALS, SVD++ và SGD 58](#_Toc149441979)

[Hình 2. 12. Mô hình biểu diễn so sánh giá trị RMSE giữa các phương pháp ALS, SVD++ và SGD 58](#_Toc149441980)

# **LỜI MỞ ĐẦU**

**Lý do chọn đề tài**

Với sự phát triển của xã hội hiện nay, việc nhu cầu giải trí của con người ngày càng tăng cao trong số đó phải kể đến là nhu cầu giải trí thông qua phim ảnh. Với lịch sử lâu đời của ngành điện ảnh cùng với kho tài liệu phim ảnh khổng lồ đã được sản xuất qua năm tháng, cùng với đó là sự phát triển của công nghệ hiện đại ngày nay đã cho phép các bộ phim có thể được lưu trữ trong các cơ sở dữ liệu lớn và được sử dụng để phục vụ cho lượng lớn khán giả trong hầu hết các trang web xem phim trực tuyến. Nhưng với số lượng phim khổng lồ và hàng ngàn thể loại phim khác nhau như thế sẽ là một vấn đề đối với các nhà sản xuất phim ảnh, họ có thể sẽ không thể thu lại lợi nhuận từ phim của họ nếu bộ phim mà họ sản xuất không được đến được tay khán giả hoặc không được nhiều người chú ý đến. Do đó, Hệ thống Khuyến Nghị là một trong những giải pháp ứng dụng phù hợp và tốt nhất để giải quyết cho vấn đề trên, nhằm thu hút nhiều khán giả đến với các sản phẩm phim hơn, đồng thời cũng là cầu nối cho khán giả đến với các bộ phim.

Và ngày nay, hệ thống Khuyến Nghị (Recommender System) là một trong những lớp ứng dụng thành công và phổ biến nhất của trí tuệ nhân tạo. Với các nền tảng dịch vụ trực tuyến đang ngày càng phát triển mạnh mẽ trong đời sống hiện nay thì Hệ thống Khuyến Nghị đóng một vai trò rất lớn trong việc ứng dụng vào các ngành Dịch vụ của con người, Ví dụ: Thương mại điện tử, mua bán các sản phẩm dịch vụ trực tuyến, ứng dụng trực tuyến, xem phim/video trực tuyến ..v..vv.

Trên các trang web xem phim, hệ thống Khuyến nghị đóng một vai trò chính trong việc giới thiệu các bộ phim đến với các khán giả. Nó đóng vai trò phân tích và tìm hiểu khối dữ liệu cá nhân của người dùng và từ đó đưa ra những dự đoán, gợi ý, đề xuất phù hợp với sở thích của khán giả. Các trang web xem phim lớn hiện nay ứng dụng thành công hệ thống Khuyến nghị như Netflix, BiliBili, FPT Play …

**Mục đích nghiên cứu**

Đề tài của đồ án chuyên ngành tập trung phân tích và làm rõ cách thức hoạt động của hệ thống khuyến nghị một số trang web xem phim, đồng thời phân tích chi tiết các thuật toán được ứng dụng để giải quyết bài toán khuyến nghị phim.

**Nhiệm vụ nghiên cứu**

Khái quát và tổng quan về Hệ thống Khuyến nghị.

Phân tích về các thuật toán được sử dụng để giải quyết bài toán khuyến nghị phim.

Khảo sát về Hệ thống khuyến nghị trên các trang web xem phim.

**Đối tượng nghiên cứu và phạm vi nghiên cứu**

Đối tượng nghiên cứu: các thuật toán được sử dụng để giải quyết cho bài toán khuyến nghị phim trên một số trang web xem phim.

Phạm vi nghiên cứu: Trang web xem phim như Netflix, BiliBili, FPT Play.

**Phương pháp nghiên cứu**

Phương pháp quan sát: Quan sát hành vi thu thập thông tin người dùng của một số hệ thống Khuyến nghị nhằm phân tích bài toán khuyến nghị phim.

Phương pháp điều tra: Tìm hiểu cụ thể đặc điểm và tính chất của bài toán Khuyến nghị phim trên các trang web xem phim nhằm đưa ra một bài viết dễ hình dung cho người đọc và có độ chính xác về mặt nội dung cao.

**Cấu trúc đồ án chuyên ngành**

Cấu trúc của đồ án chuyên ngành gồm 3 phần chính:

**Chương 1.** **Tổng quan về hệ thống khuyến nghị**

Chương 1 trong đồ án chuyên ngành sẽ giới thiệu tổng quan về hệ thống khuyến nghị, lý thuyết của bài toán khuyến nghị và các phương pháp tiếp cận của hệ thống khuyến nghị. Bên cạnh đó, đồ án chuyên ngành còn giới thiệu thêm về một số đề tài có liên quan của hệ thống khuyến nghị. Sau cùng sẽ giới thiệu bài toán khuyến nghị phim thường được sử dụng trong các hệ thống khuyến nghị trên các trang web xem phim.

**Chương 2. Một số phương pháp giải quyết bài toán khuyến nghị phim**

Chương 2 trong đồ án chuyên ngành sẽ tập trung khảo sát và phân tích vào các thuật toán, phương pháp, hệ thống được sử dụng phổ biến trong bài toán khuyến nghị phim, trình bày chi tiết các phương pháp như PMF (Probabilistic Matrix Factorization), BPMF (Bayesian Probabilistic Matrix Factorization), ALS (Alternating Least Squares), hệ thống đề xuất phim MOVREC.

**Chương 3. Khảo sát hệ thống khuyến nghị phim trên một số trang web xem phim**

Chương 3 trong đồ án chuyên ngành sẽ tập trung vào việc khảo sát cách thức của một hệ thống khuyến nghị hoạt động trên các trang web xem phim. Các trang web được khảo sát trong đồ án chuyên ngành gồm Netflix, BiliBili và FPT Play. Sau đó ta so sánh cách thức xử lý bài toán khuyến nghị phim giữa 3 trang web xem phim trên.

# **CHƯƠNG 1. TỔNG QUAN VỀ HỆ THỐNG KHUYẾN NGHỊ**

## **1.1. Giới thiệu chung**

Hệ thống khuyến nghị (có tên gọi tiếng anh là Recommender System) hay còn được gọi là hệ thống tư vấn là một hệ thống có nhiệm vụ chọn lọc thông tin nhằm dự đoán sở thích, mức độ phù hợp, mối quan tâm và nhu cầu của người dùng để đưa ra một hoặc nhiều mục, sản phẩm, dịch vụ mà người dùng sẽ quan tâm với xác suất lớn nhất [1].

Và hiện nay, mọi hệ thống, ứng dụng có hiển thị quảng cáo trên internet đều sử dụng hệ thống khuyến nghị để đưa ra quảng cáo, đề xuất tốt nhất có thể đến cho người dùng… Một vài ví dụ phổ biến và dễ gặp nhất đó là gợi ý sản phẩm hoặc dịch vụ có liên quan trên các trang web, ứng dụng phổ biến.



Hình 1. 1. Ví dụ về hệ thống khuyến nghị của một trang web xem phim

Và để làm được điều đó, hệ thống khuyến nghị đã sử dụng những thuật toán để phân tích và dự doán đựa trên dữ liệu hành vi người dùng được lưu lại. Nhờ đó, những quảng cáo mang tính cá nhân hóa được đưa đến cho người dùng. Hệ thống sẽ biết chính xác từng người dùng sử dụng có nhu cầu gì, muốn gì để từ đó đưa ra khuyến nghị.

Trong thực tế, ý tưởng để những người lập trình xây dựng một hệ thống khuyến nghị không đâu xa lạ chính là xuất phát từ hành vi của người mua hàng và người bán hàng: Khi một người mua hàng có nhu cầu mua một sản phẩm, họ thường sẽ có hành vi hỏi người bán hàng để tư vấn cho họ về sản phẩm mà họ có ý định mua. Người bán hàng sẽ tiến hành thu thập thông tin từ người mua bao gồm: nhu cầu sử dụng, đặc điểm, mức độ phù hợp, chức năng, màu sắc, … đồng thời kết hợp với kiến thức hiểu biết của mình về sản phẩm để đưa ra đề xuất, lời khuyên sản phẩm phù hợp nhất cho người mua. Và ở một mức độ cao hơn, người bán sẽ liên hệ, liên tưởng những người đã từng mua sản phẩm mà có đặc điểm tương đồng vời người mua hiện tại, từ đó họ dự đoán người mua hiện tại có khả năng thích sản phẩm nào nhất để đưa ra khuyến nghị sản phẩm phù hợp nhất cho người mua.

## **1.2. Phát biểu về bài toán khuyến nghị**

Để có thể xây dựng được một hệ thống khuyến nghị hoàn chỉnh cho từng lĩnh vực cụ thể, các nghiên cứu trước đã phát biểu được lời giải chung cho bài toán khuyến nghị như sau:

**Định nghĩa 1: Không gian người dùng** [2]

Không gian người dùng là tập tất cả nhữg người dùng mà hệ thống quan sát được, để thực hiện phân tích, khuyến nghị. Ký hiệu là *U, U =* {}.

**Định nghĩa 2: Không gian đối tượng khuyến nghị** [2]

Không gian đối tượng khuyến nghị là tập tất cả những đối tượng sẽ được khuyến nghị cho người dùng. Tùy vào ứng dụng cụ thể, đối tượng khuyển nghị có thể là sản phẩm, dịch vụ hoặc con người … Ký hiệu là *P, P =* {}.

**Định nghĩa 3: Hàm phù hợp** [3]

Hàm phù hợp *F* là ánh xạ *F : U x P R*, dùng để ước lượng độ phù hợp của p P với u U. Với R là một ma trận có thứ tự các số nguyên hoặc thực trong một khoảng nhất định.

**Phát biểu bài toán khuyến nghị** [1]**:**

**Đầu vào (Input):**

**+** Tập người dùng *U*, mỗi người dùng thuộc *U* và có các đặc điểm I = {}.

**+** Một tập sản phẩm, dịch vụ (ở đây ta gọi chung là sản phẩm) P, mỗi sản phẩm có các đặc điểm đặc trưng J = {}.

+ Một ma trận đánh giá tổng quát R = ( với i = 1, … N và j = 1…M, thể hiện mối quan hệ giữa tập người dùng U đối với tập sản phẩm P. Trong đó là đánh giá của người dùng cho sản phẩm , N và M là lần lượt số người dùng và số sản phẩm.



Hình 1. 2. Ví dụ ma trận đánh giá tổng quát Rij

**Đầu ra (Output):**

Danh sách các sản phẩm thuộc *P* có độ phù hợp với người dùng thuộc *U* nhất.

Để giải bài toán này chúng ta cần xây dựng hàm F(, để đo độ phù hợp của sản phẩm với người dùng , từ đó ta sẽ lấy được danh sách các sản phẩm, dịch vụ phù hợp (là các sản phẩm, dịch vụ mà có khả năng được người dùng chọn) nhất.

Và cũng tùy thuộc vào phương pháp sử dụng mà ta có nhiều các xây dựng hàm F khác nhau, các cách xây dựng hàm F phụ thuộc chủ yếu bởi các yếu tố sau [1]:

* Đặc điểm của người dùng (lọc theo nội dung người dùng): đặc điểm này được đánh giá chủ quan bởi các quy luật tự nhiên hoặc các quy tắc cơ bản. Ví dụ là nữ thì sẽ có xu hướng chọn mua các sản phẩm của nữ hơn là các sản phẩm của nam [1].
* Đặc điểm của sản phẩm (lọc theo nội dung sản phẩm): cũng giống như lọc theo nội dung người dùng, các sản phẩm có đặc điểm giống nhau thì cũng có khả năng được người dùng đánh giá như nhau. Ví dụ đặc điểm của các món đồ công nghệ có thể là đặc điểm, tính năng, nhu cầu sử dụng … [1]
* Lịch sử giao dịch của người dùng : từ lịch sử giao dịch cũng có thể suy ra sản phẩm mà người dùng quan tâm đến, do đó các sản phẩm cùng thể loại, lĩnh vực sẽ có độ liên quan cao hơn. Ví dụ một người đã từng mua áo, giày bóng đá thì có thể dự đoán được được người là một người đam mê bóng đá, thích thể thao. Từ đó suy ra được người này sẽ có khả năng sử dụng dịch vụ hoặc mua các sản phẩm thể thao cao hơn các dịch vụ, sản phẩm khác [1].
* Những người dùng khác có cùng đặc điểm giống với người dùng : với quan niệm rằng những người dùng giống nhau sẽ thích, đánh giá những sản phẩm giống nhau. Các đặc điểm bao gồm tập đặc điểm *I*ban đầu, kết hợp với các đặc điểm cộng tác như cùng mua mặt hàng nào đó hoặc có các hành vi mua hàng giống nhau… Việc tìm hiểu những mặt hàng, dịch vụ mà đã từng quan tâm sẽ đưa ra được những gợi ý phù hợp cho người dùng [1].

## **1.3. Các hướng tiếp cận của bài toán khuyến nghị**

### **1.3.1. Phương pháp lọc dựa trên nội dung**

#### **1.3.1.1. Định nghĩa**

Lọc dựa trên nội dung (tên tiếng anh là Content-Base Filtering) là phương pháp thực hiện dựa trên việc so sánh nội dung thông tin hay mô tả hàng hóa, để tìm ra những sản phẩm tương tự với những gì mà người dùng đã từng quan tâm để giới thiệu cho họ những sản phẩm này [1].

A diagram of a movie

Description automatically generated

Hình 1. 3. Ví dụ mô hình kỹ thuật lọc dựa trên nội dung

Trong hình 1.3 ta có thể thấy mô hình minh họa cho việc lọc theo nội dung như sau: User A đánh giá thích Movie A và Movie A có thể loại Love, Romantic. Do đó, phương pháp lọc theo nội dung sẽ dựa theo Type (Thể loại) của Movie A và từ đó khuyến nghị Movie C có cùng Type (Thể loại) với Movie A cho User A.

#### **1.3.1.2. Khái quát bài toán**

Để thực hiện việc ước lượng xem có hay không một người dùng *u* có thích đối tượng sản phẩm *p.* Ta xây dựng một hàm phù hợp *f(u,p)* của các sản phẩm khuyến nghị *p* với người dùng *u* và ước lượng giá trị phù hợp này. Các phương pháp tiếp cận nội dung thường sẽ thực hiện các bước sau đây [4]:

* **Bước 1:** Biễu diện nội dung đối tượng khuyến nghị *pP*, ký hiệu *Content(p)*.
* **Bước 2:** Mô hình hóa sở thích người dùng *uU*, gọi tắt là hồ sơ người dùng (User’s Profile), ký hiệu *UserProfile(u)*.
* **Bước 3:** Ước lượng giá trị phù hợp dựa trên độ tương tự nội dung của sản phẩm khuyến nghị *p* với hồ sơ người dùng *u*. Hệ thống sẽ ưu tiên khuyến nghị những đối tượng sản phẩm *p* có nội dung tương tự cao so với hồ sơ người dùng *p.*

#### **1.3.1.3. Phân loại các cách tiếp cận lọc dựa trên nội dung**

Phương pháp lọc dựa trên nội dung có thể được chia ra làm 2 nhóm chính:

1. Phương pháp lọc dựa trên bộ nhớ, thực hiện tính toán độ tương tự giữa *Content(p)* và *UserProfile(u)* dùng các độ do lường tương tự Consine, Euclide [5].
2. Phương pháp lọc dựa trên mô hình, với mô hình được học từ dữ liệu dùng các kỹ thuật học máy giám sát để phân các sản phẩm khuyến nghị thành những sản phẩm được người dùng quan tâm hoặc không quan tâm như: phân lớp SVM [6], phân lớp Bayesian [7] và các phương pháp xác xuất như Pazzani và Billsus [8], Mooney và Roy [9], Gemmis và đồng nghiệp [10].
3. **Tiếp cận nội dung dựa trên bộ nhớ**

Tiếp cận nội dung dựa trên bộ nhớ(hay còn gọi là phương pháp dựa trên bộ nhớ) là phương pháp thường được thực hiện với việc ước lượng mức độ phù hợp của đối tượng huyến nghị với người dùng (tức giá trị hàm phù hợp *f(u,p)*) dựa trên việc tổng hợp mức độ quan tâm *u* đối với tập k đối tượng có nội dung tương tự với *p*, ký hiệu là = {}, *P*, hoặc tổng hợp mức độ quan tâm từ tập *k* những người dùng có sở thích tương tự *u*, = {}, *U.* Tùy thuộc vào cách biểu diễn nội dung đối tượng dữ liệu và hồ sơ người dùng, chúng ta sẽ có một hàm phù hợp để tính độ tương tự và xác định tập cũng như . Thông thường, các nghiên cứu dùng mô hình không gian vector độ đo Consine để biểu diễn nội dung và tính độ tương tự giữa các đối tượng [4].

Phương pháp dựa trên bộ nhớ có những ưu điểm và nhược điểm như sau [4]:

* **Ưu điểm:**
  + Đơn giản, dễ thực hiện.
  + Chất lượng khuyến nghị thường tốt hơn do tính toán trên cả tập dữ liệu khi thực hiện khuyến nghị.
* **Nhược điểm:**
  + Tốn bộ nhớ và tốc độ xử lý chậm do phải tính toán, trên cả tập dữ liệu thực khi thực hiện khuyến nghị.
  + Không thể tổng quát hóa tập dữ liệu.

1. **Tiếp cận nội dung dựa trên mô hình**

Với phương pháp dựa trên bộ nhớ, hệ thống thường sẽ tính giá trị hàm phù hợp dựa trên các độ do như Cosine, Euclide. Đối với các phương pháp dựa trên mô hình, một mô hình sẽ được huấn luyện từ dữ liệu để phân các đối tượng khuyến nghị thành những đối tượng được người dùng quan tâm hay không quan tâm và quan tâm nhiều hay ít dùng các phương pháp học máy giám sát: phân lớp SVM [6], phân lớp Bayesian [7] và một số phương pháp xác xuất khác. Nói cách khác, mô hình huấn luyện giúp tiên đoán giá trị hàm phù hợp *f(u,p)* của đối tượng khuyến nghị đối với người dùng [4]. Chẳng hạn, phân lớp Bayesian là một phương pháp dựa trên mô hình khá phổ biến, được dùng trong khai thác dữ liệu, phân lớp Bayesian có thể dùng để ước lượng xác xuất đối tượng khuyến nghị *p* phù hợp với *u* như thế nào. Hay nói cách khác, *p* được *u* quan tâm không hay quan tâm nhiều hay ít [7].

Ví dụ, xác xuất một tài liệu *p* được một người dùng *u* nào đó quan tâm là bao nhiêu? Tức là, giá trị hàm phù hợp *f*(*u,p*) khi đó được tính dựa trên việc ước lượng xác xuất *p* thuộc lớp (u) và (u) (u quan tâm và không quan tâm đến p) là bao nhiêu, khi cho trước một tập các từ khóa mô tả tài liệu p là {}. Giá trị hàm phù hợp *f*(*u,p*) khi đó được tính như sau [4]:

*f*(*u,p*) = P(p (u)) = P(|)

Giả sử các từ khóa mô tả tài liệu là độc lập, khi đó xác xuất P(p (u)) sẽ là [4]:

P(p (u)) = P(|) = P()

Nhìn chung phương pháp dựa trên mô hình có ưu điểm và khuyến điểm như sau [4]:

* **Ưu điểm**
  + Khả năng đáp ứng tốt khi tập dữ liệu được gia tăng.
  + Một mô hình biểu diễn tốt thế giới thực sẽ giúp tránh được vấn đề khớp (overfitting) so với phương pháp dựa trên bộ nhớ.
  + Nhanh hơn so với phương pháp dựa trên bộ nhớ do không phải tính trên cả tập dữ liệu mà chỉ dựa vào mô hình đã xây dựng để khuyến nghị.
* **Khuyết điểm**
  + Phải xây dựng và cập nhật lại mô hình khi có sự thay đổi. Đây là quá trình gây tốn tài nguyên.
  + Chất lượng tiên đoán thấp hơn so với các phương pháp dựa trên bộ nhớ vì không được tính toán trên cả tập dữ liệu. Tuy nhiên, nò tùy thuộc vào chất lượng của mô hìnnh được được xây dựng có phản ánh tốt thế giới thực hay không, tức là có đúng với thực tế hay không.

#### **1.3.1.4. Ưu điểm và khuyết điểm của lọc dựa trên nội dung**

* **Ưu điểm:**
  + Là phương pháp trực quan, dễ dàng hiểu và giải thích được [11].
  + Không bị ảnh hưởng bởi khởi đầu lạnh (cold start) [11].
  + Không bị ảnh hưởng bởi vấn đề vấn đề thưa thớt dữ liệu.
  + Có thể khuyến nghị những sản phẩm mới hoặc sản phẩm không phổ biến.
  + Có thể khuyến nghị cho những người dùng có sở thích riêng.
* **Khuyết điểm:**
  + Thường gặp các khó khăn liên quan đến phân tích nội dung.
  + Gặp các vấn đề khi khuyến nghị cho người dùng mới (Khởi động lạnh).
  + Không thể đa dạng trong khuyến nghị (bao gồm các đối tượng khuyến nghị ngoài lĩnh vực quan sát).

### **1.3.2. Phương pháp lọc cộng tác**

#### **1.3.2.1. Định nghĩa**

Lọc cộng tác (tên tiếng anh là Collaborative-Filtering) là phương pháp khai thác những khía cạnh liên quan đến thói quen sử dụng sản phẩm của một nhóm người dùng có cùng sở thích trong quá khứ để đưa ra dự đoán các sản phẩm mới phù hợp với người dùng hiện tại (có thể hình dung rằng là lọc cộng tác giả định rằng những người đồng ý trong quá khứ sẽ đồng ý trong tương lai rằng họ sẽ thích các mặt hàng tương tự như các mặt hàng mà họ đã thích trong quá khứ) [12].



Hình 1. 4. Ví dụ mô hình kỹ thuật lọc cộng tác

Trong hình minh họa 1.3.2.1 ta có thể thấy phương pháp lọc cộng tác hoạt động như thế nào trên hệ thống Netflix. User A đã xem Movie A và Movie B, User B thì đã xem Movie A, Movie B và Movie C và giữa User A và User B đều có sự tương đồng về sở thích ở một mức độ giống nhau. Suy ra phương pháp lọc cộng tác sẽ cho rằng User A cũng sẽ có thể thích xem Movie C (Movie C là movie mà User B đã xem).

#### **1.3.2.2. Khái quát bài toán**

Cũng giống như lọc dựa trên nội dung, phương pháp lọc cộng tác cũng xây dựng một ma trận đánh giá gồm danh sách các người dùng *U =* {} và danh sách các sản phẩm *P =* {}nhằm tìm kiếm những giá trị tiên đoán độ phù hợp giữa sản phẩm *p* và người dùng *u* được gọi là ma trận A(*U*,*P*). Ma trận A có kích thước là i x j và chứa các giá trị đánh giá với i 1…N và j 1…M. Những giá trị này thể hiện mức độ phù hợp của đối tượng với người dùng . Giá trị có thể là giá trị nguyên hay thực trong 1 khoảng tùy vào bài toán. Thông thường, giá trị đánh giá mức độ phù hợp trong hầu hết hệ thống ứng dụng phổ biến nhận giá trị từ 1 (không phù hợp) đến 5 (rất phù hợp). Nếu người dùng chưa thể hiện đánh giá với đối tượng thì giá trị = và cần được thu thập hoặc tính toán [4] (Ví dụ minh họa hình 1.5).

A grid of red and black question marks

Description automatically generated

Hình 1. 5. Dấu ? là những giá trị cần tiên đoán trong ma trận đánh giá

Ý tưởng chung của phương pháp lọc cộng tác là khai thác thông tin, hành vi quá khứ của người dùng dựa trên các đánh giá sẵn có từ ma trận đánh giá để tiên đoán, lượng hóa mức độ phù hợp của các đối tượng sản phẩm khuyến nghị mà người dùng chưa biết [4].

#### **1.3.2.3. Phân loại các cách tiếp cận lọc cộng tác**

Các phương pháp lọc cộng tác được phân thành hai nhóm chính [4]:

1. Lọc cộng tác với cách tiếp cận dựa trên bộ nhớ (Memory-Based) như các thuật toán tính toán lân cận, tương tự.
2. Lọc cộng tác với cách tiếp cận dựa trên mô hình (Model-Based) như các thuật toán gom cụm, phân lớp giám sát, thừa số hóa ma trận.
3. **Tiếp cận lọc cộng tác dựa trên bộ nhớ**

Các hệ thống lọc cộng tác dựa trên bộ nhớ thường dùng các kỹ thuật thống kê để tìm kiếm những người dùng, hoặc các đối tượng khuyến nghị tương tự nhau dựa trên thông tin đánh giá, hành vi quá khứ của người dùng từ ma trận đánh giá. Tiếp cận lọc cộng tác dựa trên bộ nhớ tìm cách ước lượng giá trị hàm phù hợp *f(u,p)* của đối tượng khuyến nghịp với người dùng *u* dựa trên những đánh giá của những người đồng sở thích của *u* đối với *p* (lọc dựa trên người dùng), hoặc dựa trên những đánh giá của *u* với các đối tượng khuyến nghị *p’* tương tự với *p* (lọc dựa trên đối tượng khuyến nghị). Về cơ bản, thì các thuật toán, kỹ thuật tỉnh toán cho lọc cộng tác dựa trên người dùng và lọc dựa trên đối tượng khuyến nghị từ ma trận đánh giá là tương tự nhau. Có khác chăng là kích thước của không gian người dùng và không gian đối tượng khuyến nghị sẽ ảnh hưởng đến tốc độ tính toán khi xác định nhóm các đối tượng tương tự. Phương pháp lọc cộng tác với cách tiếp cận dựa trên bộ nhớ có đặc trưng cơ bản là thường sử dụng toàn bộ dữ liệu đã có để dự đoán đánh giá của một người dùng nào đó về sản phẩm mới [4]. Cách tiếp cận dựa trên bộ nhớ thường được chia làm 2 loại: dựa trên người dùng và dựa trên sản phẩm:

* **Dựa trên người dùng**

Phương pháp này gồm 2 bước như sau:

* Bước 1: Tìm kiếm những người dùng có đánh giá tương tự với người dùng cần được dự đoán.
* Bước 2: Sử dụng đánh giá từ những người dùng được tìm thấy ở bước 1 để tính toán dự đoán cho người cần được dự đoán.
* **Dựa trên sản phẩm**

Phương pháp này gồm 2 bước như sau:

* Bước 1: Xây dựng một ma trận để xác định mối quan hệ giữa các cặp sản phẩm với nhau.
* Bước 2: Kiểm tra thị hiếu của người dùng cần dự đoán bằng cách kiểm tra ma trận và kết hợp dữ liệu của người dùng đó.

1. **Tiếp cận lọc cộng tác dựa trên mô hình**

Phương pháp lọc cộng tác với cách tiếp cận dựa trên mô hìnhchủ yếu phát triểncác mô hình bằng cách sử dụng các khai phá dữ liệu khác nhau, các thuật toán học máy để dự đoán đánh giá của người dùng về các mặt hàng chưa được đánh giá [4]. Theo quan điểm xác xuất, thì các thuật toán lọc cộng tác dựa trên mô hình cần tính toán xác suất mà người dùng *u* đánh giá cho một đối tượng khuyến nghị *p*, P(). Quá trình đó có thể xem như việc tính toán giá trị kỳ vọng cho đánh giá của người dùng *u* với đối tượng khuyến nghị *p* [13].

Khác với lọc cộng tác dựa trên bộ nhớ, các thuật toán lọc cộng tác dựa trên mô hình dùng tập các đánh giá có sẵn trong ma trận A để học một mô hình đánh giá cho mỗi người dùng. Sau đó, mô hình học được sẽ dùng để tiên đoán các đánh giá khác [4]. Một số thuật toán lọc cộng tác dựa trên mô hình được sử dụng phổ biến như Thuật toán lọc cộng tác gom cụm, Thuật toán lọc cộng tác dựa trên xác xuất Bayes [13], Thừa số hóa ma trận (Matrix Factorization)…

#### **1.3.2.4. Ưu điểm và khuyết điểm của lọc cộng tác**

* **Ưu điểm:**
  + Có khả năng dự doán sở thích và nhu cầu của người dùng để đưa ra các gợi ý sản phẩm phù hợp với từng khách hàng mà không cần hiểu sản phẩm.
  + Phù hợp với những hệ thống lớn có nhiều đánh giá từ phía người dùng.
* **Khuyết điểm:**
  + Ma trận đánh giá còn thưa.
  + Giống như lọc dựa trên nội dung, Lọc cộng tác vẫn gặp các vấn đề khi khuyến nghị cho người dùng mới (Khởi động lạnh).
  + Phương pháp này cũng không thể gợi ý được các sản phẩm mới và các sản phẩm chưa được người dùng đánh giá.
  + Độ chính xác sẽ kém nếu như sở thích của người dùng thay đổi.

### **1.3.3. Phương pháp tiếp cận lai**

Phương pháp tiếp cận lai (tên tiếng anh là Hybird Filtering) là phương pháp kết hợp các kỹ thuật khuyến nghị khác nhau [12]. Hầu hết các hương pháp tiếp cận lai đều đưa ra các dự đoán dựa trên nội dung và dựa trên cộng tác một cách riêng biệt và sau đó kết hợp chúng lại với nhau. Bằng cách thêm các tính năng của lọc dựa trên nội dung vào lọc cộng tác (có thể làm ngược lại) ta có thể đưa ra các dữ liệu thực nghiệm từ một số nghiên cứu thực nghiệm đã được chứng minh rằng hiệu suất của phương pháp tiếp cận lai có thể đưa ra các kết quả khuyến nghị chính xác hơn các phương pháp tiếp cận thuần túy [1].



Hình 1. 6. Ví dụ minh họa cho phương pháp tiếp cận lai

Trong tiếp cận lai ta có một số cách kết hợp các phương pháp như sau [1]:

* Sử dụng cả hai phương pháp lọc dựa trên nội dung và phương pháp lọc cộng tác, sau đó dùng hai kết quả thu được để quyết định:
  + Sử dụng kết quả của phương pháp nào tốt hơn (tùy thuộc vào từng thời điểm).
  + Dùng cả hai kết quả để đánh giá.
* Xây dựng hệ thống lọc dựa trên nội dung sử dụng các đặc trưng của lọc cộng tác.
* Xây dựng hệ thống lọc cộng tác sử dụng các đặc trưng của lọc dựa trên nội dung.
* Xây dựng hệ thống kết hợp cả lọc dựa trên nội dung và lọc cộng tác (Hoạt động chia làm nhiều pha, mỗi pha mỗi phương pháp hoạt động độc lập với nhau).

Ngoài ra, trong thực tế các phương pháp tiếp cận lai thường được sử dụng rất đa dạng, các phương pháp lai cho hệ khuyến nghị gồm có 7 phương pháp tiến cận lai phổ biến [14]: Lai có trọng số (Weighted Hybrid); Lai chuyển đổi (Switching Hybrid); Lai trộn (Mixed Hybrid); Lai kết hợp đặc trưng (Feature Combination Hybrid); Lai theo đợt (Cascade Hybrid); Lai tăng cường đặc trưng (Feature Augmentation Hybrid); Lai meta (Meta-Level Hybrid). Phần tiếp theo trong đồ án chuyên ngành sẽ trình bày sơ lược về các phương pháp lai đã được mêu tả trên.

1. **Lai có trọng số (Weighted Hybird)**

Mỗi phương pháp khuyến nghị phải đi tìm và xác định giá trị hàm phù hợp *f(u,p)* của một đối tượng *p* *P* với người dùng *u* *U*. Tiếp cận lai có trọng số sẽ tính toán giá trị của hàm phù hợp dựa trên kết quả của tất cả của các phương pháp khuyến nghị khác tồn tại trong hệ thống. Thông thường thì hình thức lai có trọng số đơn giản nhất là kết hợp tuyến tính các giá trị phù hợp tính được từ các phương pháp khác nhau trong hệ thống [4].

Nhìn chung ưu điểm và nhược điểm của lai có trọng số là như sau:

* **Ưu điểm:** Tất cả khả năng, phương pháp khác nhau của hệ thống được tham gia vào quá trình khuyến nghị một cách minh bạch, tự nhiên và dễ dàng thực hiện, dễ dàng điều chỉnh.
* **Nhược điểm:** Việc ước lượng trọng số lớn hay nhỏ cho phù hợp với những phương pháp khác nhau.

1. **Lai chuyển đổi (Switching Hybird)**

Các hệ thống khuyến nghị thuộc nhóm lai chuyển đổi thường sử dụng một số điều kiện để chuyển đổi qua lại giữa các phương pháp khuyến nghị khác nhau. Ví dụ: một số nghiên cứu sinh đã thực hiện nghiên cứu một hệ thống khuyến nghị sử dụng phương pháp lai chuyển đổi giữa tiếp cận nội dung và lọc cộng tác. Các tác giả đã áp dụng phương pháp lọc nội dung trước và sau đó những trường hợp là tiếp cận nội dung không thực hiện được (không thể tiếp tục thực hiện khuyến nghị và đưa ra giá trị phù hợp thấp) thì tiếp cận lọc cộng tác sẽ được áp dụng vào thay [4].

Lọc cộng tác trong phương pháp lai chuyển đổi sẽ giúp hệ thống có thể khuyến nghị được các đối tượng có nội dung, ngữ nghĩa khác với các đối tượng đã được đánh giá cao. Nói cách khác, một đối tượng có thể không được khuyến nghị với cách tiếp cận nội dung nhưng sau khi áp dụng lọc cộng tác thì đối tượng đó có thể được ưu tiên khuyến nghị [4].

Nhìn chung ưu điểm và nhược điểm của lai chuyển đổi là như sau:

* **Ưu điểm:** Phương pháp tiếp cận này rất “nhạy” với các điểm mạnh và điểm yếu của các phương pháp khác nhau.
* **Nhược điểm:** Tuy “nhạy” với điểm mạnh và điểm yếu của các phương pháp khác nhau, nhưng lai chuyển đổi yêu cầu cần phải xác định điều kiện để chuyển đổi giữa các phương pháp. Điều đó làm quá trình chuyển đổi trở nên phức tạp hơn.

1. **Lai trộn (Mixed Hybird)**

Tiếp cận lai trộn là phương pháp thực hiện các phương pháp khuyến nghị khác nhau một cách độc lập và kết hợp các kết quả từ phương pháp sẽ được chuyển thành danh sách đề xuất và được chuyển đến cho người dùng. Tiếp cận lai trộn có thể tránh được vấn đề người dùng mới (Khởi động lạnh – Cold Start). Giống như trường hợp trên, lọc dựa trên nội dung trong tiếp cận lai trộn cũng giúp đề xuất các đối tượng khuyến nghị mới (là các đối tượng vừa được khởi tạo, chưa có một đánh giá từ phía người dùng) trong danh sách sau cùng dựa trên mô tả nội dung của đối tượng này, trong khi đó phương pháp lọc cộng tác thông thường không thể làm được. Bù lại, lọc cộng tác trong lai trộn chỉ giúp đề xuất các đối tượgn khuyến nghị tiềm năng nhưng lại không tương tự về nội dung [4].

Nhìn chung ưu điểm và nhược điểm của lai trộn là như sau:

* **Ưu điểm:** Có thể giúp đề xuất các đối tượng tiềm năng mà bản thân một phương pháp riêng biệt không thể xác định được. Trộn lọc nội dung và lọc cộng tác có thể giúp giải quyết được vấn đề khởi động lạnh (Cold Start) và cũng có thể đa dạng hóa khuyến nghị.
* **Nhược điểm:** Vì tiếp cận này sử dụng nhiều đề xuất từ các phương pháp khác nhau. Do đó hệ thống cần được xử lý, lọc các đề xuất đụng độ, trùng lắp từ các phương pháp khác nhau.

1. **Lai kết hợp đặc trưng (Feature Combination Hybird)**

Lai kết hợp đặc trưng là tiếp cận phát triển phương pháp khuyến nghị bằng cách sử dụng kết hợp thông tin đánh giá của người dùng với nội dung của đối tượng khuyến nghị [4].

Ưu điểm và nhược điểm của lai kết hợp đặc trưng là như sau:

* **Ưu điểm:** Lai kết hợp đặc trưng cho phép hệ thống xem xét dữ liệu cộng tác, nhưng không chỉ phụ thuộc duy nhất vào dữ liệu cộng tác trong ma trận đánh giá. Ngược lại, hệ thống cũng có được thông tin về sự tương tự vốn có giữa các đối tượng khuyến nghị (dựa trên đặc trưng nội dung) mà không bị ảnh hưởng bởi dữ liệu cộng tác.
* **Nhược điểm:** Khó khăn trong việc xác định các đặc trưng cộng tác và đặc trưng nội dung phù hợp.

1. **Lai theo đợt (Cascade Hybrid)**

Lai theo đợt là tiếp cận mà các phương pháp khuyến nghị khác nhau được lần lượt áp dụng theo một thứ tự ưu tiên được xác định trước tùy vào mỗi ứng dụng cụ thể.Ví dụ, phương pháp khuyến nghị thứ nhất sinh ra một danh sách xếp hạng các Ứng viên (danh sách thô). Tiếp đó, những phương pháp khác với độ ưu tiên thấp hơn sẽ được áp dụng để lọc lại danh sách thô này. Lai theo đợt giúp phương pháp thứ hai tránh những đối tượng có thể không bao giờ cần khuyến nghị vì những đối tượng này đã được lọc qua phương pháp thứ nhất. Đồng thời, các đối tượng được ưu tiên chọn với phương pháp thứ nhất sẽ được tinh lọc, chứ không bị loại bỏ thông qua phương pháp thứ hai [4].

Ưu điểm và nhược điểm của lai theo đợt là như sau:

* **Ưu điểm:** So với tiếp cận lại có trọng số (Weighted Hybrid) và một số tiếp cận lai khác thì việc lọc lại danh sách thô làm cho tiếp cận này hiệu quả hơn bởi vì các phương pháp tiếp theo chỉ thực hiện lọc trên một không gian nhỏ hơn, thay vì trên cả không gian tất cả các đối tượng khuyến nghị.
* **Nhược điểm:** Khó khăn trong việc xác định độ ưu tiên giữa các phương pháp khác nhau cho mỗi ứng dụng cụ thể.

1. **Lai tăng cường đặc trưng (Feature Augmentation Hybird)**

Với tiếp cận lai tăng cường đặc trưng, phương pháp đầu sẽ học một mô hình để sinh ra đặc trưng tăng cường cho đầu vào của phương pháp tiếp theo (Nhìn chung nó khá giống lai theo đợt). Nhìn chung, lai theo đợt và lai tăng cường đặc trưng đều là những tiếp cận mà hai phương pháp khác nhau sẽ được thực hiện một cách trình tự, tức là kết quả từ phương pháp thứ nhất sẽ ảnh hưởng đến phương pháp thứ hai. Tuy nhiên, về cơ bản thì hai tiếp cận lai này là hoàn toàn khác nhau. Với lai tăng cường đặc trưng thì những đặc trưng được dùng trong phương pháp thứ hai bao gồm những đặc trưng sinh ra bởi phương pháp thứ nhất, còn đối với lai theo đợt thì phương pháp thứ hai được dùng với độ ưu tiên thấp hơn phương pháp thứ nhất, nhằm lọc lại danh sách ứng viên mà phương pháp thứ nhất đã sinh ra [4].

Ưu điểm và nhược điểm của lai tăng cường đặc trưng là như sau:

* **Ưu điểm**: Việc tăng cường đặc trưng dùng các phương pháp khác giúp hệ thống có thể cải tiến độ chính xác khuyến nghị mà không thay đổi, ảnh hưởng đến phương pháp khuyến nghị chính.
* **Nhược điểm:** Khó khăn trong việc xác định đặc trưng tăng cường phù hợp.

1. **Lai meta (Meta-Level Hybird)**

Lai meta dùng mô hình được tạo ra bởi phương pháp trước làm đầu vào cho phương pháp sau. Với lại tăng cường đặc trưng (Feature Augmentation Hybird) thì phương pháp đầu sẽ học một mô hình để sinh ra đặc trưng làm đầu vào cho phương pháp tiếp theo. Trong khi lai meta thì cả mô hình của phương pháp thứ nhất sẽ làm đầu vào cho phương pháp thứ hai. Lai meta giữa lọc nội dung và lọc cộng tác phần nào giải quyết được vẫn đề ma trận thưa trong tiếp cận lọc cộng tác bởi vì lai meta sẽ tìm kiếm những người dùng tương tự dựa trên các đặc trưng nội dung trước khi áp dụng phương pháp lọc cộng tác. Đối với những người dùng có quá ít đánh giá thì việc xác định nhóm những người đồng sở thích thông qua lọc cộng khác sẽ không được chính xác [4].

Ưu điểm và nhược điểm của lai meta là như sau:

* **Ưu điểm**: Với lai meta giữa lọc nội dung và cộng tác, phương pháp lọc cộng tác sẽ dễ dàng thực hiện tính toán trên “Dữ liệu dày” hơn so với dữ liệu thô trong ma trận đánh giá.
* **Nhược điểm:** Khó khăn trong việc chọn phương pháp để thực hiện trước. Mỗi phương pháp được chọn vẫn phải gặp những hạn chế vốn có của nó.

Tóm lại, mỗi phương pháp tiếp cận lai đều có những ưu và nhược điểm vốn có của nó. Tiếp cận lai sẽ giúp giảm bớt phần nào hạn chế của các phương pháp khác nhau và đồng thời cũng được sử dụng cho nhiều trường hợp, mục đích khác nhau.

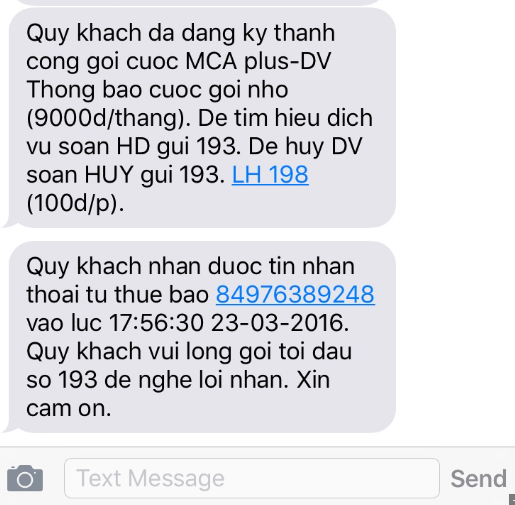
## **1.4. Khảo sát một số công trình nghiên cứu khoa học có liên quan**

### **1.4.1. Nghiên cứu và xây dựng hệ thống khuyến nghị cho bài toán dịch vụ giá trị gia tăng trong ngành viễn thông**

#### **1.4.1.1. Tổng quan về công trình nghiên cứu**

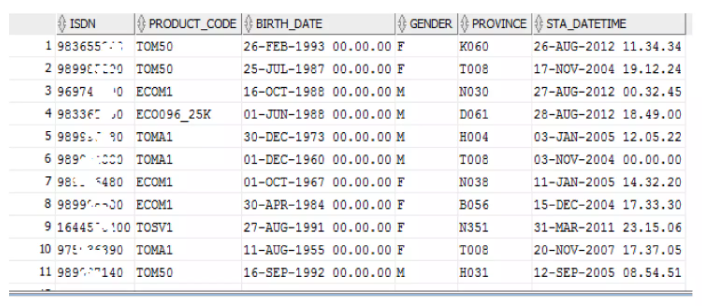
Đề tài “Nghiên cứu và xây dựng hệ thống khuyến nghị cho bài toán dịch vụ giá trị gia tăng trong ngành viễn thông” là một công trình luận văn thạc sĩ của tác giả Kiều Xuân Chấn, được viết và trình bày vào năm 2017 tại trường Đại học Công nghệ thuộc Đại học Quốc gia Hà Nội. Là một công trình nghiên cứu, xây dựng hệ thống khuyến nghị cho bài toán dịch vụ giá trị gia tăng (tên tiếng anh là VAS – Valued Added Service) ứng dụng, nghiên cứu 3 phương pháp khuyến nghị phổ biến là lọc dựa trên nội dung, lọc cộng tác và cả kết hợp cả hai phương pháp trên, cũng như là sử dụng các kỹ thuật, thuận toán nhằm giải quyết bài toán và xây dựng một hệ thống khuyến nghị dành cho bài toán giá trị dịch vụ gia tăng trong ngành điện tử viễn thông.

Vậy trước khi bắt đầu nội dung của công trình, ta cần hiểu thế nào là dịch vụ giá trị gia tăng VAS. Theo đề tài đã mô tả “*Các dịch vụ giá trị gia tăng là một thuật ngữ được sử dụng để chỉ các phụ trợ cho một dịch vụ cơ bản. Dịch vụ giá trị gia tăng thường được giới thiệu đến khách hàng sau khi khách hàng đã mua các dịch vụ cơ bản. Dịch vụ cơ bản đóng vai trò trung tâm và các dịch vụ giá trị gia tăng thường là những dịch vụ phụ thuộc vào nó*” [1] thì dịch vụ giá trị gia tăng (VAS) là một trong những dịch vụ được sinh ra nhằm xây dựng một mối quan hệ mạnh mẽ giữa khách hàng và các nhà cung cấp và các nhà cung cấp cũng xem các dịch vụ này như là một nguồn thu nhập bổ sung cho ngân sách của họ. Một trong những ví dụ dễ hình dung đó là dịch vụ thông báo cuộc gọi nhỡ MCA (Miss Call Alert) của nhà mạng Viettel và Mobifone, đây là một hệ thống cho phép thuê bao di động nhận được tin nhắn SMS thông báo thông tin về các cuộc gọi nhỡ tới số thuê bao của mình khi điện thoại di động của họ đang tắt máy, hết pin hoặc nằm ngoài vùng phủ sóng.



Hình 1. 7. Minh họa cho dịch vụ MCA của nhà mạng Viettel

Công trình nghiên cứu của tác giả Kiều Xuân Chấn đã chỉ ra rằng dịch vụ giá trị gia tăng VAS trong ngành viễn thông là một trong những dịch vụ tiện ích mang lại lợi ích lớn và mối quan hệ dài lâu của khách hàng và các nhà cung cấp, ví dụ: Tiện ích của các nhà mạng, các dịch vụ trên nền data ... Cụ thể hơn trong công trình đã chỉ ra rằng người dùng (user) trong bài toán khuyến nghị VAS chính là các thuê bao di động và thông tin (profile) của người dùng ảnh hưởng tới việc sử dụng dịch vụ đặc trưng bởi các thông tin như: Loại thuê bao, thông tin nhân thân, gói cước thuê bao, tiêu dùng hằng tháng của thuê bao, thông tin địa điểm dịch vụ, lịch sử giao dịch …



Hình 1. 8. Hình ảnh được lấy từ công trình nghiên cứu minh họa cho thông tin người dùng viễn thông

#### **1.4.1.2. Đề xuất thuật toán**

Trong công trình của mình, tác giả Kiểu Xuân Chấn đã đề cập đến một số kỹ thuật thường dùng trong hệ thống khuyến nghị cùng với lời giải thích như sau:

1. **Lọc cộng tác dựa trên bộ nhớ**
   1. **Phương pháp tự tính độ đo tương tự** là phương pháp tìm ra độ đo tương tự giữa người dùng và sản phẩm, công trình cũng đã chỉ ra một số phương pháp tính độ do tương tự như Khoảng cách manhattan, hhoảng cách Euclidean, hệ số tương quan Person, hệ số tương tự Cosine
   2. **Phương pháp K-Láng giềng gần nhất** **(KNN)** là phương pháp cổ điển phổ biến, đơn giản được sử dụng trong phương pháp lọc cộng tác dựa trên bộ nhớ và phương pháp này cũng chia ra thành 2 phương pháp cơ bản là KNN dựa trên người dùng và KNN dựa trên sản phẩm.
2. **Lọc cộng tác dựa trên mô hình:** Sử dụng dữ liệu đã đánh giá của người dùng để huấn luyện và xây dựng một mô hình đánh giá, từ đó tính toán ước lượng đánh giá của người dùng cho các sản phẩm chưa được đánh giá.
3. **Mô hình nhân tố ẩn**
   1. **Phương pháp thừa số hóa ma trận (Matrix Factorization – MF)** là một trong những phương pháp thành công nhất của mô hình nhân tố ẩn (theo tác giả thì mô hình nhân tố ẩn là mô hình biến đổi người dùng vào các mục không gian đặc trưng tiềm ẩn, có thể hiểu rằng nó xác định các yếu tố ẩn của người dùng và cả yếu tố ẩn của sản phẩm). Nó cho phép kết hợp các thông tin đã có với các thông tin bổ sung và khi thông tin phản hồi rõ ràng không có sẵn, hệ thống tư vấn có thể suy ra sở thích của người dùng bằng cách sử dụng thông tin phản hồi ngầm hoặc gián tiếp phản ánh ý kiến bằng cách quan sát hành vi người dùng.
   2. **Phương pháp sử dụng các đặc trưng ưu tiên (Biased Matrix Factorization)** là một biến thể liên quan đến phương pháp thừa số hóa ma trận. Công trình đã chỉ ra rằng “*nhiều biến thể được quan sát thấy trong các giá trị xếp hạng là do các hiệu ứng liên quan đến người dùng và sản phẩm, được gọi là đặc trưng ưu tiên và các đặc trưng này không phụ thuộc vào bất kì sự tương tác nào. Ví dụ: trong một hệ thống lớn, một số người dùng có xếp hạng cao hơn những người khác và đối với sản phẩm có xu hướng được xếp hạng cao hơn sản phẩm khác. Do đó một số sản phẩm có thể được xem là tốt hơn hoặc tệ hơn*”. Chính vì thế ta có thể hiểu rằng thành phần đặc trưng ưu tiên này là một phần không thể thiếu đối với đặc trưng của người dùng và đặc trưng của sản phẩm để mô hình hóa.
4. **Tiêu chuẩn đánh giá**
   1. **Mean absolute error (MAE)** là một phương pháp đơn giản để đo chất lượng khuyến nghị bằng cách đo lường sai số tuyệt đối trung bình (MAE), đôi khi còn được gọi là độ lệch tuyệt đối. Phương pháp này chỉ đơn giản mang ý nghĩa của sự khác biệt tuyệt đối nằm giữa dự đoán và xếp hạng cho tất cả các xếp hạng được giữ lại của người dùng trong tập kiểm tra.
   2. **Root mean square error (RMSE)** là biện pháp liên quan có ảnh hưởng của việc nhấn mạnh nhiều hơn vào các lỗi lớn. Nó được tính như MAE nhưng bình phương lỗi trước khi cộng tổng lại.
   3. **Normalized Mean absolute error (NMAE)** là phương pháp được sinh ra để giải quyết lỗi thiếu hụt của MAE và tác giả cho rằng “*MAE có cùng tỷ lệ đánh giá ban đầu, ví dụ đánh giá ở thang 5 sao được biểu diễn bằng số nguyên trong đoạn [1,5], một MAE là 0,7 có nghĩa là thuật toán trung bình bị giảm 0,7 sao. Điều này hữu ích cho việc hiểu kết quả trong một ngữ cảnh cụ thể, nhưng làm sao cho việc so sánh các kết quả trên các bộ dữ liệu rất khó khăn vì chúng có các phạm vi đánh giá khác nhau (sai số 0,7 sẽ có ý nghĩa hơn khi xếp hạng ở [1,5] hơn khi chúng ở [-10,10])”* do đó việc sai số như trên sẽ làm ảnh hưởng đến xếp hạng cho các tập dữ liệu và làm ảnh hưởng ít nhiều đến vấn đề xử lý dữ liệu. Do đó NMAE đôi khi được các nhà phát triển sử dụng để giải quyết trường hợp này, phương thức này chuẩn hóa lỗi bằng cách phân chia phạm vi xếp hạng đơn giản nhất có thể (kết quả là một giá trị trong khoảng [0,1] cho hầu hết thang đánh giá hiện tại).

Trong đề tài của mình, tác giả Kiều Xuân Chấn đã đề xuất một số giải thuật để giải quyết bài toán dịch vụ giá trị gia tăng trong ngành viễn thông, tác giả đã tập trung phân tích ,thực nghiệm và đánh giá đối với **giải pháp KNN** và **giải pháp thừa số hóa ma trận (MF)** trên tập dữ liệu mô phỏng thuê bao di động đăng kí dịch vụ VAS. Đồng thời tác giả cũng cho rằng “*KNN là phương pháp đơn giản và chạy nhanh, nó tỏ ra hiệu quả khi tập dữ liệu lớn và có nhiều thông tin. Phương pháp thừa số hóa ma trận thì có độ chính xác cao và phù hợp với các tập dữ liệu thưa*”.

#### **1.4.1.3. Kết quả thực nghiệm của công trình**

Kết quả thực nghiệm của công trình đã chỉ ra rằng “*Phương pháp KNN cho sai số RMSE rất lớn, điều đó cho thấy dữ liệu tiêu dùng thuê bao (thoại, sms, vas, data) không phải là yếu tố có giá trị đối với việc thuê bao đó đăng ký sử dụng dịch vụ VAS hay không*”. Và trong kết quả thực nghiệm của công trình đã cho rằng **Phương pháp thừa số hóa ma trận (MF) cho kết quả tốt hơn nhiều so với phương pháp KNN**. Do đó phương pháp MF phù hợp để xây dựng hệ thống khuyến nghị dịch vụ VAS.

#### **1.4.1.4. Tóm tắt công trình nghiên cứu và hướng đi cuối**

Tóm lại, Kết quả đạt được của công trình luận văn thạc sĩ của tác giả Kiều Xuân Chấn trên đã chỉ ra được tầm quan trọng của dịch vụ VAS cũng như đã chỉ ra được phương pháp tối ưu nhất cho việc xây dựng một hệ thống khuyến nghị trong ngành viễn thông. Và cũng theo tác giả, hướng đi tiếp theo cho luận văn sẽ là thử nghiệm nhiều đặc trưng của bài toán khuyến nghị VAS trên thuật toán KNN để tìm kiếm kết quả tốt hơn, đồng thời kết hợp nhiều phương pháp lọc lai cùng với deep learning để xử lý bài toán và tìm kiếm kết quả thực tế tốt hơn.

### **1.4.2. Giải quyết vấn đề phân phối trong hệ thống khuyến nghị dựa trên đặc trưng nội dung của đối tượng**

#### **1.4.2.1. Tổng quan về công trình nghiên cứu**

Đề tài “Giải quyết vấn đề phân phối trong hệ thống khuyến nghị dựa trên đặc trưng nội dung của đối tượng”là một công trình luận văn thạc sĩ của tác giả Nguyễn Văn Đạt, công trình này được viết và trình bày vào năm 2021 tại trường Đại học Công nghệ thuộc Đại học Quốc gia Hà Nội. Trong công trình của mình, tác giả đã đề cập tới 2 vấn đề trong phương pháp lọc dựa trên nội dung rằng “*Mặc dù thuật toán lọc dựa trên nội dung (Content Base – CB) là một thuật toán tốt. Tuy nhiên, trong một số trường hợp, tính chất bắt buộc khác nhau do đó kết quả gợi ý từ thuật toán lọc dựa trên nội dung vẫn chưa đáp ứng được độ chính xác cao khi bài toán liên quan đến độ tương tự về phân phối giữa các thành phần giữa các thuộc tính của đối tượng. Thêm nữa, các phương pháp để đo mức độ tương đồng giữa các sản phẩm cũng là một vấn đề quan trọng ảnh hưởng đến độ chính xác của các thuật toán lọc dựa trên nội dung trong các bài toán về độ tương đồng giữa các phân phối*”. Và để giải quyết 2 vấn đề đã được đề cập trên, tác giả của công trình đã đề xuất một thuật toán lọc dựa trên nội dung mới dựa trên mô hình hỗn hợp Gaussian(Gaussian Mixture Model – GMM) nhằm tăng độ chính xác cho đầu ra. Ngoài ra, tác giả còn đề xuất mô hình thực nghiệm trên một bộ dữ liệu về rượu bao gồm 6 mùi vị, dữ liệu tag mô tả về rượu và một số trường hợp thông tin khác.

Như vậy, công trình luận văn của tác giả hướng đến việc xây dựng một một thuật toán khuyến nghị dựa trên nội dung mới nhằm khắc phục những nhược điểm của các thuật toán lọc dựa trên nội dung đã được công bố trước đó đối với dạng đặc trưng phân phối. Đầu vào nhận bộ sản phẩm có đặc trưng chính là phân phối thuộc tính, thuật toán cần phát triển một *Bộ biểu diễn đặc trưng* thích hợp có khả năng chọn ra các đặc trưng tốt nhất làm tiền đề được sử dụng trong các thuật toán cốt lõi trong *Bộ lọc sản phẩm* [15]. Các thuật toán này cần tận dụng và phát huy tối đa được các đặc điểm đặc biệt, quan trọng của đặc trưng phân phối so với các loại đặc trưng khác như văn bản, số nguyên, dữ liệu rời rạc [15]. Đồng thời, tác giả đã khẳng định rằng “*Đối với thuật toán đề xuất, nó không chỉ hoạt động tốt trên bộ dữ liệu thực nghiệm, mà hoàn toàn còn có thể áp dụng trên các bộ dữ liệu khác có độ tương tự về đặc trưng phân phối đối với bộ dữ liệu được phân tích*” nhằm đảm bảo rằng công trình của ông có thể hoạt động tốt đối với nhiều dạng dữ liệu khác nhau.

Trong công trình của mình, tác giả cũng đã liệu kê các thuật toán mà ông đã phân tích như sau:

* Thuật toán khuyến nghị CFRS
* Thuật toán gợi ý dựa trên nội dung (CB)
* Thuật toán so sánh độ tương đồng
  + Euclidean Distance (ED)
  + Gaussian Mixture Model (GMM)
  + Word Embeddings (WE)

#### **1.4.2.2. Đề xuất thuật toán**

Trong công trình của mình, tác giả đề xuất và thu bộ dữ liệu đầu vào làm dữ liệu thực nghiệm cho bài toán. Bộ dữ liệu mà tác giả thu thập để thực nghiệm là một bộ dữ liệu về rượu, cụ thể hơn là rượu Sake và được thu thập từ bộ dữ liệu của Sakenowa (Là một trang web uy tín và nổi tiếng chuyên bán rượu sake tại Nhật). Bên cạnh đó, tác giả còn phải đánh giá lại bộ dữ liệu dựa trên đặc tính như tên, thương hiệu, năm sản xuất… và cũng đồng thời nêu ra các khó khăn trong việc chuẩn hóa bộ dữ liệu thực nghiệm cho bài toán, “*Đây thực sự là một bộ dữ liệu với nhiều thách thức, với kích thước nhỏ, thiếu dữ liệu hoặc dữ liệu không đồng đều ở một số thuộc tính dẫn đến sự rời rạc trong dữ liệu* *Cụ thể hơn, hơn 30% các giá trị trường 6 chỉ số là rỗng, ~2% không tồn tại chỉ số mùi vị tags. Thêm nữa, nhiều giá trị tags là không chính xác, không tin cậy cần được tiền xử lý và xoá bỏ nhiễu*” - theo tác giả chia sẻ. Tiếp đó, Xây dựng thành phần thuật toán gồm 3 phần chính: Bước 1 chọn các đặc trưng quan trọng để xây dựng các vector biểu diễn cho từng sản phẩm, và khi đã có 1 tập vector thì thay vì sử dụng các công thức từ thuật toán để ra luôn kết quả xử lý thì ma trận vector này dẽ được đưa vào một mô hình phân cụm GMM (Gaussian Mixture Model) để tìm ra các cụm riêng biệt, nơi mà các sản phẩm có xu hướng giống nhau về đặc trưng vector sẽ được chọn. Bước 2 và cũng là bước cuối cùng sẽ trả về các kết quả gợi ý đối với mỗi sản phẩm truy vấn. Các giai đoạn thực hiện được mô tả như sau:

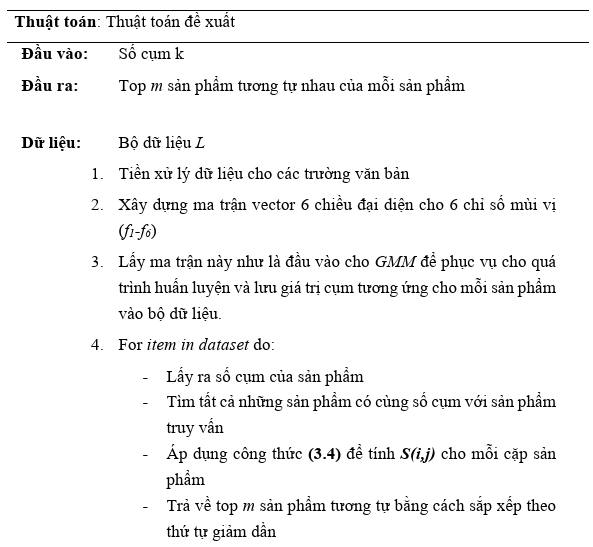
1. Tiền xử lý dữ liệu đối với dữ liệu thực nghiệm, theo tác giả đã đề cập trong công trình thì việc khai thác dữ liệu văn bản (text mining) là vô cùng quan trọng trong mọi bài toán liên quan đến văn bản, và các thuật toán lọc dựa trên nội dung cũng tương tự. Tác giả đã đề xuất phương pháp đối với bộ dữ liệu thực nghiệm của mình như sau “*thuật toán chỉ chọn tags mùi vị và 6 chỉ số mùi vị như là các đặc trưng chính cho việc tính toán mức độ tương đồng giữa các sản phẩm. Trong đó, tags mùi vị là một tập các văn bản được viết bằng tiếng Nhật và cần được làm sạch và cấu trúc lại trước khi đưa vào mô hình tính toán. Thuật toán chuyển đổi 6 chỉ số mùi vị thành số thực, và cần thực hiện một số thuật toán làm sạch và cấu trúc lại cho dữ liệu văn bản như tokenization, stemmings, stop word removal, tìm và thay thế từ đồng nghĩa, lemmatization, … trước khi sử dụng. Trong đó, trường chỉ số tags mùi vị đã được tách thành các từ có nghĩa, vì vậy, thuậ toán có thể bỏ qua bước tokenization và thực hiện các bước tiếp theo”*, việc đề xuất trên sẽ giúp cho dữ liệu được sạch hơn và tái cấu trúc giúp cho việc thao tác với dữ liệu thực nghiệm một cách dễ dàng hơn.
2. Tiếp đến là phân cụm dữ liệu, tác giả cho rằng “*các thuật toán lọc dựa trên nội cổ điển sẽ xây dựng một vector biểu diễn cho các thuộc tính được chọn của từng loại rượu, rồi tận dụng các ông thức so sánh độ tương đồng như Cosine hoặc Euclidean để sắp xếp và trả về top m kết quả*”. Tuy nhiên trong thực tế, đối khi nội dung của các bộ dữ liệu không được chính xác, có thể bị nhiễu nhiều, hoặc cũng có thể bị ảnh hưởng bởi các yếu tố ngoại lại dẫn đến việc làm ảnh hưởng đến kết quả cuối cùng và tác giả cũng nhấn mạnh rằng nếu áp dụng công thức so sánh trên toàn bộ tập dữ liệu sẽ làm giảm hiệu xuất, tốc độ tính toán của thuật toán, nhất là trong các bộ dữ liệu lớn. Do đó, đề xuất của tác giả trong trường hợp này là nhóm tất cả sản phẩm dựa theo phân phối của chúng (trong bộ dữ liệu thực nghiệm của tác giả là 6 mùi vị khác nhau) thành các nhóm riêng biệt, làm tiền đề cho giai đoạn tiếp theo.

A colorful dots and dots on a grid

Description automatically generated

Hình 1. 9. Hình ảnh được lấy từ công trình minh họa cho việc dữ liệu được phân cụm

1. Sau khi phân cụm dữ liệu ở giai đoạn trước, để trả về kết quả gợi ý cho sản phẩm truy vấn nhiệm vụ cần làm là áp dụng một thuật toán sắp xếp lên các cụm nơi các sản phẩm truy vấn được tìm ra từ mô hình huấn luyện GMM. Tác giả công trình đã phân tích bộ dữ liệu này như sau “*Đối với 6 chỉ số mùi vị, có thể nhận thấy chúng tuân theo phân phối Gaussian, tận dụng lợi thế này đồng thời dựa trên nguyên lý của hàm Gaussian Filter để tạo ra một công thức tính trọng số giữa các dãy 6 chỉ số mùi vị trong cụm. Còn đối với tags mùi vị, công thức so sánh độ tương đồng sẽ dựa trên khoảng cách Levenshtein (LD) giữa 2 chuỗi.*”. Theo đề xuất trong công trình của tác giả, ông đã đề xuất các công thức tính độ tương đồng như: 1. Độ tương đồng trong phân phối với GFF, 2. Độ tương đồng chuỗi với LD, 3. Công thức sắp xếp tổng hợp.
2. Sau đó để tường minh hơn tác giả đã đưa ra tiến trình xử lý thuật toán đề xuất dưới dạng giả mã để người đọc dễ dàng hiểu và hình dung hơn về toàn bộ thuật toán, giả mã của tác giả được trình bày ở hình dưới 1.10.



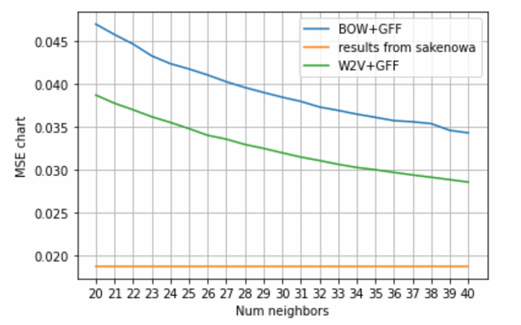
Hình 1. 10. Mô hình giả mã của tiến trình xử lý thuật toán của tác giả Nguyễn Văn Đạt

Trong thực tế, giải thuật của tác giả Nguyễn Văn Đạt hiện tại là giải thuật đang được sử dụng để giải quyết cho một hệ thống gợi ý rượu cho một công ty lớn tên là **Asian Frontier** tại Nhật Bản, điểu đó cũng là đủ để cho chúng ta thấy độ tin tưởng của thuật toán là như thế nào. Đánh giá đối với thuật toán của chính bản thân mình, ông khẳng định *“Dựa trên kết quả thực nghiệm so sánh với các thuật toán CB phổ biến, mạnh mẽ khác, thuật toán này không chỉ tốt hơn về chỉ số mùi vị trên kết quả trả về mà còn có khả năng xử lý, phản hồi người dùng nhanh hơn, đáp ứng hoàn toàn điều kiện cho một ứng dụng thời gian thực. Đặc biệt, thời gian được yêu cầu cho việc huấn luyện định kì cho thuật toán sau một khoảng thời gian các sản phẩm mới được thêm vào là nhanh, và không đáng kể so với các mô hình huấn luyện học sâu hiện tại.”* đủ để ta thấy lời nói cùng độ tin tưởng với công trình của mình là răng thép.

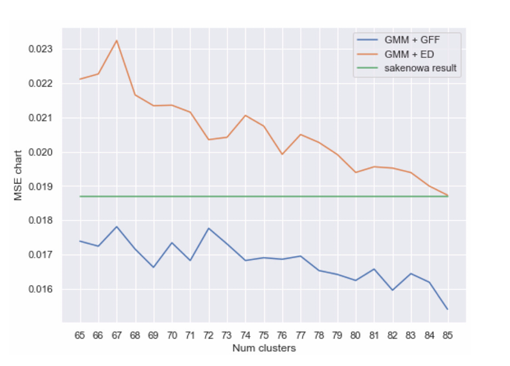
#### **1.4.2.3. Kết quả thực nghiệm của công trình**

Để chứng minh cho thuật toán đề xuất của mình hiệu quả đến như thế nào, tác giả đã đề xuất chi tiết kịch bản thử nghiệm đối với thuật toán đề xuất và 3 thuật toán lọc dựa trên nội dung phổ biến khác như BOW + GFF, W2V + GFF, và GMM + ED. Các kết quả thực nghiệm sẽ được tổng hợp, phân tích và so sánh để chứng minh độ hiệu quả, mạnh mẽ để chứng minh độ hiệu quả của thuật toán đã được tác giả đề xuất (GMM + GFF). Và để đánh giá, tác giả sử dụng phương pháp đánh giá MSE (Mean Square Error) để tính trung bình của bình phương lỗi giữa các kết quả để so sánh.

Sau cùng kết quả thực nghiệm trong công trình đã chỉ ra rằng, thuật toán đề xuất (GMM + GFF) cho ra kết quả tốt hơn 3 thuật toán lọc dựa trên nội dung phổ biến còn lại.



Hình 1. 11. Kết quả thực nghiệm từ công trình của tác giả Nguyễn Văn Đạt (MSE áp dụng BOW+GFF, W2V+GFF)



Hình 1. 12. Kết quả thực nghiệm từ công trình của tác giả Nguyễn Văn Đạt (MSE áp dụng GMM+GFF, GMM+ED)

Bên cạnh đó, thời gian xử lí trên mỗi truy vấn của thuật toán đề xuất cũng nhanh hơn 3 thuật toán còn lại:

A white square with black letters

Description automatically generated

Hình 1. 13. Thời gian thực hiện truy vấn từ công trình của tác giả Nguyễn Văn Đạt

#### **1.4.2.4. Tóm tắt công trình nghiên cứu và hướng đi cuối**

Tóm lại, công trình nghiên cứu của tác giả Nguyễn Văn Đạt đã đề xuất được một thuật toán lọc dựa trên nội dung mới có độ hiệu quả cho các bài toán gợi ý dựa tên phân phối thuộc tính trong các hệ thống khuyến nghị, mấu chốt sử dụng các đặc trưng nội dung với thuật toán GMM và nó cũng đang được áp dụng để giải quyết một hệ thống khuyến nghị rượu hiện đang được triển khai tại Nhật Bản. Thuật toán khuyến nghị dựa trên phân phối này không chỉ đạt được độ chính xác cao, mà còn đạt được tốc độ xử lý nhanh, phù hợp với các ứng dụng thực tế, song nhược điểm của thuật toán là cần huấn luyện lại mô hình định kỳ sau khi có thêm một lượng sản phẩm mới được thêm vào hệ thống. Và cũng theo tác giả Nguyễn Văn Đạt, Hướng nghiên cứu trong tương lai sau luận văn là tìm cách cải thiện mô hình GMM trong khâu phân cụm sản phẩm để đạt được kết quả tốt hơn nữ

### **1.4.3. Kết luận rút ra từ khảo sát các công trình nghiên cứu có liên quan**

Sau khi khảo sát các công trình nghiêm cứu ở 2 phần trên, kết luận đối với hệ thống khuyến nghị được rút ra rằng, các hệ thống khuyến nghị đều dựa trên các phương pháp cổ điển như lọc dựa trên nội dung (Contend Base Filtering – CB), lọc cộng tác (Colaborative Filtering – CF) và cả phương pháp lai (Hybird Filtering) để nhằm mục đích xây dựng bài toán để xử lý các vấn đề gặp phải của các hệ thống cũng như là tiền đề để xây dựng, cải tiến các bài toán khác. Bên cạnh đó các thuật toán đã được xây dựng cũng dựa vào tiền đề là các tiếp cận của các phương pháp trên. Như vậy để có thể hiểu rõ việc nghiên cứu, xây dựng hay khảo sát một hệ thống khuyến nghị hay các bài toán liên quan thì trước tiên ta phải hiểu bản chất của hệ thống khuyến nghị, các vấn đề mà các phương pháp đang gặp phải và cả các thuật toán, thuật giải, phương pháp mà những người xây dựng công trình tiên phong trước kia đã đạt được.

## **1.5. Giới thiệu bài toán khuyến nghị phim trên các trang web xem phim**

Bài toán khuyến nghị phim là một bài toán gồm các thuộc tính là sản phẩm và người dùng. Sản phẩm ở đây chính là các bộ phim và người dùng ở đây chính là những khán giả có nhu cầu thưởng thức bộ phim, và trong mỗi bộ phim đều có các thuộc tính như thể loại, thời lượng, loại phim, đánh giá, tác giả… và đối với mỗi đối tượng khán giả đều có các thuộc tính riêng biệt nhau về sở thích và nhu cầu xem phim. Bài toán khuyến nghị phim không hệ xa lạ nó đã tồn tại rất lâu trong các hệ thống xem phim nhằm hỗ trợ cho khách hàng tìm kiếm thông tin về một bộ phim mà họ yêu thích. Trong đồ án chuyên ngành này sẽ tập trung phân tích và khảo sát các phương pháp giải quyết bài toán khuyến nghị phim này, các phương pháp đó được biết đến như một thuật toán hay hệ thống nhằm giải quyết bài toán khuyến nghị phim. Và các phương pháp đó sẽ được trình bày riêng trong chương tiếp theo trong đồ án chuyên ngành này.

## **1.6. Tóm tắt chương 1**

Chương này đã trình bày tổng quan và bài toán chung của hệ thống khuyến nghị, đồng thời trình bày chi tiết các phương pháp và các hướng tiếp cận cụ thể của từng phương pháp (thông tin được trình bày được tham khảo từ một số tài liệu công trình luận văn thạc sĩ, tiến sĩ trong nước). Bên cạnh đó cũng tiến hành khảo sát cụ thể các công trình nghiên cứu luận văn thạc sĩ của 2 tác giả Kiều Xuân Chấn và Nguyễn Văn Đạt nhằm mở rộng tầm mắt về những gì làm được trong hệ thống khuyến nghị của 2 tác giả trên. Cuối cùng là giới thiệu sơ lược về chủ đề chính của đồ án chuyên ngành này đó là bài toán khuyến nghị phim.

# **CHƯƠNG 2. MỘT SỐ PHƯƠNG PHÁP GIẢI QUYẾT BÀI TOÁN KHUYẾN NGHỊ PHIM**

## **2.1. PMF**

### **2.1.1. Tổng quan**

PMF (tên tiếng anh là Probabilistic Matrix Factorization) là một phương pháp dựa trên một mô hình tuyến tính với xác xuất nhiễu Gaussian [11]. PMF sử dụng một phương pháp lọc cộng tác dựa trên xác suất để lập mô hình xếp hạng của người dùng cho các sản phẩm, trong đồ án chuyên ngành này thì sản phẩm được đề cập là phim. PMF giả định rằng ma trận xếp hạng có thể được phân tích thành hai ma trận cấp thấp hơn, một cho các đặc trưng của người dùng và một cho các đặc trưng của sản phẩm. PMF sử dụng các phân phối xác suất để biểu diễn các ma trận đặc trưng và các xếp hạng quan sát được, và sử dụng các kỹ thuật học máy để ước lượng các tham số của mô hình. PMF có thể mở rộng tốt cho các tập dữ liệu lớn và thưa thớt, và có thể cải thiện hiệu suất dự đoán bằng cách sử dụng các ưu tiên thích ứng hoặc các ràng buộc về sự tương đồng của người dùng.

Trong thực tế các phương pháp tiếp cận lọc cộng tác được sử dụng trong thực tế đều thường gặp phải vấn đề không thể xử lý các tập dữ liệu rất lớn cũng như không thể dễ dàng xử lý những người dùng có rất ít xếp hạng [16]. Trong một số nghiên cứu gần đây của tác giả Ruslan Salakhutdinov, Andriy Mnih đã chỉ ra rằng phương pháp PMF hoạt động tỷ lệ tuyến tính với số lượng quan sát và quan trọng hơn là hoạt động tốt trên tập dữ liệu Netflix Prize. Tập dữ liệu này được mô tả là rất lớn, thưa thớt và rất mất cân bằng [16]. Nhiều thuật toán lọc cộng tác hiện tại đã được áp dụng để lập mô hình xếp hạng của người dùng trên tập dữ liệu Netflix Prize bao gồm 480.189 người dùng, 17.770 phim và hơn 100 triệu lượt quan sát (bộ ba người dùng/phim/xếp hạng) [16]. Tuy nhiên, không có phương pháp nào trong số này tỏ ra đặc biệt thành công vì hai lý do: Đầu tiên, không có cách tiếp cận nào hiện tại có thể lập chính xác được mô hình xếp hạng người dùng, ngoại trừ các cách tiếp cận dựa trên hệ số ma trận vì chúng có khả năng mở rộng tốt cho các tập dữ liệu lớn; Thứ hai, hầu hết các thuật toán hiện tại đều gặp khó khăn trong việc đưa ra dự đoán chính xác cho những người dùng có rất ít xếp hạng [16]. Thực tiễn phổ biến trong hầu hết phương pháp lọc cộng tác là xóa tất cả người dùng có ít hơn một số xếp hạng tối thiểu, do đó kết quả được báo cáo trên các bộ dữ liệu tiêu chuẩn chẳng hạn như MovieLens và EachMovie trông có vẻ ấn tượng vì những trường hợp khó nhất đã được loại bỏ. Ví dụ: tập dữ liệu Netflix rất mất cân bằng, với người dùng “không thường xuyên” xếp hạng dưới 5 phim, trong khi người dùng “thường xuyên” xếp hạng trên 10.000 phim [16].

### **2.1.2. Mô hình bài toán**

#### **2.1.2.1. PMF không ràng buộc**

Giả sử chúng ta có *M* bộ phim, *N* người dùng và giá trị xếp hạng bằng số nguyên dương trong khoảng 1 đến *K*. Hãy cho rằng là giá trị biểu thị xếp hạng của người dùng *i* đối với phim *j*, và là ma trận vector đặc trưng giữa phim và người dùng, với từng cột trong vector và lần lượt biểu thị là vector đặc trưng của người dùng cụ thể *i* và phim cụ thể *j* [16]*.* Hiệu suất của mô hình PMF được đo bằng cách tính sai số bình phương trung bình gốc (gọi tắt là RMSE) trên tập kiểm tra, một số nghiên cứu đi trước [16] đã áp dụng mô hình tuyến tính xác xuất với nhiễu quan sát Gaussian. Công thức định nghĩa sự phân bố có điều kiện trên các xếp hạng được quan sát:

Trong đó là hàm mật độ xác xuất của phân bố Gaussian với giá trị trung bình với phương sai và và là hàm chỉ thị bằng 1 nếu người dùng *i* xếp hạng phim *j* và bằng 0 nếu ngược lại [16]. Một số công thức đặc trưng vector ở người dùng và phim:

Công thức tính log phân phối sau đối với đặc trưng của người dùng và phim:

Trong đó C là hằng số không phụ thuộc vào các tham số. Và cực đại hóa log-posterior (Log-posterior là một khái niệm trong thống kê Bayes, nó là hàm logarit của xác suất hậu nghiệm. Xác suất hậu nghiệm là xác suất của một sự kiện sau khi đã có thêm thông tin mới.) trên các đặc trưng phim và người dùng với các siêu tham số (tức là phương sai nhiễu quan sát và phương sai tiên nghiệm) được giữ cố định tương đương với việc tối thiểu hóa hàm mục tiêu tổng bình phương sai số với các điều kiện chuẩn hóa bậc hai như sau [16]:

Trong đó và biểu thị chuẩn Frobenius (Chuẩn Frobenius của một ma trận là căn bậc hai của tổng các bình phương của tất cả các phần tử trong ma trận đó). Bằng phương pháp gradient, một giá trị cực tiểu được sinh ra bởi công thức trên có thể được tìm thấy bằng cách giảm giá trị của *U* và *V* [16]. Trong một số nghiên cứu gần đây về về PMF, tác giả Ruslan Salakhutdinov và Andriy Mnih [16] đã đưa ra nhận định rằng thay vì sử dụng một mô hình tuyến tính Gaussian đơn giản, ta có thể tạo ra các dự đoán ngoài phạm vi của các giá trị đánh giá hợp lệ, tích vô hướng giữa các vector đặc trưng cụ thể cho người dùng và phim được truyền qua hàm logistic, giới hạn phạm vi của các dự đoán được thể hiện qua công thức sau:

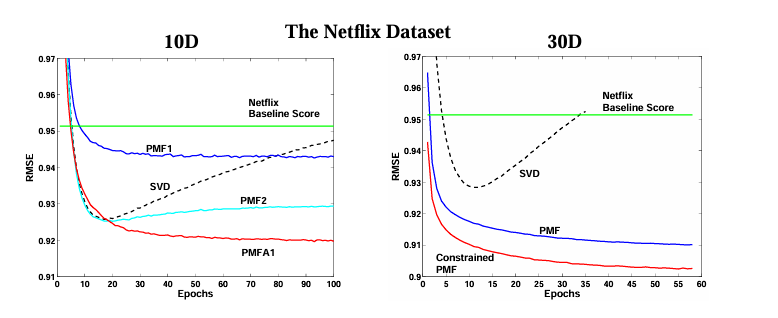
Công thức ánh xạ các đánh giá 1…K vào khoảng [0,1] bằng cách sử dụng hàm , để phạm vi của các giá trị đánh giá hợp lệ khớp với phạm vi của các dự đoán mà mô hình 2 tác giả trên tạo ra. Việc giảm thiểu hàm mục tiêu được đưa ra ở trên bằng cách sử dụng phương pháp giảm dốc cần có thời gian tuyến tính theo số lượng quan sát. Bằng cách này việc triển khai một mô hình huấn luyện trên matlab cho tập dữ liệu của netflix chỉ tốn 1 giờ cho việc huấn luyện 30 mô hình [16].

#### **2.1.2.2. PMF ràng buộc**

PMF ràng buộc là một cách bổ sung để ràng buộc các vector đặc trưng của người dùng, đặc biệt có ảnh hưởng mạnh với người dùng có ít đánh giá. Cách này giả định rằng những người dùng có rất ít đánh giá sẽ có các vector đặc trưng gần với giá trị trung bình tiên nghiệm (hay còn gọi là người dùng trung bình), do đó các đánh giá dự đoán cho những người dùng này sẽ gần với đánh giá trung bình của phim. Ta gọi là ma trận ràng buộc tương tự tiềm ẩn. Ta định nghĩa vector đặc trưng cho người dùng *i* qua công thức sau [16]:

Trong đó *I* là ma trận chỉ số quan sát với có giá trị bằng 1 nếu người dùng *i* đánh giá phim *j* và bằng 0 nếu ngược lại. Theo một cách trực quan nhất, cột thứ i của ma trận W thể hiện ảnh hưởng của việc một người dùng đã đánh giá một phim cụ thể lên giá trị trung bình tiên nghiệm của vector đặc trưng của người dùng đó. Kết quả là, những người dùng đã xem các phim giống nhau (hoặc tương tự) sẽ có các phân phối tiên nghiệm tương tự cho các vector đặc trưng của họ. Lưu ý rằng có thể được coi là độ lệch được cộng vào giá trị trung bình của phân phối tiên nghiệm để thu được vector đặc trưng cho người dùng *i*. Trong mô hình PMF không ràng buộc, và là bằng nhau vì giá trị trung bình tiên nghiệm được cố định bằng 0. Công thức xác định phân phối điều kiện các đánh giá như sau [16]:

Một số kết quả thực nghiệm của 2 tác giả Ruslan Salakhutdinov và Andriy Mnih [16] ở hình dưới đã cho thấy rằng PMF ràng buộc hoạt động tốt hơn nhiều so với PMF không ràng buộc, đặc biệt là trên những người dùng có ít đánh giá.



Hình 2. 1. Kết quả thực nghiệm thu được bằng cách sử dụng PMF (Bản số 1)

Mô tả hình 2.1: Bảng bên trái: Hiệu suất của SVD, PMF không ràng buộc sử dụng các vector đặc trưng 10D trên toàn bộ dữ liệu kiểm tra của Netflix; Bảng bên phải: Hiệu suất của SVD, PMF ràng buộc sử dụng các vector đặc trưng 30D trên dữ liệu kiểm tra của Netflix. Trục y hiển thị RMSE (sai số bình phương trung bình căn), và trục x cho biết số lần lặp, hay lượt, qua toàn bộ tập dữ liệu huấn luyện [16].

### **2.1.3. Thực nghiệm và đánh giá**

#### **2.1.3.1. Mô tả tập dữ liệu thực nghiệm**

Tập dữ liệu thực nghiệm trong đồ án chuyên ngành được tham khảo từ tập thử nghiệm trong đề tài [16] của tác giả Ruslan Salakhutdinov và Andriy Mnih nhằm mô tả tính thực nghiệm của phương pháp PMF trong dữ liệu thực. Theo tập dữ liệu đến từ Netflix, dữ liệu được thu thập từ tháng 10 năm 1998 đến tháng 12 năm 2005 và đại diện cho phân bố của tất cả các đánh giá mà Netflix nhận được trong khoảng thời gian này. Tập dữ liệu huấn luyện bao gồm 100.480.507 đánh giá từ 480.189 người dùng được chọn ngẫu nhiên và vô danh trên 17.770 tựa phim. Đồng thời, Netflix cũng cung cấp dữ liệu kiểm tra chứa khoảng 1.408.395 đánh giá. Ngoài ra, Netflix cũng cung cấp một tập dữ liệu thử nghiệm chứa 2.817.131 cặp người dùng/phim với các đánh giá bị giấu đi. Các cặp này được chọn từ các đánh giá gần đây nhất cho một tập hợp con của các người dùng trong tập dữ liệu huấn luyện. Để giảm tối thiểu trình trạng quá tải hiệu suất ngoài ý muốn cho các bộ thử nghiệm trên học máy, Netflix đã hỗ trợ sai số bình phương trung bình căn (RMSE) trên một nửa không biết của tập dữ liệu thử nghiệm. Do đó để làm mốc cho đánh giá thực nghiệm, Netflix đã cung cấp điểm số thử nghiệm của hệ thống của riêng họ được huấn luyện trên cùng một dữ liệu, là 0,9514 [16].

Và để cung cấp thêm thông tin về hiệu suất của các thuật toán khác nhau, Ruslan Salakhutdinov và Andriy Mnih đã tạo ra một tập dữ liệu nhỏ hơn và khó hơn nhiều từ dữ liệu của Netflix bằng cách chọn ngẫu nhiên 50.000 người dùng và 1.850 phim. Tập dữ liệu thực nghiệm chứa 1.082.982 cặp người dùng/phim huấn luyện và 2.462 cặp kiểm tra. Bên cạnh đó hơn 50% người dùng trong tập dữ liệu huấn luyện có ít hơn 10 đánh giá [16].

Để tăng tốc độ huấn luyện mô hình thay vì thực hiện học theo lô, Ruslan Salakhutdinov và Andriy Mnih đã chia nhỏ dữ liệu của Netflix thành các lô nhỏ có kích thước 100.000 (cặp người dùng/phim/đánh giá) và cập nhật các vector đặc trưng sau mỗi 5 lô nhỏ. Sau khi thử nghiệm nhiều giá trị khác nhau cho tốc độ học và động lượng, Ruslan Salakhutdinov và Andriy Mnih chọn sử dụng tốc độ học là 0,005 và động lượng là 0,9, vì họ cho rằng cài đặt này hoạt động tốt cho tất cả các giá trị của *D* mà họ đã thử nghiệm [16].

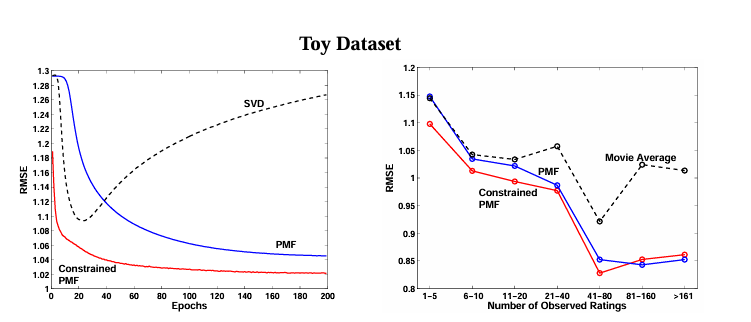
#### **2.1.3.2. Kết quả thực nghiệm của PMF không ràng buộc**

Để đánh giá hiệu suất của mô hình PMF không ràng buộc, Ruslan Salakhutdinov và Andriy Mnih [16] sử dụng các mô hình có 10D đặc trưng. Số chiều này được chọn để minh họa rằng ngay cả khi số chiều của các đặc trưng khá thấp thì các mô hình giống SVD vẫn có thể quá khớp và có thể cải thiện hiệu suất bằng cách điều chuẩn các mô hình một cách tự động. Ruslan Salakhutdinov và Andriy Mnih đã so sánh tập các mô hình gồm một mô hình SVD, hai mô hình PMF tiên nghiệm cố định, và hai mô hình PMF có tiên nghiệm thích nghi. Mô hình SVD được huấn luyện để tối thiểu hóa khoảng cách bình phương chỉ đối với các phần tử quan sát được của ma trận mục tiêu. Các vector đặc trưng của mô hình SVD không được điều chuẩn theo bất kỳ cách nào. Hai mô hình PMF không ràng buộc khác nhau về các tham số điều chuẩn: một (PMF1) có = 0,01 và = 0,001, trong khi đó (PMF2) có = 0,001 và = 0,0001. Mô hình PMF đầu tiên với tiên nghiệm thích nghi (PMFA1) có tiên nghiệm Gaussian với các ma trận hiệp phương sai cầu trên các vector đặc trưng của người dùng và phim, trong khi mô hình thứ hai (PMFA2) có các ma trận hiệp phương sai chéo. Trong cả hai trường hợp, các tiên nghiệm thích nghi có các giá trị trung bình có thể điều chỉnh. Các tham số tiên nghiệm và phương sai nhiễu được cập nhật sau mỗi 10 và 100 lần cập nhật ma trận đặc trưng tương ứng. Các mô hình được so sánh dựa trên RMSE trên tập dữ liệu kiểm tra [16].

Kết quả của việc so sánh được hiển thị trên **Hình 2.1** (bảng bên trái). Lưu ý rằng đường cong cho mô hình PMF với các ma trận hiệp phương sai cầu không được hiển thị vì nó gần như giống với đường cong cho mô hình với các ma trận hiệp phương sai chéo. So sánh các mô hình dựa trên RMSE thấp nhất đạt được trong quá trình huấn luyện, Ruslan Salakhutdinov và Andriy Mnih thấy rằng mô hình SVD hoạt động gần như tốt bằng mô hình PMF được điều chuẩn vừa phải (PMF2) (0,9258 so với 0,9253) trước khi quá khớp xấu vào cuối quá trình huấn luyện. Trong khi PMF1 không quá khớp, nó rõ ràng là thiếu khớp vì nó chỉ đạt RMSE là 0,9430. Các mô hình với tiên nghiệm thích nghi rõ ràng vượt trội so với các mô hình cạnh tranh, đạt RMSE là 0,9197 (ma trận hiệp phương sai chéo) và 0,9204 (ma trận hiệp phương sai cầu). Những kết quả này cho thấy việc điều chuẩn tự động thông qua các tiên nghiệm thích nghi hoạt động tốt trong thực tế. Hơn nữa, kết quả sơ bộ thực nghiệm của Ruslan Salakhutdinov và Andriy Mnih cho các mô hình với các vector đặc trưng có số chiều cao hơn cho thấy khoảng cách về hiệu suất do sử dụng các tiên nghiệm thích nghi có thể tăng lên khi số chiều của các vector đặc trưng tăng lên. Trong khi việc sử dụng các ma trận hiệp phương sai chéo không mang lại sự cải thiện đáng kể so với các ma trận hiệp phương sai cầu, các ma trận hiệp phương sai chéo có thể phù hợp để điều chuẩn tự động phiên bản tham lam của thuật toán huấn luyện PMF, nơi các vector đặc trưng được học theo từng chiều [16].

#### **2.1.3.3. Kết quả thực nghiệm của PMF ràng buộc**

Đối với các thí nghiệm liên quan đến mô hình PMF ràng buộc, Ruslan Salakhutdinov và Andriy Mnih [16] sử dụng các đặc trưng 30D (D = 30), vì lựa chọn này mang lại hiệu suất mô hình tốt nhất trên tập dữ liệu kiểm tra. Các giá trị của D trong khoảng [20, 60] cho kết quả tương tự. Kết quả hiệu suất của SVD, PMF không ràng buộc và PMF ràng buộc trên tập dữ liệu thực nghiệm được hiển thị trên **Hình 2.2**. Các vector đặc trưng được khởi tạo với các giá trị giống nhau trong tất cả ba mô hình. Đối với cả hai mô hình PMF không ràng buộc và PMF ràng buộc, các tham số điều chuẩn được đặt là = = = = 0.002. Rõ ràng là mô hình SVD quá phù hợp. Mô hình PMF ràng buộc hoạt động tốt hơn nhiều và hội tụ nhanh hơn rất nhiều so với mô hình PMF không ràng buộc. **Hình 2.2** (bảng bên phải) cho thấy ảnh hưởng của việc ràng buộc các đặc trưng cụ thể cho người dùng lên các dự đoán cho những người dùng ít đánh giá. Hiệu suất của mô hình PMF cho một nhóm người dùng có ít hơn 5 đánh giá trong các tập dữ liệu huấn luyện gần như giống với thuật toán trung bình phim vì luôn dự đoán điểm số tru ng bình của mỗi phim. Tuy nhiên, mô hình PMF ràng buộc hoạt động tốt hơn đáng kể đối với những người dùng có ít đánh giá. Khi số lượng đánh giá tăng lên, cả PMF không ràng buộc và PMF ràng buộc đều có hiệu suất tương tự [16].



Hình 2. 2. Kết quả thực nghiệm thu được bằng cách sử dụng PMF (Bản số 2)

Mô tả cho hình 2.2: Bảng bên trái: Hiệu suất của SVD, PMF không ràng buộc và PMF có ràng buộc trên dữ liệu xác thực. Trục y hiển thị RMSE (sai số bình phương trung bình), và trục x cho biết số lần lặp qua toàn bộ tập dữ liệu huấn luyện; Bảng bên phải: Hiệu suất của PMF có ràng buộc, PMF không ràng buộc và thuật toán trung bình phim (Movie Average) luôn dự đoán điểm số trung bình của mỗi phim. Người dùng được nhóm theo số lượng đánh giá được quan sát trong tập dữ liệu huấn luyện [16].

A graph of a number of downward-moving bars

Description automatically generated with medium confidence

Hình 2. 3. Kết quả thực nghiệm thu được bằng cách sử dụng PMF (Bản số 3)

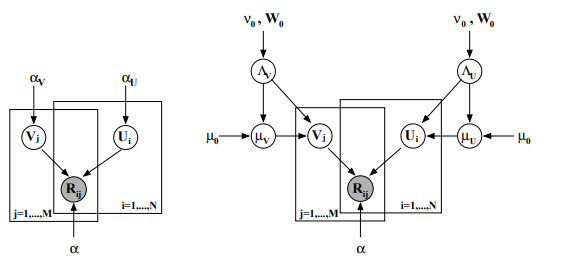
Mô tả cho hình 2.3: Bảng bên trái: Hiệu suất của PMF có ràng buộc, PMF không ràng buộc và thuật toán trung bình phim (Movie Average) luôn dự đoán điểm số trung bình của mỗi phim. Người dùng được nhóm theo số lượng đánh giá được quan sát trong tập dữ liệu huấn luyện, với trục x cho biết các nhóm đó, và trục y cho biết RMSE trên tập dữ liệu xác thực Netflix đầy đủ cho mỗi nhóm; Bảng bên giữa: Phân bố người dùng trong tập dữ liệu huấn luyện; Bảng bên phải: Hiệu suất của PMF có ràng buộc và PMF có ràng buộc sử dụng thông tin đánh giá/không đánh giá bổ sung thu được từ tập dữ liệu kiểm tra [16].

Kết quả thực nghiệm trên toàn bộ dữ liệu Netflix tương tự như kết quả trên tập dữ liệu trên hình 2.2. Đối với cả hai mô hình PMF không ràng buộc và PMF có ràng buộc, các tham số điều chuẩn được đặt là = = = = 0.001. Hình 2.1 (bảng bên phải) cho thấy PMF có ràng buộc vượt trội hơn nhiều so với mô hình PMF không có ràng buộc, đạt được RMSE là 0.9016. Một mô hình SVD đơn giản đạt được RMSE khoảng 0.9280 và sau khoảng 10 lần lặp bắt đầu quá khớp. Hình 2.3 (bảng bên trái) cho thấy mô hình PMF có ràng buộc có khả năng tổng quát hóa tốt hơn nhiều cho những người dùng có ít đánh giá. Lưu ý rằng hơn 10% người dùng trong tập dữ liệu huấn luyện có ít hơn 20 đánh giá. Bên cạnh đó netflix cho chúng ta biết trước những cặp người dùng/phim xuất hiện trong tập kiểm tra, vì vậy chúng ta có một danh mục bổ sung: những bộ phim đã được xem nhưng không biết điểm số. Đây là một nguồn thông tin quý giá về những người dùng xuất hiện nhiều lần trong tập kiểm tra, đặc biệt là nếu họ chỉ có một số lượng nhỏ đánh giá trong tập huấn luyện. Mô hình PMF có ràng buộc có thể dễ dàng sử dụng thông tin này. Hình 2.3 (bảng bên phải) cho thấy nguồn thông tin bổ sung này cải thiện thêm hiệu suất của mô hình [16].

## **2.2. BPMF**

### **2.2.1. Tổng quan**

Bayesian Probabilistic Matrix Factorization (BPMF) là một phương pháp tiếp cận Bayes hoàn toàn cho bài toán phân tích nhân tố ma trận xác suất (PMF). PMF là một mô hình hóa sở thích của người dùng dựa trên tích vô hướng của hai vector đặc trưng người dùng và sản phẩm. BPMF khác với PMF ở chỗ nó không chỉ tối đa hóa hậu nghiệm trên các vector đặc trưng với các siêu tham số cố định mà còn đặt các tiên nghiệm liên hợp cho các vector đặc trưng và các siêu tham số và sử dụng phương pháp Markov chain Monte Carlo (MCMC) để tính toán xấp xỉ phân phối hậu nghiệm. BPMF có thể kiểm soát tự động khả năng của mô hình và cung cấp phân phối dự đoán cho các giá trị đánh giá mới. BPMF được cho là có hiệu quả dự đoán cao hơn so với PMF được huấn luyện bằng ước lượng điểm tối đa hậu nghiệm (MAP).



Hình 2. 4. Hình bên trái là mộ hình đồ thị PMF, Hình bên phải là mô hình đồ thị BPMF.

### **2.2.2. Mô hình bài toán**

#### **2.2.2.1. Lý thuyết**

Mô hình đồ thị biểu diễn BPMF được thể hiện trong hình 2.4 (Hình bên phải). Cũng giống như PMF, khả năng đánh giá quan sát được biểu diễn bởi công thức bên dưới [17]:

Các phân phối tiên nghiệm trên vector đặc trưng của người dùng và phim được cho bởi 2 công thức sau [17]:

Các tiên nghiệm Gaussian-Wishart cũng được định nghĩa trên các siêu tham số và và được củng cố bằng 2 công thức bên dưới [17]:

Trong đó, W nghĩa là Wishart với bậc tự do và ma trận tỷ lệ D × D là [17]:

Trong đó, C là hằng số chuẩn hóa. Để thuận tiện cho việc nghiên cứu tác giả của nghiên cứu tác giả của công thức trên đã định nghĩa . Họ cũng đã đặt và là ma trận đơn vị cho cả hai siêu tham số người dùng và phim và đồng thời chọn theo tính đối xứng [17].

Giả định: Ta có phân phối dự đoán của giá trị đánh giá cho người dùng *i* và phim *j* được thu được bằng cách tích phân theo các tham số mô hình và siêu tham số theo công thức như sau [17]:

Tuy nhiên,việc đánh giá chính xác phân phối dự đoán này là không khả thi phân tích do sự phức tạp của hậu nghiệm, chúng ta cần phải sử dụng suy luận xấp xỉ. Một lựa chọn là sử dụng các **phương pháp biến thiên** để cung cấp các kịch bản xấp xỉ xác định cho các hậu nghiệm. Cụ thể, chúng ta có thể xấp xỉ hậu nghiệm thực bằng một phân phối có tính chất phân tích, với mỗi yếu tố có một dạng tham số cụ thể như phân phối Gaussian. Hậu nghiệm xấp xỉ này sẽ cho phép chúng ta xấp xỉ các tích phân trong công thức số 7 trên. Các phương pháp biến thiên đã trở thành phương pháp lựa chọn, vì chúng thường có thể mở rộng tốt cho các ứng dụng quy mô lớn. Tuy nhiên, chúng có thể tạo ra kết quả không chính xác vì chúng có xu hướng liên quan đến các xấp xỉ quá đơn giản cho hậu nghiệm [17].

Các phương pháp dựa trên MCMC (Markov Chain Monte Carlo) thì ngược lại, chúng sử dụng phép tính gần đúng Monte Carlo cho phân phối dự đoán của công thức số 7 như sau [17]:

Các mẫu {} được sinh ra bằng cách chạy một đoạn Markov có phân phối cân bằng là phân phối hậu nghiệm trên các tham số và siêu tham số mô hình . Lợi thế của các phương pháp dựa trên Monte Carlo là chúng cho kết quả chính xác khi tiến tới giới hạn. Tuy nhiên, trong thực tế, các phương pháp MCMC thường được coi là rất tốn thời gian tính toán nên việc sử dụng chúng bị giới hạn ở các vấn đề quy mô nhỏ [17].

#### **2.2.2.2. Giải thuật và mã giả**

Một trong những thuật toán MCMC (Markov Chain Monte Carlo) đơn giản nhất là thuật toán lấy mẫu Gibbs (Gibbs Sampling Algorithm), nó hoạt động bằng cách lặp lại qua các biến tiềm ẩn, lấy mẫu từng biến từ phân phối có điều kiện dựa trên các giá trị hiện tại của tất cả các biến khác. Thuật toán lấy mẫu Gibbs thường được sử dụng khi các phân phối có điều kiện này có thể được lấy mẫu một cách dễ dàng [17].

Do sử dụng các tiên nghiệm liên hợp cho các tham số và siêu tham số trong mô hình BPMF, các phân phối có điều kiện được suy ra từ phân phối hậu nghiệm rất dễ để lấy mẫu. Cụ thể hơn, phân phối có điều kiện trên vector đặc trưng người dùng , có điều kiện trên các đặc trưng phim, ma trận đánh giá người dùng quan sát *R*, và các giá trị của siêu tham số là Gaussian [17]:

Trong đó, và được định nghĩa như sau:

Lưu ý rằng phân phối có điều kiện trên ma trận đặc trưng người dùng tiềm ẩn *U* được phân tích thành tích của các phân phối có điều kiện trên các vector đặc trưng người dùng riêng lẻ như sau [17]:

Do đó, chúng ta có thể dễ dàng tăng tốc bộ lấy mẫu bằng cách lấy mẫu từ các phân phối có điều kiện này một cách song song. Sự tăng tốc có thể rất đáng kể, đặc biệt khi số lượng người dùng lớn [17].

Công thức phân phối có điều kiện trên các siêu tham số người dùng có điều kiện trên ma trận đặc trưng người dùng U được cho bởi phân phối Gaussian-Wishart như sau [17]:

Trong đó, các biến được mô tả như sau:

Mã giả cho thuật toán lấy mẫu Gibbs được mô tả như sau [17]:

|  |
| --- |
| 1. Khởi tạo các tham số mô hình {, } 2. Vòng lặp For: t = 1…, T  * Lấy mẫu siêu tham số (Công thức 12) * Với mỗi i = 1…, N mẫu đặc trưng của người dùng (Công thức 9) * Với mỗi i = 1…, N mẫu đặc trưng của phim |

### **2.2.3. Quá trình và đánh giá thực nghiệm**

#### **2.2.3.1. Mô tả tập dữ liệu thực nghiệm**

Tập dữ liệu thực nghiệm trong đồ án chuyên ngành được tham khảo từ tập thử nghiệm trong đề tài [17] của tác giả Ruslan Salakhutdinov và Andriy Mnih nhằm mô tả tính thực nghiệm của phương pháp BPMF trong dữ liệu thực và đồng thời trong phần này cũng so sánh mức độ hiệu quả giữa PMF và BPMF trong thực tế. Bộ dữ liệu được thu thập bởi Netflix, đại diện cho phân bố của tất cả các đánh giá Netflix nhận được từ tháng 10 năm 1998 đến tháng 12 năm 2005. Bộ dữ liệu huấn luyện bao gồm 100.480.507 đánh giá từ 480.189 người dùng được chọn ngẫu nhiên, ẩn danh trên 17.770 tựa phim. Bên cạnh đó, Netflix cũng cung cấp thêm dữ liệu kiểm tra, chứa 1.408.395 đánh giá. Ngoài ra, Netflix cũng cung cấp một tập dữ liệu kiểm tra chứa 2.817.131 cặp người dùng/phim với các đánh giá được giữ lại. Các cặp này được chọn từ các đánh giá gần đây nhất từ một tập hợp con của các người dùng trong tập dữ liệu huấn luyện. Để giảm tối thiểu trình trạng quá tải hiệu suất ngoài ý muốn cho các bộ thử nghiệm trên học máy, Netflix đã hỗ trợ sai số bình phương trung bình căn (RMSE) trên một nửa không biết của tập dữ liệu thử nghiệm. Do đó để làm mốc cho đánh giá thực nghiệm, Netflix đã cung cấp điểm số thử nghiệm của hệ thống của riêng họ được huấn luyện trên cùng một dữ liệu, là 0,9514.

#### **2.2.3.2. Mô tả quá trình huấn luyện mô hình PMF**

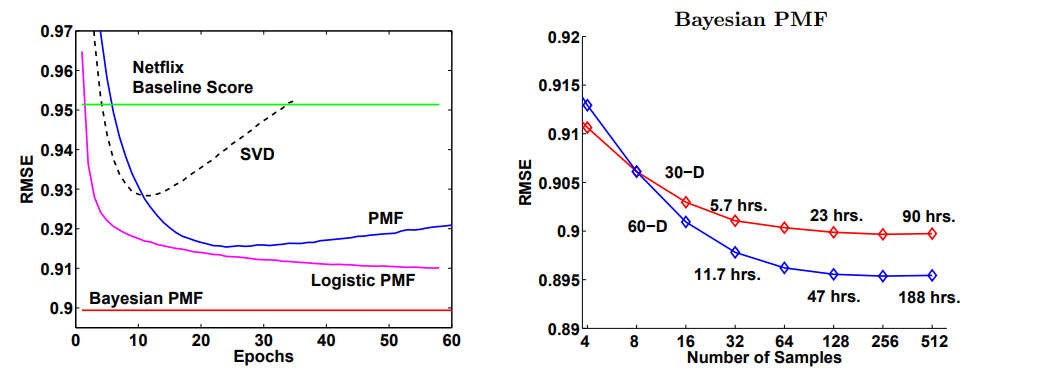
Để so sánh, Ruslan Salakhutdinov và Andriy Mnih đã huấn luyện nhiều mô hình PMF tuyến tính sử dụng MAP, chọn các tham số điều chuẩn bằng cách sử dụng tập kiểm định. Ngoài các mô hình PMF tuyến tính, Ruslan Salakhutdinov và Andriy Mnih cũng đã huấn luyện các mô hình PMF logistic, trong đó họ truyền tích vô hướng giữa các vector đặc trưng cụ thể cho người dùng và phim qua hàm logistic σ(x) = 1/(1 + exp(−x)) để giới hạn phạm vi của các dự đoán:

Trong đó, các xếp hạng từ 1 đến 5 được ánh xạ vào khoảng [0, 1] bằng cách sử dụng hàm t(x) = (x−1)/4 và để phạm vi của các giá trị xếp hạng hợp lệ khớp với phạm vi của các dự đoán mô hình của Ruslan Salakhutdinov và Andriy Mnih có thể tạo ra. Các mô hình PMF logistic có thể đôi khi cung cấp kết quả tốt hơn một chút so với các mô hình tuyến tính cùng loại. Để tăng tốc độ huấn luyện, thay vì thực hiện toàn bộ học theo lô, Ruslan Salakhutdinov và Andriy Mnih đã chia dữ liệu Netflix thành các lô nhỏ có kích thước 100.000 (bộ ba người dùng/phim/xếp hạng) và cập nhật các vector đặc trưng sau mỗi lô nhỏ. Họ sử dụng tốc độ học là 0.005 và động lượng là 0.9 để huấn luyện các mô hình PMF tuyến tính cũng như logistic.

#### **2.2.3.3. Mô tả quá trình huấn luyện mô hình BPMF**

Trong quá trình thực hiện, Ruslan Salakhutdinov và Andriy Mnih khởi tạo bộ lấy mẫu Gibbs bằng cách đặt các tham số mô hình *U* và *V* bằng các ước lượng MAP thu được bằng cách huấn luyện một mô hình PMF tuyến tính. Họ cũng đặt = 0, = D, và trong đó là ma trận đơn vị cho cả siêu tiên nghiệm của người dùng và phim. Độ chính xác của nhiễu quan sát α được cố định là 2. Phân phối dự đoán được tính toán thông qua công thức số 8 bằng cách chạy bộ lấy mẫu Gibbs với các mẫu thu thập sau mỗi bước chạy giải thuật này.

#### **2.2.3.4. Kết quả thực nghiệm thu được**

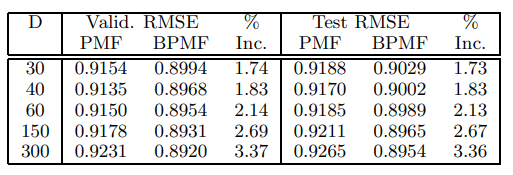


Hình 2. 5. Kết quả thực nghiệm thu được bằng cách sử dụng phương pháp BPMF

Mô tả cho hình 2.5: Hình 2. Bảng bên trái: Hiệu suất của SVD, PMF, logistic PMF và Bayesian PMF sử dụng vector đặc trưng 30D, trên dữ liệu kiểm tra của Netflix. Trục tung hiển thị RMSE (sai số bình phương trung bình), và trục hoành cho biết số lần lặp, hay số lần duyệt qua toàn bộ tập huấn luyện; Bảng bên phải: RMSE cho các mô hình Bayesian PMF trên tập dữ liệu kiểm tra theo số lượng mẫu được tạo ra. Hai đường cong là cho các mô hình có vector đặc trưng 30D và 60D.

Trong một số thí nghiệm đầu tiên, Ruslan Salakhutdinov và Andriy Mnih [17] so sánh một mô hình BPMF với một mô hình SVD, một mô hình PMF tuyến tính và một mô hình PMF logistic, tất cả sử dụng vector đặc trưng 30D. Mô hình SVD được huấn luyện để cực tiểu hóa tổng bình phương khoảng cách đến các mục quan sát của ma trận mục tiêu, không áp dụng điều chuẩn cho các vector đặc trưng. Lưu ý rằng mô hình này có thể được coi là một mô hình PMF được huấn luyện bằng phương pháp khả năng tối đa (Maximum Likelihood Method). Đối với các mô hình PMF, các tham số điều chuẩn và được đặt là 0.002. Hiệu suất dự đoán của các mô hình này trên tập kiểm định được hiển thị trong Hình 2.5 (bảng bên trái). Giá trị trung bình của phân phối dự đoán của mô hình BPMF đạt RMSE là 0.8994, so với RMSE là 0.9174 của một mô hình PMF tuyến tính có điều chuẩn vừa phải, cải thiện hơn 1.7% [17].

Mô hình PMF logistic có hiệu suất cao hơn mô hình tuyến tính, đạt RMSE là 0.90971. Tuy nhiên, hiệu suất của nó vẫn kém hơn nhiều so với mô hình BPMF. Một mô hình SVD đơn giản đạt RMSE khoảng 0.9280 và sau khoảng 10 epoch thì bắt đầu xuất hiện hiện tượng overfit nặng. Thí nghiệm này minh chứng rõ ràng rằng các mô hình PMF được huấn luyện bằng SVD và MAP có thể xảy ra hiện tượng overfit và độ chính xác dự đoán có thể được cải thiện bằng cách tích phân các tham số và siêu tham số của mô hình [17].



Hình 2. 6. Bảng dữ liệu thực nghiệm so sánh PMF và BPMF

Ruslan Salakhutdinov và Andriy Mnih [17] sau đó đã huấn luyện các mô hình PMF lớn hơn với D = 40 và D = 60. Việc kiểm soát dung lượng cho những mô hình này trở thành một nhiệm vụ khá thách thức. Ví dụ, một mô hình PMF với D = 60 có khoảng 30 triệu tham số. Do dó, việc tìm kiếm các giá trị điều chuẩn phù hợp trở nên rất tốn kém về tính toán. Bảng 2.4 cũng cho thấy rằng đối với các vector đặc trưng 60 chiều, BPMF vượt trội so với PMF MAP hơn 2%. Ruslan Salakhutdinov và Andriy Mnih đã chỉ ra rằng ngay cả phần mở rộng Bayesian đơn giản nhất cũng có thể có đặc trưng của mô hình PMF, trong đó các ưu tiên Gamma được đặt trên các siêu tham số chính xác là và (xem Hình 2.4, mô hình bên trái), do dó nó cũng có hiệu suất cao hơn nhiều so với các mô hình PMF được huấn luyện bằng MAP, mặc dù nó không hoạt động tốt bằng các mô hình BPMF [17].

Ruslan Salakhutdinov và Andriy Mnih [17] đã nhận định thông qua kết quả thực nghiệm rằng khi kích thước chiều của đặc trưng tăng lên, độ chính xác của mô hình PMF được huấn luyện bằng MAP không cải thiện và việc kiểm soát overfit trở thành một vấn đề quan trọng. Tuy nhiên, độ chính xác dự đoán của mô hình BPMF được cải thiện đều đặn khi độ phức tạp của mô hình tăng lên. Do đó, họ đã thử nghiệm với các mô hình BPMF có D = 150 và D = 300 vector đặc trưng và kết quả RMSE cho hai mô hình này trên tập kiểm định là 0.8931 và 0.8920 (kết quả từ Hình 2.6) và điều đó cho thấy những mô hình này không chỉ vượt trội hơn các mô hình PMF MAP tương ứng mà còn vượt trội hơn các mô hình BPMF có ít tham số hơn. Những kết quả này rõ ràng cho thấy rằng phương pháp Bayes không yêu cầu giới hạn độ phức tạp của mô hình dựa trên số lượng mẫu huấn luyện. Tuy nhiên trong thực tế, chúng ta thường sẽ bị giới hạn bởi tài nguyên máy tính hiện tại [17].

## **2.3. ALS**

### **2.3.1. Tổng quan và mô hình bài toán**

Alternating Least Squares (ALS) là một phương pháp lọc cộng tác, sử dụng kỹ thuật phân tích ma trận để giải quyết vấn đề quá khớp (overfitting) trong dữ liệu thưa thớt và tăng độ chính xác của dự đoán trong hệ thống gợi ý.

Trong một số hệ thống đánh giá hiện tại, dữ liệu đánh giá có thể được mô tả và biểu diễn dưới dạng một ma trận R gồm m × n phần tử, trong đó n là số người dùng và m là số sản phẩm. Phần tử trong ma trận R có nghĩa là , nó có nghĩa là đánh giá của người dùng *u* cho sản phẩm *i*. Do đó, Ma trận R là ma trận thưa, vì vậy các sản phẩm không nhận được nhiều đánh giá từ người dùng. Cho nên, ma trận R có nhiều giá trị bị thiếu nhất. Phân tích ma trận là giải pháp cho vấn đề ma trận thưa này. Có hai vector *k* chiều được gọi là “nhân tố” trong đó định nghĩa về chúng là [18]:

* là ma trận k chiều tượng trưng cho mọi người dùng *u*
* là ma trận k chiều tượng trưng cho các sản phẩm *i*

Một phương trình đã được chứng minh như sau:

Trong đó, λ được gọi là hệ số điều chuẩn và nó được sử dụng để giải quyết vấn đề quá khớp (overfitting) và còn được gọi là điều chuẩn trọng số λ. Giá trị của λ có thể được điều chỉnh để giải quyết vấn đề quá khớp, giá trị mặc định thường là 1 [18].

Giả sử rằng, tập hợp các biến là một hằng số thì hàm mục tiêu của là hàm lồi và tập hợp các biến là hằng số thì hàm mục tiêu của là hàm lồi. Do đó, giá trị tối ưu của và có thể được tìm thấy bằng cách lặp lại phương pháp đã nêu trên cho đến khi hội tụ, nói cho dễ hiểu thì ALS hoạt động bằng cách thay phiên cố định một ma trận nhân tố và giải cho ma trận nhân tố còn lại, cho đến khi hội tụ. Đây còn được gọi là phương pháp lặp xen kẽ bình phương nhỏ nhất (Alternating Least Squares - ALS) [18]

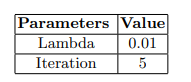
ALS có thể xử lý dữ liệu phản hồi rõ ràng (explicit feedback), nơi người dùng cung cấp đánh giá cho các sản phẩm hoặc dữ liệu phản hồi không rõ ràng (implicit feedback), nơi người dùng chỉ thể hiện sự quan tâm bằng cách xem, mua, hoặc tương tác với các sản phẩm, trong đồ án chuyên ngành này thì sản phẩm mà ta để cập đến là phim.

Mã giả của ALS được mô tả như sau [18]:

|  |
| --- |
| Thuật toán: Alternating Least Squares (ALS) |
| * Hàm thủ tục ALS(,)  1. Khởi tạo 0 2. Khởi tạo ma trận với giá trị ngẫu nhiên 3. Lặp lại cho đến số lần lặp tối đa    * Tìm kiếm và giải quyết bằng cách giảm thiểu hàm mục tiêu    * Hàm (tổng các sai số bình phương)    * Tiếp tục tìm kiếm và giải quyết bằng cách giảm thiểu hàm mục tiêu    * Lặp lại hoạt động tương tự 4. Trả về giá trị , 5. Kết thúc |

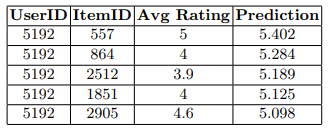
### **2.3.2. Thực nghiệm và đánh giá**

Tập dữ liệu thực nghiệm trong đồ án chuyên ngành được tham khảo từ tập thử nghiệm trong đề tài [18] của nhóm tác giả gồm Subasish Gosh, Nazmun Nahar, Mohammad Abdul Wahab, Munmun Biswas, Mohammad Shahadat Hossain và Karl Andersson. Dữ liệu thực nghiệm đến từ bộ dữ liệu của MovieLens gồm 1 triệu dataset gồm có 71567 người dùng xếp hạng và 10681 bộ phim, trong đó những người dùng trong tập dữ liệu này đã thực hiện xếp hạng cho tối thiểu 20 bộ phim khác nhau. Và có khoảng 10000054 xếp hạng có sẵn trong tập dữ liệu này. Do đó, tập dữ liệu MovieLens được cho là tập dữ liệu có phản hồi rõ ràng từ người dùng. Nhóm tác giả chia bộ dữ liệu thành các tập huấn luyện và tập kiểm tra, sử dụng thuật toán ALS để huấn luyện mô hình gợi ý và dự đoán các đánh giá còn thiếu trong tập kiểm tra. Tác giả sử dụng chỉ số RMSE để đo lường sai số giữa các đánh giá thực tế và dự đoán. Hệ thống thực nghiệm của nhóm tác giả được triển khai trên Apache Spark và cụm Apache Hadoop [18].



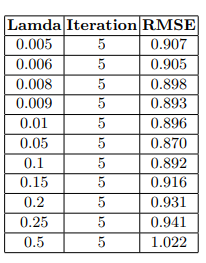
Hình 2. 7. Bảng cài đặt tham số tham khảo cho thực nghiệm

Trong quá trình xây dựng hệ thống nhóm tác giả [18] đã chia các tập dữ liệu thành 20 RDD (Viết tắt của Resilient Distributed Dataset – hay còn gọi là Tập dữ liệu phân tán linh hoạt) cho các yếu tố người dùng và các yếu tố sản phẩm, mỗi loại 10 RDD. Do đó, 2 yếu tố này được tổng hợp để tạo ra các gợi ý sản phẩm Tham số ban đầu cho. Tham số cho hệ thống được hiển thị trên hình 2.7 [18].

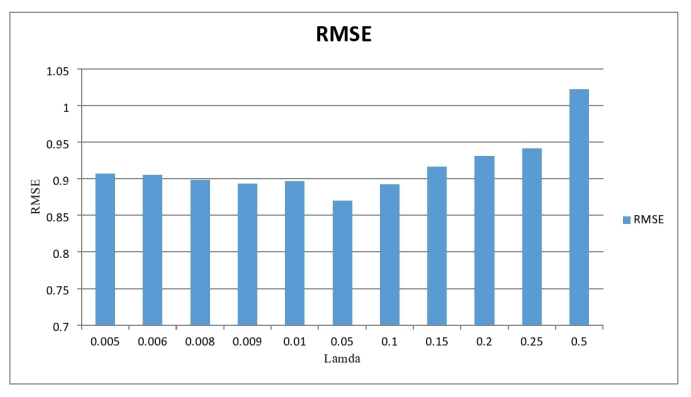


Hình 2. 8. Tập khuyến nghị sử dụng thuật toán ALS

Từ hình 2.8 cho thấy 5 gợi ý hàng đầu của sản phẩm cho một người dùng cụ thể cùng với điểm trung bình (Avg Rating) và giá trị dự đoán (Prediction) trong tập dữ liệu và giá trị thực nghiệm RMSE thu được là 0.9 đối với bảng trên [18].

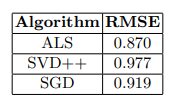


Hình 2. 9. Giá trị RMSE tương ứng với số lần lặp trong thực nghiệm

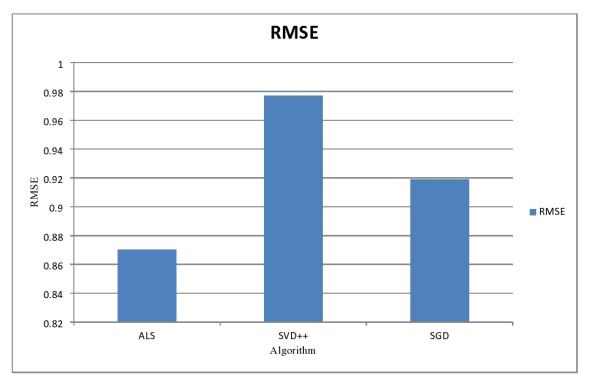


Hình 2. 10. Mô hình biễu diễn giá trị RMSE

Trong hình 2.9, quá trình huấn luyện của nhóm tác giả [18] bao gồm một bước kiểm tra ngưỡng RMSE trong đó giá trị Lamda(λ) được thay đổi để tối ưu hóa mô hình. Trong hình 2.10, bằng cách giữ nguyên số vòng lặp là 5 và thay đổi giá trị lambda trong khoảng từ 0.005 đến 1 do đó giá trị RMSE nhỏ nhất được tìm thấy là 0.870 đối với giá trị lambda là 0.05. Kết quả thực nghiệm từ hình 2.9 đã chỉ ra rằng khi giá trị lambda nằm trong khoảng từ 0.005 đến 0.006, giá trị RMSE lần lượt là 0.907 và 0.905. Và khi giá trị lambda tăng lên thì giá trị RMSE sẽ giảm so với giá trị trước đó. Sau đó, giá trị RMSE sẽ thay đổi khi giá trị lambda tăng lên và giá trị RMSE nhỏ nhất được tìm thấy khi giá trị lambda là 0.05. Giá trị RMSE nhỏ nhất là 0.870 khi số vòng lặp là 5 [18].



Hình 2. 11. So sánh giá trị RMSE giữa các phương pháp ALS, SVD++ và SGD



Hình 2. 12. Mô hình biểu diễn so sánh giá trị RMSE giữa các phương pháp ALS, SVD++ và SGD

Do đó, trong kết quả nghiên cứu của nhóm tác giả [18], họ đã đưa ra một phân tích so sánh được tiến hành để đánh giá hiệu suất của phương pháp ALS. Do đó, một sự so sánh được tiến hành giữa ALS, SVD++ và các phương pháp phân tích ma trận song song SGD. Kết quả cho thấy rằng giá trị RMSE được giảm đáng kể bằng cách sử dụng phương pháp phân tích ma trận ALS cho kết quả giá trị RMSE là 0.870. Do đó, thí nghiệm cho thấy rằng thuật toán ALS phù hợp để huấn luyện tập dữ liệu phản hồi rõ ràng nơi người dùng cung cấp đánh giá cho các sản phẩm [18].

## **2.4. Hệ thống khuyến nghị phim MOVREC**

### **2.4.1. Giới thiệu**

Ngày này, các phương pháp để giải quyết bài toán khuyến nghị phim thường rất đa dạng, một trong số đó đã được xây dựng và phát triển thành một hệ thống khuyến nghị phim hoàn chỉnh. Do đó, trong đồ án chuyên ngành này em xin trình bày một hệ thống khuyến nghị phim hoàn chỉnh được sinh ra để giải quyết bài toàn của đề tài. Hệ thống khuyến nghị phim đó có tên là MORVEC, nó được nghiên cứu và xây dựng bởi nhóm tác giả Manoj Kumar, D.K. Yadav, Ankur Singh và Vijay Kr. Gupta [19], hệ thống hoạt động dựa trên phương pháp lọc cộng tác (Collaborative Filtering) bằng cách sử dụng thông tin do người dùng cung cấp, sau đó phân tích chúng và đề xuất những bộ phim phù hợp nhất với người dùng tại thời điểm đó. Danh sách các phim đề xuất được sắp xếp theo xếp hạng của người dùng đã đánh giá trước đó và nó sử dụng thuật toán K-means cho mục đích này. MOVREC cũng giúp người dùng tìm thấy những bộ phim mà họ lựa chọn dựa trên trải nghiệm phim của những người dùng khác một cách hiệu quả và hiệu quả mà không lãng phí nhiều thời gian vào việc duyệt web vô ích.

### **2.4.2. Mô tả cấu trúc hệ thống**

#### **2.4.2.1. Tổng quan về thuật toán K-Means**

Thuật toán K-Means ban đầu được đề xuất bởi MacQueen [20]. Trong kỹ thuật phân cụm K-Means, ta chọn K là trọng tâm ban đầu trong đó K là số cụm mong muốn. Sau đó, mỗi điểm được gán cho cụm có giá trị trung bình gần nhất, tức là tâm của cụm. Tiếp đó, ta cập nhật trọng tâm của từng cụm dựa trên các điểm được gán cho cụm. Tiếp tục lặp đi lặp lại cho đến khi không có thay đổi nào ở trung tâm cụm (Centroid). Suy cho cùng, thuật toán này nhằm mục đích giảm thiểu hàm mục tiêu, trong trường hợp này là hàm lỗi bình phương. Hàm mục tiêu được mô tả bằng công thức như sau:

Trong đó, *k* được định nghĩa là số cụm, *n* là số trường hợp mà là thước đo khoảng cách được chọn giữa điểm dữ liệu và tâm cụm . Nói cách khác, chúng là chỉ báo về khoảng cách của *n* điểm dữ liệu tâm cụm tương ứng của chúng. Thuật toán K-Means được tuân thủ theo các bước sau đây:

|  |
| --- |
| 1. Chọn K làm điểm ban đầu 2. Lặp lại 3. Tìm K cụm bằng cách gán từng điểm cho trọng tâm gần nhất của nó 4. Tính lại trọng tâm của mỗi cụm 5. Cho đến khi Trung tâm cụm (Centroid) không thay đổi |

#### **2.4.2.2. Mô tả dữ liệu mô phỏng**

Dữ liệu trong mô hình đề xuất được lọc từ bộ lọc trước khi tiến hành áp dụng thuật toán K-Mean lên chúng [19]. Các thuộc tính được sử dụng để tính khoảng cách từ mỗi điểm đến tâm là:

1. Thể loại (Genre)
2. Diễn viên (Actor)
3. Director (Đạo diễn)
4. Year (Năm sản xuất)
5. Rating (Xếp hạng)

Trong quá trình nhóm tác giả [19] đã nhận định rằng các thuộc tính khác nhau có trọng số khác nhau. Và trong quá trình nghiên cứu, họ đã nhận thấy rằng những đề xuất phù hợp nhất có thể được tạo ra phải dựa trên xếp hạng của những người dùng trước đó dành cho phim. Do dó, họ đã coi trọng thuộc tính xếp hạng hơn các thuộc tính khác. Những xếp hạng này được lấy từ trang web www.imdb.com vì trang web đó có bộ sưu tập phim lớn nhất cùng với xếp hạng dành cho những bộ phim này được đánh giá bởi một số lượng lớn người dùng khác nhau từ các nơi khác nhau trên thế giới [19].

Tham số quan trọng khác trong mô hình đề xuất của nhóm tác giả [19] là tổng số phiếu bầu mà một bộ phim cụ thể nhận được. Họ đã chia số phiếu bầu thành ba loại nhỏ hơn hoặc bằng 1000, lớn hơn 1000 nhưng nhỏ hơn hoặc bằng 10.000 và lớn hơn 10.000. Và trong nghiên cứu họ [19], họ nhận thấy rằng khi số lượng phiếu bầu tăng thì trọng số xếp hạng cũng sẽ tăng tương ứng. Vì vậy, họ đã sử dụng tỷ lệ 1:1, 1:2 và 1:3 tùy thuộc vào tổng số phiếu bầu mà một bộ phim nhận được. Họ cũng nhận thấy rằng những bộ phim có xếp hạng dưới 5 là những bộ phim ít phù hợp nhất để đề xuất và ít được người dùng mong muốn nhất. Người dùng thường muốn xem một bộ phim hay và xếp hạng cao hơn và để đảm bảo rằng bộ phim được dự đoán trong dữ liệu thực nghiệm của họ là những bộ phim được nhiều người dùng yêu thích. Do đó, trọng số được gán cho các thuộc tính khác thường dựa trên mức trung bình của tổng số phim được liên kết với thuộc tính cụ thể đó với tổng số phim trong tập dữ liệu của nhóm tác giả [19].

#### **2.4.2.3. Mô phỏng quá trình hoạt động của MOVREC**

Để có thể đánh giá quá trình hoạt động của hệ thống MORVEC trong môi trường giả lập. Nhóm tác giả [19] đã giả định rằng: Khi bất kì người dùng nào vào hệ thống MOVREC họ đều có một số tùy chọn nhất định. Người dùng có thể tìm kiếm một bộ phim cụ thể hoặc xem danh sách phim đang ra mắt hoặc cũng có thể truy cập vào trang đề xuất phim. Và trên trang đề xuất, người dùng được lựa chọn hoặc nhập vào các thuộc tính giá trị khác nhau về phim mà họ muốn tìm. Và dựa trên các cơ sở dữ liệu đầu vào này, hệ thống sẽ tìm kiếm trên cơ sở dữ liệu và chuẩn bị một loạt phim phù hợp, những bộ phim được hệ thống chuẩn bị đều là những bộ phim có thuộc tính khớp với yêu cầu đầu vào của người dùng. Sau đó, hệ thống sẽ tiếp tục tính toán số lượng phim trong mảng đã chuẩn bị, nếu số lượng bộ phim nhỏ hơn hoặc bằng 20 thì hệ thống sẽ hiển thị danh sách phim được sắp xếp theo xếp hạng của phim. Còn nếu trong trường hợp số lượng phim lớn hơn 20 thì hệ thống sẽ áp dụng bộ lọc và chọn 20 phim hàng đầu theo xếp hạng, nếu có 2 phim có cùng xếp hạng (rating) giống nhau thì phim nào có số lượng bình chọn lớn sẽ được ưu tiên hiển thị trước. Sau khi lọc danh sách phim, hệ thống sẽ khớp giá trị thuộc tính với trọng số tương ứng của chúng và tính tổng trọng số của từng phim. Sau khi tính toán tổng trọng số của từng phim, hệ thống sẽ tiến hàng áp dụng thuật toán phân cụm K-mean trên nhóm phim này. Và cũng trong nghiên cứu của nhóm tác giả [19] cũng nhận thấy rằng nhìn chung người dùng thích một danh sách có 5 phim nên họ đã giả sử K bằng 4, sao cho trung bình mỗi K có 5 phim, trong đó K là số cụm được hình thành [19].

Đối với mỗi cụm K từ K1, K2, K3, K4, nhóm tác giả giả [19] sử trung tâm cụm (Centroid) ban đầu rằng C1, C2, C3, C4 tương ứng với phim thứ 1, 6, 10 và 16 trong mảng n phim. Sau khi xác định tâm ban đầu, nhóm tác giả [19] tính toán khoảng cách của tất cả các điểm dữ liệu khác từ mỗi tâm và gán các điểm dữ liệu (phim) còn lại cho các cụm hình thành và tâm gần nhất. Thước đo khoảng cách mà họ đã sử dụng để tính khoảng cách giữa các điểm dữ liệu và tâm là Khoảng cách Euclide (Euclidean Distance). Sau khi hình thành các cụm ban đầu, họ tiến hành thu thập từng cụm một. Sau đó tiếp tục tính toán trung tâm cụm (Centroid) nhưng lần này với mỗi trung tâm cụm (Centroid) tương ứng với giá trị trung bình của các điểm trong cụm đó. Sau khi tính toán lại trọng tâm, nhóm tác giả [19] tính toán khoảng cách của tất cả các điểm dữ liệu đối với các trọng tâm mới được hình thành này và gán lại chúng để tạo thành cụm. Họ lặp lại quá trình này cho đến khi không có sự thay đổi nào về trọng tâm. Điều này đảm bảo rằng các cụm được hình thành cuối cùng được tối ưu hóa và không thể phân nhóm thêm nữa. Sau khi cụm cuối cùng được hình thành, nhóm tác giả [19] tính toán xếp hạng trung bình của tất cả các điểm thuộc cụm đó, tức là xếp hạng cụm, sau đó theo truy vấn của người dùng đầu vào đồng thời hiển thị cụm có xếp hạng cao nhất.

#### **2.4.2.4. Mã giả thuật toán**

* Đầu vào (Input): số lượng phim gọi là m.
* Đầu ra (Output): số lượng cụm gọi là K.
* Bước 1: Chọn n phim từ m phim sao cho thỏa n < m.
* Bước 2: Nếu n > 20 thì chọn 20 phim hay nhất từ ​​n phim dựa trên xếp hạng. Ngược lại, hiển thị phim đầu ra được sắp xếp theo xếp hạng.
* Bước 3: Nếu xếp hạng của phim x và y bằng nhau () thì chọn phim mà có số lượng người dùng bình chọn lớn hơn.
* Bước 4: Giả sử K = 4.
* Bước 5: Lặp lại(6, 7).
* Bước 6: Bắt đầu chọn trung tâm cụm ban đầu gồm , , , .
* Bước 7: Tính khoảng cách Euclide của tất cả điểm dữ liệu , , , và tính lại trọng tâm của từng cụm.
* Bước 8: Lặp lại cho đến khi trọng tâm không thay đổi.

Trong đó:

**m**: Tổng số phim có trong cơ sở dữ liệu.

**n**: Số lượng phim sau truy vấn của người dùng.

**x, y**: 2 phim ngẫu nhiên x và y bất kì.

: Xếp hạng của phim x và y

**K**: Số lượng cụm

**, , ,** : Trung tâm cụm

**2.4.2.5. Thách thức gặp phải**

Trong bất kì quá trình phát triển hệ thống nào cũng thế, thách thức lớn nhất được đặt ra đó là làm hài lòng người dùng. Trong MOVREC cũng không ngoại lệ, nhóm tác giả [19] đã cũng phải đối mặt với những thách thức nhất định trong quá trình phát triển MOVREC của mình. Một số thách thức mà học gặp phải là [19]:

* Đảm bảo rằng hệ thống thân thiện với người dùng, dễ hiểu và dễ sử dụng.
* Đảm bảo tập dữ liệu được tạo ra phải đầy đủ thông tin cụ thể.
* Thử thách lớn nhất vẫn là đề xuất phim một cách phù hợp nhất.
* Đảm bảo hệ thống đa dạng hóa, có thể hoạt động ở nhiều nơi khác nhau, nhiều vùng địa lý, ngôn ngữ khác nhau.
* Đảm bảo trọng số được gán cho các thuộc tính là đẩy đủ.

#### **2.4.2.6. Thành công đạt được**

Bên cạnh những khó khăn khi phát triển hệ thống MOVREC mà nhóm tác giả [19] đã gặp phải thì, những thành công mà họ đạt được xứng đáng cho kì vọng của họ:

* Hệ thống đề xuất đã được thử nghiệm trên một nhóm nhỏ người và nhóm tác giả [19] đã nhận được phản hồi tích cực từ họ. Để giữ cho hệ thống thật là đơn giản và có tính tương tác cao, họ đề xuất các nhà phát triển nên chọn PHP hoặc JAVA để phát triển.
* Để có thể thu thập thông tin, nhóm tác giả [19] đã tìm kiếm kỹ lưỡng các cơ sở dữ liệu phim trực tuyến miễn phí và trích xuất thông tin hữu ích cho hệ thống được đề xuất của họ.
* Áp dụng thuật toán K-Means và Tiền xử lý dữ liệu để đảm bảo hoạt động giới thiệu phim một cách chính xác nhất.
* Đảm bảo hệ thống hoạt động ổn định trên bất kì ngôn ngữ và vị trí nào.
* Để đảm bảo cho việc gán trọng số cho các thuộc tính và ưu tiên một cách tối ưu , nhóm tác giả [19] đã tiến hành một cuộc khảo sát trên một nhóm người và trên cơ sở kết quả thu được họ đã ưu tiên các thuộc tính của mình như sau:

Bảng trích xuất một phần khảo sát của nhóm tác giả [19] được mô tả như sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tên | Diễn viên/Dàn diễn viên | Thể loại | Xếp hạng | Dạo diễn | Năm |
| Malti Singh | 2 | 5 | 1 | 4 | 3 |
| K.K. Singh | 5 | 3 | 1 | 2 | 4 |
| Kritika Baranwal | 1 | 3 | 2 | 5 | 4 |
| Ranjana Rai | 3 | 2 | 1 | 4 | 5 |
| Siddharth Singh | 2 | 3 | 1 | 4 | 5 |
| Alok Kumar | 1 | 4 | 2 | 5 | 3 |
| Shashank Pandey | 3 | 2 | 1 | 5 | 4 |
| Pankaj Agarwal | 3 | 2 | 1 | 4 | 5 |
| Girjesh Mishra | 3 | 1 | 2 | 4 | 5 |

Trong cuộc khảo sát của nhóm tác giả [19], họ nhận thấy rằng hầu hết mọi người dùng đều dành mức độ ưu tiên tối đa cho xếp hạng liên quan đến bộ phim, vì vậy họ đã dành mức ưu tiên tối đa cho thuộc tính xếp hạng trong hệ thống MOVREC. Ngoài ra, nhóm tác giả [19] đã suy luận từ cuộc khảo sát của mình rằng xếp hạng phim sẽ có sức nặng hơn nếu bộ phim đó nhận được số lượng phiếu bầu lớn. Vì vậy, nhóm tác giả [19] đã chia xếp hạng và phiếu bầu thành ba loại gồm tối thiểu, trung bình và tối đa.

## **2.5. Tóm tắt chương 2**

Chương này trong đồ án chuyên ngành đã trình bày chi tiết thông tin, bài toán, công thức và một số thực nghiêm có liên quan đến bài toán khuyến nghị phim của các phương pháp như PMF (Probabilistic Matrix Factorization), BPMF (Bayesian Probabilistic Matrix Factorization), ALS (Alternating Least Squares), ngoài ra trong phần này còn mô tả chi tiết một hệ thống khuyến nghị phim được xây dựng hoàn chỉnh là MORVEC (toàn bộ dữ liệu thực nghiệm và lý thuyết liên quan đến các phương pháp, hệ thống đều được tham khảo từ các sách và tạp chí khoa học ngoài nước),.

# **TÀI LIỆU THAM KHẢO**

# **Tiếng anh**

|  |  |  |
| --- | --- | --- |
| [2] |  | D. Jannach, M. Zanker, A. Felfernig and G. Friedrich, Recommender System: An Introduction, New York: Cambridge University Press, 2010. |
| [3] |  | A. Gediminas and A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE Transactions on Knowledge and Data Engineering, 2005. |
| [5] |  | R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval, Boston – USA: Addison Wesley Longman Publishing Co. Inc., 1999. |
| [6] |  | T. Joachims , Text categorization with Support Vector Machines: Learning with many relevant features, London, UK: In procedings of the 10th Europen Conference on Machine Learning, 1998. |
| [7] |  | K. Nigam, A. K. Mccallum, S. Thrun and T. Mitchell, Text Classification from Labeled and Unlabeled Documents using EM, Machine Learning, 2000. |
| [8] |  | M. Pazzani and D. Billsus, Learning and Revising User Profiles: The Identification of Interesting Web Sites, Machine Learning, 1997. |
| [9] |  | R. J. Mooney and L. Roy, Content-based book recommending using learning for text categorization, New York: In proceedings of the Fifth ACM Conference on Digital Libraries, 2000. |
| [10] |  | M. de Gemmis, P. Lops, G. Semeraro and P. Basile, Integrating tags in a semantic content-based recommender, New York: Proceedings of the 2008 ACM conference on Recommender systems, 2008. |
| [13] |  | J. S. Breese, D. Heckerman and C. Kadie, Empirical Analysis of Predictive Algorithms for Collaborative Filtering, Fan Francisco: In Proceeding of the Fourteenth Conference on Uncertainty in Artificial Intelligence, 1998. |
| [14] |  | R. Burke, Hybrid Recommender Systems: Survey and Experiments, User Modeling and User-adapted Interaction, 2002. |
| [16] |  | R. Salakhutdinov and . A. Mnih, Probabilistic Matrix Factorization, Canada: Department of Computer Science, University of Toronto, 2007. |
| [17] |  | R. Salakhutdinov and A. Mnih, Bayesian Probabilistic Matrix Factorization using Markov Chain Monte Carlo, Canada: Proceedings of the 25th international conference on Machine learning, 2008. |
| [18] |  | S. Gosh, N. Nahar, M. A. Wahab, M. Biswas, M. S. Hossain and K. Andersson, Recommendation System for E-commerce using Alternating Least Squares (ALS) on Apache Spark, Intelligent Computing and Optimization, Proceedings of the 3rd International Conference on Intelligent Computing and Optimization 2020, 2021. |
| [19] |  | M. Kumar, V. K. Gupta, A. Singh and D. S. Yadav, "A Movie Recommender System: MOVREC," *International Journal of Computer Applications,* vol. Volume 124 – No.3, p. 0975 – 8887, 2015. |
| [20] |  | J. Macqueen, "Some methods for classification and analysis of multivariate observations," *In Proc. Of the 5th Berkeley Symp. On Mathematical Statistics and Probability,* vol. University of California Press, pp. 281- 297, 1967. |

# **Tiếng việt**

|  |  |
| --- | --- |
| [1] | C. X. Kiều, “Nghiên cứu và xây dựng Hệ thống Khuyến nghị cho bài toán dịch vụ giá trị gia tăng trong ngành viễn thông,” Đại học Công Nghệ, Hà Nội, 2017. |
| [4] | T. N. Huỳnh, “Phát triển một số phương pháp khuyến nghị hỗ trợ tìm kiếm thông tin học thuật dựa trên tiếp cận phân tích mạng xã hội,” Trường Đại học Công nghệ thông tin, Hồ Chí Minh, 2016. |
| [11] | M. V. Bùi, “Nghiên cứu, Xây dựng Hệ thống Khuyến nghị phim tự động,” Học viện Bưu chính viễn thông, Hà Nội, 2017. |
| [12] | L. T. Đỗ, “Phát triển một số phương pháp xây dựng hệ tư vấn,” Học viện Bưu chính viễn thông, 2020, 2020. |
| [15] | Đ. V. Nguyễn, “Giải quyết vấn đề phân phối trong hệ thống khuyến nghị dựa trên đặc trưng nội dung của đối tượng,” Trường đại học Công nghệ, 2021, 2021. |