

## **LỜI CAM ĐOAN**

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi. Các kết quả được viết chung với các tác giả khác đều được sự đồng ý của đồng tác giả trước khi đưa vào luận án. Các kết quả nêu trong luận án là trung thực và chưa từng được công bố trong các công trình nào khác.

**Tác giả**

*Nguyễn Duy Phương*

## **Lời cảm ơn**

Thực hiện luận án tiến sĩ là một thử thách lớn, đòi hỏi sự kiên trì và tập trung cao độ. Tôi thực sự hạnh phúc với kết quả đạt được trong đề tài nghiên cứu của mình. Những kết quả đạt được không chỉ là nỗ lực cá nhân, mà còn có sự hỗ trợ và giúp đỡ của tập thể giáo viên hướng dẫn, nhà trường, bộ môn, đồng nghiệp và gia đình. Tôi muốn bày tỏ tình cảm của mình đến với họ.

Trước tiên, tôi xin bày tỏ sự biết ơn sâu sắc đến tập thể giáo viên hướng dẫn PGS TS Từ Minh Phương và PGS TS Đinh Mạnh Tường. Được làm việc với hai thầy là một cơ hội lớn cho tôi học hỏi phương pháp nghiên cứu. Cảm ơn hai thầy rất nhiều vì sự hướng dẫn tận tình, nghiêm túc và khoa học.

Tôi xin trân trọng cảm ơn Bộ môn Khoa học máy tính, Khoa Công nghệ thông tin, Phòng Đào tạo, Ban giám hiệu trường Đại học Công nghệ đã tạo điều kiện thuận lợi cho tôi trong suốt quá trình thực hiện luận án.

Tôi xin cảm ơn tập thể Lãnh đạo Học Viện Công nghệ Bưu chính Viễn thông, cán bộ, giảng viên khoa Công nghệ thông tin – Học Viện Công nghệ Bưu chính Viễn thông đã cổ vũ động viên tôi trong quá trình nghiên cứu.

Tôi cảm ơn tất cả những người bạn của tôi, những người luôn chia sẻ và cổ vũ tôi trong những lúc khó khăn và tôi luôn ghi nhớ điều đó.

Cuối cùng, tôi xin bày tỏ lòng biết ơn vô hạn đối với cha mẹ và gia đình đã luôn bên cạnh ủng hộ, giúp đỡ tôi.

## MỤC LỤC

PHẦN MỞ ĐẦU .....	
1. Tính cấp thiết của luận án .....	11
2. Mục tiêu của luận án .....	12
3. Các đóng góp của luận án .....	13
4. Bố cục của luận án .....	15
CHƯƠNG 1. TỔNG QUAN VỀ LỌC THÔNG TIN CHO HỆ TƯ VẤN .....	16
1.1. GIỚI THIỆU CHUNG .....	16
1.1.1. Kiến trúc tổng quát của hệ thống lọc thông tin .....	17
1.1.2. Lọc thông tin và truy vấn thông tin.....	18
1.1.3. Học máy và lọc thông tin.....	19
1.1.4. Lọc thông tin và các hệ tư vấn.....	21
1.2. PHƯƠNG PHÁP LỌC THEO NỘI DUNG.....	24
1.2.1. Bài toán lọc theo nội dung .....	25
1.2.2. Các phương pháp lọc theo nội dung.....	25
1.2.2.1. Lọc nội dung dựa vào bộ nhớ.....	25
1.2.2.2. Lọc nội dung dựa vào mô hình.....	28
1.2.3. Những vấn đề tồn tại.....	29
1.3. PHƯƠNG PHÁP LỌC CỘNG TÁC .....	30
1.3.1. Bài toán lọc cộng tác.....	30
1.3.2. Các phương pháp lọc cộng tác.....	32
1.3.2.1. Lọc cộng tác dựa trên bộ nhớ .....	32
1.3.2.2. Lọc cộng tác dựa vào mô hình .....	35
1.3.3. Những vấn đề tồn tại.....	38
1.4. PHƯƠNG PHÁP LỌC KẾT HỢP.....	39
1.4.1. Bài toán lọc kết hợp .....	39
1.4.2. Các phương pháp lọc kết hợp.....	40
1.4.3. Những vấn đề còn tồn tại.....	42
1.5. KẾT LUẬN .....	42

CHƯƠNG 2. LỌC CỘNG TÁC BẰNG PHƯƠNG PHÁP HỌC ĐA NHIỆM.....	
2.1. ĐẶT VẤN ĐỀ.....	44
2.1.1. Vấn đề dữ liệu thừa của lọc cộng tác .....	44
2.1.2. Ảnh hưởng của vấn đề dữ liệu thừa .....	45
2.1.3. Các phương pháp hạn chế vấn đề dữ liệu thừa.....	46
2.2. LỌC CỘNG TÁC BẰNG PHÂN LOẠI .....	48
2.2.1. Phát biểu bài toán lọc cộng tác bằng phân loại .....	48
2.2.2. Phân loại bằng phương pháp Boosting .....	51
2.3. PHÂN LOẠI VỚI CÁC ĐẶC TRƯNG CHUNG .....	56
2.3.1. Phương pháp học đa nhiệm .....	56
2.3.2. Boosting đồng thời cho nhiều bài toán phân loại .....	59
2.3.2.1. Xây dựng hàm mục tiêu .....	59
2.3.2.2. Xây dựng bộ phân loại yếu .....	60
2.3.2.3. Độ phức tạp thuật toán .....	63
2.4. THỬ NGHIỆM VÀ KẾT QUẢ .....	65
2.4.1. Phương pháp thử nghiệm.....	65
2.4.2. Dữ liệu thử nghiệm .....	65
2.4.3. So sánh và đánh giá dựa vào giá trị MAE .....	67
2.4.4. Kết quả thử nghiệm.....	67
2.4.5. Phân tích kết quả .....	69
2.5. KẾT LUẬN .....	72
CHƯƠNG 3. LỌC KẾT HỢP DỰA TRÊN MÔ HÌNH ĐỒ THỊ.....	
3.1. VẤN ĐỀ LỌC KẾT HỢP.....	73
3.2. LỌC CỘNG TÁC DỰA TRÊN MÔ HÌNH ĐỒ THỊ .....	75
3.2.1. Phương pháp biểu diễn đồ thị .....	75
3.2.2. Phương pháp dự đoán trên đồ thị Người dùng- Sản phẩm .....	76
3.2.2.1. Tách đồ thị Người dùng- Sản phẩm thành các đồ thị con .....	78
3.2.2.2. Phương pháp dự đoán trên đồ thị $G^+$ .....	80
3.2.2.3. Phương pháp dự đoán trên đồ thị $G^-$ .....	83

3.2.2.4. Phương pháp dự đoán theo tất cả đánh giá.....	85
3.3. KẾT HỢP LỌC CỘNG TÁC VÀ LỌC NỘI DUNG .....	88
3.3.1. Biểu diễn đồ thị kết hợp.....	88
3.3.2. Xây dựng liên kết người dùng và nội dung sản phẩm .....	91
3.3.3. Phương pháp dự đoán .....	95
3.3.3.1. Lọc cộng tác dựa trên mô hình đồ thị kết hợp.....	95
3.3.3.2. Lọc nội dung dựa trên mô hình đồ thị kết hợp .....	95
3.3.3.3. Phương pháp lọc kết hợp đơn giản.....	96
3.3.3.4. Phương pháp kết hợp đề xuất .....	96
3.3.4. Thuật toán lan truyền mạng .....	102
3.4. THỬ NGHIỆM VÀ KẾT QUẢ .....	103
3.4.1. Dữ liệu thử nghiệm .....	104
3.4.2. Phương pháp thử nghiệm.....	105
3.4.3. So sánh và đánh giá dựa vào Precision, Recall và F-measure.....	105
3.4.4. Phân tích kết quả .....	107
3.4.5. Trường hợp dữ liệu thưa.....	110
3.5. KẾT LUẬN .....	111
KẾT LUẬN.....	113
DANH MỤC CÁC CÔNG TRÌNH CÔNG BỐ.....	116
TÀI LIỆU THAM KHẢO (TIẾNG VIỆT):.....	117
TÀI LIỆU THAM KHẢO (TIẾNG ANH): .....	117
PHỤ LỤC 1 XÂY DỰNG HỆ THỐNG TƯ VẤN LỰA CHỌN PHIM DỰA TRÊN MÔ HÌNH ĐỒ THỊ KẾT HỢP.....	127

## DANH MỤC CÁC CHỮ VIẾT TẮT

KÝ HIỆU	DIỄN GIẢI
AM	Aspect Model (Mô hình định hướng)
AU	Active User (Người dùng hiện thời)
CBF	Content-Based Filtering (Lọc dựa trên nội dung)
CF	Collaborative Filtering (Lọc cộng tác)
DAC	Data Analyser Component (Thành phần phân tích dữ liệu)
DBC	Data-Based Concept (Nguyên lý dựa vào dữ liệu)
DF	Degree of Freedom (Số bậc tự do)
EM	Expectation Maximization (Cực đại kỳ vọng)
FC	Filtering Component (Thành phần lọc)
FMM	Flexible Mixture Model (Mô hình pha trộn linh hoạt)
IBL	Instance-Based Learning (Học dựa trên ví dụ)
IDF	Inverse Document Frequency (Tần suất xuất hiện ngược)
IE	Information Extraction (Tách thông tin)
IF	Information Filtering (Lọc thông tin)
IO	Information Overload (Quá tải thông tin)
IR	Information Retrieval (Truy vấn thông tin)
KNN	K Nearest Neighbor (K người láng giềng gần nhất)
KPC	KNN Pearson Correlation (Phương pháp K người láng giềng gần nhất dựa trên độ tương quan Pearson)
LC	Learning Component (Thành phần học)
LL	Lazy Learning (Học lười)
LSE	Least Square Estimation (Ước lượng bình phương tối thiểu)
LSM	Latent Semantic Model (Mô hình ngữ nghĩa ẩn)
MAE	Mean Absolute Error (Trung bình giá trị tuyệt đối lỗi)
MBF	Memory-Based Filtering (Lọc dựa vào bộ nhớ)
MC	Multiclass Classification (Phân loại nhiều lớp)
MDBF	Model-Based Filtering (Lọc dựa vào mô hình)
ML	Machine Learning (Học máy)
MM	Multinomial Model (Mô hình đa thức)

MMM	Multinomial Mixture Model (Mô hình pha trộn đa thức)
MTL	Multi Task Learning (Học đa nhiệm)
PCA	Principal Components Analysis (Phân tích thành phần chính)
RS	Recommender System (Hệ thống tư vấn)
SD	Standard Deviation (Độ lệch chuẩn)
SDP	Sparsity Data Problem (Vấn đề dữ liệu thưa)
SE	Standard Error (Lỗi chuẩn)
STL	Single Task Learning (Phương pháp học đơn lẻ)
SVD	Singular Value Decomposition (Phân rã giá trị riêng)
SVM	Support Vector Machine (Máy hỗ trợ véctơ)
TF	Term Frequency (Tần suất)
UMC	User-Model Component (Thành phần mô hình người dùng)
URP	User Rating Profile (Hồ sơ đánh giá người dùng)

## DANH MỤC CÁC HÌNH

<b>Hình 1.1.</b> Kiến trúc tổng quát của hệ thống lọc thông tin. ....	17
<b>Hình 1.2.</b> Các thành phần của hệ thống lọc cộng tác .....	31
<b>Hình 2.1.</b> Thuật toán GentleBoost. ....	52
<b>Hình 2.2.</b> Phương pháp STL cho bốn bài toán phân loại độc lập nhau.....	58
<b>Hình 2.3.</b> Phương pháp học MTL cho bốn bài toán phân loại đồng thời.....	58
<b>Hình 2.4.</b> Thuật toán MC-Boost cải tiến sử dụng đặc trưng chung cho nhiều bài toán.....	62
<b>Hình 2.5.</b> Phương pháp duyệt tập con các bài toán phân loại .....	64
<b>Hình 3.1.</b> Đồ thị Người dùng- Sản phẩm .....	76
<b>Hình 3.2.</b> Đồ thị $G^+$ biểu diễn các đánh giá thích hợp .....	79
<b>Hình 3.3.</b> Đồ thị $G^-$ biểu diễn các đánh giá không thích hợp. ....	80
<b>Hình 3.4.</b> Thuật toán dự đoán trên đồ thị $G^+$ .....	81
<b>Hình 3.5.</b> Thuật toán dự đoán trên đồ thị $G^-$ .....	84
<b>Hình 3.6.</b> Thuật toán dự đoán trên tất cả đánh giá.....	86
<b>Hình 3.7.</b> Đồ thị kết hợp người dùng và nội dung sản phẩm .....	90
<b>Hình 3.8.</b> Đồ thị thiết lập liên kết giữa người dùng và đặc trưng nội dung .....	94
<b>Hình 3.9.</b> Thuật toán dự đoán trên đồ thị kết hợp.....	99
<b>Hình 3.10.</b> Thuật toán lan truyền mạng.....	103
<b>Hình 3.11.</b> Giá trị F-Measure ở các mức độ thừa thớt dữ liệu.....	111



## DANH MỤC CÁC BẢNG

Bảng 1.1. Phân loại các phương pháp tư vấn và một số nghiên cứu điển hình...	23
Bảng 1.2. Ví dụ về ma trận đánh giá của lọc cộng tác .....	31
Bảng 2.1. Ma trận đánh giá người dùng.....	45
Bảng 2.2. Ma trận đầu vào của lọc cộng tác .....	49
Bảng 2.3. Ma trận đầu vào bài toán phân loại theo người dùng.....	50
Bảng 2.4. Ma trận đầu vào bài toán phân loại theo sản phẩm .....	50
Bảng 2.5. Kết quả thử nghiệm với MovieLens .....	68
Bảng 2.6. Kết quả thử nghiệm với EachMovie .....	68
Bảng 2.7. Các tham số thống kê với $K=5$ đánh giá biết trước.....	70
của tập dữ liệu MovieLens.....	70
Bảng 2.8. Các tham số thống kê với $K=10$ đánh giá biết trước.....	70
của tập dữ liệu MovieLens.....	70
Bảng 2.9. Các tham số thống kê với $K=20$ đánh giá biết trước.....	71
của tập dữ liệu MovieLens.....	71
Bảng 2.10. Các tham số thống kê với $K=5$ đánh giá biết trước.....	71
của tập dữ liệu EachMovie .....	71
Bảng 2.11. Các tham số thống kê với $K=10$ đánh giá biết trước .....	71
của tập dữ liệu EachMovie .....	71
Bảng 2.12. Các tham số thống kê với $K=20$ đánh giá biết trước .....	72
của tập dữ liệu EachMovie .....	72
Bảng 3.1. Ma trận đánh giá $R$ .....	74
Bảng 3.2. Ma trận Sản phẩm – Nội dung $Y$ .....	74
Bảng 3.3. Ma trận $X$ biểu diễn đánh đồ thị Người dùng- Sản phẩm .....	76
Bảng 3.4. Ma trận $X^+$ biểu diễn các đánh giá thích hợp.....	79
Bảng 3.5. Ma trận $X^-$ biểu diễn các đánh giá không thích hợp.....	80
Bảng 3.6. Ma trận đánh giá $R$ .....	89
Bảng 3.7. Ma trận Người dùng- Sản phẩm $X$ .....	89

Bảng 3.8. Ma trận Sản phẩm- Nội dung Y .....	90
Bảng 3.9. Giá trị Precision, Recall, F-Measure kiểm nghiệm trên tập MovieLens1 .....	106
Bảng 3.10. Giá trị Precision, Recall, F-Measure kiểm nghiệm trên tập MovieLens2 .....	107
Bảng 3.11. Kết quả kiểm nghiệm paired t-test với K=10 sản phẩm cần tư vấn ..... trên tập MovieLens1 .....	108
Bảng 3.12. Kết quả kiểm nghiệm paired t-test với K=20 sản phẩm cần tư vấn ..... trên tập MovieLens1 .....	109
Bảng 3.13. Kết quả kiểm nghiệm paired t-test với K=50 sản phẩm cần tư vấn ..... trên tập MovieLens1.....	109
Bảng 3.14. Kết quả kiểm nghiệm paired t-test với K=10 sản phẩm cần tư vấn ..... trên tập MovieLens2 .....	109
Bảng 3.15. Kết quả kiểm nghiệm paired t-test với K=20 sản phẩm cần tư vấn ..... trên tập MovieLens2 .....	110
Bảng 3.16. Kết quả kiểm nghiệm paired t-test với K=50 sản phẩm cần tư vấn ..... trên tập MovieLens2 .....	110

## PHẦN MỞ ĐẦU

### 1. Tính cấp thiết của luận án

Vấn đề quá tải thông tin (*Information Overload*) được J.Denning nêu ra lần đầu tiên vào năm 1982 [49]. Với những lý lẽ và bằng chứng thuyết phục, Denning khẳng định khả năng lựa chọn thông tin hữu ích của người dùng máy tính sẽ gặp khó khăn nghiêm trọng bởi sự gia tăng không ngừng lượng thông tin khổng lồ đến từ hàng trăm kênh truyền hình, hàng triệu băng hình, sách, báo, tạp chí, tài liệu thông qua các hệ thống giao dịch điện tử. Vấn đề Denning công bố ngay lập tức được cộng đồng các nhà khoa học máy tính nhiệt tình hưởng ứng và tập trung nghiên cứu phương pháp hạn chế ảnh hưởng của vấn đề quá tải thông tin đối với người dùng, thúc đẩy một lĩnh vực nghiên cứu mới đó là lọc thông tin.

Lọc thông tin (*Information Filtering*) là lĩnh vực nghiên cứu các quá trình lọc bỏ những thông tin không thích hợp và cung cấp thông tin thích hợp đến với mỗi người dùng. Lọc thông tin được xem là phương pháp hiệu quả hạn chế tình trạng quá tải thông tin được quan tâm nhiều nhất hiện nay.

Lọc thông tin được tiếp cận theo hai xu hướng chính, đó là lọc dựa trên tri thức và lọc dựa trên dữ liệu. Trong trường hợp dựa vào tri thức, hệ thống thực hiện lọc thông tin bằng cách sử dụng tập luật xây dựng trước. Nhược điểm của phương pháp này là để có được một tập luật đủ tốt đòi hỏi chi phí nhiều thời gian và kinh nghiệm của chuyên gia; việc cập nhật các luật không thể thực hiện được tự động vì nguồn dữ liệu vào thường không có cấu trúc và luôn trong trạng thái biến động. Chính vì vậy, lọc dựa trên tri thức có xu hướng ít được sử dụng.

Đối với các hệ thống lọc dựa trên dữ liệu, các quy tắc lọc được xây dựng từ dữ liệu mà hệ thống thu thập được bằng các kỹ thuật thống kê hoặc các thuật toán học máy. Cách tiếp cận này cho phép tự động cập nhật các quy tắc lọc và không lệ thuộc vào tri thức chuyên gia. Hệ thống lọc dựa trên dữ liệu có khả năng thích nghi cao và tận dụng được nguồn dữ liệu. Chính vì vậy, cách tiếp cận này được quan tâm nghiên cứu hơn so với phương pháp dựa vào tri thức.

Hệ tư vấn (*Recommender System*) là hệ thống có khả năng tự động phân tích, phân loại, lựa chọn và cung cấp cho người dùng những thông tin, hàng hóa hay dịch vụ mà họ quan tâm. Hệ tư vấn được xem như một biến thể điển hình có vai trò quan trọng trong lọc thông tin. Nhiều hệ tư vấn đã được thương mại hóa và triển khai thành công, tiêu biểu là hệ tư vấn của các hãng Amazon.com, Netflix.com, Procter & Gamble.

Hệ tư vấn được xây dựng dựa trên hai kỹ thuật lọc thông tin chính: Lọc theo nội dung (*Content-Based Filtering*) và lọc cộng tác (*Collaborative Filtering*). Lọc theo nội dung khai thác những khía cạnh liên quan đến nội dung thông tin sản phẩm người dùng đã từng sử dụng hay truy nhập trong quá khứ để tạo nên tư vấn. Trái lại, lọc cộng tác khai thác những khía cạnh liên quan đến thói quen sử dụng sản phẩm của cộng đồng người dùng có cùng sở thích để tạo nên tư vấn.

Trong quá trình nghiên cứu và ứng dụng, bên cạnh những vấn đề chung của bài toán lọc thông tin thông thường, xuất hiện một số vấn đề mang tính đặc thù đối với thông tin tư vấn như tính thừa thớt dữ liệu huấn luyện, xử lý người dùng mới, hàng hóa mới, yêu cầu kết hợp các dạng thông tin khác nhau, làm việc với dữ liệu kích thước lớn được cập nhật thường xuyên. Mặc dù đã có nhiều nghiên cứu nhắm tới nội dung này, nhưng đây vẫn là những vấn đề nghiên cứu mở, có tính thời sự và thu hút sự qua tâm của cộng đồng nghiên cứu.

Đề tài “*Phát triển một số phương pháp lọc thông tin cho hệ tư vấn*” được thực hiện trong khuôn khổ luận án tiến sĩ chuyên ngành khoa học máy tính nhằm góp phần giải quyết một số vấn đề còn tồn tại của lọc thông tin cho các hệ tư vấn.

## **2. Mục tiêu của luận án**

Mục tiêu của luận án là nghiên cứu áp dụng, cải tiến một số kỹ thuật học máy nhằm cải thiện độ chính xác của lọc thông tin trong các hệ tư vấn. Đặc biệt, nghiên cứu tập trung vào việc nâng cao kết quả dự đoán nhu cầu người dùng trong trường hợp dữ liệu thưa, cũng như trong trường hợp có cả dữ liệu sở thích người dùng và thông tin nội dung sản phẩm.

### 3. Các đóng góp của luận án

Đóng góp thứ nhất của luận án là đề xuất áp dụng một kỹ thuật Boosting cải tiến cho nhiều bài toán phân loại vào lọc cộng tác [3, 81], bao gồm:

- Đề xuất phương pháp giải quyết bài toán lọc cộng tác bằng kỹ thuật Boosting dựa trên biểu diễn dữ liệu phù hợp cho bài toán phân loại của học máy;
- Áp dụng kỹ thuật Boosting cải tiến cho nhiều bài toán phân loại bằng phương pháp học đa nhiệm dựa trên gốc quyết định (*Decision Stump*) cho lọc cộng tác nhằm hạn chế ảnh hưởng của vấn đề dữ liệu thưa;
- Thử nghiệm và đánh giá kết quả phương pháp cải tiến, đặc biệt chú trọng đánh giá kết quả dự đoán trong trường hợp dữ liệu thưa của lọc cộng tác.

Hầu hết các phương pháp học máy cho lọc cộng tác hiện nay đều thực hiện những nhiệm vụ học đơn lẻ (*Single Task Learning*) với giả thiết dữ liệu huấn luyện và dữ liệu kiểm tra được mô tả trong cùng một không gian các giá trị đặc trưng với cùng một phân bố. Khi phân bố thay đổi, tập dữ liệu huấn luyện và dữ liệu kiểm tra phải xây dựng lại. Trên thực tế, việc làm này không phải lúc nào cũng thực hiện được làm cho kết quả dự đoán các phương pháp kém tin cậy.

Mặt khác, tại mỗi thời điểm, phương pháp chỉ thực hiện một nhiệm vụ đơn lẻ, kết quả của mỗi nhiệm vụ cụ thể hoàn toàn độc lập với các nhiệm vụ khác. Chính vì vậy, phương pháp tiếp cận này sẽ gặp khó khăn khi dữ liệu huấn luyện thưa thớt. Để giải quyết vấn đề này, luận án đề xuất áp dụng phương pháp học đa nhiệm (*Multi-Task Learning*) cho lọc cộng tác nhằm sử dụng tập thông tin chung giữa các nhiệm vụ học đơn lẻ. Tập thông tin chung tìm được đóng vai trò chia sẻ và bổ sung thông tin vào quá trình huấn luyện cho mỗi người dùng khác nhau, góp phần nâng cao kết quả dự đoán và hạn chế được ảnh hưởng của tình trạng dữ liệu thưa trong lọc cộng tác.

*Đóng góp thứ hai của luận án là đề xuất một phương pháp lọc kết hợp dựa trên mô hình đồ thị [2, 80], bao gồm:*

- Biểu diễn mối liên hệ giữa các đối tượng tham gia hệ thống lọc (Người dùng, sản phẩm và nội dung sản phẩm) dựa vào mô hình đồ thị;
- Xây dựng phương pháp dự đoán cho lọc cộng tác dựa trên mô hình đồ thị.
- Xây dựng phương pháp trích chọn đặc trưng nội dung sản phẩm dựa trên thói quen sử dụng sản phẩm của người dùng;
- Cá nhân hóa ảnh hưởng của các đặc trưng nội dung đối với thói quen sử dụng sản phẩm của người dùng;
- Áp dụng thuật toán lan truyền mạng trên đồ thị kết hợp để dự đoán, phân bổ các sản phẩm cho mỗi người dùng;
- Thử nghiệm và đánh giá kết quả phương pháp đề xuất.

Để tận dụng lợi thế của mỗi phương pháp lọc, luận án đề xuất phương pháp kết hợp giữa lọc cộng tác và lọc nội dung dựa trên biểu diễn đồ thị các đối tượng tham gia quá trình lọc, bao gồm: người dùng, sản phẩm, đánh giá người dùng và nội dung sản phẩm.

Để tránh những hạn chế của các phương pháp lọc kết hợp trước đây (phương pháp trích chọn đặc trưng nội dung chỉ dựa vào nội dung sản phẩm), luận án đề xuất phương pháp trích chọn đặc trưng nội dung dựa vào thói quen người dùng đối với sản phẩm. Dựa trên phương pháp này, những đặc trưng nội dung được xem là quan trọng với mỗi người dùng được giữ lại để phục vụ mục tiêu dự đoán. Việc tìm ra những đặc trưng có ảnh hưởng quan trọng đến thói quen người dùng không chỉ làm giảm chi phí tính toán của phương pháp (vì số lượng các đặc trưng nội dung quan trọng đối với mỗi người dùng còn lại rất ít), mà còn loại bỏ được những đặc trưng không ảnh hưởng hoặc ảnh hưởng không tốt đến thói quen sử dụng sản phẩm của người dùng.

Phương pháp dự đoán được đưa về bài toán tìm kiếm trên đồ thị không chỉ tận dụng được các thuật toán hiệu quả trên đồ thị mà còn tận dụng được mối liên hệ gián tiếp giữa các đối tượng tham gia hệ thống.

Phương pháp lọc kết hợp đề xuất được thử nghiệm và áp dụng cho hệ thống tư vấn lựa chọn phim đã cho lại kết quả dự đoán tốt. Hệ thống cho phép xem, đánh giá, bình luận và gợi ý những phim được xem hợp với sở thích ứng với mỗi người dùng. Hệ thống gồm bốn chức năng chính: Chức năng cập nhật, phân tích thông tin người dùng và sản phẩm; chức năng học; chức năng lọc và chức năng tư vấn. Trong đó, chức năng học và lọc được thực hiện theo phương pháp lọc kết hợp đề xuất.

#### **4. Bố cục của luận án**

Nội dung luận án được xây dựng thành ba chương và một phụ lục, trong đó:

**Chương 1.** giới thiệu tổng quan về lọc thông tin. Trình bày những nghiên cứu cơ bản của lọc thông tin, các phương pháp lọc thông tin cho hệ tư vấn và những vấn đề cần tiếp tục nghiên cứu của mỗi phương pháp. Trên cơ sở những nghiên cứu cơ bản, xác định rõ hướng nghiên cứu của đề tài. Một kết quả nghiên cứu cơ bản của đề tài được công bố trong [4].

**Chương 2.** trình bày phương pháp hạn chế ảnh hưởng của vấn đề dữ liệu thừa trong lọc cộng tác bằng phương pháp học đa nhiệm. Nội dung trình bày trong chương này được tổng hợp dựa trên kết quả nghiên cứu đã công bố trong [3, 81].

**Chương 3.** trình bày phương pháp kết hợp giữa lọc cộng tác và lọc nội dung dựa trên mô hình đồ thị. Nội dung trình bày trong chương này được tổng hợp từ kết quả nghiên cứu đã công bố trong [2, 80]. Cuối cùng là một số kết luận và đề xuất các nghiên cứu tiếp theo.

**Phụ lục.** trình bày thiết kế và xây dựng ứng dụng cho phương pháp lọc kết hợp được đề xuất trong Chương 3.

# CHƯƠNG 1

## TỔNG QUAN VỀ LỌC THÔNG TIN CHO HỆ TƯ VẤN

Chương này trình bày những vấn đề tổng quan về lọc thông tin, các phương pháp lọc thông tin cho hệ tư vấn cùng với những hạn chế tồn tại mỗi phương pháp. Trên cơ sở những nghiên cứu cơ bản, xác định rõ hướng nghiên cứu cụ thể của đề tài. Những kết quả nghiên cứu của đề tài sẽ được trình bày trong các chương tiếp theo của luận án.

Do lọc thông tin là lĩnh vực nghiên cứu có phạm vi rộng lớn, sau khi trình bày ngắn về lọc thông tin nói chung, luận án tập trung trình bày vào chủ đề nghiên cứu chính của luận án đó là vấn đề lọc trong các hệ tư vấn.

### 1.1. GIỚI THIỆU CHUNG

Lọc thông tin (*IF*) là lĩnh vực nghiên cứu các quá trình cung cấp thông tin thích hợp, ngăn ngừa và gỡ bỏ thông tin không thích hợp cho mỗi người dùng [75, 99]. Thông tin được cung cấp (còn được gọi là sản phẩm) có thể là văn bản, trang web, phim, ảnh, dịch vụ hoặc bất kỳ dạng thông tin nào được sản sinh ra từ các phương tiện truyền thông. Phạm vi ứng dụng của lọc thông tin trải rộng trong nhiều ứng dụng thực tế khác nhau của khoa học máy tính. Ứng dụng tiêu biểu nhất của lọc thông tin được kể đến là lọc kết quả tìm kiếm trong các máy tìm kiếm (*Search Engine*), lọc e-mail dựa trên nội dung thư và hồ sơ người dùng, lọc thông tin văn bản trên các máy chủ để cung cấp thông tin cho tập thể hoặc cá nhân thích hợp, loại bỏ những trang thông tin có ảnh hưởng không tốt đối với người dùng. Đặc biệt, lọc thông tin có vai trò quan trọng cho các hệ thống tư vấn (*RS*) ứng dụng trong thương mại điện tử.

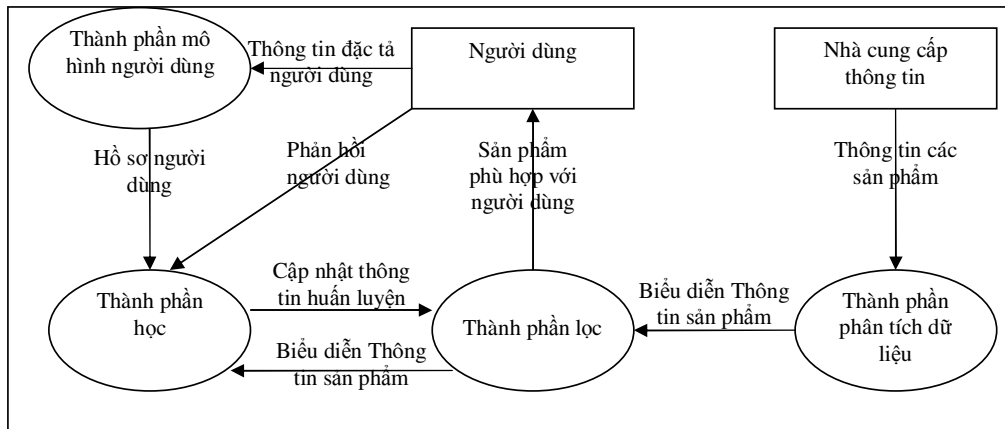
Các hệ thống lọc thông tin có thể khác nhau về nguyên lý, phương pháp, kỹ thuật, phạm vi ứng dụng nhưng đều thực hiện mục tiêu cung cấp cho người dùng những thông tin cần thiết nhất, loại bỏ những thông tin không có giá trị hoặc không thích hợp đối với người dùng. Nguyên lý phổ biến được dùng trong



lọc thông tin là nguyên lý dựa vào dữ liệu (*Data-Based*) và nguyên lý dựa vào tri thức (*Knowledge-Based*) [99]. Các phương pháp lọc có thể được thực hiện dựa vào nội dung thông tin sản phẩm hoặc lọc dựa trên thói quen sở thích người dùng. Các kỹ thuật lọc được phát triển dựa trên nền tảng từ lĩnh vực truy vấn thông tin (*Information Retrieval*), tách thông tin (*Information Extraction*), phân loại thông tin (*Information Classification*). Phạm vi ứng dụng của các hệ thống lọc được áp dụng cho tất cả các mô hình thương mại điện tử thực tế: Khách hàng - Khách hàng (*Customer to Customer*), Nhà cung cấp - Khách hàng (*Business to Customer*), Nhà cung cấp - Nhà cung cấp (*Business to Business*) [75].

### 1.1.1. Kiến trúc tổng quát của hệ thống lọc thông tin

Một hệ thống lọc thông tin tổng quát bao gồm bốn thành phần cơ bản [99]: Thành phần phân tích dữ liệu (*Data Analyser Component*), thành phần mô hình người dùng (*User Model Component*), thành phần học (*Learning Component*) và thành phần lọc (*Filtering Component*).



**Hình 1.1.** Kiến trúc tổng quát của hệ thống lọc thông tin.

- Thành phần phân tích dữ liệu (*DAC*) có nhiệm vụ thu thập dữ liệu về sản phẩm từ các nhà cung cấp thông tin (ví dụ tài liệu, thư điện tử, sách, báo, tạp chí, phim, ảnh...). Dữ liệu về sản phẩm được phân tích và biểu diễn theo một khuôn dạng thích hợp, sau đó chuyển đến bộ phận lọc như Hình 1.1.

- *Thành phần mô hình người dùng (UMC)* có thể “hiện” hoặc “ẩn” dùng để lấy thông tin về người dùng, như giới tính, tuổi, nơi sinh sống và thông tin người dùng đã truy vấn trước đó để tạo nên hồ sơ người dùng. Hồ sơ người dùng sau khi tạo ra được chuyển đến thành phần học để thực hiện nhiệm vụ huấn luyện.
- *Thành phần học (LC)* thực hiện huấn luyện trên tập hồ sơ và phản hồi của người dùng theo một thuật toán học máy cụ thể. Thuật toán học lấy dữ liệu từ thành phần mô tả người dùng; lấy dữ liệu về sản phẩm đã được biểu diễn từ thành phần lọc kết hợp với thông tin phản hồi người dùng để thực hiện nhiệm vụ huấn luyện. Kết quả quá trình học được chuyển lại cho bộ phận lọc để thực hiện nhiệm vụ tiếp theo.
- *Thành phần lọc (FC)* là thành phần quan trọng nhất của hệ thống, có nhiệm vụ xem xét sự phù hợp giữa hồ sơ người dùng và biểu diễn dữ liệu sản phẩm để đưa ra quyết định phân bổ sản phẩm. Nếu dữ liệu sản phẩm phù hợp với hồ sơ người dùng, sản phẩm sẽ được cung cấp cho người dùng đó. Trong trường hợp ngược lại, hệ thống loại bỏ sản phẩm khỏi danh sách những sản phẩm phân bổ cho người dùng. Người dùng nhận được những sản phẩm thích hợp, xem xét, đánh giá, phản hồi lại cho thành phần học để phục vụ quá trình lọc tiếp theo.

### **1.1.2. Lọc thông tin và truy vấn thông tin**

Belkin và Croft [75] nhìn nhận lọc thông tin và truy vấn thông tin như hai mặt của cùng một vấn đề. Chính vì vậy, nhiều đặc trưng cơ bản của lọc thông tin có thể tìm thấy trong lĩnh vực truy vấn thông tin (*IR*). Tuy nhiên, ta có thể phân biệt sự khác biệt giữa hai hệ thống này thông qua việc so sánh một số đặc trưng cơ bản dưới đây.

- *Kiểu người dùng.* Hệ thống truy vấn thông tin đáp ứng nhu cầu cho tất cả người dùng tại mọi thời điểm mà không cần quan tâm đến họ là ai. Trái

lại, lọc thông tin quan tâm đến những người dùng thường xuyên sử dụng hệ thống dùng, có hồ sơ rõ ràng, có mối quan tâm dài hạn đối với hệ thống và luôn nhận được thông tin thích hợp từ hệ thống ở mọi thời điểm.

- *Biểu diễn nhu cầu thông tin.* Hệ thống truy vấn thông tin biểu diễn nhu cầu người dùng bất kỳ dưới dạng một câu truy vấn. Lọc thông tin biểu diễn nhu cầu người dùng lâu dài hệ thống dưới dạng một hồ sơ người dùng. Hồ sơ người dùng không chỉ ghi lại các đặc trưng thông tin cá nhân, mà còn bao hàm các đặc trưng liên quan đến lịch sử truy cập hay thói quen sử dụng thông tin của người dùng này.
- *Mục tiêu hệ thống.* Hệ thống truy vấn thông tin quan tâm đến các phương pháp cung cấp thông tin thích hợp cho mỗi người dùng phù hợp với truy vấn của người dùng này. Lọc thông tin quan tâm đến các phương pháp gỡ bỏ dữ liệu hơn là việc nỗ lực tìm kiếm thêm dữ liệu. Cũng vì lý do này, lọc thông tin được xem là phương pháp giảm tải thông tin chính được quan tâm nhất hiện nay.
- *Cơ sở dữ liệu.* Hệ thống truy vấn thông tin thực hiện cung cấp thông tin trên các cơ sở dữ liệu tĩnh. Lọc thông tin cung cấp thông tin trên cơ sở dữ liệu động, có cấu trúc khác nhau và thường xuyên biến đổi.
- *Phạm vi tương tác.* Hệ thống truy vấn không quan tâm đến sự tương tác giữa những người dùng khác nhau. Lọc thông tin quan tâm đến sự tương đồng theo sở thích, thói quen hay những đặc trưng xã hội, tự nhiên khác nhau của tập người dùng. Hệ thống luôn có một mô hình người dùng để giữ lại những đặc trưng cần thiết cho mỗi người dùng.

### 1.1.3. Học máy và lọc thông tin

**Học máy** (*Machine Learning*). Học máy là lĩnh vực nghiên cứu của trí tuệ nhân tạo tập trung vào việc ra quyết định hoặc phát hiện tri thức dựa trên dữ liệu [1, 85, 97]. Các kỹ thuật học máy được sử dụng trong việc dự đoán (ví

dự đoán nhu cầu người dùng), phân loại, xếp hạng (ví dụ phân loại, xếp hạng thông tin, phân loại người dùng).

Lọc thông tin có cùng chung mục tiêu với học máy (*ML*) đó là cung cấp thông tin cần thiết cho mỗi người dùng dựa trên những gì có thể học từ những kinh nghiệm của cộng đồng trong quá khứ. Chính vì vậy, thành phần lọc thông tin được xây dựng theo hai cách tiếp cận chính của học máy: *lọc dựa trên tri thức* và *lọc dựa trên dữ liệu*.

**Lọc dựa trên tri thức (KBC).** Thông tin được lọc bằng cách sử dụng các luật. Mỗi luật biểu diễn nhu cầu thông tin người dùng hoặc một mẫu thông tin cần lọc. Mỗi quyết định lọc sẽ được thực hiện nếu những điều kiện của luật đưa ra được thỏa mãn. Ví dụ trong hệ thống lọc thư điện tử, mỗi luật có thể được định nghĩa và áp dụng cho các trường tiêu đề thư (Người gửi, ngày gửi, chủ đề...).

Điểm quan trọng của cách tiếp cận này là các luật do người dùng (chuyên gia) cung cấp dựa trên kinh nghiệm hay tri thức của mình. Ưu điểm của cách tiếp cận này là hệ thống sẽ đơn giản hơn do không cần sử dụng các kỹ thuật học tự động. Nhược điểm là việc xây dựng các luật lọc tốt đòi hỏi nhiều thời gian, kinh nghiệm của chuyên gia. Việc cập nhật các luật cũng không thể thực hiện tự động. Do nhược điểm này, lọc dựa trên tri thức có xu hướng ít được sử dụng.

**Lọc dựa trên dữ liệu (DBC).** Khác với lọc dựa trên tri thức, trong cách tiếp cận dựa trên dữ liệu, các quy tắc cho thành phần lọc được xây dựng từ dữ liệu mà hệ thống thu thập được bằng cách sử dụng kỹ thuật thống kê hoặc các thuật toán học máy. Cách tiếp cận này cho phép tạo ra và cập nhật quy tắc lọc thông tin mà không cần tới tri thức chuyên gia, đồng thời chất lượng lọc có thể tốt hơn so với cách tiếp cận dựa trên tri thức, đặc biệt khi có lượng dữ liệu lớn và thường xuyên biến động.

Do việc thu thập dữ liệu ngày càng nhanh và dễ, lọc dựa trên dữ liệu đang dần trở thành cách tiếp cận chính trong lọc thông tin. *Chính vì vậy, luận án sẽ tập trung nghiên cứu kỹ thuật lọc thông tin cho hệ tư vấn dựa trên cách tiếp cận này.*

#### **1.1.4. Lọc thông tin và các hệ tư vấn**

**Hệ tư vấn (RS)** là trường hợp riêng của các hệ thống lọc thông tin. Dựa trên thông tin đã có về người dùng, hệ tư vấn xem xét trong số lượng rất lớn hàng hóa hay thông tin và tư vấn cho người dùng một danh sách ngắn gọn nhưng đầy đủ những hàng hóa mà người dùng có khả năng quan tâm [25, 26, 40, 51, 53, 54, 67, 70, 83].

Sử dụng hệ tư vấn trong các ứng dụng thương mại điện tử sẽ hỗ trợ khách hàng không cần thực hiện các thao tác tìm kiếm sản phẩm, mà chỉ cần lựa chọn hàng hóa hoặc dịch vụ ưa thích do hệ thống cung cấp. Điều này sẽ làm gia tăng năng lực mua, bán của toàn bộ hệ thống. Chính vì lý do này, hàng loạt các công ty đa quốc gia (*Amazon.com, Netflix.com, CDNOW, J.C. Penney, Procter & Gamble..*) đã đầu tư và phát triển thành công công nghệ tư vấn để gia tăng hệ thống khách hàng và bán hàng qua mạng [7].

Do là trường hợp riêng của hệ thống lọc tin, hệ tư vấn có nhiều đặc điểm của hệ lọc tin tiêu biểu. Tuy nhiên, do đặc điểm của dữ liệu, người dùng và nội dung, hệ tư vấn cũng như các kỹ thuật được sử dụng có một số khác biệt nhất định. Tùy vào phương pháp lọc tin, các hệ tư vấn được phân loại thành ba loại: *Tư vấn dựa vào phương pháp lọc theo nội dung (Content-Based Filtering Recommendation), tư vấn dựa vào phương pháp lọc cộng tác (Collaborative Filtering Recommendation) và tư vấn dựa vào phương pháp lọc kết hợp (Hybrid Filtering Recommendation)*[36, 107].

- *Phương pháp tư vấn dựa vào lọc nội dung*: Hệ thống tư vấn cho người dùng những sản phẩm mới có nội dung tương tự với một số sản phẩm họ đã từng mua hoặc từng truy nhập trong quá khứ.
- *Phương pháp tư vấn dựa vào lọc cộng tác*: Người dùng sẽ được tư vấn một số sản phẩm của những người có sở thích giống họ đã từng ưa thích trong quá khứ.
- *Phương pháp tư vấn dựa vào lọc kết hợp*: Hệ thống tư vấn cho người dùng những sản phẩm tương tự với một số sản phẩm họ đã từng mua hoặc từng truy nhập trong quá khứ và sản phẩm của những người có sở thích giống họ đã từng ưa thích trong quá khứ.

Mỗi phương pháp lọc áp dụng cho các hệ tư vấn được phân thành hai hướng tiếp cận [36, 107]: lọc dựa vào bộ nhớ (*Memory-Based Filtering*) và lọc dựa vào mô hình (*Model-Based Filtering*).

- *Các phương pháp lọc dựa vào bộ nhớ (MBF)* [21, 22, 29, 52, 57, 63, 64, 69]: Đây là phương pháp lưu lại toàn bộ các ví dụ huấn luyện. Khi cần dự đoán, hệ thống tìm các ví dụ huấn luyện giống trường hợp cần dự đoán nhất và đưa ra tư vấn dựa trên các ví dụ này. Trường hợp tiêu biểu của lọc dựa vào bộ nhớ là thuật toán K người láng giềng gần nhất (KNN). Ưu điểm chính của phương pháp tiếp cận này là đơn giản, dễ cài đặt. Tuy nhiên, phương pháp này có thời gian lọc chậm do việc dự đoán đòi hỏi so sánh và tìm kiếm trên toàn bộ lượng người dùng và sản phẩm.
- *Phương pháp lọc dựa trên mô hình (MDBF)* [27, 30, 32, 33, 34, 35, 37, 41, 43, 45, 90, 95, 96, 108, 109, 121]. Trong phương pháp này, dữ liệu được sử dụng để xây dựng mô hình rút gọn, ví dụ mô hình xác suất hay cây quyết định. Mô hình này sau đó được sử dụng để đưa ra các tư vấn. Phương pháp này cho phép thực hiện việc dự đoán nhanh, do quá trình dự đoán thực hiện trên mô hình đã học trước đó.

Bảng 1.1 thống kê một số nghiên cứu tiêu biểu các phương pháp lọc thông tin cho hệ tư vấn [36].

**Bảng 1.1.** Phân loại các phương pháp tư vấn và một số nghiên cứu điển hình

<b>PHƯƠNG PHÁP TƯ VẤN DỰA VÀO LỌC NỘI DUNG</b>	
<b>Lọc nội dung dựa vào bộ nhớ</b>	<b>Lọc nội dung dựa vào mô hình</b>
<p><i>Các kỹ thuật thông dụng:</i></p> <ul style="list-style-type: none"> <li>• Tần suất xuất hiện ngược</li> <li>• Phân cụm (Clustering)</li> </ul> <p><i>Những nghiên cứu điển hình:</i></p> <ul style="list-style-type: none"> <li>• Balabanovic và Shoham [69]</li> <li>• Pazzani và Billsus [73]</li> </ul>	<p><i>Các kỹ thuật thông dụng:</i></p> <ul style="list-style-type: none"> <li>• Mô hình mạng Bayes</li> <li>• Mô hình phân cụm</li> <li>• Mô hình cây quyết định</li> <li>• Mô hình mạng nơ ron nhân tạo</li> </ul> <p><i>Những nghiên cứu điển hình:</i></p> <ul style="list-style-type: none"> <li>• Pazzani [74]</li> <li>• Mooney và Roy [92]</li> <li>• Billsus và Pazzani [30]</li> <li>• Zhang và các cộng sự [113]</li> </ul>
<b>PHƯƠNG PHÁP TƯ VẤN DỰA VÀO LỌC CỘNG TÁC</b>	
<b>Lọc cộng tác dựa vào bộ nhớ</b>	<b>Lọc cộng tác dựa vào mô hình</b>
<p><i>Các kỹ thuật thông dụng:</i></p> <ul style="list-style-type: none"> <li>• K người láng giềng gần nhất (<i>K-Nearest Neighbour</i>) sử dụng độ tương tự cosin hoặc các độ tương quan.</li> <li>• Phân cụm</li> <li>• Độ tương quan gián tiếp (Indirect Similarity)</li> </ul> <p><i>Những nghiên cứu điển hình:</i></p> <ul style="list-style-type: none"> <li>• Resnick và các cộng sự [83]</li> <li>• Breese và các cộng sự [52]</li> <li>• Nakamura và Abe [11]</li> <li>• M. Deshpande and G. Karypis [72]</li> <li>• Sarwar và các cộng sự [21]</li> <li>• Yu và các cộng sự [63, 64]</li> <li>• Herlocker và các cộng sự [55]</li> <li>• Wang và các cộng sự [57]</li> <li>• Bell và Koren [86]</li> <li>• Desrosiers và Karypis [24]</li> </ul>	<p><i>Các kỹ thuật thông dụng:</i></p> <ul style="list-style-type: none"> <li>• Mô hình mạng Bayes</li> <li>• Mô hình phân cụm</li> <li>• Mô hình cây quyết định</li> <li>• Mô hình mạng nơ ron nhân tạo</li> <li>• Mô hình hội qui tuyến tính</li> <li>• Mô hình thống kê</li> <li>• Mô hình đồ thị</li> </ul> <p><i>Những nghiên cứu điển hình:</i></p> <ul style="list-style-type: none"> <li>• Nakamura và Abe [11]</li> <li>• Umyarov và Alexander Tuzhilin [15, 16, 17]</li> <li>• Ungar và Foster [68]</li> <li>• Aggarwal và các cộng sự [24]</li> <li>• Chien và George [114]</li> <li>• Condliff và các cộng sự [71]</li> <li>• Kumar và các cộng sự [89]</li> <li>• Shani và các cộng sự [41]</li> <li>• Hofmann [95, 96]</li> <li>• Marlin [18]</li> </ul>

<ul style="list-style-type: none"> <li>• Goldberg và các cộng sự [62]</li> </ul>	<ul style="list-style-type: none"> <li>• Si và Jin [66]</li> <li>• Getoor và Sahami [65]</li> <li>• Huang và các cộng sự [119]</li> <li>• DeCoste [31]</li> <li>• Nikovski và Kulev [33]</li> <li>• Su và các cộng sự [105, 106, 107]</li> </ul>
<b>PHƯƠNG PHÁP TƯ VẤN DỰA VÀO LỌC KẾT HỢP</b>	
<b>Lọc kết hợp dựa vào bộ nhớ</b>	<b>Lọc kết hợp dựa vào mô hình</b>
<p><i>Các kỹ thuật thông dụng:</i></p> <ul style="list-style-type: none"> <li>• Tổ hợp tuyến tính kết quả dự đoán của cả hai phương pháp.</li> <li>• Kết hợp các đặc tính của lọc cộng tác vào lọc nội dung.</li> <li>• Kết hợp các đặc tính của lọc nội dung vào lọc cộng tác.</li> <li>• Hợp nhất lọc cộng tác và lọc nội dung trong cùng mô hình.</li> </ul> <p><i>Những nghiên cứu điển hình:</i></p> <ul style="list-style-type: none"> <li>• Basu và các cộng sự [23]</li> <li>• Claypool và các cộng sự [70]</li> <li>• Soboroff và Nicolas [46]</li> <li>• Billsus và Pazzani [30]</li> <li>• Tran và Cohen [98]</li> <li>• Melville và các cộng sự [82]</li> <li>• Adomavicius và các cộng sự [37, 38, 39]</li> <li>• Anand và Bharadwaj [28]</li> </ul>	<p><i>Các kỹ thuật thông dụng:</i></p> <ul style="list-style-type: none"> <li>• Hợp nhất mô hình biểu diễn dữ liệu.</li> <li>• Hợp nhất mô hình dự đoán.</li> <li>• Hợp nhất mô hình biểu diễn dữ liệu và mô hình dự đoán.</li> </ul> <p><i>Những nghiên cứu điển hình:</i></p> <ul style="list-style-type: none"> <li>• Gunawardana và Meek [8]</li> <li>• Billsus và Pazzani [29]</li> <li>• Lazanas và Karacapilidis [10]</li> <li>• Popescul và các cộng sự [12]</li> <li>• Hofmann [96]</li> <li>• Huang và các cộng sự [120, 121, 122]</li> <li>• Su và các cộng sự [104]</li> <li>• Balisico và Hofmann [47]</li> <li>• Good và các cộng sự [76]</li> </ul>

Formatted: Indent: Left: 0,63 cm

## 1.2. PHƯƠNG PHÁP LỌC THEO NỘI DUNG

Lọc theo nội dung là phương pháp thực hiện dựa trên việc so sánh nội dung thông tin hay mô tả hàng hóa, nhằm 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 [4, 6, 19, 69, 73, 84, 92]. Các phương pháp tiếp cận cho lọc theo nội dung có nguồn gốc từ lĩnh vực truy vấn thông tin, trong đó mỗi sản phẩm được biểu diễn bằng một hồ sơ sản phẩm, mỗi người dùng được biểu diễn bằng một



hồ sơ người dùng. Phương pháp dự đoán nội dung nguyên bản của sản phẩm thực hiện dựa vào việc xem xét các hồ sơ sản phẩm có mức độ phù hợp cao với hồ sơ người dùng [84].

### 1.2.1. Bài toán lọc theo nội dung

Bài toán lọc theo nội dung được phát biểu như sau. Cho  $P = \{p_1, p_2, \dots, p_N\}$  là tập gồm  $N$  sản phẩm. Nội dung sản phẩm  $p \in P$  được ký hiệu là  $Content(p)$  được biểu diễn thông qua tập  $K$  đặc trưng nội dung của  $P$ . Tập các đặc trưng sản phẩm  $p$  được xây dựng bằng các kỹ thuật truy vấn thông tin để thực hiện mục đích dự đoán những sản phẩm khác tương tự với  $p$ .

Cho  $U = \{u_1, u_2, \dots, u_M\}$  là tập gồm  $M$  người dùng. Với mỗi người dùng  $u \in U$ , gọi  $ContentBasedProfile(u)$  là hồ sơ người dùng  $u$ . Hồ sơ của người dùng  $u$  thực chất là lịch sử truy cập hoặc đánh giá của người đó đối với các sản phẩm.  $ContentBasedProfile(u)$  được xây dựng bằng cách phân tích nội dung các sản phẩm mà người dùng  $u$  đã từng truy cập hoặc đánh giá dựa trên các kỹ thuật truy vấn thông tin.

Bài toán lọc theo nội dung khi đó là dự đoán những sản phẩm mới có nội dung thích hợp với người dùng dựa trên tập hồ sơ sản phẩm  $Content(p)$  và hồ sơ người dùng  $ContentBasedProfile(u)$ .

### 1.2.2. Các phương pháp lọc theo nội dung

Như đã trình bày ở trên, lọc theo nội dung được tiếp cận theo hai xu hướng: lọc dựa trên bộ nhớ và lọc dựa trên mô hình. Nội dung cụ thể các phương pháp được thực hiện như dưới đây.

#### 1.2.2.1. Lọc nội dung dựa vào bộ nhớ

Lọc nội dung dựa vào bộ nhớ là phương pháp sử dụng toàn bộ tập hồ sơ sản phẩm và tập hồ sơ người dùng để thực hiện huấn luyện và dự đoán. Trong phương pháp này, các sản phẩm mới được tính toán và so sánh với tất cả hồ sơ người dùng. Những sản phẩm mới có mức độ tương tự cao nhất với hồ sơ người dùng sẽ

được dùng để tư vấn cho người dùng này. Phương pháp này còn được gọi là *học lười* (*Lazy Learning*) hay *học dựa trên ví dụ* (*Instance-Based Learning*) trong các tài liệu về học máy [97].

Để thực hiện lọc theo nội dung, ta cần giải quyết hai vấn đề: thứ nhất là biểu diễn  $Content(p)$  dưới dạng vector trọng số các đặc trưng nội dung, thứ hai là tính độ tương tự giữa hồ sơ người dùng và hồ sơ sản phẩm.

#### **Phương pháp biểu diễn hồ sơ sản phẩm:**

Phương pháp ước lượng trọng số các đặc trưng thông dụng nhất thường được sử dụng là phép đo tần suất kết hợp với tần suất xuất hiện ngược (*Term Frequency / Inverse Document Frequency*). Phương pháp được thực hiện như sau.

Gọi  $f_{i,j}$  là số lần đặc trưng nội dung  $k_i$  xuất hiện trong sản phẩm  $p_j$ . Khi đó tần suất  $TF_{i,j}$  của đặc trưng nội dung  $k_i$  trong sản phẩm  $p_j$  được xác định theo công thức (1.1).

$$TF_{i,j} = \frac{f_{i,j}}{\max_z f_{z,j}} \quad (1.1)$$

Ở đây,  $\max_z f_{z,j}$  là số lần xuất hiện nhiều nhất của đặc trưng nội dung  $k_z$  trong sản phẩm  $p_j$ .

Tuy nhiên, những đặc trưng nội dung xuất hiện trong nhiều sản phẩm không được dùng để xem xét mức độ tương tự giữa các sản phẩm, thậm chí những đặc trưng nội dung này không chứa đựng nhiều thông tin phản ánh nội dung sản phẩm. Chính vì vậy, tần suất xuất hiện ngược  $IDF_i$ , kết hợp với tần suất  $TF_{i,j}$  cho phép ta chú ý nhiều hơn đến những đặc trưng nội dung có trong sản phẩm này nhưng ít xuất hiện trong các sản phẩm khác.

Phương pháp xác định tần suất xuất hiện ngược được thực hiện như sau. Giả sử hệ có  $N$  sản phẩm cần được phân bổ hoặc tư vấn cho người dùng và đặc trưng nội dung  $k_i$  xuất hiện trong  $n_i$  sản phẩm. Tần suất xuất hiện ngược  $IDF_i$  của đặc trưng nội dung  $k_i$  có tần suất xuất hiện trong sản phẩm  $p_j$  là  $TF_{i,j}$  được xác định theo công thức (1.2), mức độ quan trọng hay trọng số của đặc trưng nội dung  $k_i$  được xác định theo công thức (1.3).

$$IDF_i = \log \frac{N}{n_i} \quad (1.2)$$

$$w_{i,j} = TF_{i,j} \times IDF_i \quad (1.3)$$

Trong công thức 1.2, nếu  $n_i \equiv N$  hay đặc trưng nội dung  $k_i$  xuất hiện trong đa số các sản phẩm cần phân bổ đến người dùng thì trọng số  $w_{i,j} \equiv 0$ . Nói cách khác, những đặc trưng nội dung có trong mọi sản phẩm thì đặc trưng đó không chứa nhiều nội dung thông tin phản ánh sản phẩm. Ngược lại, nếu đặc trưng nội dung chỉ xuất hiện trong một sản phẩm thì  $n_i = 1$ , khi đó  $w_{i,j} = TF_{i,j}$ . Như vậy, những đặc trưng nội dung chỉ xuất hiện ở một loại sản phẩm và không xuất hiện ở những sản phẩm khác thì những đặc trưng nội dung này chứa nhiều nội dung quan trọng đối với sản phẩm.

Bằng cách ước lượng này, mỗi sản phẩm  $p_j \in P$  được biểu diễn như một véc tơ trọng số các đặc trưng nội dung  $Content(p_j) = (w_{1,j}, w_{2,j}, \dots, w_{K,j})$ . Trong đó,  $K$  là số lượng đặc trưng nội dung của toàn bộ sản phẩm.

#### **Phương pháp biểu diễn hồ sơ người dùng:**

Mỗi hồ sơ người dùng  $ContentBasedProfile(u)$  cũng được biểu diễn bằng một véc tơ trọng số các đặc trưng nội dung  $(w_{1,u}, w_{2,u}, \dots, w_{K,u})$ , trong đó mỗi  $w_{k,u}$  biểu thị mức độ quan trọng của đặc trưng nội dung  $k$  đối với người dùng  $u$ . Véc tơ trọng số  $(w_{1,u}, w_{2,u}, \dots, w_{K,u})$  được tính toán bằng các kỹ thuật khác nhau từ véc tơ hồ sơ sản phẩm đã được người dùng thường xuyên truy cập hoặc đánh giá. Balabanovic [69] tính toán véc tơ trọng số mỗi hồ sơ người dùng  $ContentBasedProfile(u)$  bằng cách lấy trung bình cộng véc tơ trọng số  $Content(p_j)$  trên các tài liệu  $p_j \in P$  mà người dùng đã từng truy cập hoặc đánh giá. Pazzani [74] sử dụng bộ phân loại Bayes ước lượng khả năng giống nhau của sản phẩm và đề xuất thuật toán Winnow thực hiện trong những trường hợp có nhiều đặc trưng nội dung.

### Xác định mức độ tương tự:

Với cách biểu như trên, vectơ trọng số các đặc trưng nội dung sản phẩm  $ContentBasedProfile(u)$  và  $Content(p)$  có cùng số chiều và ước lượng theo cùng một phương pháp (trong trường hợp này là TF-IDF). Việc xác định mức độ thích hợp của mỗi sản phẩm  $p \in P$  cho người dùng  $u$  được xem xét theo mức độ giống nhau giữa véc tơ hồ sơ người dùng  $u \in U$  và véc tơ hồ sơ sản phẩm  $p \in P$ .

$$r(u, p) = Sim(ContentBased Profile(u), Content(p)) \quad (1.4)$$

Phương pháp ước lượng mức độ giống nhau giữa véc tơ hồ sơ người dùng  $u \in U$  và véc tơ hồ sơ sản phẩm  $p \in P$  được dùng phổ biến là tìm cosin của hai véc tơ trọng số  $\vec{w}_u$  và  $\vec{w}_p$ .

$$\begin{aligned} r(u, p) &= \cos(\vec{w}_u, \vec{w}_p) = \frac{\vec{w}_u \cdot \vec{w}_p}{\|\vec{w}_u\|_2 \times \|\vec{w}_p\|_2} \\ &= \frac{\sum_{i=1}^K w_{i,u} w_{i,p}}{\sqrt{\sum_{i=1}^K w_{i,u}^2} \sqrt{\sum_{i=1}^K w_{i,p}^2}}, \end{aligned} \quad (1.5)$$

Ở đây,  $K$  là số lượng đặc trưng nội dung của hệ thống. Trong công thức 1.5, nếu cosin của hai véc tơ gần với 1, hay góc tạo bởi hai véc tơ này nhỏ thì mức độ tương tự giữa hồ sơ người dùng và hồ sơ sản phẩm càng cao. Ngược lại, nếu cosin của hai véc tơ gần với 0, hay góc tạo bởi hai véc tơ lớn thì mức độ phù hợp của sản phẩm với hồ sơ người dùng càng thấp. Với cách đo này, nếu người dùng  $u$  truy nhập nhiều sản phẩm liên quan đến một chủ đề nào đó thì hệ thống lọc theo nội dung sẽ phân bổ những sản phẩm của chủ đề đó cho người dùng  $u$ .

Ngoài cosin, các độ đo tương tự khác như khoảng cách Euclid hay độ tương quan Pearson cũng được sử dụng trong những nghiên cứu khác nhau.

#### 1.2.2.2. Lọc nội dung dựa vào mô hình

Lọc nội dung dựa trên mô hình là phương pháp sử dụng tập hồ sơ sản phẩm và tập hồ sơ người dùng để xây dựng nên mô hình huấn luyện. Mô hình dự đoán sau đó sẽ sử dụng kết quả của mô hình huấn luyện để sinh ra tư vấn cho người

dùng. Trong cách tiếp cận này, lọc nội dung có thể sử dụng các kỹ thuật học máy như mạng Bayes, phân cụm, cây quyết định, mạng nơron nhân tạo để tạo nên dự đoán.

*Pazzani và Billsus* [73] sử dụng bộ phân loại Bayes dựa trên những đánh giá “*thích*” hoặc “*không thích*” của người dùng để phân loại các sản phẩm. Trong đó, phương pháp ước lượng xác suất sản phẩm  $p_j$  có thuộc lớp  $C_i$  hay không dựa vào tập các đặc trưng nội dung  $k_{1,j}, \dots, k_{n,j}$  của sản phẩm đó.

$$P(C_i | k_{1,j} \& k_{2,j} \& \dots \& k_{n,j}) \quad (1.6)$$

*Panzanni và Billsus* giả thiết các đặc trưng nội dung xuất hiện độc lập nhau, vì vậy xác suất ở trên tương ứng với:

$$P(C_i) \prod_x P(k_{x,j} | C_i) \quad (1.7)$$

Vì  $P(k_{x,j} | C_i)$  và  $P(C_i)$  có thể ước lượng dựa vào tập dữ liệu huấn luyện. Do vậy, sản phẩm  $p_j$  được xem là thuộc lớp  $C_i$  nếu xác suất  $P(C_i | k_{1,j} \& k_{2,j} \& \dots \& k_{n,j})$  có giá trị cao nhất thuộc lớp này.

Solombo [42] đề xuất mô hình lọc thích nghi, trong đó chú trọng đến việc quan sát mức phù hợp của tất cả các sản phẩm. Zhang [112] đề xuất mô hình tối ưu tập các sản phẩm tương tự dựa vào giá trị ngưỡng. Trong đó, giá trị ngưỡng được ước lượng dựa trên tập sản phẩm thích hợp và tập tài liệu không thích hợp với mỗi hồ sơ người dùng.

### 1.2.3. Những vấn đề tồn tại

Mặc dù lọc theo nội dung đã áp dụng thành công cho nhiều ứng dụng lọc văn bản, tuy vậy phương pháp vẫn tồn tại một số vấn đề cần tiếp tục nghiên cứu giải quyết [36, 107].

- *Vấn đề trích chọn đặc trưng.* Lọc theo nội dung kế thừa và phát triển dựa chủ yếu vào các phương pháp trích chọn đặc trưng trong lĩnh vực truy vấn thông tin. Để có một tập các đặc trưng đầy đủ, nội dung tài liệu phải được biểu diễn dưới dạng phù hợp để máy tính có thể tự động phân tích, tính toán trọng số các đặc trưng nội dung hoặc phải được thực hiện bán tự động. Phương pháp sẽ khó áp dụng trong những trường hợp việc trích

chọn nội dung phức tạp, chẳng hạn trích chọn đặc trưng nội dung các đối tượng dữ liệu đa phương tiện (hình ảnh, âm thanh, dịch vụ).

- *Vấn đề người dùng mới.* Các hệ thống lọc theo nội dung chỉ thực hiện hiệu quả khi người dùng đánh giá hoặc truy nhập một số lượng sản phẩm đủ lớn. Trong trường hợp người dùng mới, véc tơ hồ sơ người dùng có các thành phần là  $\emptyset$ , vì vậy hệ thống sẽ không thể thực hiện dự đoán và phân bổ những sản phẩm thích hợp cho người dùng.

### 1.3. PHƯƠNG PHÁP LỌC CỘNG TÁC

Không giống như lọc theo nội dung, lọc cộng tác khai thác những khía cạnh liên quan đến thói quen sở thích của người sử dụng sản phẩm để đưa ra dự đoán các sản phẩm mới cho người dùng này. So với lọc theo nội dung, lọc cộng tác không phải phân tích, bóc tách, hiểu, đánh chỉ mục cho các đặc trưng nội dung sản phẩm. Chính vì vậy, lọc cộng tác có thể lọc hiệu quả trên nhiều dạng sản phẩm khác nhau như hàng hóa, phim, ảnh, tài liệu [55]. Cùng trên một hệ tư vấn, người dùng sẽ được tư vấn nhiều loại mặt hàng khác nhau cho dù các mặt hàng này có thể biểu diễn trên không gian các đặc trưng nội dung khác nhau.

#### 1.3.1. Bài toán lọc cộng tác

Ký hiệu  $U = \{u_1, u_2, \dots, u_N\}$  là tập gồm  $N$  người dùng,  $P = \{p_1, p_2, \dots, p_M\}$  là tập gồm  $M$  sản phẩm mà người dùng có thể lựa chọn. Mỗi sản phẩm  $p_i \in P$  có thể là hàng hóa, phim, ảnh, tạp chí, tài liệu, sách, báo, dịch vụ hoặc bất kỳ dạng thông tin nào mà người dùng cần đến.

Tiếp theo, ký hiệu  $R = \{r_{ij}\}$ ,  $i = 1..N, j = 1..M$  là ma trận đánh giá, trong đó mỗi người dùng  $u_i \in U$  đưa ra đánh giá của mình cho một số sản phẩm  $p_j \in P$  bằng một số  $r_{ij}$ . Giá trị  $r_{ij}$  phản ánh mức độ ưa thích của người dùng  $u_i$  đối với sản phẩm  $p_j$ . Giá trị  $r_{ij}$  có thể được thu thập trực tiếp bằng cách hỏi ý kiến người dùng hoặc thu thập gián tiếp thông qua cơ chế phản hồi của người dùng. Giá trị  $r_{ij} = \emptyset$  trong trường hợp người dùng  $u_i$  chưa đánh giá hoặc chưa bao giờ biết đến sản phẩm  $p_j$ .

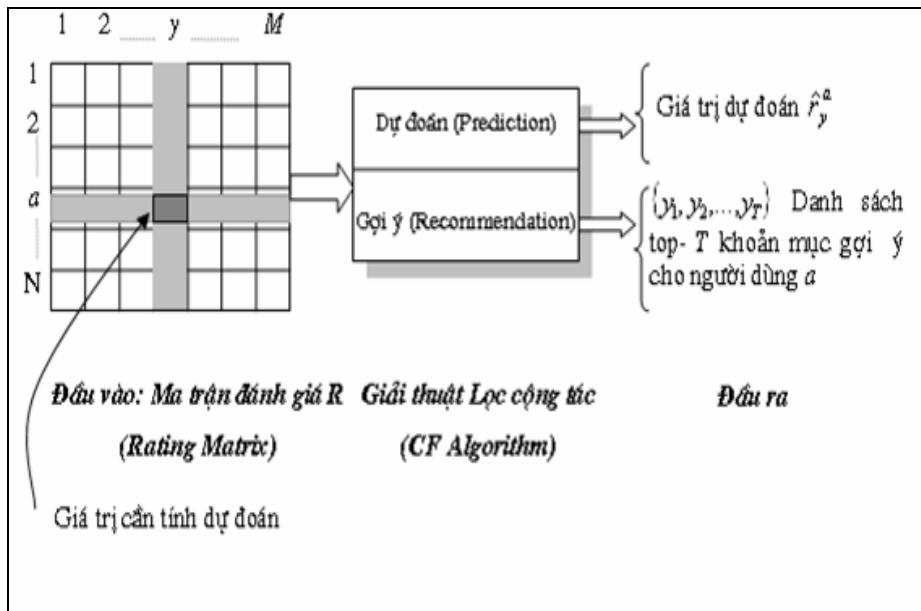
Với một người dùng cần được tư vấn  $u_a$  (được gọi là người dùng hiện thời, người dùng cần được tư vấn, hay người dùng tích cực), bài toán lọc cộng

tác là bài toán dự đoán đánh giá của  $u_a$  đối với những mặt hàng mà  $u_a$  chưa đánh giá ( $r_{aj} = \emptyset$ ), trên cơ sở đó tư vấn cho  $u_a$  những sản phẩm được đánh giá cao.

Bảng 1.2 thể hiện một ví dụ với ma trận đánh giá  $R = (r_{ij})$  trong hệ gồm 5 người dùng  $U = \{u_1, u_2, u_3, u_4, u_5\}$  và 4 sản phẩm  $P = \{p_1, p_2, p_3, p_4\}$ . Mỗi người dùng đều đưa ra các đánh giá của mình về các sản phẩm theo thang bậc  $\{\emptyset, 1, 2, 3, 4, 5\}$ . Giá trị  $r_{ij} = \emptyset$  được hiểu là người dùng  $u_i$  chưa đánh giá hoặc chưa bao giờ biết đến sản phẩm  $p_j$ . Các giá trị  $r_{5,2} = ?$  là sản phẩm hệ thống cần dự đoán cho người dùng  $u_5$ .

**Bảng 1.2.** Ví dụ về ma trận đánh giá của lọc cộng tác

	$p_1$	$p_2$	$p_3$	$p_4$
$u_1$	2	1	3	5
$u_2$	4	2	1	$\emptyset$
$u_3$	3	$\emptyset$	2	4
$u_4$	4	4	$\emptyset$	$\emptyset$
$u_5$	4	?	5	5



**Hình 1.2.** Các thành phần của hệ thống lọc cộng tác

Ma trận đánh giá  $R = (r_{ij})$  là thông tin đầu vào duy nhất của các phương pháp lọc cộng tác. Dựa trên ma trận đánh giá, các phương pháp lọc cộng tác thực hiện hai tác vụ: Dự đoán quan điểm của người dùng hiện thời (*Active User*) về các sản phẩm mà họ chưa đánh giá, đồng thời đưa ra một danh sách các sản phẩm có đánh giá cao nhất phân bổ cho người dùng hiện thời. Hình 1.2 mô tả các thành phần của hệ thống lọc cộng tác.

### 1.3.2. Các phương pháp lọc cộng tác

Cũng giống như lọc theo nội dung, lọc cộng tác tiếp cận theo hai xu hướng chính: Lọc cộng tác dựa trên bộ nhớ và lọc cộng tác dựa trên mô hình. Mỗi phương pháp tiếp cận có những ưu điểm và hạn chế riêng, khai thác các mối liên hệ trên ma trận đánh giá người dùng. Cách tiếp cận cụ thể mỗi phương pháp được thực hiện như sau.

#### 1.3.2.1. Lọc cộng tác dựa trên bộ nhớ

Các phương pháp lọc dựa trên bộ nhớ [21, 52, 56, 83, 100, 101, 102] sử dụng toàn bộ ma trận đánh giá để sinh ra dự đoán các sản phẩm cho người dùng hiện thời (*AU*). Về thực chất, đây là phương pháp học lười (*LL*) hay học dựa trên ví dụ (*IBL*) được sử dụng trong học máy. Phương pháp được thực hiện theo hai bước: Tính toán mức độ tương tự và bước tạo nên dự đoán.

- **Tính toán mức độ tương tự  $sim(x, y)$ :** Mô tả khoảng cách, sự liên quan, hay trọng số giữa hai người dùng  $x$  và  $y$  (hoặc giữa hai sản phẩm  $x$  và  $y$ ).
- **Dự đoán:** Đưa ra dự đoán cho người dùng cần được tư vấn bằng cách xác định tập láng giềng của người dùng này. Tập láng giềng của người dùng cần tư vấn được xác định dựa trên mức độ tương tự giữa các cặp người dùng hoặc sản phẩm.

#### Các phương pháp tính toán mức độ tương tự

Việc tính toán mức độ tương tự giữa hai người dùng  $x$  và  $y$  được xem xét dựa vào tập sản phẩm cả hai người dùng đều đánh giá. Tương tự, việc tính toán mức độ tương tự giữa hai sản phẩm  $x$  và  $y$  được xem xét dựa vào tập người dùng



cùng đánh giá cả hai sản phẩm. Sau đó, sử dụng một độ đo cụ thể để xác định mức độ tương tự giữa hai người dùng hoặc sản phẩm.

Có nhiều phương pháp khác nhau tính toán mức độ tương tự  $sim(x, y)$  giữa các cặp người dùng [56, 72]. Hai phương pháp phổ biến nhất được sử dụng là độ tương quan Pearson và giá trị cosin giữa hai vectơ.

- *Độ tương quan Pearson giữa hai người dùng  $x, y$  (User-Based Similarity)* được tính toán theo công thức (1.8). Trong đó,  $P_{xy} = \{p \in P \mid r_{x,p} \neq \emptyset \wedge r_{y,p} \neq \emptyset\}$  là tập tất cả các sản phẩm người dùng  $x$  và người dùng  $y$  cùng đánh giá,  $\bar{r}_x, \bar{r}_y$  là trung bình cộng các đánh giá khác  $\emptyset$  của người dùng  $x$  và người dùng  $y$ .

$$sim(x, y) = \frac{\sum_{p \in P_{xy}} (r_{x,p} - \bar{r}_x)(r_{y,p} - \bar{r}_y)}{\sqrt{\sum_{p \in P_{x,y}} (r_{x,p} - \bar{r}_x)^2 \sum_{p \in P_{xy}} (r_{y,p} - \bar{r}_y)^2}} \quad (1.8)$$

- *Độ tương quan Pearson giữa hai sản phẩm  $x, y$  (Item-Based Similarity)* được tính toán theo công thức (1.9). Trong đó,  $U_{xy} = \{u \in U \mid r_{u,x} \neq \emptyset \wedge r_{u,y} \neq \emptyset\}$  là tập tất cả người dùng cùng đánh giá sản phẩm  $x$  và sản phẩm  $y$ . Giá trị  $\bar{r}_x, \bar{r}_y$  là đánh giá trung bình cho sản phẩm  $x$  và sản phẩm  $y$ .

$$sim(x, y) = \frac{\sum_{u \in U_{xy}} (r_{u,x} - \bar{r}_x)(r_{u,y} - \bar{r}_y)}{\sqrt{\sum_{u \in U_{x,y}} (r_{u,x} - \bar{r}_x)^2 \sum_{u \in U_{xy}} (r_{u,y} - \bar{r}_y)^2}} \quad (1.9)$$

- *Độ tương tự vectơ giữa hai người dùng  $x, y$*  là cosin của hai vectơ  $x$  và  $y$  theo công thức (1.10). Trong đó, hai người dùng  $x$  và  $y$  được xem xét như hai véc tơ  $m$  chiều,  $m=|P_{xy}|$  là số lượng các sản phẩm cả hai người dùng cùng đánh giá.

$$sim(x, y) = \cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\| \times \|\vec{y}\|} = \frac{\sum_{p \in P_{xy}} r_{x,p} r_{y,p}}{\sqrt{\sum_{p \in P_{xy}} r_{x,p}^2} \sqrt{\sum_{p \in P_{xy}} r_{y,p}^2}} \quad (1.10)$$

- Độ tương tự vectơ giữa hai sản phẩm  $x, y$  là cosin của hai vectơ  $x$  và  $y$  theo công thức (1.10). Trong đó, hai sản phẩm  $x$  và  $y$  được xem xét như hai vectơ cột  $n$  chiều,  $n=|U_{xy}|$  là số lượng các người dùng cùng đánh giá sản phẩm  $p$ .

$$sim(x, y) = \cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\| \times \|\vec{y}\|} = \frac{\sum_{u \in U_{xy}} r_{u,x} r_{u,y}}{\sqrt{\sum_{u \in U_{xy}} r_{u,x}^2} \sqrt{\sum_{u \in U_{xy}} r_{u,y}^2}} \quad (1.11)$$

Chú ý rằng cả hai phương pháp lọc theo nội dung và lọc cộng tác đều sử dụng độ đo cosin giống nhau trên tập các sản phẩm. Tuy nhiên, lọc theo nội dung sử dụng độ tương tự cosin cho các vectơ của trọng số được tính theo độ đo TF-IDF, lọc cộng tác sử dụng cosin giữa hai vectơ biểu diễn đánh giá của người dùng.

Một số độ tương tự khác cũng được sử dụng trong lọc cộng tác như: *Constrained Pearson correlation*, *Root Mean Square*, *Spearman rank correlation*, *Kendall's  $\tau$  correlation*. Về bản chất, những độ đo tương tự này là biến đổi của độ tương quan Pearson [56].

### Các phương pháp dự đoán

Phương pháp dự đoán mức độ thích hợp của sản phẩm  $p$  chưa được người dùng  $u$  đánh giá được tính toán dựa trên tập những người dùng khác đã đánh giá  $p$ . Gọi  $\hat{U}$  là tập  $N$  người dùng tương tự nhất đối với  $u$ . Khi đó, mức độ phù hợp của người dùng  $u$  đối với sản phẩm mới  $p$  được xác định như một hàm các đánh giá của tập láng giềng. Dưới đây là một số phương pháp thông dụng nhất để dự đoán mức độ phù hợp của sản phẩm  $p$  đối với người dùng  $u$ .

$$\begin{aligned}
(a) \quad r_{u,p} &= \frac{1}{N} \sum_{u' \in \tilde{U}} r_{u',p} \\
(b) \quad r_{u,p} &= k \sum_{u' \in \tilde{U}} \text{sim}(u, u') \times r_{u',p} \\
(c) \quad r_{u,p} &= \bar{r}_u + k \sum_{u' \in \tilde{U}} \text{sim}(u, u') \times (r_{u',p} - \bar{r}_{u'})
\end{aligned} \tag{1.12}$$

Trong công thức (1.12),  $k$  được gọi là nhân tố chuẩn hóa,  $\bar{r}$  là trung bình các đánh giá của người dùng  $u$  được xác định theo (1.13).

$$\begin{aligned}
k &= 1 / \sum_{u' \in \tilde{U}} |\text{sim}(u, u')| \\
\bar{r}_u &= \frac{1}{|P_u|} \sum_{p \in P_u} r_{u,p} \\
P_u &= \{p \in P \mid r_{u,p} \neq \phi\}
\end{aligned} \tag{1.13}$$

### 1.3.2.2. Lọc cộng tác dựa vào mô hình

Khác với phương pháp dựa trên bộ nhớ, phương pháp lọc dựa trên mô hình [3, 11, 18, 34, 41, 59, 65, 68, 71, 77, 81, 88, 93, 94, 95, 103, 106, 117, 118, 119] sử dụng tập đánh giá để xây dựng mô hình huấn luyện. Kết quả của mô hình huấn luyện được sử dụng để sinh ra dự đoán quan điểm của người dùng về các sản phẩm chưa được họ đánh giá. Ưu điểm của phương pháp này là mô hình huấn luyện có kích thước nhỏ hơn rất nhiều so với ma trận đánh giá và thực hiện dự đoán nhanh. Mô hình chỉ cần cập nhật lại khi có những thay đổi lớn và chỉ thực hiện lại pha xây dựng mô hình.

#### Mô hình mạng Bayes:

Mô hình mạng Bayes biểu diễn mỗi sản phẩm như một đỉnh của đồ thị, trạng thái của đỉnh tương ứng với giá trị đánh giá của người dùng đối với sản phẩm đã được đánh giá. Cấu trúc của mạng được nhận biết từ tập dữ liệu huấn luyện.

Breese [52] đề xuất phương pháp mạng Bayes đơn giản cho lọc cộng tác, trong đó những đánh giá chưa biết được tính toán theo công thức (1.14). Breese giả thiết các giá trị đánh giá được xem xét như những số nguyên nằm giữa 0 và  $n$ . Đánh giá chưa biết của người dùng  $u$  đối với sản phẩm  $p$  là  $r_{u,p}$  được ước

lượng thông qua những đánh giá trước đó của người dùng  $u$ . Gọi  $P_u = \{ p' \in P \mid r_{u,p'} \neq \emptyset \}$ . Khi đó, đánh giá chưa biết của người dùng  $u$  đối với sản phẩm  $p$  được tính theo công thức (1.14)

$$r_{u,p} = E(r_{u,p}) = \sum_{i=0}^n i \times \Pr(r_{u,p} = i \mid r_{u,p'}, p' \in P_u) \quad (1.14)$$

Billsus và Pazzani [29, 30] chuyển đổi dữ liệu có nhiều mức đánh giá thành dữ liệu nhị phân. Khi đó, ma trận đánh giá được chuyển đổi thành ma trận bao gồm đặc trưng nhị phân. Việc chuyển đổi này làm cho việc sử dụng mô hình mạng Bayes trở nên thuận tiện hơn. Tuy nhiên, kết quả phân loại theo các đặc trưng nhị phân không phản ánh đúng các bộ dữ liệu thực.

Su và Khoshgoftaar [103] mở rộng mô hình mạng Bayes cho các tập dữ liệu thực gồm nhiều lớp đánh giá khác nhau. Kết quả dự đoán của mô hình tốt hơn so với các phương pháp dựa trên độ tương quan Pearson và mô hình mạng Bayes đơn giản.

### **Mô hình phân cụm:**

Một cụm là tập các đối tượng dữ liệu có các phần tử trong cụm giống nhau nhiều nhất, và khác nhau nhiều nhất đối với các phần tử thuộc các cụm khác [107]. Các phương pháp phân cụm cho lọc cộng tác được sử dụng để phân chia tập người dùng (hoặc tập sản phẩm) thành các cụm người dùng (hoặc sản phẩm) có sở thích tương tự nhau. Khi đó, người dùng (hoặc sản phẩm) thuộc cụm nào sẽ được dự đoán và tư vấn các sản phẩm được đánh giá cao trong cụm đó [55, 107].

Độ đo dùng để ước lượng mức độ giống nhau giữa các đối tượng dữ liệu thường được sử dụng là khoảng cách Minkowski và độ tương quan Pearson [107].

Cho hai đối tượng dữ liệu  $X = (x_1, x_2, \dots, x_n)$ ,  $Y = (y_1, y_2, \dots, y_n)$ . Khi đó, khoảng cách Minkowski được định nghĩa theo công thức (1.15).

$$d(X, Y) = \sqrt[q]{\sum_{i=1}^n |x_i - y_i|^q}, \quad (1.15)$$

Trong đó,  $n$  là số chiều của  $X$  và  $Y$ ;  $x_i, y_i$  là giá trị thành phần thứ  $i$  của  $X$  và  $Y$ ;  $q$  là một số nguyên dương. Nếu  $q = 1$ , thì  $d(X, Y)$  là khoảng cách Minkowski. Nếu  $q = 2$ , thì  $d(X, Y)$  là khoảng cách Euclid.

Sarwar [20] và Herlocker [55] cùng các cộng sự sử dụng các kỹ thuật phân cụm chia tập người dùng thành các cụm. Phương pháp dự đoán sử dụng các thuật toán dựa trên bộ nhớ như độ tương quan Pearson để thực hiện trên mỗi cụm dữ liệu.

Ungar và Foster [68] sử dụng kỹ thuật K-median phân tập người dùng thành các cụm dựa vào những sản phẩm họ đã đánh giá, phân tập sản phẩm thành các cụm sản phẩm dựa vào những người dùng đánh giá sản phẩm đó. Tập người dùng sau đó được phân cụm lại dựa vào số sản phẩm họ đánh giá. Tương tự như vậy, tập sản phẩm cũng được phân cụm lại dựa vào số lượng người dùng đã đánh giá sản phẩm. Phương pháp này được đánh giá cao về ý tưởng, nhưng trên thực tế kết quả dự đoán không được như mong muốn.

Si và Jin [66] đề xuất mô hình phân cụm bằng mô hình *FMM (Flexible Mixture Model)*. Phương pháp phân cụm đồng thời cho cả người dùng và sản phẩm và cho phép mỗi người dùng hoặc sản phẩm có thể thuộc nhiều cụm khác nhau, sau đó mô hình hóa các cụm người dùng và các cụm sản phẩm độc lập nhau để thực hiện dự đoán. Kết quả thử nghiệm đã chứng tỏ phương pháp cho lại kết quả tốt hơn so với phương pháp dựa trên độ tương quan Pearson và mô hình định hướng (*Aspect Model*) [95].

### **Mô hình ngữ nghĩa ẩn:**

Mô hình ngữ nghĩa ẩn cho lọc cộng tác dựa vào các kỹ thuật thống kê, trong đó các tham biến ẩn được thiết lập trong một mô hình hỗn hợp để khám phá ra cộng đồng người dùng phù hợp với mẫu hồ sơ thích hợp. Hofmann [96] đề xuất mô hình định hướng (*AM*) cấp 3 bằng cách mở rộng mô hình định hướng cấp 2 đã được áp dụng cho bài toán phân tích ngữ nghĩa văn bản. Sau đó sử dụng thuật toán *EM (Expectation Maximization)* để ước lượng ngữ nghĩa các tham biến ẩn.

Si và Jin [66] đề xuất mô hình *MM* (*Multinomial Model*) phân loại tập người dùng với giả thiết chỉ có một kiểu người dùng duy nhất. Marlin [18] đề xuất mô hình *MMM* (*Multinomial Mixture Model*), kết hợp với mô hình định hướng (*AM*) [96] để tạo nên mô hình *URP* (*User Rating Profile*) với giả thiết có nhiều kiểu người dùng và các đánh giá mỗi người dùng độc lập nhau. Marlin khẳng định, *URP* thực hiện tốt hơn so với mô hình *AM* và *MMM* [18].

### **Mô hình phân loại và hồi qui:**

Cho tập gồm  $N$  vectơ  $M$  chiều  $\{x_i\}$ . Mục tiêu của phân loại hay hồi qui là dự đoán chính xác giá trị đầu ra tương ứng  $\{c_i\}$ . Trong trường hợp phân loại,  $c_i$  nhận một giá trị từ một tập hữu hạn gọi là tập các nhãn. Trong trường hợp hồi qui,  $c_i$  có thể nhận một giá trị thực.

Để áp dụng mô hình phân loại cho lọc cộng tác [23, 29, 84, 103, 106], mỗi sản phẩm (hoặc người dùng) được xây dựng một bộ phân loại riêng. Bộ phân loại cho sản phẩm  $y$  phân loại tập người dùng dựa trên những người dùng khác đã đánh giá sản phẩm  $y$ . Các bộ phân loại được tiến hành huấn luyện độc lập nhau trên tập các ví dụ huấn luyện.

**Một số mô hình khác:** Một số mô hình khác cũng được sử dụng trong lọc cộng tác như mô hình cực đại Entropy (*Maximization Entropy Model*) [34], mô hình đồ thị (*Graph-Based Model*) [27, 118, 119].

### **1.3.3. Những vấn đề tồn tại**

So với lọc theo nội dung, lọc cộng tác có ưu điểm là không đòi hỏi biểu diễn sản phẩm dưới dạng các đặc trưng nội dung. Ngoài ra, lọc cộng tác cho kết quả chính xác hơn trong một số ứng dụng [56, 119]. Tuy nhiên, lọc cộng tác vẫn gặp phải những hạn chế cần được tiếp tục nghiên cứu dưới đây [36, 78, 107].

- *Vấn đề người dùng mới (New User Problem)*. Cũng giống như lọc theo nội dung, để phân bổ chính xác các sản phẩm người dùng quan tâm, lọc cộng tác phải ước lượng được sở thích của người dùng đối với các sản phẩm mới thông qua những đánh giá của họ trong quá khứ. Trong trường

hợp một người dùng mới, số đánh giá của người dùng cho các sản phẩm là  $\emptyset$ , khi đó phương pháp lọc cộng tác không thể đưa ra những tư vấn chính xác cho người dùng này.

- *Vấn đề sản phẩm mới (New Item Problem)*. Trong lọc thông tin, các sản phẩm thường xuyên được bổ sung, cập nhật vào hệ thống. Khi xuất hiện một sản phẩm mới, tất cả đánh giá người dùng cho sản phẩm này đều là  $\emptyset$ . Do đó, lọc cộng tác không thể tư vấn sản phẩm cho bất kỳ người dùng nào trong hệ thống.
- *Vấn đề dữ liệu thưa (Sparsity Data Problem)*. Kết quả dự đoán của lọc cộng tác phụ thuộc chủ yếu vào số các đánh giá có trước của người dùng đối với các sản phẩm. Tuy nhiên, đối với các hệ thống thực tế, số lượng người dùng và sản phẩm là rất lớn (hàng triệu người dùng và sản phẩm), số những đánh giá biết trước thường rất nhỏ so với số lượng các đánh giá cần được dự đoán.

## 1.4. PHƯƠNG PHÁP LỌC KẾT HỢP

Lọc kết hợp hay còn gọi là phương pháp lai [ 2, 8, 10, 28, 70, 74, 80, 96, 104, 117, 122] là phương pháp kết hợp giữa cộng tác và lọc nội dung nhằm tận dụng lợi thế và tránh những hạn chế của mỗi phương pháp. So với các phương pháp khác, lọc kết hợp cho lại kết quả dự đoán tốt và có nhiều triển vọng áp dụng trong các ứng dụng thực tế. Bài toán tổng quát của lọc kết hợp được phát biểu như sau.

### 1.4.1. Bài toán lọc kết hợp

Ngoài tập người dùng  $U$ , tập sản phẩm  $P$  và ma trận lọc cộng tác  $R$  như đã được trình bày ở trên, ký hiệu  $C = \{c_1, c_2, \dots, c_K\}$  là tập  $K$  đặc trưng biểu diễn nội dung thông tin các sản phẩm  $p \in P$  hoặc người dùng  $u \in U$ . Ví dụ nếu  $p \in P$  là một bộ phim, khi đó ta có thể biểu diễn nội dung của phim thông qua các đặc trưng  $c_i$ : “thể loại”, “đạo diễn”, “diễn viên”, “hãng sản xuất” và các đặc trưng nội dung

khác của phim; nếu  $u \in U$  là một người dùng thì ta có thể xem xét đặc trưng  $c_i$  : “tuổi”, “giới tính”, “nghề nghiệp” và các đặc trưng nội dung khác phản ánh thông tin cá nhân người dùng.

Bài toán của lọc kết hợp là dự đoán cho người dùng hiện thời  $u_a$  những sản phẩm  $p_k \in P$  chưa được  $u_a$  đánh giá dựa trên ma trận đánh giá  $r_{ij}$  và các đặc trưng nội dung  $C = \{c_1, c_2, \dots, c_K\}$ .

#### 1.4.2. Các phương pháp lọc kết hợp

Lọc kết hợp được tiếp cận theo bốn xu hướng chính: Kết hợp tuyến tính, kết hợp đặc tính của lọc nội dung vào lọc cộng tác, kết hợp đặc tính của lọc cộng tác vào lọc nội dung và xây dựng mô hình hợp nhất giữa lọc cộng tác và lọc nội dung.

**Kết hợp tuyến tính** [46, 70, 74, 98] là phương pháp xây dựng hai lược đồ lọc nội dung và lọc cộng tác độc lập nhau. Kết quả dự đoán của toàn bộ mô hình có thể được lựa chọn từ phương pháp cho kết quả tốt hơn. Ưu điểm của phương pháp này là kế thừa được phương pháp biểu diễn và tính toán vốn có của các phương pháp. Nhược điểm lớn nhất của mô hình này là cho lại kết quả không cao vì chưa có sự kết hợp hiệu quả giữa nội dung và đánh giá người dùng.

**Kết hợp đặc tính của lọc nội dung vào lọc cộng tác** [23, 76, 82] là phương pháp dựa trên các kỹ thuật lọc cộng tác thuần túy nhưng vẫn duy trì hồ sơ người dùng  $ContentBasedProfile(u)$  như một tham biến tham khảo khi tính toán sự tương tự giữa các cặp người dùng. Phương pháp có thể phát hiện ra những sản phẩm tương tự với hồ sơ người dùng hoặc không tương tự với hồ sơ người dùng. Trong trường hợp dữ liệu thưa hoặc người dùng mới, mức độ tương tự giữa hồ sơ người dùng và sản phẩm sẽ được xem xét đến để tạo nên dự đoán.

**Kết hợp đặc tính của lọc cộng tác vào lọc nội dung** [10, 46, 80, 105] là phương pháp xem xét các đánh giá người dùng của lọc cộng tác như một thành phần trong mỗi hồ sơ người dùng. Phương pháp dự đoán thực hiện theo lọc nội dung thuần túy và so sánh với kết quả dựa trên biểu diễn hồ sơ người dùng mở



rộng. Phương pháp phổ biến nhất thực hiện theo mô hình này là sử dụng các kỹ thuật giảm số chiều cho hồ sơ người dùng trước khi kết hợp với đánh giá người dùng.

**Mô hình hợp nhất** (*Unifying Models*) [7, 8, 12, 23, 47, 98, 117] là phương pháp biểu diễn đặc trưng nội dung và đánh giá người dùng trên cùng mô hình. Kết quả dự đoán dựa trên mô hình dữ liệu hợp nhất của cả nội dung và đánh giá người dùng. Basu và các cộng sự [23] đề xuất sử dụng lọc cộng tác và lọc nội dung trong một bộ phân loại đơn lẻ. Schein [14] đề xuất phương pháp thống kê kết hợp hai phương pháp dựa trên mô hình phân tích ngữ nghĩa ẩn (*LSM*). Ansari [7] đề xuất mô hình hồi qui dựa trên mạng Bayes, trong đó mỗi hồ sơ người dùng và sản phẩm được biểu diễn trong cùng một mô hình thống kê. Các đánh giá chưa biết  $r_{ij}$  của người dùng  $i$  cho sản phẩm  $j$  được xác định theo công thức (1.16).

$$\begin{aligned} r_{ij} &= x_{ij}\mu + z_i\gamma_j + w_j\lambda_i + e_{ij}, \\ e_{ij} &\approx N(0, \sigma^2), \\ \lambda_i &\approx N(0, \Lambda), \\ \gamma_j &\approx N(0, \Gamma), \end{aligned} \tag{1.16}$$

Trong đó,  $i=1,2,\dots,N$  biểu diễn tập người dùng;  $j= 1, 2,\dots,M$  biểu diễn tập sản phẩm;  $e_{ij}$  là biến ngẫu nhiên điều khiển nhiễu tương tác giữa người dùng và sản phẩm,  $\lambda_i$  là biến ngẫu nhiên điều khiển nhiễu không quan sát được đối với người dùng,  $\gamma_j$  là biến ngẫu nhiên điều khiển nhiễu không quan sát được đối với sản phẩm,  $x_{ij}$  biểu diễn các đặc trưng của người dùng và sản phẩm,  $z_i$  là véc tơ các đặc trưng người dùng,  $w_j$  là véc tơ các đặc trưng của sản phẩm. Các tham biến chưa biết của mô hình là  $\mu, \sigma^2, \Lambda, \Gamma$  được ước lượng từ dữ liệu đánh giá biết trước sử dụng chuỗi Markov ẩn theo phương pháp Monte Carlo.

Tóm lại, mô hình sử dụng tập các thuộc tính người dùng  $\{z_i\}$  tạo thành một phần của hồ sơ người dùng, tập các thuộc tính sản phẩm  $\{w_j\}$  tạo thành một phần của hồ sơ sản phẩm, kết hợp với ma trận tương tác giữa người dùng với sản phẩm  $\{x_{ij}\}$  để ước lượng đánh giá chưa biết của sản phẩm.

Nhiều kết quả so sánh lọc kết hợp đã chứng tỏ phương pháp cho lại kết quả dự đoán tốt hơn so với các phương pháp lọc cộng tác và lọc nội dung thuần túy [82]. Đặc biệt, lọc kết hợp hạn chế hiệu quả vấn đề dữ liệu thưa và người dùng mới. Tuy nhiên, các phương pháp vẫn còn một số hạn chế dưới đây cần được nghiên cứu khắc phục [8, 10, 36, 38].

### 1.4.3. Những vấn đề còn tồn tại

- *Thiếu sự kết hợp hiệu quả các đặc trưng nội dung vào lọc cộng tác.*  
Không phải tất cả các đặc trưng nội dung của sản phẩm đều ảnh hưởng đến thói quen sử dụng sản phẩm của tất cả người dùng. Việc tìm ra tập các đặc trưng nội dung có ảnh hưởng quan trọng đến thói quen sử dụng sản phẩm của mỗi người dùng cụ thể, sẽ cải thiện đáng kể kết quả dự đoán của các mô hình.
- *Thiếu sự kết hợp hiệu quả các đặc tính của lọc cộng tác vào lọc nội dung.*  
Các phương pháp lọc cộng tác thực hiện dự đoán dựa trên tập đánh giá người dùng đối với sản phẩm. Trái lại, các phương pháp lọc nội dung dựa trên biểu diễn nội dung sản phẩm và hồ sơ người dùng sản phẩm. Việc thực hiện tính toán mức độ tương tự theo nội dung trên cả nội dung sản phẩm và đánh giá người dùng chưa giải quyết triệt để mâu thuẫn giữa các cách tiếp cận.

## 1.5. KẾT LUẬN

Như đã trình bày ở trên, phương pháp lọc theo nội dung thực hiện hiệu quả với các dạng thông tin được biểu diễn dưới dạng các đặc trưng nội dung nhưng lại khó thực hiện trên các dạng thông tin đa phương tiện. Lọc cộng tác cho lại kết quả tốt hơn so với lọc nội dung và có thể lọc bất kỳ dạng thông tin nào nhưng gặp phải vấn đề dữ liệu thưa, người dùng mới và sản phẩm mới. Lọc kết hợp chỉ phát huy hiệu quả nếu ta giải quyết được những mâu thuẫn trong khi kết hợp các đặc trưng nội dung vào lọc cộng tác. Chính vì vậy, luận án tập trung

ngiên cứu vào một số vấn đề còn tồn tại trong lọc cộng tác và lọc kết hợp với mục tiêu cụ thể sau:

- Nghiên cứu và đề xuất phương pháp hạn chế ảnh hưởng tình trạng dữ liệu thừa của lọc cộng tác. Phương pháp đề xuất được trình bày trong Chương 2.
- Nghiên cứu và đề xuất phương pháp kết hợp giữa lọc cộng tác và lọc nội dung để nâng cao chất lượng tư vấn. Mô hình kết hợp đề xuất được trình bày trong Chương 3.
- Xây dựng một ứng dụng dựa trên phương pháp đề xuất. Kết quả cài đặt ứng dụng được trình bày trong Phụ lục 1.

## CHƯƠNG 2

### LỌC CỘNG TÁC BẰNG PHƯƠNG PHÁP HỌC ĐA NHIỆM

Nội dung chính của chương này trình bày một phương pháp lọc cộng tác dựa trên kỹ thuật học đa nhiệm (*Multi-task Learning*). Học đa nhiệm thực hiện đồng thời nhiều bài toán phân loại nhằm phát hiện ra những đặc trưng chung cho một hoặc nhiều bài toán phân loại. Tập đặc trưng chung tìm được đóng vai trò chia sẻ và bổ sung thông tin giữa các bài toán phân loại khác nhau, góp phần nâng cao kết quả dự đoán và hạn chế ảnh hưởng của vấn đề dữ liệu thưa trong lọc cộng tác.

Để thuận tiện cho việc trình bày, vấn đề dữ liệu thưa được trình bày trong Mục 2.1. Mục 2.2 trình bày phương pháp phân loại bằng kỹ thuật Boosting. Mục 2.3 trình bày phương pháp học đa nhiệm dựa trên kỹ thuật Boosting. Mục 2.4 trình bày về phương pháp thử nghiệm và đánh giá kết quả phân loại trong trường hợp dữ liệu thưa.

#### 2.1. ĐẶT VẤN ĐỀ

Như đã trình bày trong Chương 1, một khó khăn các phương pháp lọc cộng tác gặp phải là vấn đề dữ liệu thưa [9, 14, 24, 36, 107]. Vấn đề dữ liệu thưa ảnh hưởng trực tiếp đến kết quả tính toán mức độ tương tự, xác định tập láng giềng và nhiều vấn đề liên quan khác trong lọc cộng tác. Chính vì vậy, hạn chế ảnh hưởng vấn đề dữ liệu thưa là một trong những trọng tâm nghiên cứu của lọc cộng tác [9, 24, 115]. Vấn đề dữ liệu thưa của lọc cộng tác có thể được mô tả như sau.

##### 2.1.1. Vấn đề dữ liệu thưa của lọc cộng tác

Giả sử hệ gồm  $N$  người dùng  $U = \{u_1, u_2, \dots, u_N\}$ ,  $M$  sản phẩm  $P = \{p_1, p_2, \dots, p_M\}$  với ma trận đánh giá  $R = (r_{ij})$  như đã được trình bày trong Mục 1.3.1. Trong các hệ thống lọc cộng tác, số lượng người dùng  $|U|$  và số lượng sản phẩm

$|P|$  là rất lớn. Tuy vậy, mỗi người dùng chỉ đưa ra một số rất ít các đánh giá của mình trong tập các sản phẩm. Điều này làm cho ma trận đầu vào  $r_{ij}$  có số lượng các đánh giá  $r_{ij} \neq \emptyset$  nhỏ hơn rất nhiều lần số các đánh giá  $r_{ij} = \emptyset$ . Tỷ lệ dữ liệu chưa được đánh giá trong tập dữ liệu EachMovie là 97.6% và MovieLens là 95.7%. Loại cộng tác gọi vấn đề này là vấn đề dữ liệu thưa (*SDP*).

Nhiệm vụ của các phương pháp lọc cộng tác là dự đoán cho người dùng hiện thời (*AC*) dựa trên ma trận đánh giá  $R$  với hầu hết các giá trị  $r_{ij} = \emptyset$ .

**Bảng 2.1.** Ma trận đánh giá người dùng

	p <sub>1</sub>	p <sub>2</sub>	p <sub>3</sub>	p <sub>4</sub>	p <sub>5</sub>
u <sub>1</sub>	5	∅	∅	4	4
u <sub>2</sub>	∅	4	∅	3	5
u <sub>3</sub>	∅	4	5	2	3
u <sub>4</sub>	5	∅	5	∅	∅

### 2.1.2. Ảnh hưởng của vấn đề dữ liệu thưa

- *Vấn đề dữ liệu thưa đánh giá làm cho nhiều cặp người dùng không xác định được mức độ tương tự.* Loại cộng tác gọi vấn đề này là vấn đề dữ liệu bao phủ yếu (*Reduced Coverage Problem*)[107]. Ví dụ ta cần xác định mức độ tương tự giữa người dùng  $u_4$  và  $u_2$  trong Bảng 2.1. Vì số các sản phẩm cả  $u_4$  và  $u_2$  đều đánh giá không phủ nhau hay không giao nhau, do đó độ tương tự giữa  $u_4$  và  $u_2$  tính toán theo các độ đo tương tự là 0. Điều này ảnh hưởng trực tiếp đến phương pháp huấn luyện và kết quả dự đoán vì các đánh giá khác  $\emptyset$  của người dùng  $u_2$  không bao giờ được xem xét đến trong quá trình huấn luyện và tham gia đóng góp vào kết quả dự đoán cho người dùng  $u_4$ .
- *Vấn đề dữ liệu thưa làm cho việc xác định tập hàng xóm cho người dùng hiện thời kém tin cậy.* Ví dụ ta cần dự đoán các sản phẩm cho người dùng

$u_4$  trong Bảng 2.1, dựa trên các độ đo tương tự ta sẽ tính toán được  $u_4$  tương tự với  $u_1$  vì  $r[u_1, p_1] = r[u_4, p_4] = 5$ . Kết quả là các sản phẩm  $p_4, p_5$  sẽ được phân bổ cho  $u_4$  vì  $u_4$  tương tự với  $u_1$  và  $u_1$  “*thích*”  $p_4, p_5$ . Tuy nhiên, ta cũng tính toán được  $u_4$  tương tự với  $u_3$  vì  $r[u_3, p_3] = r[u_4, p_3] = 5$ , do đó  $p_4, p_5$  sẽ bị gỡ bỏ trong danh mục các sản phẩm phân bổ cho  $u_4$  vì  $u_4$  tương tự với  $u_3$  và  $u_3$  “*không thích*”  $p_4, p_5$ . Như vậy, nếu coi hoặc  $u_1$  hoặc  $u_3$  là láng giềng của  $u_4$  thì kết quả dự đoán trở nên kém tin cậy, nếu xem xét cả  $u_1$  và  $u_3$  đều là láng giềng của  $p_4$  thì xảy ra mâu thuẫn vì  $u_1$  và  $u_3$  hoàn toàn không tương tự nhau.

- *Vấn đề dữ liệu thừa làm cho việc giải quyết bài toán đánh giá ban đầu (The First Rater Problem) gặp nhiều khó khăn.* Khi hệ thống có thêm một người dùng mới, người dùng này cần có một số đánh giá ban đầu cho một vài sản phẩm thì hệ thống mới có thể dự đoán cho họ những sản phẩm tiếp theo. Tương tự như vậy đối với các sản phẩm mới chưa được bất kỳ người dùng nào đánh giá, sản phẩm này chỉ được tư vấn đến người dùng khi có một vài người dùng đánh giá. Loại cộng tác còn gọi những vấn đề này là *vấn đề xuất phát chậm (Cold Start Problem)*.

### 2.1.3. Các phương pháp hạn chế vấn đề dữ liệu thừa

Hướng tiếp cận phổ biến để hạn chế ảnh hưởng vấn đề dữ liệu thừa dựa vào các phương pháp giảm số chiều của ma trận đánh giá. Về bản chất, những phương pháp này hạn chế vấn đề dữ liệu thừa bằng cách tạo nên ma trận tương tác đặc hơn, sau đó sử dụng ma trận này để tính toán mức độ tương quan giữa người dùng hoặc sản phẩm.

Chiến lược đơn giản nhất để giảm số chiều của ma trận đánh giá là tạo lập nên các cụm sản phẩm hoặc cụm người dùng, sau đó sử dụng những cụm này như những đơn vị cơ bản để sinh ra dự đoán [14, 20, 24, 55, 103]. Ungar và Foster [68] sử dụng kỹ thuật K-median phân cụm người dùng và sản phẩm độc lập nhau, sau đó các cụm người dùng và sản phẩm được phân cụm lại để tạo nên các

cụm có mức độ tương tự cao theo cả người dùng và sản phẩm. Si và Jin [66] thực hiện phân cụm đồng thời cho cả người dùng và sản phẩm. Mô hình cho phép người dùng hoặc sản phẩm có thể ở những cụm khác nhau. Kết quả dự đoán được thực hiện trong cụm người dùng hoặc sản phẩm có mật độ đánh giá cao nhất.

Phương pháp giảm số chiều của ma trận đánh giá bằng các kỹ thuật thống kê được quan tâm nhiều hơn so với các kỹ thuật phân cụm [20, 29, 62, 79]. Billsus và Pazzani [29] đề xuất việc sử dụng phương pháp phát hiện ngữ nghĩa ẩn (*LSM*) dựa trên kỹ thuật phân rã giá trị riêng (*SVD*). K.Goldberg cùng các cộng sự [62] cải tiến phương pháp phân cụm sử dụng kỹ thuật phân tích thành phần chính (*PCA*). Tuy nhiên, trong nhiều trường hợp thông tin hữu ích có thể bị mất trong quá trình giảm chiều ma trận làm cho kết quả dự đoán gặp nhiều hạn chế.

Một hướng tiếp cận khác hạn chế vấn đề dữ liệu thưa dựa vào việc khai thác các mối liên hệ gián tiếp trên ma trận đánh giá. Huang [119] biểu diễn người dùng và sản phẩm như một đồ thị hai phía (*Bipart Graph Model*), một phía là tập người dùng, phía còn lại là tập sản phẩm, mỗi cạnh nối từ đỉnh người dùng đến đỉnh sản phẩm được thiết lập nếu người dùng đã mua hoặc đánh giá cao cho sản phẩm tương ứng. Dựa trên biểu diễn mối quan hệ người dùng và sản phẩm, dữ liệu được điền vào các ô còn trống trong ma trận đánh giá thực hiện bằng cách lan truyền có trọng số trên đồ thị hai phía.

Desrosiers và Karypis [24] hạn chế vấn đề dữ liệu thưa bằng độ tương quan gián tiếp (*Indirect Similarity*). Trong phương pháp này, mức độ tương tự giữa các cặp người dùng không chỉ được tính toán dựa trên tập sản phẩm cả hai người dùng cùng đánh giá, mà còn được tăng cường thêm giá trị tương tự gián tiếp được tính dựa trên tập sản phẩm hai người dùng đánh giá không giao nhau.

Phương pháp hạn chế vấn đề dữ liệu thưa của lọc cộng tác đề xuất trong chương này được thực hiện dựa trên kỹ thuật học đa nhiệm [3, 81]. Học đa nhiệm cho phép phát hiện ra các đặc trưng chung cho một hoặc nhiều người dùng khác nhau. Các đặc trưng chung tìm được đóng vai trò chia sẻ, bổ sung

thông tin cho những người dùng khác sẽ làm tăng dữ liệu huấn luyện, vì vậy nâng cao kết quả dự đoán và hạn chế được ảnh hưởng của tình trạng dữ liệu thưa của lọc cộng tác.

## 2.2. LỌC CỘNG TÁC BẰNG PHÂN LOẠI

Lọc cộng tác có thể phát biểu như bài toán phân loại tự động của học máy [23, 29, 81, 106, 108]. Dựa trên đánh giá của người dùng về những sản phẩm khác nhau, một mô hình phân loại sẽ được xây dựng và huấn luyện cho mỗi người dùng. Mô hình này sau đó được sử dụng để phân chia sản phẩm mới thành các loại khác nhau, ví dụ như loại “*thích*” và “*không thích*”. Tương tự như vậy, có thể thay đổi vai trò giữa người dùng và sản phẩm, cho phép ta xây dựng được các bộ phân loại cho mỗi sản phẩm để dự đoán một sản phẩm cụ thể có “*thích*” hay “*không thích*” đối với người dùng. Bài toán lọc cộng tác bằng phân loại được phát biểu như sau.

### 2.2.1. Phát biểu bài toán lọc cộng tác bằng phân loại

Cho ma trận đánh giá người dùng  $R = (r_{ij})$  như được trình bày ở trên. Các hàng của ma trận tương ứng với tập người dùng, các cột của ma trận tương ứng với tập sản phẩm, các phần tử  $r_{ij}$  của ma trận tương ứng với đánh giá của người dùng đối với sản phẩm. Thông thường, mỗi người dùng chỉ đánh giá một tập rất nhỏ các mặt hàng và do vậy đa số các giá trị  $r_{ij}$  được để trống ( $r_{ij} = \emptyset$ ). Nhiệm vụ của lọc cộng tác là điền vào hay dự đoán các giá trị thích hợp vào các ô trống cho mỗi hàng của ma trận đánh giá.

Tiếp cận cho lọc cộng tác bằng phân loại, ta cần cá nhân hóa mô hình học cho mỗi người dùng. Mỗi người dùng sẽ được xây dựng riêng một bộ phân loại. Mỗi bộ phân loại dự đoán các giá trị trống cho một hàng của ma trận đánh giá. Ví dụ với ma trận đầu vào của lọc cộng tác  $R = (r_{ij})$  mô tả hệ gồm 4 người dùng và 5 sản phẩm trong Bảng 2.2, ta cần xây dựng 4 bộ phân loại khác nhau cho 4 người dùng  $u_1, u_2, u_3, u_4$ . Giả sử ta cần dự đoán cho người dùng  $u_4$  về các sản



phẩm  $p_4$  và  $p_5$ . Ta cần huấn luyện một thuật toán học dựa vào thông tin đánh giá trước đó của người dùng  $u_4$  cho các sản phẩm. Trong Bảng 2.2, người dùng  $u_4$  đã đánh giá 3 sản phẩm  $p_1, p_2, p_3$ . Điều này chỉ ra 3 ví dụ huấn luyện  $p_1, p_2, p_3$  sẽ được dùng để sinh ra dự đoán cho người dùng  $u_4$ .

**Bảng 2.2.** Ma trận đầu vào của lọc cộng tác

	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$
$u_1$	5	2	$\emptyset$	4	4
$u_2$	$\emptyset$	4	5	3	$\emptyset$
$u_3$	4	5	2	$\emptyset$	3
$u_4$	5	3	4	?	?

Mỗi ví dụ huấn luyện được biểu diễn dưới dạng một véc tơ đặc trưng. Mỗi đặc trưng tương ứng với một người dùng khác người dùng cần dự đoán (người dùng  $u_1, u_2, u_3$ ). Giá trị khác rỗng của ma trận đánh giá là giá trị các đặc trưng (ví dụ  $r_{1,1}, r_{1,2}, r_{2,3}, r_{2,4}$  là các giá trị đặc trưng ứng với người dùng  $u_1, u_2$ ). Nhãn phân loại cho các ví dụ huấn luyện là những đánh giá khác  $\emptyset$  của người dùng hiện thời (ví dụ  $r_{4,1}, r_{4,2}, r_{4,3}$  là các nhãn phân loại cho người dùng  $u_4$ ).

Một vấn đề đặt ra trong biểu diễn này là nhiều giá trị đặc trưng có giá trị rỗng ( $r_{ij} = \emptyset$ ) chưa được điền giá trị (ví dụ  $r_{1,3}, r_{2,1}$ ). Để khắc phục điều này, ta chỉ cần thực hiện một biến đổi đơn giản đưa ma trận đánh giá  $R = \{ r_{ij} \mid r_{ij} = \emptyset, 1, 2, \dots, V \}$  thành ma trận  $R = \{ r_{ij} \mid r_{ij} = -1, 0, 1 \}$ . Trong đó, các giá trị  $r_{ij} > \theta$  được biến đổi thành +1; các giá trị  $r_{ij} \leq \theta$  được biến đổi thành -1;  $r_{ij} = \emptyset$  được biến đổi thành 0;  $\theta$  là một giá trị ngưỡng được xác định tùy thuộc vào tập dữ liệu kiểm nghiệm. Ở đây, giá trị  $r_{ij} = 1$  biểu diễn người dùng  $u_i$  “*thích*” sản phẩm  $p_j$ ,  $r_{ij} = -1$  biểu diễn người dùng  $u_i$  “*không thích*” sản phẩm  $p_j$ ,  $r_{ij} = 0$  biểu diễn người dùng  $u_i$  chưa đánh giá hoặc chưa bao giờ biết đến sản phẩm  $p_j$ .

Ví dụ với ma trận đánh giá được cho trong Bảng 2.2, ma trận đầu vào cho các bài toán phân loại được chuyển đổi thành ma trận trong Bảng 2.3. Các giá trị

$r_{ij} > 3$  được chuyển đổi thành +1, các giá trị  $r_{ij} \leq 3$  được chuyển đổi thành -1, những giá trị  $\emptyset$  còn lại được điền là giá trị 0.

**Bảng 2.3.** Ma trận đầu vào bài toán phân loại theo người dùng

	p <sub>1</sub>	p <sub>2</sub>	p <sub>3</sub>	p <sub>4</sub>	p <sub>5</sub>
u <sub>1</sub>	1	-1	0	1	1
u <sub>2</sub>	0	1	1	-1	0
u <sub>3</sub>	1	1	-1	0	-1
u <sub>4</sub>	1	-1	1	?	?

Tương tự như trên, ta có thể thay đổi vai trò giữa người dùng và sản phẩm để xây dựng nên các bộ phân loại cho các sản phẩm. Mỗi bộ phân loại thực hiện dự đoán sản phẩm tương ứng phù hợp hoặc không phù hợp với những người dùng nào. Ví dụ ta có thể thay đổi vai trò giữa người dùng và sản phẩm trong Bảng 2.2 và thực hiện biến đổi như trên ta được ma trận đầu vào cho bài toán phân loại cho các sản phẩm trong Bảng 2.4.

**Bảng 2.4.** Ma trận đầu vào bài toán phân loại theo sản phẩm

	u <sub>1</sub>	u <sub>2</sub>	u <sub>3</sub>	u <sub>4</sub>
p <sub>1</sub>	1	0	1	1
p <sub>2</sub>	-1	1	1	-1
p <sub>3</sub>	0	1	-1	1
p <sub>4</sub>	1	-1	0	0
p <sub>5</sub>	1	?	-1	?

Với ví dụ huấn luyện như trên, bài toán phân loại có thể thực hiện bằng những phương pháp phân loại thông dụng, ví dụ mạng nơron nhân tạo, cây quyết định, máy hỗ trợ vectơ (SVM). Tuy nhiên, trước khi sử dụng trực tiếp dữ liệu huấn luyện và phân loại, một vấn đề cần giải quyết là trích chọn đặc trưng.

Trong trường hợp trình bày ở đây, mỗi đặc trưng chính là đánh giá của một người dùng khác với người dùng đang xét (trong ví dụ ở Bảng 2.2, bài toán phân loại cho người dùng  $u_4$  có 3 đặc trưng là đánh giá của người dùng  $u_1, u_2, u_3$ ). Trên thực tế, số lượng đặc trưng rất lớn và không phải đặc trưng nào cũng liên quan tới đánh giá của người dùng đang xét. Việc sử dụng cả các đặc trưng không liên quan làm tăng độ phức tạp tính toán đồng thời làm giảm độ chính xác phân loại.

Để giải quyết vấn đề trích chọn đặc trưng, Billsus và Pazzani [29] sử dụng phương pháp SVD để phân tích ma trận đánh giá thành tích của ma trận bao gồm các vectơ riêng và ma trận đường chéo bao gồm các giá trị riêng, sau đó rút gọn kích thước ma trận bằng cách chỉ giữ lại những vectơ riêng tương ứng với những giá trị riêng lớn nhất. Nhờ vậy, những đặc trưng ban đầu được biến đổi thành đặc trưng mới. Đặc điểm của đặc trưng mới là số lượng đặc trưng ít hơn, nhưng sau khi chiếu dữ liệu xuống đặc trưng mới sẽ cho phương sai lớn hơn so với khi chiếu xuống đặc trưng gốc, do vậy dễ phân loại dữ liệu hơn.

Trong phạm vi luận án, chúng tôi đề xuất một cách tiếp cận khác dựa trên việc sử dụng kỹ thuật Boosting cho bài toán phân loại của lọc cộng tác. Nội dung cụ thể của phương pháp được trình bày trong Mục 2.2.2.

### **2.2.2. Phân loại bằng phương pháp Boosting**

Boosting là phương pháp học máy cho phép tạo ra bộ phân loại mạnh (*Strong Classifier*) có độ chính xác cao bằng cách kết hợp nhiều bộ phân loại có độ chính xác kém hơn hay còn được gọi là bộ phân loại yếu (*Weak Classifier*). Dựa trên nguyên tắc chung này, nhiều phiên bản khác nhau của kỹ thuật Boosting đã được đề xuất và sử dụng [50, 91, 110, 111]. Trong đó, phiên bản Gentle AdaBoost (viết tắt là GentleBoost) được Friedman đề xuất có nhiều ưu điểm như đơn giản, ổn định và cho kết quả phân loại tốt trong nhiều ứng dụng [13].

Phương pháp GentleBoost cho trường hợp phân loại hai lớp có thể mô tả tóm tắt như sau. Cho tập dữ liệu huấn luyện bao gồm  $M$  ví dụ  $(x_1, y_1), \dots, (x_M, y_M)$

với  $x_i$  là vectơ các đặc trưng và  $y_i$  là nhãn phân loại nhận giá trị  $y_i = +1$  hoặc  $y_i = -1$  (tương ứng với “*thích*” và “*không thích*”). Bộ phân loại mạnh  $F(x)$  được tạo thành bằng cách tổ hợp tuyến tính các bộ phân loại yếu.

$$F(x) = \sum_{k=1}^K f_k(x) \quad (2.1)$$

Trong đó,  $x$  là vectơ đặc trưng đầu vào,  $f_k(x)$  là bộ phân loại yếu có khả năng dự đoán nhãn phân loại cho vectơ đầu vào  $x$ ,  $K$  là số vòng lặp. Kết quả phân loại cuối cùng được tạo ra bằng cách tính  $\text{sign}(F(x))$ .

Thuật toán bao gồm  $K$  vòng lặp. Tại vòng lặp thứ  $k$ , các ví dụ huấn luyện sẽ được đánh trọng số lại sao cho những ví dụ bị phân loại sai trong vòng trước nhận được trọng số cao hơn và do vậy được bộ phân loại chú ý hơn. Bộ phân loại  $f_k(x)$  được huấn luyện trên dữ liệu có trọng số trong vòng thứ  $k$ . Thuật toán GentleBoost được thể hiện trong Hình 2.1.

**Đầu vào:**

- Tập dữ liệu huấn luyện gồm  $M$  ví dụ  $(x_1, y_1), \dots, (x_M, y_M)$  với  $x_i$  là vectơ các đặc trưng và  $y_i$  là nhãn phân loại nhận giá trị  $y_i = +1$  hoặc  $y_i = -1$ .
- $K$  là số vòng lặp ( $k \leq 200$ ).

**Đầu ra:**

- Trả lại  $\text{sign}[F(x)] = \text{sign}[\sum_{k=1}^K f_k(x)]$

**Các bước thực hiện:**

1. Khởi tạo các trọng số  $w_i = 1/M$ ,  $i = 1..M$ ,  $w_i$  là trọng số của ví dụ huấn luyện thứ  $i$ .  
 Khởi tạo  $F(x) = 0$
2. Lặp với  $k = 1, 2, \dots, K$ 
  - a. Huấn luyện  $f_k(x)$  sử dụng dữ liệu huấn luyện có trọng số
  - b. Cập nhật  $F(x) \leftarrow F(x) + f_k(x)$
  - c. Cập nhật trọng số  $w_i \leftarrow w_i e^{-y_i f_k(x)}$  ( $i=1, 2, \dots, M$ ) và chuẩn tắc hoá trọng số
3. Trả về bộ phân loại  $\text{sign}[F(x)] = \text{sign}[\sum_{k=1}^K f_k(x)]$

**Hình 2.1.** Thuật toán GentleBoost.

Tại bước (a) của mỗi vòng lặp, thuật toán lựa chọn  $f_k(x)$  sao cho sai số phân loại (2.2) nhỏ nhất:

$$J = \sum_{i=1}^M w_i (y_i - f_k(x_i))^2 \quad (2.2)$$

Để tìm được bộ phân loại cho phép cực tiểu hoá (2.2), cần xây dựng bộ phân loại yếu  $f_k(x)$  cho phép cực tiểu hoá bình phương lỗi phân loại có tính tới trọng số. Có nhiều phương pháp xây dựng bộ phân loại yếu khác nhau. Ở đây, bộ phân loại yếu được sử dụng là gốc quyết định (*Decision Stump*). Gốc quyết định là phiên bản đơn giản của cây quyết định với một nút duy nhất. Gốc quyết định lựa chọn một đặc trưng của ví dụ huấn luyện, sau đó tùy thuộc vào giá trị của đặc trưng để gán cho nhãn giá trị 1 hay  $-1$ .

$$f_k(x) = a\delta(x^f > t) + b\delta(x^f \leq t) \quad (2.3)$$

Trong đó  $\delta(e) = 1$  nếu  $e$  đúng và  $\delta(e) = 0$  nếu ngược lại,  $t$  là một giá trị ngưỡng,  $a$  và  $b$  là tham số,  $x^f$  là giá trị đặc trưng thứ  $f$  của vectơ  $x$ . Trong trường hợp dữ liệu đánh giá chỉ bao gồm giá trị 1 và 0 hoặc 1 và  $-1$ , có thể chọn ngưỡng  $t = 0$ . Như vậy, ngoài việc phân loại, gốc quyết định còn thực hiện trích chọn đặc trưng, mỗi gốc chỉ chọn một đặc trưng duy nhất.

Quá trình huấn luyện để chọn ra gốc tốt nhất được thực hiện bằng cách thử tất cả đặc trưng  $f$ . Với mỗi giá trị của  $f$ , giá trị tối ưu của  $a$  và  $b$  được ước lượng theo kỹ thuật bình phương tối thiểu (*Least Square Estimation*) mà bản chất là tính giá trị tham số tại điểm có đạo hàm bằng 0.

$$a = \frac{\sum_i w_i y_i \delta(x^f > 0)}{\sum_i w_i \delta(x^f > 0)} \quad (2.4)$$

$$b = \frac{\sum_i w_i y_i \delta(x^f \leq 0)}{\sum_i w_i \delta(x^f \leq 0)} \quad (2.5)$$

Giá trị  $f$ ,  $a$  và  $b$  tính được cho sai số dự đoán (2.2) nhỏ nhất sẽ được chọn để tạo ra bộ phân loại  $f_k(x)$  cho vòng lặp thứ  $k$ . Bộ phân loại yếu  $f_k(x)$  sau đó được cộng thêm vào bộ phân loại chính  $F(x)$  tại bước b.

Tại bước (c), thuật toán tiến hành cập nhật lại trọng số  $w_i \leftarrow w_i e^{-y_i f_k(x_i)}$ . Các ví dụ phân loại sai có  $y_i f_k(x_i) < 0$  được GentleBoost tăng trọng số, các ví dụ phân loại đúng có  $y_i f_k(x_i) > 0$  bị giảm trọng số. Với cách làm này, thuật toán sẽ khiến bộ phân loại chú ý hơn tới những ví dụ hiện đang bị phân loại sai.

Friedman và các cộng sự chứng minh việc thêm  $f_k(x)$  vào  $F(x)$  tại bước  $k$  đáp ứng được kỳ vọng mong muốn thông qua các bước cập nhật của phép khai triển Niuton [50].

**Mệnh đề 2.1.** *Thuật toán GentleBoost cực tiểu hóa hàm lỗi phân loại thông qua các bước của phép khai triển Niuton.*

**Chứng minh.** Thực chất, GentleBoost xây dựng bộ phân loại yếu bằng cách sử dụng phương pháp Niuton để cực tiểu hóa hàm lỗi khi phân loại. Dưới đây là những phân tích chứng minh cho khẳng định này.

Trong bài toán phân loại nói chung, giá trị hàm lỗi được tính bằng kỳ vọng của số lượng ví dụ bị phân loại sai, tức là bằng:

$$E[\delta(F(x) \neq y)] = \frac{1}{M} \sum_{i=1}^M \delta(F(x_i) \neq y_i) \quad (2.6)$$

Trong đó,  $E[.]$  là kỳ vọng và được tính bằng tỷ lệ lỗi trên dữ liệu huấn luyện. Do hàm (2.6) không khả vi, nên thay vào đó các thuật toán Boosting sử dụng hàm lỗi theo tiêu chuẩn hàm mũ dưới dạng

$$J = E[e^{-yF(x)}] \quad (2.7)$$

Dễ dàng nhận thấy hàm (2.6) là giới hạn trên của (2.5) (do  $e^{-y_i F(x_i)} > 1$  khi  $F(x_i) \neq y_i$ ), vì vậy cực tiểu hóa (2.6) cho phép giảm giá trị hàm lỗi (2.5).

Thuật toán GentleBoost xây dựng hàm phân loại sao cho giá trị lỗi  $J$  nhỏ nhất bằng cách cải thiện dần hàm phân loại. Tại bước thứ  $k$ , ta cần cải thiện hàm

$F(x)$  bằng cách cập nhật  $F(x) + f_k(x)$ . Thuật toán sẽ lựa chọn  $f_k(x)$  sao cho giá trị hàm lỗi  $J$  giảm nhiều nhất. Khai triển hàm  $J$  dưới dạng chuỗi Taylor bậc 2, ta có:

$$\begin{aligned} J(F(x) + f_k(x)) &= E[e^{-y(F(x)+f_k(x))}] \\ &\approx E[e^{-yF(x)}(1 - yf_k(x) + y^2 f_k(x)^2 / 2)] \\ &= E[e^{-yF(x)}(1 + (y - f_k(x))^2 / 2)] \quad \text{do } y^2 = 1 \end{aligned}$$

Do hằng số và hệ số  $1/2$  không ảnh hưởng tới  $f_k(x)$ , ta có thể viết:

$$f_k(x) = \arg \min_{f_k} E[e^{-yF(x)}(y - f_k(x))^2] \quad (2.8)$$

Thay kỳ vọng bằng giá trị trung bình trên dữ liệu huấn luyện và sử dụng ký hiệu trọng số  $w_i = e^{-y_i F(x_i)}$ , (2.8) được viết lại như sau:

$$f_k(x) = \arg \min_{f_k} \left[ \sum_{i=1}^M w_i (y_i - f_k(x_i))^2 \right] \quad (2.9)$$

Giá trị cần tối ưu chính là hàm lỗi (2.2) đã nói ở trên. Công thức này giải thích việc lựa chọn  $f_k(x)$  bằng kỹ thuật  $LSE$  và cách thức cập nhật trọng số  $w_i$  tại bước  $k$ .

Tiếp theo, ta thấy việc cập nhật  $F(x)$  sử dụng  $f_k(x)$  theo (2.8) tương ứng với các bước trong phương pháp Newton để cực tiểu hoá hàm lỗi. Thật vậy, sử dụng một số biến đổi đơn giản, ta có:

$$\left. \frac{\partial J(F(x) + f_k(x))}{\partial (f_k(x))} \right|_{f_k(x)=0} = -E[e^{-yF(x)} y] \quad (2.10)$$

$$\left. \frac{\partial^2 J(F(x) + f_k(x))}{\partial^2 (f_k(x))} \right|_{f_k(x)=0} = E[e^{-yF(x)} y^2] = E[e^{-yF(x)}] \quad (2.11)$$

Các bước cập nhật của phương pháp Newton khi đó được tính bởi:

$$F(x) \leftarrow F(x) + \frac{E[e^{-yF(x)} y]}{E[e^{-yF(x)}]} \quad (2.12)$$

Thay trọng số  $w_i = e^{-y_i F(x_i)}$  và xác định  $f_k(x)$  từ (2.9) bằng cách cho đạo hàm bằng 0, ta có:

$$f_k(x) = \frac{E[e^{-yF(x)} y]}{E[e^{-yF(x)}]} \quad (2.13)$$

Như vậy, việc thêm  $f_k(x)$  vào  $F(x)$  tại bước  $k$  đáp ứng được kỳ vọng mong muốn tương ứng với các bước cập nhật của phương pháp Niuton và thuật toán sẽ hội tụ nếu chọn  $K$  đủ lớn. Trên thực tế, các thử nghiệm cho thấy thuật toán GentleBoost cho kết quả tốt với  $K=200$  vòng lặp [50].

## 2.3. PHÂN LOẠI VỚI CÁC ĐẶC TRƯNG CHUNG

Với phương pháp trình bày ở trên, quá trình trích chọn đặc trưng và huấn luyện bộ phân loại cho người dùng  $u_i$  được thực hiện trên dữ liệu tạo thành từ những sản phẩm mà người dùng này đã có đánh giá. Thông thường, mỗi người dùng chỉ đánh giá một tập rất nhỏ các sản phẩm, do vậy mỗi bộ phân loại chỉ được huấn luyện trên một lượng dữ liệu nhỏ. Đây là yếu tố dẫn tới hiệu quả phân loại thấp. Để giải quyết nhược điểm nói trên, mục này sẽ trình bày phương pháp huấn luyện và trích chọn đặc trưng đồng thời cho tất cả người dùng thay vì cho từng người riêng rẽ như vừa mô tả ở trên. Việc huấn luyện đồng thời cho phép kết hợp thông tin và dữ liệu huấn luyện từ những người dùng khác, nhờ vậy giảm bớt yêu cầu có nhiều sản phẩm được đánh giá trước cho mỗi người dùng. Đây là một kỹ thuật thường được gọi là học đa nhiệm.

### 2.3.1. Phương pháp học đa nhiệm

Để thuận lợi cho việc trình bày, trong phần này chúng tôi tóm tắt về phương pháp học đa nhiệm trước khi chuyển sang trình bày về đề xuất sử dụng học đa nhiệm dựa trên Boosting cho bài toán lọc cộng tác.

Hầu hết các phương pháp học máy cho lọc cộng tác hiện nay đều thực hiện những nhiệm vụ học đơn lẻ. Kết quả của mỗi nhiệm vụ cụ thể hoàn toàn độc lập với các nhiệm vụ khác. Trên thực tế, kết quả của mỗi bài toán phân loại cho từng người dùng không hoàn toàn độc lập nhau, kết quả của bài toán phân loại này có thể được dùng làm ví dụ huấn luyện cho bài toán phân loại khác. Chẳng hạn, ta có thể sử dụng kết quả phương pháp học nhận ra quả táo để áp

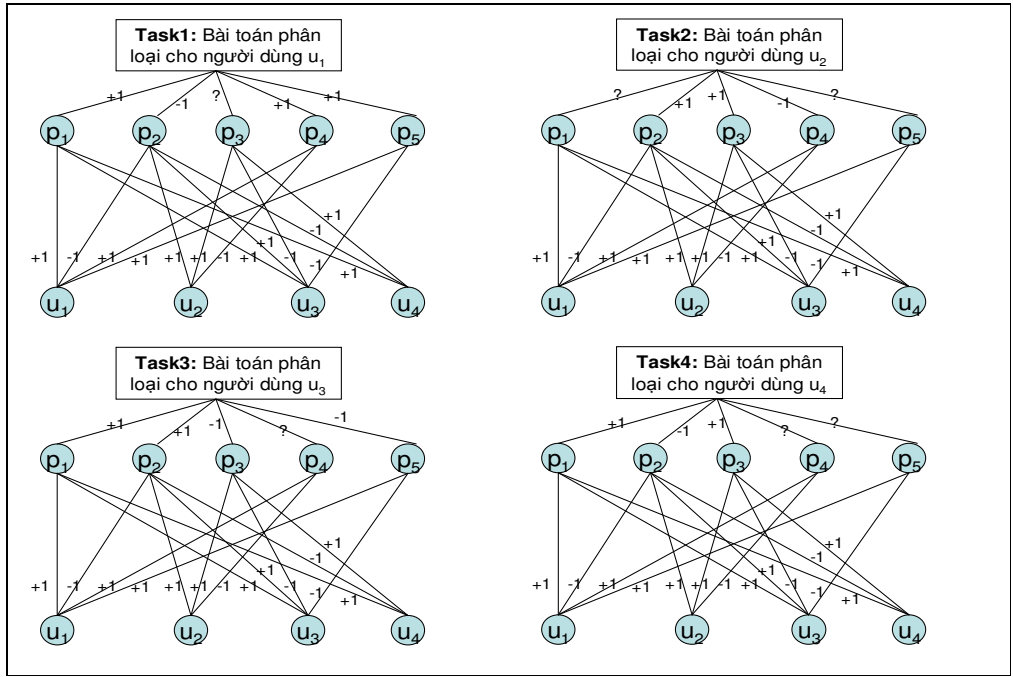


dụng cho việc học nhận ra quả lê, sử dụng phương pháp học chơi đàn Violin để học cách chơi đàn Organ. Để thực hiện điều này, trước khi thực hiện nhiệm vụ nào đó ta thường nhớ lại và chuyển giao những tri thức nhận được để thực hiện những nhiệm vụ khác.

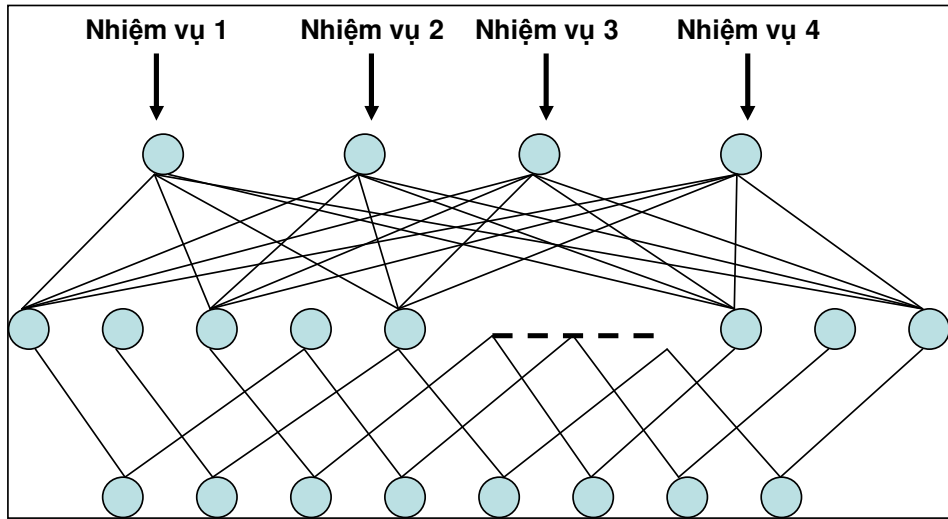
Phương pháp học máy thực hiện đồng thời từ nhiều nhiệm vụ liên quan để nâng cao kết quả dự đoán được gọi là phương pháp học đa nhiệm [3, 48, 81, 87]. Bằng việc suy diễn đồng thời giữa các nhiệm vụ, học đa nhiệm phát hiện ra những tri thức từ nhiều nhiệm vụ để tăng cường vào kết quả dự đoán cho mỗi nhiệm vụ đơn lẻ. Với những bài toán có số lượng nhiệm vụ lớn nhưng có số ví dụ huấn luyện ít, học đa nhiệm nâng cao kết quả dự đoán cho mỗi nhiệm vụ bằng cách chia sẻ thông tin chung giữa các nhiệm vụ.

Hình 2.2 mô tả phương pháp học đơn lẻ cho bốn bài toán phân loại với ma trận đầu vào được cho trong Bảng 3.3. Mỗi bài toán phân loại được xem xét như một nhiệm vụ cần dự đoán. Mỗi nhiệm vụ được biểu diễn như một đồ thị, trong đó các cạnh nối từ đỉnh người dùng  $u_i$  đến đỉnh sản phẩm  $p_j$  được đánh trọng số là giá trị đặc trưng tương ứng. Các cạnh nối từ đỉnh sản phẩm đến đỉnh nhiệm vụ tương ứng (Task1, Task2, Task3, Task4) được đánh trọng số là các nhãn phân loại, trọng số được đánh dấu “?” là nhãn chưa biết cần được dự đoán.

Trong ví dụ này, bài toán phân loại cho các người dùng khác nhau được tiến hành huấn luyện độc lập nhau trên cùng một tập thông tin vào. Kết quả mỗi bài toán phân loại là tập các nhãn đầu ra cho các sản phẩm chưa được người dùng đánh giá. Quá trình huấn luyện và dự đoán không xử lý bất kỳ mối quan hệ nào giữa các nhiệm vụ. Điều này làm cho các phương pháp học đơn lẻ cho lại kết quả thấp trong trường hợp có ít dữ liệu huấn luyện.



Hình 2.2. Phương pháp STL cho bốn bài toán phân loại độc lập nhau.



Hình 2.3. Phương pháp học MTL cho bốn bài toán phân loại đồng thời

Trái lại, phương pháp học đa nhiệm tiến hành thực hiện huấn luyện đồng thời cho các nhiệm vụ (Hình 2.3). Các nhiệm vụ chia sẻ các giá trị đặc trưng và kết quả huấn luyện thông qua một lớp ẩn. Lớp ẩn được xây dựng từ việc xử lý

các mối quan hệ giữa các nhiệm vụ để tăng cường cho các nhiệm vụ đơn lẻ. Đây cũng là trọng tâm nghiên cứu của học đa nhiệm.

Nhiều đề xuất xem xét mối liên hệ giữa các nhiệm vụ trong học đa nhiệm. Hướng tiếp cận phổ biến xem xét các mối liên hệ như những ràng buộc cứng biết trước để thực hiện quá trình huấn luyện. Những đề xuất khác xem xét các nhiệm vụ chia sẻ với nhau thông qua một phần của tham biến [13]. Trong mục tiếp theo, chúng tôi xem xét các bài toán phân loại chia sẻ với nhau thông qua một tập các giá trị đặc trưng chung và tìm chiến lược tối ưu phương pháp dự đoán dựa trên giá trị của các đặc trưng chung.

### **2.3.2. Boosting đồng thời cho nhiều bài toán phân loại**

Lọc cộng tác có thể được thực hiện theo phương pháp học đa nhiệm bằng kỹ thuật Boosting như đã trình bày trong Mục 2.2.2. Để thực hiện điều này, ta có thể xem xét mỗi bài toán phân loại như một nhiệm vụ. Sau đó, thay thế việc thực hiện từng bài toán phân loại đơn lẻ cho mỗi người dùng bằng việc thực hiện đồng thời cho tập con các bài toán phân loại. Thay việc giảm sai số cho một bài toán phân loại bằng việc giảm sai số đồng thời cho tập con các bài toán phân loại. Với cách làm này, tại mỗi vòng lặp thuật toán Boosting tìm được một *đặc trưng chung* cho tất cả các bài toán trong tập con các bài toán phân loại. Đặc trưng chung tìm được đóng vai trò chia sẻ và bổ sung thông tin giữa các bài toán phân loại để tăng cường thêm vào kết quả dự đoán. Phương pháp Boosting cải tiến cho nhiều bài toán phân loại được thực hiện như sau.

#### **2.3.2.1. Xây dựng hàm mục tiêu**

Với tập  $N$  người dùng  $U$ ,  $M$  sản phẩm  $P$ , và giá trị đánh giá  $r_{ij}$  như đã trình bày ở trên, ta có tất cả  $N$  bài toán phân loại, bài toán thứ  $n$ ,  $n = 1, \dots, N$  được cho bởi  $M$  ví dụ huấn luyện  $(x_1^n, y_1^n), \dots, (x_M^n, y_M^n)$ , trong đó  $y_j^n = r_{nj}$  tức là đánh giá của người dùng  $n$  cho sản phẩm  $j$ , và  $x_j^n = (r_{1j}, \dots, r_{(n-1)j}, r_{(n+1)j}, \dots, r_{Nj})$  tức là đánh giá của tất cả người dùng cho sản phẩm  $j$  trừ người dùng  $n$ . Cần lưu ý rằng, chỉ những cột có  $r_{nj} \neq \emptyset$  mới được sử dụng làm ví dụ huấn luyện trong bài toán thứ

$n$  (cột của sản phẩm 1, 2, 3 trong ví dụ ở Bảng 2.3). Tuy nhiên, ta vẫn liệt kê cả những ví dụ có  $r_{nj} = \emptyset$ . Những ví dụ này sau đó sẽ được gán trọng số bằng 0 và do vậy không ảnh hưởng tới kết quả huấn luyện.

Mỗi ví dụ huấn luyện thứ  $j$  tương ứng với  $n$  trọng số  $w_j^n$ ,  $n = 1, \dots, N$ . Mỗi trọng số được sử dụng khi ví dụ đó được dùng với bộ phân loại thứ  $n$ ;  $w_j^n = 0$  nếu  $r_{nj} = 0$  tức là ví dụ  $j$  không tham gia vào huấn luyện bộ phân loại  $n$ . Khi đó sai số phân loại được tính bằng tổng sai số cho tất cả  $N$  bộ phân loại:

$$J = \sum_{n=1}^N \sum_{i=1}^M w_i^n (y_i^n - f_k^n(x_i))^2 \quad (2.14)$$

Mục tiêu của các phương pháp Boosting là lựa chọn các bộ phân loại yếu trong mỗi vòng lặp sao cho (2.14) đạt giá trị nhỏ nhất.

### 2.3.2.2. Xây dựng bộ phân loại yếu

Tại mỗi vòng lặp  $k$ , gọi  $S(t)$  là tập con các bài toán. Thay vì xác định đặc trưng  $f$  tốt nhất cho từng bài toán riêng lẻ như ở phần trước, thuật toán cần xác định đặc trưng chung cho tất cả bài toán thuộc  $S(t)$  và chọn gốc quyết định tương ứng sao cho sai số (2.14) là nhỏ nhất. Gốc cây quyết định sẽ có dạng như sau:

$$f_k^n(x, t) = \begin{cases} a_s & \text{if } \delta(x^f > 0) \wedge n \in S(t) \\ b_s & \text{if } \delta(x^f \leq 0) \wedge n \in S(t) \\ c^n & \text{if } n \notin S(t) \end{cases} \quad (2.15)$$

Trong đó,  $a_s$ ,  $b_s$ ,  $c_n$  được xác định sao cho hàm lỗi phân loại (2.14) đạt giá trị nhỏ nhất. Vì giá trị gốc cây quyết định phụ thuộc vào việc tập con  $S(t)$  nào được chọn, do đó  $f_k$  là một hàm của  $t$ . Ký hiệu  $f_k^n(x, t)$  được hiểu là hàm phân loại yếu tại bước thứ  $k$  cho bài toán thứ  $n$  và hàm này chung cho tập con  $S(t)$  các bài toán phân loại. Do giá trị hàm lỗi (2.14) cũng phụ thuộc vào tập con  $S(t)$  nên hàm lỗi (2.14) cũng cần viết lại thành hàm của tham số  $t$  như sau:

$$J(t) = \sum_{n=1}^N \sum_{i=1}^M w_i^n (y_i^n - f_k^n(x_i, t))^2 \quad (2.16)$$

Điểm khác nhau cơ bản so với gốc quyết định ở phần trước là gốc quyết định (2.15) phân biệt trường hợp bài toán  $n$  thuộc tập con  $S(t)$  và trường hợp không thuộc tập con  $S(t)$ . Trong trường hợp  $n$  không thuộc  $S(t)$ , hàm  $f_k^n(x, t)$  sẽ được đặt bằng hằng số  $c^n$  để tránh trường hợp lựa chọn bộ phân loại một cách tình cờ do chênh lệch số lượng giữa ví dụ huấn luyện 1 và  $-1$  (Chẳng hạn trong trường hợp quá nhiều ví dụ 1 thì có thể luôn dự đoán nhãn là 1 không cần quan tâm tới đặc trưng).

Với mỗi tập con  $S(t)$ , giải bài toán cực tiểu hoá sai số (2.15) ta nhận được:

$$a_s(f) = \frac{\sum_{n \in S(t)} \sum_{i=1}^M w_i^n y_i^n \delta(x_i^f > 0)}{\sum_{n \in S(t)} \sum_{i=1}^M w_i^n \delta(x_i^f > 0)}, \quad (2.17)$$

$$b_s(f) = \frac{\sum_{n \in S(t)} \sum_{i=1}^M w_i^n y_i^n \delta(x_i^f \leq 0)}{\sum_{n \in S(t)} \sum_{i=1}^M w_i^n \delta(x_i^f \leq 0)}, \quad (2.18)$$

$$c^n = \frac{\sum_{i=1}^M w_i^n y_i^n}{\sum_{i=1}^M w_i^n}, \quad n \notin S(t) \quad (2.19)$$

Tại mỗi bước lặp, thuật toán sẽ lựa chọn tập con  $S(t)$  tốt nhất, tức là tập con cho giá trị hàm lỗi (2.16) nhỏ nhất và gốc quyết định tốt nhất cho tập con đó. Ký hiệu  $F^n(x)$  là bộ phân mạnh cho bài toán phân loại thứ  $n$ , thuật toán được thể hiện trong Hình 2.4.

Một chi tiết cần làm rõ đối với thuật toán MC-Boost (Hình 2.4) là cách xác định tập con  $S(t)$ . Nếu liệt kê tất cả tập con  $S(t)$  từ  $N$  bài toán thì số lượng tập con là  $O(2^N)$ . Do vậy, thay vì liệt kê ta có thể sử dụng phương pháp tìm kiếm tham lam để giảm số các tập con cần duyệt từ  $O(2^N)$  thành  $O(KN^2)$  theo Mệnh đề 2.3 dưới đây.

**Đầu vào:**

- Tập ví dụ huấn luyện của  $N$  bài toán phân loại, bài toán thứ  $n$ ,  $n = 1, \dots, N$  được cho bởi  $M$  ví dụ huấn luyện  $(x^n_1, y^n_1), \dots, (x^n_M, y^n_M)$ .
- $K$  là số vòng lặp ( $K \leq 200$ )

**Đầu ra:**

- Trả về bộ phân loại  $\text{sign}[F^n(x)]$

**Các bước thực hiện:**

1. Khởi tạo  $w^n_j = 1$  nếu  $r_{nj} \neq \emptyset$  và  $w^n_j = 0$  nếu  $r_{nj} = \emptyset$ ,  $j = 1, \dots, M$ ;  $n = 1, \dots, N$ .

Khởi tạo  $F^n(x) = 0$

2. Lặp với  $k = 1, \dots, K$

a. Lặp với tập con các bài toán  $S(t)$

i. Tính tham số  $a_s, b_s$ , và  $c^n$  theo (2.17), (2.18), (2.19).

ii. Tính sai số  $J(t) = \sum_{n=1}^N \sum_{i=1}^M w_i^n (y_i^n - f_k^n(x_i, t))^2$

b. Chọn tập  $S(t)$  tốt nhất  $t^* = \arg \min_t J(t)$

c. Cập nhật  $F^n(x) \leftarrow F^n(x) + f_k^n(x_i, t^*)$

d. Cập nhật trọng số  $w_i^n \leftarrow w_i^n e^{-y_i^n f_k^n(x_i, t^*)}$ ,  $i=1, 2, \dots, M$

3. Trả về bộ phân loại  $\text{sign}[F^n(x)]$

**Hình 2.4.** Thuật toán MC-Boost cải tiến sử dụng đặc trưng chung cho nhiều bài toán.

**Mệnh đề 2.2.** Thuật toán MC-Boost cực tiểu hóa hàm lỗi phân loại thông qua các bước của phép khai triển Niuton.

**Chứng minh.** Các phân tích lý thuyết đối với thuật toán GentleBoost ở Mục 2.2.2 vẫn đúng đối với thuật toán MC-Boost. Thực vậy, thay hàm lỗi (2.14) theo tiêu chuẩn hàm ta nhận được (2.20).

$$J = E[e^{-y^n F^n(x)}] \quad (2.20)$$

Xấp xỉ  $J(F^n(x) + f_k^n(x))$  dưới dạng khai triển Taylor bậc 2 ta có

$$f_k^n(x) = \arg \min E[e^{y_i^n F^n(x)} (y_i^n - f_k^n(x))^2] \quad (2.21)$$

Thay kỳ vọng bằng giá trị trung bình trên  $M$  ví dụ huấn luyện và đặt  $w_i^n = e^{-y_i^n F^n(x_i)}$ , khi đó (2.21) được viết lại thành (2.22).

$$f_k^n(x) = \arg \min_{f_k^n} [\sum_{i=1}^M w_i^n (y_i^n - f_k^n(x_i))^2] \quad (2.22)$$

Triển khai các bước cập nhật của phương pháp Niuton tương tự trong Mục 2.2.2, thay trọng số  $w_i^n = e^{-y_i^n F^n(x_i)}$  và xác định  $f_k^n(x)$  từ (2.21) bằng cách cho đạo hàm bằng 0 ta nhận được (2.23) là điều cần thực hiện.

$$f_k^n(x_i) = \frac{E[e^{y_i^n F^n(x)} y_i]}{E[e^{y_i^n F^n(x)}]} \quad (2.23)$$

Điểm khác nhau duy nhất ở đây là tại mỗi bước lặp, thuật toán có thể không tìm được hàm  $f_k^n()$  cho sai số nhỏ nhất do phải xác định tập con  $S(t)$  một cách tham lam. Do vậy, tốc độ hội tụ sẽ chậm hơn so trường hợp tìm được  $f_k^n()$  tối ưu. Tuy nhiên, thuật toán vẫn cho phép giảm dần lỗi phân loại tại mỗi bước lặp và cho kết quả tốt trong các thử nghiệm.

### 2.2.2.3. Độ phức tạp thuật toán

**Mệnh đề 2.3.** Số lượng các tập con  $S(t)$  cần duyệt của thuật toán MC-Boost là  $O(KN^2)$ . Trong đó,  $K$  là số vòng lặp,  $N$  là số lượng người dùng.

**Chứng minh.** Phương pháp tìm kiếm tham lam trong thuật toán MC-Boost được tiến hành như sau:

- Trước tiên, xác định tập con  $t$  chỉ bao gồm một bài toán có sai số (2.16) nhỏ nhất. Số lượng các tập con cần so sánh trong  $N$  tập con để tìm ra tập con có sai số (2.16) nhỏ nhất là  $N$ .
- Tiếp theo trong số  $(N-1)$  bài toán còn lại, thêm một bài toán khác vào tập con  $t$  trước đó sao cho  $t$  mới có sai số (2.16) nhỏ nhất. Số lượng các tập con cần so sánh  $(N-1)$  bài toán để tìm ra bài toán có sai số (2.16) nhỏ nhất là  $(N-1)$ .

- Tổng quát, giả sử ta đã thêm được  $i$  bài toán có sai số (2.16) nhỏ nhất trong số  $(N-i)$  bài toán phân loại còn lại. Số lượng các tập con cần so sánh  $(N-i)$  bài toán để tìm ra bài toán có sai số (2.16) nhỏ nhất là  $(N-i)$ .
- Quá trình được tiếp tục cho đến khi chỉ còn một bài toán phân loại với số lượng các tập con cần so sánh là 1.

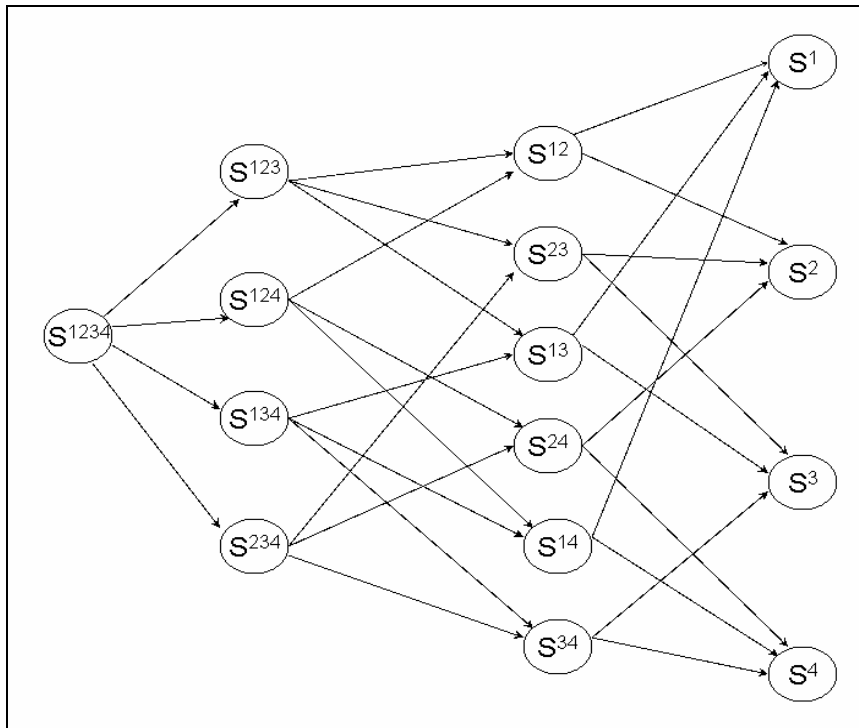
Như vậy tổng toàn bộ các tập con cần duyệt của thuật toán là:

$$(N + (N-1) + \dots + (N-i) + \dots + 1) = (N * (N-1)) / 2 = O(N^2)$$

Thuật toán bao gồm  $K$  vòng lặp, mỗi vòng lặp duyệt  $O(N^2)$  tập con  $S(t)$ .

Do vậy, số lượng các tập con  $S(t)$  cần duyệt của thuật toán là  $O(KN^2)$ .

Hình 2.5 dưới đây, mô tả phương pháp duyệt tập con các bài toán phân loại theo MC-Boost.



**Hình 2.5.** Phương pháp duyệt tập con các bài toán phân loại



## 2.4. THỬ NGHIỆM VÀ KẾT QUẢ

Như đã trình bày trong Chương 1, hiệu quả các phương pháp lọc cộng tác được xác định dựa trên khả năng thuật toán dự đoán chính xác đánh giá của khách hàng. Phương pháp Boosting đồng thời được trình bày ở trên được đánh giá và so sánh với các phương pháp khác theo thủ tục mô tả dưới đây.

### 2.4.1. Phương pháp thử nghiệm

Trước tiên, toàn bộ khách hàng được chia thành hai phần, một phần  $U_{tr}$  được sử dụng làm dữ liệu huấn luyện, phần còn lại  $U_{te}$  được sử dụng để kiểm tra. Dữ liệu huấn luyện được sử dụng để xây dựng mô hình theo thuật toán mô tả ở trên. Với mỗi khách hàng thuộc tập dữ liệu kiểm tra  $u$ , các đánh giá (đã có) của khách hàng được chia làm hai phần  $O_u$  và  $P_u$ .  $O_u$  được coi là đã biết, trong khi đó  $P_u$  là đánh giá cần dự đoán từ dữ liệu huấn luyện và  $O_u$ .

Sai số dự đoán  $MAE_u$  với mỗi khách hàng  $u$  thuộc tập dữ liệu kiểm tra được tính bằng trung cộng sai số tuyệt đối giữa giá trị dự đoán và giá trị thực đối với tất cả sản phẩm thuộc tập  $P_u$ .

$$MAE_u = \frac{1}{|P_u|} \sum_{y \in P_u} |\hat{r}_y^u - r_y^u| \quad (2.24)$$

Sai số dự đoán trên toàn tập dữ liệu kiểm tra được tính bằng trung bình cộng sai số dự đoán cho mỗi khách hàng thuộc  $U_{te}$ .

$$MAE = \frac{\sum_{u \in U_{te}} MAE_u}{|U_{te}|} \quad (2.25)$$

Giá trị  $MAE$  càng nhỏ, phương pháp cho lại kết quả càng chính xác.

### 2.4.2. Dữ liệu thử nghiệm

Thuật toán lọc cộng tác được thử nghiệm trên hai bộ dữ liệu EachMovie ([www.research.compaq.com/SRC/eachmovie/](http://www.research.compaq.com/SRC/eachmovie/)) và MovieLens ([www.grouplens.org/node/12](http://www.grouplens.org/node/12)). Đây là hai bộ dữ liệu được cộng đồng nghiên

cứ sử dụng phổ biến trong những nghiên cứu về lọc cộng tác [11, 15, 16, 17, 18, 21, 27, 29, 38, 39, 52, 56, 57, 95, 96, 106, 107].

Mặc dù có lọc cộng tác là vấn đề được quan tâm nghiên cứu và đã có ứng dụng thương mại, tuy nhiên số bộ dữ liệu chuẩn dùng để đánh giá thuật toán lọc cộng tác không nhiều như đối với trường hợp học máy nói chung. Lý do chủ yếu là do việc thu thập sở thích khách hàng đòi hỏi nhiều thời gian, phương pháp lấy ý kiến hợp lý, đồng thời không được vi phạm hạn chế về thông tin cá nhân. Do vậy, hầu hết nghiên cứu về lọc cộng tác chỉ có thể tiếp cận và sử dụng hai bộ dữ liệu EachMovie và MovieLens để đánh giá thuật toán lọc. Trong phạm vi luận án, đây cũng là hai bộ dữ liệu được sử dụng cho các thử nghiệm.

EachMovie được xây dựng bởi trung tâm nghiên cứu hệ thống thông tin của hãng Compaq. Bộ dữ liệu này gồm 72916 người dùng, 1628 bộ phim với 2811983 đánh giá, các mức đánh giá được cho từ 1 đến 6 (chính xác là 0.0, 0.2, 0.4, 0.6, 0.8, 1.0) , trung bình số lượng phim người dùng chưa đánh giá là 97.6%. Hai mức đánh giá cao nhất (0.8 và 1.0) được biến đổi thành “*thích*” (+1), bốn mức đánh giá còn lại được biến đổi thành “*không thích*” (-1). Phương pháp biến đổi này dựa trên những phân tích thực nghiệm của Billsus và Pazzani [29]. Vào tháng 10 năm 2004, hãng Compaq sát nhập với HP và đã ngừng cung cấp bộ dữ liệu *EachMovie* phục vụ công việc nghiên cứu. Bộ dữ liệu thực hiện ở đây được chúng tôi thu thập trước tháng 10 năm 2004.

MovieLens là cơ sở dữ liệu được xây dựng bởi nhóm nghiên cứu GroupLens của trường đại học Minnesota. MovieLens có 6040 người dùng, 3900 bộ phim, 1000209 đánh giá, các mức đánh giá cho từ 1 đến 5, trung bình số lượng phim người dùng chưa đánh giá là 95.7%. Dựa trên những phân tích thực nghiệm của Billsus và Pazzani [29], hai mức đánh giá cao nhất (4, 5) được biến đổi thành “*thích*”, các mức còn lại thành “*không thích*”.

### 2.4.3. So sánh và đánh giá dựa vào giá trị MAE

Phương pháp Boosting với đặc trưng chung (ký hiệu là MC-Boost) trình bày trong Mục 2.3.2 được so sánh với những phương pháp sau:

- Phương pháp K hàng xóm gần nhất sử dụng độ tương quan Pearson (ký hiệu là *KPC*) [52]. Đây là phương pháp lọc cộng tác thông dụng nhất thường được sử dụng trong khi so sánh.
- Phương pháp Boosting không sử dụng đặc trưng chung (GentleBoost) như trình bày trong Mục 2.2.2. Việc so sánh với Boosting thông thường cho phép làm rõ ảnh hưởng của Boosting đa nhiệm đối với hiệu quả lọc cộng tác.

Cách thức tiến hành thử nghiệm và so sánh như sau: Lần lượt lấy 100, 200, và 300 người dùng được lựa chọn ngẫu nhiên từ bộ dữ liệu MovieLens và được dùng làm dữ liệu huấn luyện, 200 người dùng lựa chọn ngẫu nhiên trong số còn lại để làm tập kiểm tra. Đối với bộ dữ liệu EachMovies, lần lượt lấy 1000, 2000, và 6000 người dùng được chọn ngẫu nhiên làm dữ liệu huấn luyện, 4000 người dùng còn lại được dùng để kiểm tra. Việc tạo tập huấn luyện và tập người dùng như vậy được thực hiện 10 lần, độ chính xác được lấy trung bình trên 10 lần thử nghiệm.

### 2.4.4. Kết quả thử nghiệm

Để thử nghiệm khả năng của phương pháp mới đề xuất so với những phương pháp khác trong việc hạn chế ảnh hưởng của vấn đề dữ liệu thưa, ta thay đổi số lượng đánh giá của mỗi người dùng trong tập kiểm tra sao cho số lượng đánh giá đã biết lần lượt là 5, 10 và 20, phần còn lại là những đánh giá cần dự đoán. Giá trị *MAE* cho từng bộ được thể hiện trong Bảng 2.5 và Bảng 2.6. Kết quả này được lấy trung bình của 10 lần thử nghiệm ngẫu nhiên dữ liệu của tập kiểm tra.

**Bảng 2.5.** Kết quả thử nghiệm với MovieLens

Kích thước tập huấn luyện	Phương pháp	Số đánh giá cho trước của tập kiểm tra		
		5	10	20
100 người dùng	KPC	0.378	0.337	0.328
	GentleBoost	0.350	0.322	0.291
	MC-Boost	<b>0.329</b>	<b>0.305</b>	0.292
200 người dùng	KPC	0.361	0.330	0.318
	GentleBoost	0.333	0.314	0.284
	MC-Boost	<b>0.314</b>	<b>0.299</b>	0.289
300 người dùng	KPC	0.348	0.336	0.317
	GentleBoost	0.325	0.304	0.279
	MC-Boost	<b>0.308</b>	<b>0.298</b>	0.283

**Bảng 2.6.** Kết quả thử nghiệm với EachMovie

Kích thước tập huấn luyện	Phương pháp	Số đánh giá cho trước của tập kiểm tra		
		5	10	20
1000 người dùng	KPC	0.559	0.474	0.449
	GentleBoost	0.515	<b>0.455</b>	0.421
	MC-Boost	<b>0.492</b>	0.460	0.429
2000 người dùng	KPC	0.528	0.450	0.422
	GentleBoost	0.495	0.424	0.393
	MC-Boost	<b>0.484</b>	<b>0.419</b>	0.393
6000 người dùng	KPC	0.521	0.437	0.378
	GentleBoost	0.477	0.408	0.362
	MC-Boost	<b>0.452</b>	<b>0.397</b>	0.365

Kết quả sai số phân loại  $MAE$  trên cả hai bộ dữ liệu cho thấy, dù số lượng dữ liệu biết trước nhiều hay ít (số đánh giá biết trước trong tập dữ liệu kiểm tra  $N=5, 10, 20$ ) phương pháp GentleBoost và MC-Boost đều cho lại giá trị  $MAE$

nhỏ hơn phương pháp *KPC*. Điều đó chứng tỏ, việc sử dụng gốc quyết định trong trích chọn đặc trưng của thuật toán GentleBoost lựa chọn được phương án phân loại tốt hơn so với *KPC*.

Trong trường hợp đủ dữ liệu, cụ thể là khi biết trước nhiều đánh giá của người dùng trong tập kiểm tra ( $N=20$ ), phương pháp GentleBoost cho kết quả tốt hơn so với MC-Boost. Có thể giải thích kết quả này là do GentleBoost chọn được đặc trưng tối ưu hơn đối với từng bài toán phân loại, trong khi MC-Boost chỉ chọn được đặc trưng tối ưu cho cả nhóm bài toán phân loại.

Tuy nhiên, khi dữ liệu ít đi, cụ thể là khi chỉ biết trước 5 hoặc 10 đánh giá của người dùng kiểm tra thì MC-Boost cho sai số *MAE* nhỏ hơn so với GentleBoost trong đa số trường hợp. Lý do chủ yếu là do MC Boost cho phép kết hợp thông tin từ những người dùng tương tự với người dùng kiểm tra thông qua các đặc trưng chung và do vậy giảm được ảnh hưởng của việc thiếu nhãn phân loại.

#### **2.4.5. Phân tích kết quả**

Để thấy rõ sự nổi trội của mô hình, chúng tôi lấy giá trị trung bình *MAE* của 10 lần kiểm nghiệm ngẫu nhiên trong tập dữ liệu kiểm tra để tiến hành một paired t-test [58]. Các tham số thống kê so sánh giữa *KPC* với GentleBoost và MC-Boost bao gồm:

- Tham số *DF* (*Degree of Freedom*) là số bậc tự do của paired t-test.
- Tham số *Mean* là trung bình độ lệch giữa *KPC* và phương pháp so sánh.
- Tham số *SD* (*Standard Deviation*) là độ lệch chuẩn giữa *KPC* và phương pháp so sánh.
- Tham số *SE* (*Standard Error*) là lỗi chuẩn được tính theo độ lệch chuẩn của *KPC* và phương pháp so sánh.
- Tham số *t* và *p* là giá trị kiểm nghiệm của pair t-test được tính theo bậc tự do, trung bình độ lệch, và lỗi chuẩn giữa các phương pháp. Giá trị

$p < 0.05$  phản ánh pháp KPC có giá trị MAE lớn hơn GentleBoost và MC-Boost ít nhất 5%.

Kết quả kiểm nghiệm các tham số thống kê giữa KPC với GentleBoost và MC-Boost ứng với từng bộ dữ liệu huấn luyện của tập MovieLens được thể hiện trong các Bảng 2.7, Bảng 2.8 và Bảng 2.9. Kết quả kiểm nghiệm các tham số thống kê giữa KPC với GentleBoost và MC-Boost ứng với từng bộ dữ liệu huấn luyện của tập EachMovie được thể hiện trong các Bảng 2.10, Bảng 2.11 và Bảng 2.12. Giá trị  $p$  ( $p$ -value) trong tất cả các bộ dữ liệu huấn luyện đều nhỏ hơn 0.05. Điều đó chứng tỏ, trên 5% giá trị MAE của phương pháp KPC lớn hơn GentleBoost và MC-Boost. Nói cách khác, GentleBoost và MC-Boost cho lại kết quả phân loại tốt hơn KPC.

**Bảng 2.7.** Các tham số thống kê với  $K=5$  đánh giá biết trước của tập dữ liệu MovieLens

Kích thước tập dữ liệu	Phương pháp so sánh	DF	Mean	SD	SE	t-value	p-value
100 người dùng	GentleBoost	9	0.028	0.035	0.011	2.549	0.031
	MC-Boost	9	0.049	0.052	0.016	2.982	0.015
200 người dùng	GentleBoost	9	0.028	0.034	0.011	2.628	0.027
	MC-Boost	9	0.047	0.046	0.015	3.214	0.011
300 người dùng	GentleBoost	9	0.023	0.026	0.008	2.822	0.020
	MC-Boost	9	0.040	0.037	0.012	3.425	0.008

**Bảng 2.8.** Các tham số thống kê với  $K=10$  đánh giá biết trước của tập dữ liệu MovieLens

Kích thước tập dữ liệu	Phương pháp so sánh	DF	Mean	SD	SE	t-value	P-value
100 người dùng	GentleBoost	9	0.015	0.017	0.005	2.766	0.022
	MC-Boost	9	0.032	0.028	0.009	3.623	0.006
200 người dùng	GentleBoost	9	0.016	0.019	0.006	2.713	0.024
	MC-Boost	9	0.031	0.029	0.009	3.378	0.008
300 người dùng	GentleBoost	9	0.032	0.034	0.011	2.973	0.012
	MC-Boost	9	0.038	0.032	0.010	3.725	0.005

**Bảng 2.9.** Các tham số thống kê với K=20 đánh giá biết trước của tập dữ liệu MovieLens

Kích thước tập dữ liệu	Phương pháp so sánh	DF	Mean	SD	SE	t-value	p-value
100 người dùng	GentleBoost	9	0.037	0.040	0.013	2.916	0.017
	MC-Boost	9	0.036	0.048	0.015	2.341	0.044
200 người dùng	GentleBoost	9	0.034	0.039	0.012	2.741	0.023
	MC-Boost	9	0.029	0.037	0.012	2.469	0.036
300 người dùng	GentleBoost	9	0.038	0.033	0.011	3.629	0.027
	MC-Boost	9	0.034	0.045	0.014	2.377	0.041

**Bảng 2.10.** Các tham số thống kê với K=5 đánh giá biết trước của tập dữ liệu EachMovie

Kích thước tập dữ liệu	Phương pháp so sánh	DF	Mean	SD	SE	t-value	p-value
1000 người dùng	GentleBoost	9	0.044	0.600	0.019	2.335	0.044
	MC-Boost	9	0.067	0.066	0.021	3.215	0.011
2000 người dùng	GentleBoost	9	0.033	0.035	0.011	2.943	0.016
	MC-Boost	9	0.044	0.046	0.016	3.018	0.015
6000 người dùng	GentleBoost	9	0.044	0.054	0.017	2.589	0.029
	MC-Boost	9	0.069	0.070	0.022	3.112	0.013

**Bảng 2.11.** Các tham số thống kê với K=10 đánh giá biết trước của tập dữ liệu EachMovie

Kích thước tập dữ liệu	Phương pháp so sánh	DF	Mean	SD	SE	t-value	p-value
1000 người dùng	GentleBoost	9	0.019	0.021	0.007	2.812	0.020
	MC-Boost	9	0.014	0.018	0.006	2.451	0.037
2000 người dùng	GentleBoost	9	0.026	0.027	0.008	3.083	0.013
	MC-Boost	9	0.031	0.31	0.010	3.203	0.011
6000 người dùng	GentleBoost	9	0.029	0.035	0.011	2.641	0.027
	MC-Boost	9	0.040	0.043	0.014	2.915	0.017

**Bảng 2.12.** Các tham số thống kê với  $K=20$  đánh giá biết trước của tập dữ liệu EachMovie

Kích thước tập dữ liệu	Phương pháp so sánh	DF	Mean	SD	SE	t-value	P-value
1000 người dùng	GentleBoost	9	0.028	0.028	0.009	3.218	0.011
	MC-Boost	9	0.020	0.025	0.008	2.541	0.032
2000 người dùng	GentleBoost	9	0.029	0.033	0.011	2.758	0.022
	MC-Boost	9	0.029	0.035	0.011	2.615	0.028
6000 người dùng	GentleBoost	9	0.016	0.020	0.006	2.497	0.034
	MC-Boost	9	0.013	0.018	0.006	2.312	0.046

## 2.5. KẾT LUẬN

Chương này đã trình bày một phương pháp học đa nhiệm cho lọc cộng tác. Phương pháp được phát triển dựa trên nền tảng của kỹ thuật phân loại Boosting kết hợp với trích chọn đặc trưng dựa vào gốc cây quyết định. Đây là một cải tiến của thuật toán Boosting, trong đó việc lựa chọn đặc trưng cho mỗi bộ phân loại yếu được thực hiện đồng thời trên một nhóm người dùng tương tự nhau.

Ưu điểm chủ yếu của phương pháp này là việc phân loại đồng thời từng nhóm người dùng cho phép sử dụng thông tin từ những người dùng tương tự nhau và nhờ vậy cải thiện độ chính xác phân loại khi dữ liệu thưa thớt, ví dụ khi người dùng cần dự đoán chỉ đánh giá rất ít sản phẩm trước đó.

Hiệu quả của phương pháp đề xuất được kiểm nghiệm trên hai bộ dữ liệu MovieLens và EachMovie với những độ thưa thớt dữ liệu khác nhau. Kết quả cho thấy, phương pháp đề xuất cải thiện đáng kể độ chính xác so với hai phương pháp khác (với giá trị  $p$  nhỏ ở 0.05), và có ưu thế rõ ràng khi độ thưa thớt của dữ liệu tăng lên.



### CHƯƠNG 3

## LỘC KẾT HỢP DỰA TRÊN MÔ HÌNH ĐỒ THỊ

Như đã đề cập trong Chương 1, lọc cộng tác và lọc nội dung là hai phương pháp cơ bản được sử dụng trong các hệ tư vấn lựa chọn. Mỗi phương pháp có những ưu điểm riêng, khai thác những khía cạnh riêng theo nội dung sản phẩm hoặc thói quen người dùng. Kết hợp hai phương pháp trong cùng một mô hình cho phép ta tận dụng được lợi thế mỗi phương pháp trong việc nâng cao kết quả dự đoán.

Mô hình kết hợp giữa lọc cộng tác và lọc nội dung được trình bày trong chương này thực hiện dựa trên biểu diễn đồ thị quan hệ giữa người dùng, sản phẩm và nội dung sản phẩm. Sử dụng biểu diễn đồ thị cho phép tận dụng được các mối quan hệ gián tiếp giữa những đối tượng nói trên vào kết quả dự đoán.

Để thuận tiện cho việc trình bày, Mục 3.1 phát biểu lại bài toán lọc kết hợp cùng với các mở rộng khi thêm thông tin nội dung. Mục 3.2 trình bày một phương pháp lọc cộng tác bằng đồ thị làm cơ sở để kết hợp với lọc nội dung. Mục 3.3 trình bày phương pháp kết hợp giữa lọc cộng tác và lọc nội dung. Mục 3.4 trình bày về kết quả thử nghiệm, so sánh và đánh giá với các phương pháp lọc khác. Mục cuối cùng là kết luận và hướng nghiên cứu tiếp theo.

### 3.1. VẤN ĐỀ LỘC KẾT HỢP

Giả sử hệ có  $N$  người dùng  $U = \{u_1, u_2, \dots, u_N\}$ ,  $M$  sản phẩm  $P = \{p_1, p_2, \dots, p_M\}$  với ma trận đánh giá  $\{r_{ij}\}$  như phát biểu trong phần 2.1. Gọi  $C = \{c_1, c_2, \dots, c_K\}$  là  $K$  đặc trưng thể hiện nội dung các sản phẩm  $P$ . Ví dụ nếu  $c_i$  là một phim, khi đó ta có thể xem xét đặc trưng  $c_i$  là “*thể loại*”, “*đạo diễn*”, “*diễn viên*”, “*hãng sản xuất*”.... Ký hiệu ma trận  $Y = (y_{ij})$  biểu thị mối quan hệ giữa sản phẩm và đặc trưng nội dung sản phẩm được xác định theo công thức (3.1).

$$y_{ij} = \begin{cases} 1 & \text{nếu sản phẩm } p_i \text{ có đặc trưng nội dung } c_j \\ 0 & \text{nếu sản phẩm } p_i \text{ không có đặc trưng nội dung } c_j \end{cases} \quad (3.1)$$

Bài toán của lọc kết hợp là dự đoán cho người dùng hiện thời  $u_a$  những sản phẩm  $p_j \in P$  chưa được  $u_a$  đánh giá dựa trên ma trận đánh giá  $r_{ij}$  và các đặc trưng nội dung  $C = \{c_1, c_2, \dots, c_K\}$ .

Để minh họa cho mô hình trình bày trong các mục tiếp theo, ta xem xét ví dụ sau. Giả sử hệ gồm 5 người dùng  $U = \{u_1, u_2, u_3, u_4, u_5\}$ , 7 phim  $P = \{p_1, p_2, p_3, p_4, p_5, p_6, p_7\}$ . Ma trận đánh giá  $r_{ij}$  được cho trong Bảng 3.1. Giả sử  $p_1, p_2, p_4, p_5, p_6$  có đặc trưng nội dung phim  $c_1$  (“hành động”);  $p_3, p_4, p_5, p_7$  có đặc trưng nội dung phim  $c_2$  (“tình cảm”). Khi đó, ma trận nội dung  $Y = (y_{ij})$  được thể hiện trong Bảng 3.2.

**Bảng 3.1.** Ma trận đánh giá R

Người dùng	Sản phẩm						
	p <sub>1</sub>	p <sub>2</sub>	p <sub>3</sub>	p <sub>4</sub>	p <sub>5</sub>	p <sub>6</sub>	p <sub>7</sub>
u <sub>1</sub>	1	∅	-1	1	∅	-1	∅
u <sub>2</sub>	∅	1	-1	1	-1	∅	-1
u <sub>3</sub>	-1	1	1	-1	∅	∅	1
u <sub>4</sub>	-1	-1	∅	∅	1	-1	1
u <sub>5</sub>	?	1	?	-1	1	1	?

**Bảng 3.2.** Ma trận Sản phẩm – Nội dung Y

Sản phẩm	Nội dung	
	c <sub>1</sub>	c <sub>2</sub>
p <sub>1</sub>	1	0
p <sub>2</sub>	1	0
p <sub>3</sub>	0	1
p <sub>4</sub>	1	1
p <sub>5</sub>	1	1
p <sub>6</sub>	1	0
p <sub>7</sub>	0	1

Nhiệm vụ của lọc kết hợp là dự đoán và phân bổ các sản phẩm thích hợp, gỡ bỏ những sản phẩm không thích hợp cho người dùng hiện thời  $u_a$ . Ví dụ ta cần dự đoán các sản phẩm cho người dùng  $u_5$  các sản phẩm  $u_5$  chưa đánh giá đó là  $p_1, p_3, p_7$ .

## 3.2. LỌC CỘNG TÁC DỰA TRÊN MÔ HÌNH ĐỒ THỊ

Lọc cộng tác có thể xem xét như bài toán tìm kiếm trên đồ thị dựa trên biểu diễn mối quan hệ đánh giá của người dùng đối với các sản phẩm. Mục này trình bày một mô hình đồ thị cho lọc cộng tác. Đây là một thành phần quan trọng của mô hình đồ thị kết hợp sẽ được trình bày trong mục tiếp theo.

### 3.2.1. Phương pháp biểu diễn đồ thị

Mô hình đồ thị cho lọc cộng tác có thể mô tả như sau. Cho ma trận đánh giá đầu vào của lọc cộng tác  $R = (r_{ij})$ . Gọi  $X=(x_{ij})$  là ma trận cấp  $N \times M$  có các phần tử được xác định theo công thức (3.2). Trong đó,  $x_{ij} = 1$  tương ứng với trạng thái người dùng  $u_i$  đã đánh giá sản phẩm  $p_j$ ,  $x_{ij} = 0$  tương ứng với trạng thái người dùng chưa đánh giá sản phẩm  $p_j$ .

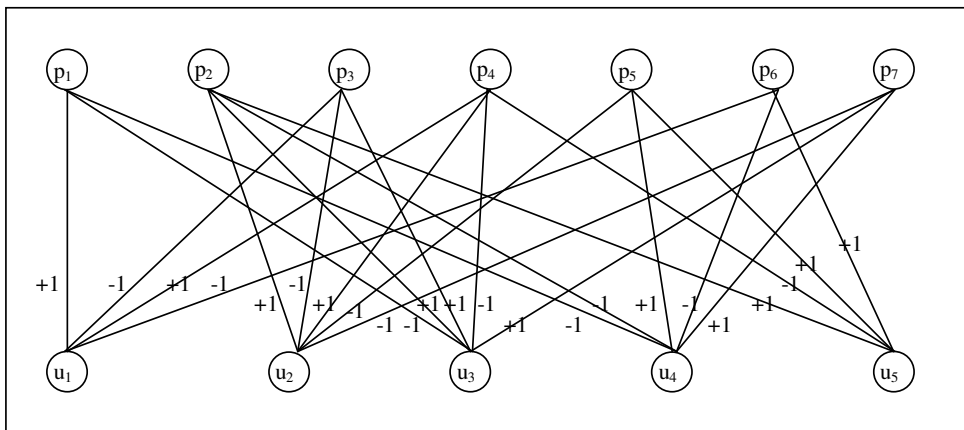
$$x_{ij} = \begin{cases} 1 & \text{if } r_{ij} \neq \phi \\ 0 & \text{otherwise} \end{cases} \quad (3.2)$$

Đồ thị biểu diễn đánh giá của người dùng đối với các sản phẩm (Gọi tắt là Người dùng - Sản phẩm)  $G = (V, E)$  được biểu diễn theo ma trận  $X$ , trong đó tập đỉnh  $V = U \cup P$  ( $U$  là tập người dùng,  $P$  là tập sản phẩm); tập cạnh  $E$  bao gồm tập các cạnh biểu diễn đánh giá của người dùng đối với sản phẩm. Cạnh nối giữa đỉnh  $u_i \in U$  và đỉnh  $p_j \in P$  được thiết lập nếu người dùng  $u_i$  đã đánh giá sản phẩm  $p_j$  ( $x_{ij} = 1$ ). Trọng số của mỗi cạnh được lấy tương ứng là  $r_{ij}$ . Như vậy, trong biểu diễn này, đồ thị Người dùng- Sản phẩm có hai loại cạnh: Cạnh có trọng số dương  $r_{ij}=+1$  biểu diễn người dùng  $u_i$  “*thích*” sản phẩm  $p_j$ , cạnh có trọng số âm  $r_{ij}=-1$  biểu diễn người dùng  $u_i$  “*không thích*” sản phẩm  $p_j$ .

Ví dụ với ma trận đánh giá  $R$  được cho trong Bảng 3.1 thì ma trận  $X$  được thể hiện như Bảng 3.3. Khi đó, đồ thị được biểu diễn như Hình 3.1.

**Bảng 3.3.** Ma trận  $X$  biểu diễn đánh đồ thị Người dùng- Sản phẩm

Người dùng	Sản phẩm						
	p <sub>1</sub>	p <sub>2</sub>	p <sub>3</sub>	p <sub>4</sub>	p <sub>5</sub>	p <sub>6</sub>	p <sub>7</sub>
u <sub>1</sub>	1	0	1	1	0	1	0
u <sub>2</sub>	0	1	1	1	1	0	1
u <sub>3</sub>	1	1	1	1	0	0	1
u <sub>4</sub>	1	1	0	0	1	1	1
u <sub>5</sub>	0	1	0	1	1	1	0



**Hình 3.1.** Đồ thị Người dùng- Sản phẩm

### 3.2.2. Phương pháp dự đoán trên đồ thị Người dùng- Sản phẩm

Các phương pháp lọc cộng tác dựa trên độ tương quan [52, 56] thực hiện bằng cách xác định những người dùng tương tự nhất với người dùng hiện thời để tạo nên tư vấn. Trong ví dụ trên, dễ dàng nhận thấy  $u_5$  tương tự nhất với  $u_2, u_3$  và  $u_4$  vì  $u_5, u_2, u_3$  cùng “thích”  $p_2$  và  $u_5, u_4$  cùng “thích”  $p_5$ . Dựa trên mức độ tương tự này, các sản phẩm  $p_3, p_4$  và  $p_7$  sẽ được tư vấn cho người dùng  $u_5$ .

Cách làm trên có thể được thực hiện dễ dàng trên mô hình đồ thị bằng cách xem xét các đường đi độ dài 3 từ đỉnh người dùng đến đỉnh sản phẩm,

những sản phẩm nào có nhiều số đường đi độ dài 3 từ đỉnh người dùng hiện thời đến đỉnh sản phẩm sẽ được dùng để tạo nên tư vấn. Ví dụ ta cần phân bổ sản phẩm cho người dùng  $u_5$ , các đường đi  $u_5-p_5-u_4-p_7$ ,  $u_5-p_2-u_2-p_4$ ,  $u_5-p_2-u_3-p_3$ ,  $u_5-p_2-u_3-p_7$  được xem xét đến trong khi dự đoán các sản phẩm cho  $u_5$ . Những sản phẩm có nhiều đường đi nhất đến  $u_5$  sẽ được dùng để tư vấn. Ví dụ  $p_7$  có nhiều đường đi độ dài 3 hơn so với  $p_3$  và  $p_4$  ( $u_5-p_5-u_4-p_7$ ,  $u_5-p_2-u_3-p_7$ ) sẽ được tư vấn cho  $u_5$ .

Hơn thế nữa, phương pháp lọc dựa trên độ tương quan sẽ không bao giờ được xem xét đến  $p_1$  trong các khả năng tư vấn vì  $u_5$  và  $u_1$  được xác định là không tương tự nhau. Điều này không đúng trong trường hợp dữ liệu thưa của lọc cộng tác,  $u_5$  và  $u_1$  không tương tự nhau vì chúng có quá ít dữ liệu đánh giá để thực hiện tính toán. Nhược điểm này có thể khắc phục trên mô hình đồ thị bằng cách mở rộng phương pháp dự đoán đến các đường đi độ dài lẻ lớn hơn 3 (5, 7, 9...). Những sản phẩm có nhiều đường đi nhất đến nó được dùng để tư vấn cho người dùng hiện thời. Với cách làm này,  $p_1$  cũng được xem xét đến vì có đường đi độ dài 5 ( $u_5-p_2-u_2-p_4-u_1-p_1$ ). Phương pháp dự đoán trên đồ thị Người dùng-Sản phẩm có thể được thực hiện thông qua các bước sau:

**Tách đồ thị Người dùng- Sản phẩm thành các đồ thị con.** Trong số các đường đi từ  $u_i$  đến  $p_j$ , ta xem xét đến hai loại đường đi: Đường đi theo các cạnh có trọng số dương (ví dụ đường đi  $u_5-p_2-u_3-p_3$ ) và đường đi theo các cạnh có trọng số âm (ví dụ đường đi  $u_5-p_4-u_3-p_1$ ). Để tính toán hiệu quả cho mỗi loại đường đi, ta tách đồ thị Người dùng- Sản phẩm thành hai đồ thị con: Đồ thị con chỉ bao gồm các cạnh có trọng số dương và đồ thị con chỉ bao gồm các cạnh có trọng số âm. Phương pháp tách cụ thể được trình bày trong Mục 3.2.2.1.

**Dự đoán trên đồ thị con chỉ bao gồm các cạnh có trọng số dương.** Trọng số đường đi từ đỉnh người dùng  $u_i$  đến đỉnh sản phẩm  $p_j$  theo các cạnh có trọng số dương được ghi nhận là một số dương phản ánh mức độ “thích” của sản phẩm đối với người dùng. Những đường đi có độ dài lớn sẽ được đánh trọng số thấp, những đường đi có độ dài nhỏ được đánh trọng số cao. Những sản phẩm

nào có trọng số cao sẽ được dùng để tư vấn cho người dùng hiện thời. Đây chính là mô hình đồ thị được Huang đề xuất trong [119]. Phương pháp dự đoán cụ thể được trình bày trong Mục 3.2.2.2.

**Dự đoán trên đồ thị con chỉ bao gồm các cạnh có trọng số âm.** Trọng số đường đi từ đỉnh người dùng  $u_i$  đến đỉnh sản phẩm theo các cạnh có trọng số âm được ghi nhận là một số âm phản ánh mức độ “*không thích*” của người dùng đối với sản phẩm. Những đường đi có độ dài lớn sẽ được đánh trọng số cao, những đường đi có độ dài nhỏ được đánh trọng số thấp. Những sản phẩm nào có trọng số thấp được loại bỏ ra khỏi danh sách các sản phẩm cần tư vấn cho người dùng hiện thời. Phương pháp dự đoán cụ thể được trình bày trong Mục 3.2.2.3.

**Dự đoán trên tất cả đánh giá.** Một sản phẩm người dùng “*thích*” vẫn có thể xuất hiện trong danh sách các sản phẩm loại bỏ khỏi quá trình tư vấn, một sản phẩm người dùng “*không thích*” vẫn có thể xuất hiện trong danh sách các sản phẩm cần tư vấn. Để ngăn ngừa tình trạng này, luận án đề xuất phương pháp dự đoán trên tất cả đánh giá được trình bày trong Mục 3.2.2.4.

### 3.2.2.1. Tách đồ thị Người dùng- Sản phẩm thành các đồ thị con

Cho đồ thị Người dùng - Sản phẩm  $G = (V, E)$  được biểu diễn theo ma trận  $X = (x_{ij})$  cấp  $N \times M$  như đã trình bày trong Mục 3.2.1. Ký hiệu  $X^+ = (x_{ij}^+)$  là ma trận cấp  $N \times M$  được xác định theo công thức (3.3). Ký hiệu  $X^- = (x_{ij}^-)$  là ma trận cấp  $N \times M$  được xác định theo công thức (3.4).

$$x_{ij}^+ = \begin{cases} 1 & \text{if } r_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.3)$$

$$x_{ij}^- = \begin{cases} 1 & \text{if } r_{ij} < 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.4)$$

Đồ thị  $G^+ = (V, E^+)$  được biểu diễn theo ma trận  $X^+$  có tập đỉnh đúng bằng tập đỉnh của  $G$ , có tập cạnh  $E^+$  bao gồm các cạnh có trọng số dương của  $G$ .

$$E^+ = \{e = (u_i, p_j) \in E \mid r_{ij} = 1\} \quad (3.5)$$

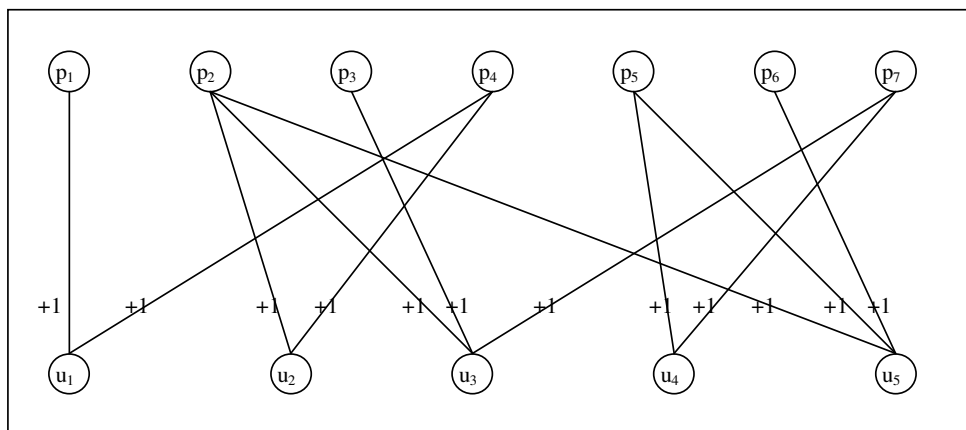
Đồ thị  $G^+ = (V, E^+)$  được biểu diễn theo ma trận  $X$  có tập đỉnh đúng bằng tập đỉnh của  $G$ , có tập cạnh  $E^+$  bao gồm các cạnh có trọng số âm của  $G$ .

$$E^- = \{e = (u_i, p_j) \in E \mid r_{ij} = -1\} \quad (3.6)$$

Ví dụ với ma trận đánh giá  $R$  được cho trong Bảng 3.1, đồ thị  $G$  được biểu diễn theo ma trận  $X$  trong Bảng 3.3 thì ma trận  $X^+$ ,  $X^-$  được thể hiện trong Bảng 3.4 và Bảng 3.5. Đồ thị  $G^+$ ,  $G^-$  tương ứng được biểu diễn theo Hình 3.2 và Hình 3.3.

**Bảng 3.4.** Ma trận  $X^+$  biểu diễn các đánh giá thích hợp

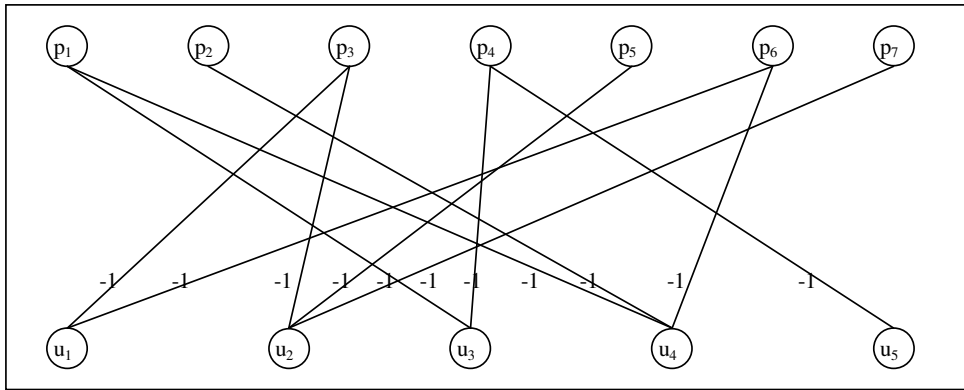
Người dùng	Sản phẩm						
	p <sub>1</sub>	p <sub>2</sub>	p <sub>3</sub>	p <sub>4</sub>	p <sub>5</sub>	p <sub>6</sub>	p <sub>7</sub>
u <sub>1</sub>	1	0	0	1	0	0	0
u <sub>2</sub>	0	1	0	1	0	0	0
u <sub>3</sub>	0	1	1	0	0	0	1
u <sub>4</sub>	0	0	0	0	1	0	1
u <sub>5</sub>	0	1	0	0	1	1	0



**Hình 3.2.** Đồ thị  $G^+$  biểu diễn các đánh giá thích hợp

**Bảng 3.5.** Ma trận  $X$  biểu diễn các đánh giá không thích hợp

Người dùng	Sản phẩm						
	p <sub>1</sub>	p <sub>2</sub>	p <sub>3</sub>	p <sub>4</sub>	p <sub>5</sub>	p <sub>6</sub>	p <sub>7</sub>
u <sub>1</sub>	0	0	1	0	0	1	0
u <sub>2</sub>	0	0	1	0	1	0	1
u <sub>3</sub>	1	0	0	1	0	0	0
u <sub>4</sub>	1	1	0	0	0	1	0
u <sub>5</sub>	0	0	0	1	0	0	0



**Hình 3.3.** Đồ thị  $G$  biểu diễn các đánh giá không thích hợp.

### 3.2.2.2. Phương pháp dự đoán trên đồ thị $G^+$

Phương pháp dự đoán trên đồ thị  $G^+$  được Huang đề xuất dựa trên việc tính toán trọng số các đường đi từ đỉnh người dùng đến đỉnh sản phẩm [119]. Những sản phẩm nào có trọng số cao nhất sẽ được dùng để tư vấn cho người dùng hiện thời.

Đề ý rằng, đồ thị  $G$ ,  $G^+$ ,  $G^-$  đều là những đồ thị hai phía, một phía là các đỉnh người dùng, phía còn lại là các đỉnh sản phẩm. Do vậy, các đường đi từ đỉnh người dùng đến đỉnh sản phẩm luôn có độ dài lẻ.

Đối với đồ thị hai phía, số các đường đi độ dài  $L$  xuất phát từ một đỉnh bất kỳ thuộc phía người dùng đến đỉnh bất kỳ thuộc phía sản phẩm được xác định



theo công thức (3.7), trong đó  $X$  là ma trận biểu diễn đồ thị hai phía,  $X^T$  là ma trận chuyển vị của  $X$ ,  $L$  là độ dài đường đi.

$$X = \begin{cases} X & \text{if } L=1 \\ X.X^T.X^{L-2} & \text{if } L=3,5,7,\dots \end{cases} \quad (3.7)$$

Để ghi nhận trọng số của các đường đi từ đỉnh sản phẩm đến đỉnh người dùng trên đồ thị  $G^+$  sao cho những đường đi dài có trọng số thấp, những đường đi ngắn có trọng số cao, ta sử dụng hằng khử nhiễu  $\alpha$  ( $0 < \alpha \leq 1$ ) theo công thức (3.8), trong đó  $X^+$  là ma trận biểu diễn đồ thị  $G^+$ ,  $(X^+)^T$  là ma trận chuyển vị của  $X^+$ ,  $L$  là độ dài đường đi. Thuật toán dự đoán trên đồ thị  $G^+$  được thể hiện trong Hình 3.4.

$$(X^+)_\alpha^L = \begin{cases} \alpha.X^+ & \text{if } L=1 \\ \alpha^2.X^+.(X^+)^T.(X^+)_\alpha^{L-2} & \text{if } L=3,5,7,\dots \end{cases} \quad (3.8)$$

**Đầu vào:**

- Ma trận  $X^+$  là biểu diễn của đồ thị  $G^+$ ;
- $\alpha$  là hằng số ( $0 < \alpha \leq 1$ ),  $L$  là độ dài đường đi;
- $K$  là số sản phẩm cần tư vấn.

**Đầu ra:**

- $K$  sản phẩm có trọng số cao nhất chưa được người dùng đánh giá

**Các bước thực hiện:**

**Bước 1.** Tìm trọng số các đường đi độ dài lẻ  $L$  trên đồ thị  $G^+$  sao cho các đường đi độ dài nhỏ được đánh trọng số cao, các đường đi có độ dài lớn được đánh trọng số thấp.

$$(X^+)_\alpha^L = \begin{cases} \alpha.X^+ & \text{if } L=1 \\ \alpha^2.X^+.(X^+)^T.(X^+)_\alpha^{L-2} & \text{if } L=3,5,7,\dots \end{cases}$$

**Bước 2.** Sắp xếp các sản phẩm theo thứ tự giảm dần của trọng số  $x_\alpha^{+L}$ .

**Bước 3.** Chọn  $K$  sản phẩm có trọng số  $x_\alpha^{+L}$  cao nhất chưa được đánh giá để tư vấn cho người dùng hiện thời.

**Hình 3.4.** Thuật toán dự đoán trên đồ thị  $G^+$

**Mệnh đề 3.1.** Độ phức tạp thuật toán dự đoán trên đồ thị  $G^+$  là  $O(L.N^{2.376})$ . Trong đó,  $L$  là độ dài đường đi từ đỉnh người dùng đến đỉnh sản phẩm,  $N$  là số lượng người dùng.

**Chứng minh.** Thực vậy, tại bước 1 của thuật toán tích của  $L$  ma trận  $(X^+)$  và  $(X^+)^T$  sẽ có độ phức tạp là  $O(L \cdot X^+ \cdot (X^+)^T)$ . Sử dụng thuật toán nhân hai ma trận hiệu quả nhất hiện nay của *Coppersmith–Winograd* sẽ cho ta độ phức tạp là  $O(N^{2.376})$  khi  $X^+$  là ma trận vuông ( $N=M$ ) [44]. Như vậy, độ phức tạp thuật toán thực hiện tại bước 1 là  $O(L.N^{2.376})$ .

Độ phức tạp thuật toán thực hiện tại bước 2 chính bằng độ phức tạp của thuật toán sắp xếp được sử dụng. Trong trường hợp sử dụng thuật toán Heap Sort, độ phức tạp thuật toán tại bước 2 là  $O(M.log_2M)$ . Khi trường hợp xấu nhất xảy ra tại bước 1, ta có  $M = N$ . Do đó, độ phức tạp tại bước 2 của thuật toán là  $O(N.log_2N)$ .

Độ phức tạp tại bước 3 của thuật toán là hằng số  $O(K)$  vì ta chỉ đơn thuần chọn  $K$  sản phẩm  $x_{\alpha}^{+L}$  có trọng số cao nhất chưa được người dùng đánh giá, đã được sắp xếp ở bước 2 để tư vấn cho người dùng hiện thời.

Như vậy, độ phức tạp của toàn bộ thuật toán là

$$O(L.N^{2.376} + N.log_2N + K) = O(L.N^{2.376}).$$

**Ví dụ minh họa:**

Ví dụ với ma trận  $X^+$  biểu diễn đồ thị  $G^+$  trong Bảng 3.4, lấy  $\alpha = 0.5, L=5$ . Giả sử ta cần tư vấn  $K=2$  sản phẩm cho người dùng  $u_5$ , khi đó thuật toán thực hiện như sau:

Tại bước 1 của thuật toán, số các đường đi độ dài 5 từ đỉnh người dùng đến đỉnh sản phẩm được xác định theo công thức (3.8). Khi đó,

$$(X^+)_{0.5}^3 = \begin{bmatrix} 0.250 & 0.125 & 0.000 & 0.375 & 0.000 & 0.000 & 0.000 \\ 0.125 & 0.500 & 0.125 & 0.375 & 0.125 & 0.125 & 0.125 \\ 0.000 & 0.625 & 0.375 & 0.125 & 0.250 & 0.125 & 0.500 \\ 0.000 & 0.250 & 0.125 & 0.000 & 0.375 & 0.125 & 0.375 \\ 0.000 & 0.625 & 0.125 & 0.125 & 0.500 & 0.375 & 0.250 \end{bmatrix}$$

$$(X^+)_{0.5}^5 = \begin{bmatrix} 0.15625 & 0.18750 & 0.03125 & 0.28125 & 0.03125 & 0.03215 & 0.03215 \\ 0.12500 & 0.59375 & 0.18750 & 0.34375 & 0.25000 & 0.18750 & 0.25000 \\ 0.03125 & 0.81250 & 0.37500 & 0.21875 & 0.43750 & 0.25000 & 0.56250 \\ 0.00000 & 0.43750 & 0.18750 & 0.06250 & 0.37500 & 0.18750 & 0.37500 \\ 0.03125 & 0.81250 & 0.25000 & 0.21875 & 0.56250 & 0.37500 & 0.43750 \end{bmatrix}$$

Tại bước 2 của thuật toán: Sắp xếp các sản phẩm theo thứ tự giảm dần của trọng số cho người dùng  $u_5$  ta nhận được:  $p_2, p_5, p_7, p_6, p_3, p_4, p_1$ .

Bước cuối cùng của thuật toán: Chọn  $K=2$  sản phẩm chưa được người dùng đánh giá có trọng số cao để tư vấn cho  $u_5$  ta nhận được:  $p_3, p_7$ .

### 3.2.2.3. Phương pháp dự đoán trên đồ thị $G^-$

Để xem xét ảnh hưởng các đánh giá “không thích” vào quá trình dự đoán, ta có thể ước lượng mức độ đóng góp của các đánh giá này trên đồ thị  $G^-$  bằng cách phủ định lại phương pháp dự đoán trên đồ thị  $G^+$ .

Cụ thể phương pháp thay thế việc dự đoán trên đồ thị  $G^+$  bằng đồ thị  $G^-$ . Thay việc ước lượng trọng số đường đi từ đỉnh người dùng đến đỉnh sản phẩm dài sẽ có trọng số thấp, đường đi ngắn có trọng số cao bằng việc ước lượng trọng số các đường đi dài có trọng số cao, đường đi ngắn có trọng số thấp. Thay việc sử dụng hằng số khử nhiễu  $+\alpha$  bằng hằng số khử nhiễu  $-\alpha$  để trọng số các đường đi luôn âm và tăng dần theo độ dài đường đi. Thay việc sắp xếp các sản phẩm theo thứ tự giảm dần của trọng số bằng việc sắp xếp các sản phẩm theo thứ tự tăng dần của trọng số. Thay quá trình phân bổ các sản phẩm có trọng số cao cho người dùng hiện thời bằng việc loại bỏ các sản phẩm có trọng số thấp. Thuật toán dự đoán trên đồ thị  $G^-$  được thể hiện trong Hình 3.5.

**Đầu vào:**

- Ma trận  $X$  là biểu diễn của đồ thị  $G$ ;
- $\alpha$  là hằng số ( $0 < \alpha \leq 1$ ),  $L$  là độ dài đường đi;
- $K$  là số sản phẩm cần tư vấn.

**Đầu ra:**

- $K$  sản phẩm có trọng số nhỏ nhất chưa được người dùng đánh giá

**Các bước thực hiện:**

**Bước 1.** Tìm trọng số các đường đi độ dài lẻ  $L$  trên đồ thị  $G$  sao cho các đường đi có độ dài nhỏ được đánh trọng số thấp, các đường đi có độ dài lớn được đánh trọng số cao.

$$(X^-)_{\alpha}^L = \begin{cases} -\alpha \cdot X^- & \text{if } L=1 \\ (-\alpha)^2 \cdot X^- \cdot (X^-)^T \cdot (X^-)_{\alpha}^{L-2} & \text{if } L=3,5,7\dots \end{cases}$$

**Bước 2.** Sắp xếp các sản phẩm theo thứ tự tăng dần của trọng số  $x_{\alpha}^{-L}$ .

**Bước 3.** Loại bỏ  $K$  sản phẩm có trọng số  $x_{\alpha}^{-L}$  thấp nhất chưa được người dùng đánh giá ra khỏi danh sách các sản phẩm cần tư vấn cho người dùng hiện thời.

**Hình 3.5.** Thuật toán dự đoán trên đồ thị  $G$ 

**Mệnh đề 3.2.** Độ phức tạp thuật toán dự đoán trên đồ thị  $G$  là  $O(L \cdot N^{2.376})$ .

Trong đó,  $L$  là độ dài đường đi từ đỉnh người dùng đến đỉnh sản phẩm,  $N$  là số lượng người dùng.

**Chứng minh.** Thực vậy, tại bước 1 của thuật toán tích của  $L$  ma trận  $(X)$  và  $(X)^T$  sẽ có độ phức tạp là  $O(L \cdot X \cdot (X)^T)$ . Sử dụng thuật toán nhân hai ma trận hiệu quả nhất hiện nay của *Coppersmith–Winograd* sẽ cho ta độ phức tạp là  $O(N^{2.376})$  [44] khi  $X$  là ma trận vuông ( $N=M$ ). Như vậy, độ phức tạp thuật toán thực hiện tại bước 1 là  $O(L \cdot N^{2.376})$ .

Độ phức tạp thuật toán thực hiện tại bước 2 chính bằng độ phức tạp của thuật toán sắp xếp được sử dụng. Trong trường hợp sử dụng thuật toán Heap Sort, độ phức tạp thuật toán tại bước 2 là  $O(M \cdot \log_2 M)$ . Khi trường hợp xấu nhất

xảy ra tại Bước 1, ta có  $M = N$ . Do đó, độ phức tạp tại bước 2 của thuật toán là  $O(N \cdot \log_2 N)$ .

Độ phức tạp tại bước 3 của thuật toán là hằng số  $O(K)$  vì ta chỉ đơn thuần chọn  $K$  sản phẩm  $x_a^{-L}$  có trọng số thấp nhất chưa được người dùng đánh giá, đã được sắp xếp ở bước 2 để loại bỏ ra khỏi danh sách các sản phẩm cần tư vấn cho người dùng hiện thời.

Như vậy, độ phức tạp của toàn bộ thuật toán là

$$O(L \cdot N^{2.376} + N \cdot \log_2 N + K) = O(L \cdot N^{2.376}).$$

#### **Ví dụ minh họa:**

Ví dụ với ma trận  $X^-$  trong Bảng 3.5, lấy  $L = 5$  và  $\alpha = 0.5$ . Giả sử ta cần gỡ bỏ  $K = 2$  các sản phẩm cho người dùng  $u_5$ . Khi đó,

Tại bước 1 của thuật toán, ta tính được:

$$(X^-)_{0.5}^5 = \begin{bmatrix} -0.18750 & -0.15625 & -0.34375 & -0.03125 & -0.15625 & -0.34375 & -0.15625 \\ -0.03125 & -0.03125 & -0.46875 & -0.00000 & -0.31250 & -0.18750 & -0.31250 \\ -0.34375 & -0.15625 & -0.03125 & -0.28125 & -0.00000 & -0.18750 & -0.00000 \\ -0.50000 & -0.34375 & -0.18750 & -0.18750 & -0.03125 & -0.50000 & -0.03125 \\ -0.12500 & -0.03125 & -0.00000 & -0.15625 & -0.00000 & -0.03125 & -0.00000 \end{bmatrix}$$

Tại bước 2 của thuật toán, sắp xếp các sản phẩm theo thứ tự tăng dần của trọng số, ta nhận được:  $p_4, p_1, p_2, p_6, p_3, p_5, p_7$ .

Bước cuối cùng của thuật toán chọn các sản phẩm có trọng số nhỏ nhất chưa được  $u_5$  đánh giá đưa ra khỏi danh sách các sản phẩm cần tư vấn cho  $u_5$ , ta nhận được:  $p_1, p_3$ .

#### **3.2.2.4. Phương pháp dự đoán theo tất cả đánh giá**

Phương pháp dự đoán trên đồ thị  $G^+$  chỉ được thực hiện trên những đánh giá “*thích*” của người dùng đối với sản phẩm, phương pháp dự đoán trên đồ thị  $G^-$  chỉ được thực hiện trên những đánh giá “*không thích*” của người dùng đối với sản phẩm. Việc bỏ qua những đánh giá “*không thích*” của người dùng đối với sản phẩm có những ảnh hưởng không nhỏ đến chất lượng dự đoán, vì đánh giá

“thích” hay “không thích” đều phản ánh thói quen và sở thích sử dụng sản phẩm của người dùng.

Trong ví dụ trên, nếu thực hiện dự đoán trên đồ thị  $G^+$  thì  $p_3$  được xem là phương án dùng để tư vấn cho  $u_5$ . Nếu thực hiện dự đoán trên đồ thị  $G^-$  thì  $p_3$  được xem là phương án loại bỏ ra khỏi danh sách các sản phẩm dùng để tư vấn cho  $u_5$ . Để khắc phục mâu thuẫn này, ta có thể mở rộng phương pháp dự đoán cho tất cả các đánh giá “thích” và “không thích” của người dùng. Các bước cụ thể của phương pháp được tiến hành như Hình 3.6.

**Đầu vào:**

- Ma trận  $X^+$ ,  $X^-$  là biểu diễn của đồ thị  $G^+$ ,  $G^-$
- $\alpha$  là hằng số ( $0 < \alpha \leq 1$ ),  $L$  là độ dài đường đi;
- $K$  là số sản phẩm cần tư vấn.

**Đầu ra:**

- $K$  sản phẩm có trọng số cao nhất chưa được người dùng đánh giá

**Các bước thực hiện:**

**Bước 1.** Tính toán ma trận trọng số  $(X^+)_\alpha^L$  của các đường đi độ dài lẻ  $L$  trên ma trận  $X^+$  sao cho các đường đi có độ dài nhỏ được đánh trọng số cao, các đường đi có độ dài lớn được đánh trọng số thấp.

$$(X^+)_\alpha^L = \begin{cases} \alpha \cdot X^+ & \text{if } L=1 \\ \alpha^2 \cdot X^+ \cdot (X^+)^T \cdot (X^+)_\alpha^{L-2} & \text{if } L=3,5,7,\dots \end{cases}$$

**Bước 2.** Tính toán ma trận trọng số  $(X^-)_\alpha^L$  của các đường đi độ dài chẵn  $L$  trên ma trận  $X^-$  sao cho các đường đi có độ dài nhỏ được đánh trọng số thấp, các đường đi có độ dài lớn được đánh trọng số cao.

$$(X^-)_\alpha^L = \begin{cases} -\alpha \cdot X^- & \text{if } L=1 \\ (-\alpha)^2 \cdot X^- \cdot (X^-)^T \cdot (X^-)_\alpha^{L-2} & \text{if } L=3,5,7,\dots \end{cases}$$

**Bước 3.** Kết hợp ma trận trọng số  $X_\alpha^L = (X^+)_\alpha^L + (X^-)_\alpha^L$ .

**Bước 4.** Sắp xếp các sản phẩm theo thứ tự tăng dần của trọng số  $x_\alpha^L$ .

**Bước 5.** Chọn  $K$  sản phẩm có trọng số  $x_\alpha^L$  cao nhất chưa được đánh giá để tư vấn cho người dùng hiện thời.

**Hình 3.6.** Thuật toán dự đoán trên tất cả đánh giá

**Mệnh đề 3.3.** Độ phức tạp thuật toán dự đoán trên tất cả đánh giá là  $O(L.N^{2.376})$ . Trong đó,  $L$  là độ dài đường đi từ đỉnh người dùng đến đỉnh sản phẩm,  $N$  là số lượng người dùng.

**Chứng minh.** Vì ma trận  $X^+$  và  $X^-$  có cùng cấp nên theo Mệnh đề 3.1 và Mệnh đề 3.2 ta có độ phức tạp thuật toán tại bước 1 và bước 2 của thuật toán đều là  $O(L.N^{2.376})$ , trong đó,  $N$  là cấp của ma trận  $X^+$ ,  $X^-$ . Lập luận tương tự như việc chứng minh Mệnh đề 3.1 và Mệnh đề 3.2 ta có độ phức tạp của toàn bộ thuật toán là  $O(L.N^{2.376})$ .

**Ví dụ minh họa:**

Ví dụ với ma trận  $X^+$  trong Bảng 3.4,  $X^-$  trong Bảng 3.5, lấy  $L=5$  và  $\alpha=0.5$ . Giả sử ta cần tư vấn  $K=2$  sản phẩm cho người dùng  $u_5$ . Khi đó thuật toán thực hiện như sau:

*Tại bước 1 của thuật toán ta tính được:*

$$(X^+)_{0.5}^5 = \begin{bmatrix} 0.15625 & 0.18750 & 0.03125 & 0.28125 & 0.03125 & 0.03215 & 0.03215 \\ 0.12500 & 0.59375 & 0.18750 & 0.34375 & 0.25000 & 0.18750 & 0.25000 \\ 0.03125 & 0.81250 & 0.37500 & 0.21875 & 0.43750 & 0.25000 & 0.56250 \\ 0.00000 & 0.43750 & 0.18750 & 0.06250 & 0.37500 & 0.18750 & 0.37500 \\ 0.03125 & 0.81250 & 0.25000 & 0.21875 & 0.56250 & 0.37500 & 0.43750 \end{bmatrix}$$

*Tại bước 2 của thuật toán ta tính được:*

$$(X^-)_{0.5}^5 = \begin{bmatrix} -0.18750 & -0.15625 & -0.34375 & -0.03125 & -0.15625 & -0.34375 & -0.15625 \\ -0.03125 & -0.03125 & -0.46875 & -0.00000 & -0.31250 & -0.18750 & -0.31250 \\ -0.34375 & -0.15625 & -0.03125 & -0.28125 & -0.00000 & -0.18750 & -0.00000 \\ -0.50000 & -0.34375 & -0.18750 & -0.18750 & -0.03125 & -0.50000 & -0.03125 \\ -0.12500 & -0.03125 & -0.00000 & -0.15625 & -0.00000 & -0.03125 & -0.00000 \end{bmatrix}$$

*Tại bước 3 của thuật toán ta tính được:*

$$X_{0.5}^5 = \begin{bmatrix} -0.03125 & +0.03125 & -0.03125 & +0.25000 & -0.12500 & -0.32150 & -0.12500 \\ +0.09375 & +0.56250 & -0.28125 & +0.34375 & -0.06250 & +0.00000 & -0.00625 \\ -0.31250 & +0.65625 & +0.34375 & -0.06250 & +0.43750 & +0.62500 & +0.56250 \\ -0.50000 & +0.09375 & +0.00000 & -0.12500 & +0.34375 & -0.31250 & +0.34375 \\ -0.09375 & +0.78125 & +0.25000 & +0.06250 & +0.56250 & +0.37500 & +0.43750 \end{bmatrix}$$

Tại bước 4 của thuật toán, ta sắp xếp được:  $p_2, p_5, p_7, p_6, p_3, p_4, p_1$ .

Tại bước cuối cùng của thuật toán ta chọn  $p_7$  và  $p_3$  tư vấn cho  $u_5$ .

Như đã đề cập trong Chương 2, lọc cộng tác trong trường hợp dữ liệu thưa thường dựa vào phương pháp giảm số chiều ma trận đánh giá. Hạn chế lớn nhất của phương pháp này là có thể mất thông tin trong khi giảm số chiều ma trận. Hạn chế này cũng có thể khắc phục dựa trên việc xem xét và mở rộng độ dài đường đi trên mô hình đồ thị như đã được trình bày ở trên.

### 3.3. KẾT HỢP LỌC CỘNG TÁC VÀ LỌC NỘI DUNG

Mục này trình bày mô hình đồ thị kết hợp giữa lọc cộng tác và lọc nội dung. Đối với lọc cộng tác, mô hình quan tâm xem xét và biểu diễn cho tất cả các đánh giá “*thích*” hoặc “*không thích*” như đã trình bày trong Mục 3.2. Đối với các đặc trưng nội dung, mô hình đề xuất phương pháp xác định mức độ quan trọng của từng đặc trưng nội dung cụ thể đối với mỗi người dùng dựa trên ước lượng sự tương tự theo nội dung và đánh giá người dùng. Phương pháp dự đoán được thực hiện dựa trên mức độ đóng góp của các đánh giá người dùng và đặc trưng nội dung sản phẩm người dùng ưa thích.

#### 3.3.1. Biểu diễn đồ thị kết hợp

Cho ma trận đánh giá người dùng  $R = (r_{ij})$  được xác định theo công thức (3.1), ma trận nội dung sản phẩm  $Y = (y_{ij})$  được xác định theo công thức (3.2), ma trận  $X = (x_{ij})$  được xác định theo công thức (3.3). Khi đó, đồ thị kết hợp  $G = (V, E)$  được hình thành bởi tập đỉnh  $V = U \cup P \cup C$  ( $U$  là tập người dùng,  $P$  là tập sản phẩm,  $C$  là tập đặc trưng nội dung sản phẩm) và tập cạnh  $E$ . Trong đó, tập cạnh  $E$  bao gồm hai loại cạnh: Cạnh biểu diễn đánh giá của người dùng đối với sản phẩm và cạnh biểu diễn giữa sản phẩm và nội dung sản phẩm. Cạnh nối



giữa đỉnh  $u_i \in U$  và đỉnh  $p_j \in P$  được thiết lập nếu  $x_{ij} \neq 0$ , cạnh nối giữa  $p_i \in P$  và  $c_j \in C$  được thiết lập nếu  $y_{ij} \neq 0$ . Các cạnh đánh giá  $(u_i, p_j)$  được đánh trọng số là  $r_{ij} = +1$  hoặc  $r_{ij} = -1$ . Các cạnh nối giữa đỉnh người dùng và đỉnh nội dung  $(u_i, c_j)$  được xem như có trọng số bằng nhau là  $+1$ . Ví dụ với ma trận đánh giá  $R$  được cho trong Bảng 3.6, ma trận  $X$  được thể hiện trong Bảng 3.7, ma trận  $Y$  trong Bảng 3.8 thì đồ thị kết hợp được biểu diễn như Hình 3.7.

**Bảng 3.6.** Ma trận đánh giá  $R$

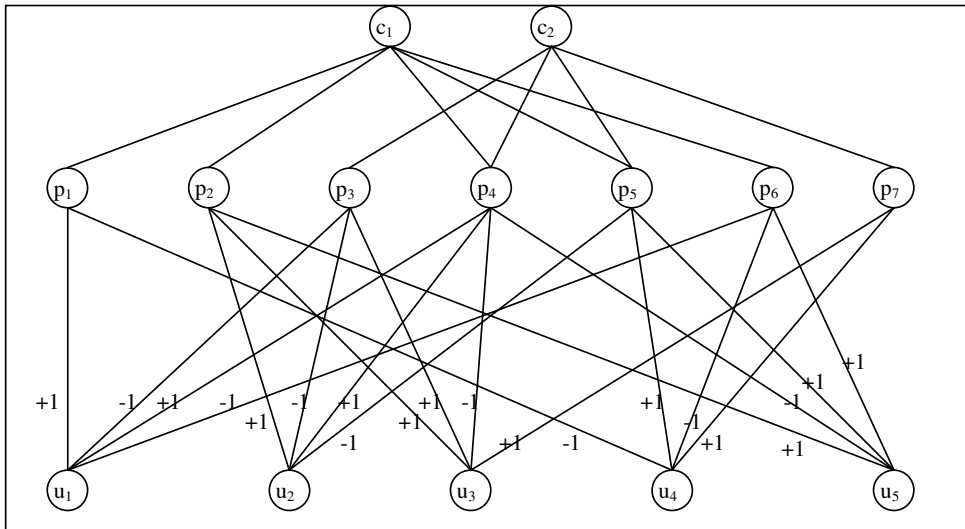
Người dùng	Sản phẩm						
	p1	p2	p3	p4	p5	p6	p7
$u_1$	1	$\emptyset$	-1	1	$\emptyset$	-1	$\emptyset$
$u_2$	$\emptyset$	1	-1	1	-1	$\emptyset$	$\emptyset$
$u_3$	$\emptyset$	1	1	-1	$\emptyset$	$\emptyset$	1
$u_4$	-1	$\emptyset$	$\emptyset$	$\emptyset$	1	-1	1
$u_5$	$\emptyset$	1	$\emptyset$	-1	1	1	$\emptyset$

**Bảng 3.7.** Ma trận Người dùng- Sản phẩm  $X$

Người dùng	Sản phẩm						
	p1	p2	p3	p4	p5	p6	p7
$u_1$	1	0	1	1	0	1	0
$u_2$	0	1	1	1	1	0	0
$u_3$	0	1	1	1	0	0	1
$u_4$	1	0	0	0	1	1	1
$u_5$	0	1	0	1	1	1	0

**Bảng 3.8.** Ma trận Sản phẩm- Nội dung Y

Sản phẩm	Nội dung	
	c <sub>1</sub>	c <sub>2</sub>
p <sub>1</sub>	1	0
p <sub>2</sub>	1	0
p <sub>3</sub>	0	1
p <sub>4</sub>	1	1
p <sub>5</sub>	1	1
p <sub>6</sub>	1	0
p <sub>7</sub>	0	1



**Hình 3.7.** Đồ thị kết hợp người dùng và nội dung sản phẩm

Trong những nghiên cứu trước đây [116, 117, 122], các tác giả chỉ quan tâm đến các đánh giá “thích” ( $r_{ij}=+1$ ) mà không quan tâm đến các đánh giá “không thích” ( $r_{ij}=-1$ ). Nói cách khác, biểu diễn đồ thị chỉ bao gồm các cạnh có trọng số  $+1$ . Điều này chưa thực sự hợp lý trong lọc cộng tác vì cả hai đánh giá “thích” và “không thích” đều phản ánh thói quen người dùng sản phẩm.

Việc áp đặt lập luận hai sản phẩm tương tự với nhau về nội dung để suy ra sự tương tự về sở thích chưa phản ánh đúng thói quen tự nhiên của người dùng. Phương pháp biểu diễn liên kết giữa người dùng và nội dung sản phẩm được

tính toán thông qua sự tương tự giữa các đặc trưng nội dung sản phẩm. Tuy nhiên, cách làm này không xem xét đến ảnh hưởng của các giá trị đánh giá của người dùng đối với các đặc trưng nội dung sản phẩm. Chính vì vậy, không điều chỉnh hợp lệ sự tương tự giữa những người dùng. Để chứng tỏ điều này, chúng ta xem xét ví dụ cho bởi Hình 3.7.

Trong ví dụ này, sản phẩm  $p_3$  và  $p_4$  có chung đặc trưng  $c_2$ . Nếu xét trên quan điểm của lọc nội dung, phương pháp tính toán sự tương tự theo nội dung sẽ cho ta kết quả  $p_3$  tương tự với  $p_4$  vì chúng cùng chung đặc trưng  $c_2$ . Tuy nhiên, điều này không đúng cho người dùng  $u_1$  vì  $u_1$  đánh giá  $p_3$  là +1 (“*thích*”) và  $p_4$  là -1 (“*không thích*”). Nói cách khác nếu xem xét ở góc độ người dùng,  $p_3$  và  $p_4$  không tương tự nhau và  $c_2$  không có ảnh hưởng gì đến thói quen sử dụng sản phẩm của  $u_1$ . Ví dụ này chứng tỏ sự cần thiết kết hợp đánh giá người dùng để tính toán mức độ tương tự giữa sản phẩm và sản phẩm thông qua nội dung của nó.

Để kết hợp đánh giá người dùng trong khi tính toán sự tương tự giữa các sản phẩm, ta coi mỗi đặc trưng sản phẩm có mức độ quan trọng riêng đối với thói quen sử dụng sản phẩm của người dùng. Mức độ quan trọng mỗi đặc trưng cụ thể có thể ước lượng được bằng cách quan sát tất cả các sản phẩm  $p_j \in P$  chứa đựng đặc trưng  $c_k \in C$  mà  $u_i \in U$  đã đánh giá trước đó. Ví dụ để xem xét đặc trưng  $c_1$  quan trọng hay không quan trọng với người dùng  $u_1$ , ta cần quan sát các sản phẩm  $p_1, p_3, p_6$  có đặc trưng  $c_1$  để thực hiện tính toán. *Phương pháp ước lượng mức độ quan trọng mỗi đặc trưng nội dung sản phẩm cho từng người dùng được trình bày chi tiết trong Mục 3.3.2. Đây cũng điểm mới khác biệt quan trọng của mô hình đề xuất so với các mô hình lọc kết hợp dựa trên đồ thị khác.*

### **3.3.2. Xây dựng liên kết người dùng và nội dung sản phẩm**

Trong phần này, chúng tôi đề xuất một phương pháp xây dựng liên kết người dùng với nội dung sản phẩm trên cơ sở cá nhân hóa các liên kết này.

Với đồ thị được biểu diễn ở trên, bằng trực quan có thể nhận thấy, người dùng  $u_i$  “*thích*” hay “*không thích*” nội dung  $c_j$  phụ thuộc vào số các sản phẩm  $p_k \in P$  có nội dung  $c_j$  mà  $u_i$  đã đánh giá ( $r_{ik} \neq 0$ ). Gọi  $s_{ik}$  là số các sản phẩm  $p_j$  có nội dung  $c_k$  mà người dùng  $u_i$  đã đánh giá. Giá trị  $s_{ik}$  chính là số đường đi độ dài 2 từ đỉnh người dùng  $u_i$  đến đỉnh đặc trưng nội dung  $c_k$  thông qua các đỉnh trung gian  $p_j$ .

$$s_{ik} = \sum_{j=1}^M x_{ij} * y_{jk} \quad (3.9)$$

Đề ý rằng,  $s_{ik}$  bao gồm số các sản phẩm  $p_j$  có nội dung  $c_k$  mà người dùng  $u_i$  đã đánh giá +1 “*thích*” và số các sản phẩm người dùng  $u_i$  đánh giá -1 “*không thích*”. Gọi  $w_{ik}$  là hiệu số giữa tập các sản phẩm  $p_j$  có nội dung  $c_k$  người dùng  $u_i$  đánh giá “*thích*” và tập các sản phẩm  $p_j$  có nội dung  $c_k$  người dùng  $u_i$  đánh giá “*không thích*”. Giá trị  $w_{ik}$  chính là tích của hai ma trận  $r_{ij}$  và  $y_{jk}$  được xác định theo công thức (3.10).

$$w_{ik} = \sum_{j=1}^M r_{ij} * y_{jk} \quad (3.10)$$

Khi giá trị của  $s_{ik}$  lớn và  $w_{ik} > 0$  (số lượng đánh giá các sản phẩm  $p_j$  của người dùng  $u_i$  có nội dung  $c_k$  “*thích*” lớn hơn nhiều số lượng đánh giá “*không thích*”), ta có thể khẳng định nội dung  $c_k$  là quan trọng đối với  $u_i$ . Nếu giá trị của  $s_{ik}$  lớn và  $w_{ik} \leq 0$  (số lượng đánh giá các sản phẩm  $p_j$  của người dùng  $u_i$  có nội dung  $c_k$  “*thích*” nhỏ hơn số lượng đánh giá “*không thích*”) ta cũng có thể khẳng định nội dung  $c_k$  là không quan trọng đối với  $u_i$ . Tuy nhiên, trong trường hợp  $s_{ik}$  nhỏ thì dù  $w_{ik} \leq 0$  hay  $w_{ik} > 0$  ta cũng không thể khẳng định nội dung  $p_k$  là quan trọng hay không quan trọng đối với người dùng  $u_i$ . Để ngăn ngừa điều này, ta sử dụng ngưỡng  $\gamma$  để phân tập đánh giá người dùng thành hai loại: Tập đánh giá của người dùng  $u_i$  cho các sản phẩm  $p_j$  có nội dung  $c_k$  lớn hơn  $\gamma$  ( $s_{ik} > \gamma$ ) và tập  $s_{ik} \leq \gamma$ ; ngưỡng  $T$  ( $0 < T \leq 1$ ) dùng để so sánh với tỷ lệ giữa những đánh giá “*thích*” trên toàn bộ đánh giá  $s_{ik}$  ( $w_{ik}/s_{ik}$ ). Công thức (3.11) dưới đây dùng để xác định mức độ quan trọng của đặc trưng nội dung  $c_k$  đối với người dùng  $u_i$ .

$$v_{ik} = \begin{cases} \frac{\min(s_{ik}, \gamma) * w_{ik}}{\gamma * s_{ik}} & \text{if } \frac{w_{ik}}{s_{ik}} > T \\ 0 & \text{otherwise} \end{cases} \quad (3.11)$$

Trong công thức (3.11), nếu  $s_{ik} > \gamma$  thì  $\frac{\min(s_{ik}, \gamma)}{\gamma} = 1$ , khi đó  $v_{ik}$  được xác định theo công thức (3.12). Nếu  $s_{ik} \leq \gamma$  thì  $\frac{\min(s_{ik}, \gamma)}{\gamma} = \frac{s_{ik}}{\gamma}$ , khi đó  $v_{ik}$  được xác định theo công thức (3.13).

$$v_{ik} = \begin{cases} \frac{w_{ik}}{s_{ik}} & \text{if } \frac{w_{ik}}{s_{ik}} > T \\ 0 & \text{otherwise} \end{cases} \quad (3.12)$$

$$v_{ik} = \begin{cases} \frac{w_{ik}}{\gamma} & \text{if } \frac{w_{ik}}{s_{ik}} > T \\ 0 & \text{otherwise} \end{cases} \quad (3.13)$$

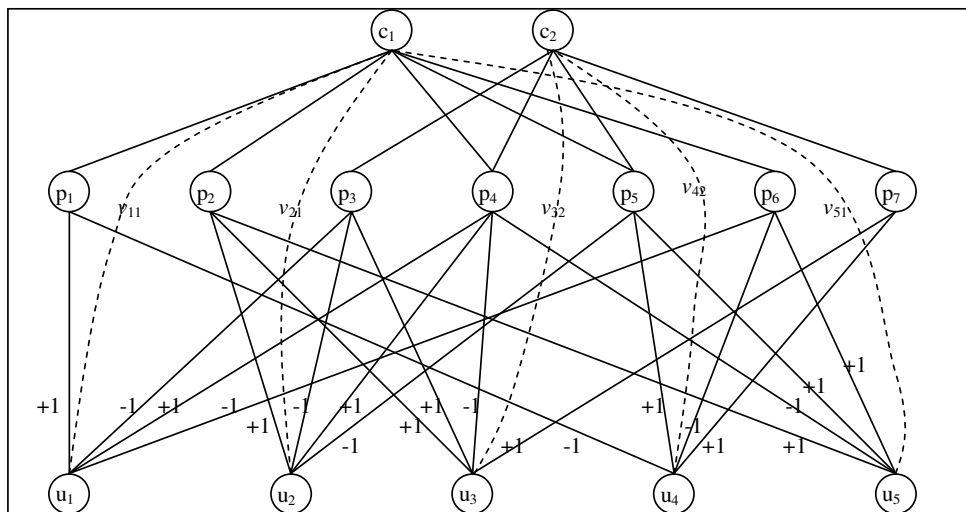
Trong thử nghiệm, ta có thể dùng ngưỡng  $\gamma = 20$ , nghĩa là nếu người dùng  $u_i$  đánh giá các sản phẩm  $p_j$  có nội dung  $c_k$  lớn hơn 20 thì  $v_{ik}$  được xác định theo (3.12), trường hợp còn lại  $v_{ij}$  được tính theo (3.13). Giá trị ngưỡng  $T$  được chọn là 0.7, nghĩa là số cạnh có trọng số dương gấp đôi số cạnh có trọng số âm đối với mỗi đặc trưng được xem là quan trọng.

Với mỗi cặp đỉnh  $(u_i, c_k)$  có  $v_{ik} > 0$ , chúng ta thiết lập một liên kết trực tiếp giữa người dùng  $u_i$  và đặc trưng  $c_k$  với trọng số  $v_{ik}$ . Ví dụ với các ma trận  $R, X, Y$  được cho trong Bảng 3.6, 3.7, 3.8, chọn  $\gamma = 2$  và  $T = 0.3$ . Khi đó,  $s_{ik}, w_{ik}, v_{ik}$  được tính toán theo như kết quả dưới đây.

$$R = \begin{vmatrix} 1 & \emptyset & -1 & 1 & \emptyset & -1 & \emptyset \\ \emptyset & 1 & -1 & 1 & -1 & \emptyset & \emptyset \\ \emptyset & 1 & 1 & -1 & \emptyset & \emptyset & 1 \\ -1 & \emptyset & \emptyset & \emptyset & 1 & -1 & 1 \\ \emptyset & 1 & \emptyset & -1 & 1 & 1 & \emptyset \end{vmatrix} \quad X = \begin{vmatrix} 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \end{vmatrix}$$

$$Y = \begin{vmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{vmatrix} \quad S = \begin{vmatrix} 3 & 2 \\ 3 & 3 \\ 2 & 3 \\ 2 & 2 \\ 4 & 2 \end{vmatrix} \quad W = \begin{vmatrix} 1 & 0 \\ 1 & -1 \\ 0 & 1 \\ 0 & 2 \\ 2 & 0 \end{vmatrix} \quad V = \begin{vmatrix} 0.33 & 0.00 \\ 0.33 & 0.00 \\ 0.00 & 0.33 \\ 0.00 & 1.00 \\ 0.50 & 0.00 \end{vmatrix}$$

Ma trận  $S$  ghi nhận số các đường đi từ đỉnh người dùng  $u_i$  đến đỉnh nội dung  $c_k$ . Ma trận  $W$  ghi nhận tổng các trọng số của tất cả các đường đi từ đỉnh người dùng  $u_i$  đến đỉnh nội dung  $c_k$ . Giá trị  $v_{ik}$  được tính toán dựa trên giá trị của  $s_{ik}$  và  $w_{ik}$ , với  $s_{ik} > \gamma$  ( $\gamma=2$ ) thì  $v_{ik}$  được tính toán theo (3.11), những giá trị còn lại được tính toán theo (3.12). Các giá trị  $v_{ik} > 0$  phản ánh đặc trưng  $c_k$  quan trọng đối với người dùng  $u_i$  và được thiết lập một cạnh nối trực tiếp từ đỉnh người dùng đến đỉnh đặc trưng nội dung. Trong ví dụ trên,  $v_{11} \neq 0$ ,  $v_{21} \neq 0$ ,  $v_{51} \neq 0$ ,  $v_{32} \neq 0$ ,  $v_{42} \neq 0$  nên ta thiết lập được các cạnh  $(u_1, c_1)$ ,  $(u_2, c_1)$ ,  $(u_5, c_1)$ ,  $(u_3, c_2)$ , và  $(u_4, c_2)$ . Đồ thị Hình 3.7 được biến đổi thành đồ thị Hình 3.8, trong đó các cạnh mới thiết lập thêm được nối bằng các nét đứt.



**Hình 3.8.** Đồ thị thiết lập liên kết giữa người dùng và đặc trưng nội dung

### 3.3.3. Phương pháp dự đoán

Các phương pháp lọc cộng tác thuần túy, lọc nội dung thuần túy, lọc kết hợp đơn giản, lọc kết hợp dựa vào ước lượng mức độ quan trọng của các đặc trưng nội dung (Ký hiệu là Combined-Graph) có thể xem như một bài toán tìm kiếm trên đồ thị kết hợp. Để thuận tiện cho việc trình bày, ta sử dụng đồ thị trong Hình 3.8 làm ví dụ minh họa. Các phương pháp có thể được thực hiện như dưới đây.

#### 3.3.3.1. Lọc cộng tác dựa trên mô hình đồ thị kết hợp

Các phương pháp lọc cộng tác thuần túy thực hiện dự đoán dựa trên việc tính toán mức độ tương tự giữa  $u_a$  với những người dùng còn lại thông qua các giá trị đánh giá  $r_{ij}$ , sau đó phân bổ  $K$  sản phẩm chưa được  $u_a$  đánh giá có mức độ tương tự cao nhất đối với  $u_a$ . Chẳng hạn, ta cần phân bổ các sản phẩm cho người dùng  $u_5$ , sự tương tự giữa  $u_5$  và  $u_3$  là cao nhất vì chúng có chung nhiều nhất các đánh giá giống nhau ( $r_{52} = r_{32} = +1$  và  $r_{54} = r_{34} = -1$ ). Dựa vào nhận xét này, các sản phẩm chưa được  $u_5$  đánh giá là  $p_3$  và  $p_7$  sẽ được phân bổ cho  $u_5$ . Kém tương tự nhất với  $u_5$  là  $u_1$  vì chúng không tương tự nhau bất kỳ đánh giá nào. Chính vì vậy  $p_1$  sẽ không bao giờ được phân bổ cho  $p_5$ .

Phương pháp dự đoán này có thể dễ dàng cài đặt bằng mô hình đồ thị thông qua việc tính toán các đường đi độ dài 3 từ đỉnh người dùng đến đỉnh sản phẩm thông qua các cạnh đánh giá. Những sản phẩm nào có số đường đi độ dài 3 nhiều nhất đến nó sẽ được phân bổ cho người dùng hiện thời.

#### 3.3.3.2. Lọc nội dung dựa trên mô hình đồ thị kết hợp

Các phương pháp lọc theo nội dung thuần túy thực hiện dự đoán dựa trên việc so sánh nội dung sản phẩm người dùng từng ưa thích và chọn ra những sản phẩm có nội dung tương tự nhất để phân bổ cho họ những mặt hàng này. Ví dụ ta cần phân bổ các sản phẩm cho người dùng  $u_5$ , vì  $u_5$  đã từng thích hợp với việc sử dụng  $p_2, p_5, p_6$  có đặc trưng nội dung  $c_1$ ,  $p_1$  có đặc trưng nội dung  $c_1$  nên  $p_1$  được xem là tương tự nhất với  $p_2, p_5, p_6$  sẽ được phân bổ cho  $u_5$ . Kém tương tự

nhất đối với  $u_5$  là  $p_7$  vì  $u_5$  đã từng không thích hợp với việc phân bổ  $p_4$  có đặc trưng nội dung  $c_2$  và  $p_7$  là sản phẩm tương tự nhất với  $p_4$  chứa đựng đặc trưng  $c_2$ . Như vậy, với lọc nội dung,  $p_1$  được xem xét như phương án ưu tiên nhất phân bổ cho  $u_5$  và  $p_7$  luôn bị gỡ bỏ ra khỏi danh sách các sản phẩm phân bổ cho  $u_5$ . Trái lại, lọc cộng tác lại xem  $p_7$  là phương án ưu tiên nhất phân bổ cho  $u_5$  còn  $p_1$  luôn là phương án gỡ bỏ ra khỏi danh sách các sản phẩm phân bổ cho  $u_5$ . Ví dụ này một lần nữa minh chứng cho sự khác biệt lớn giữa cánh tiếp cận của lọc cộng tác và lọc nội dung.

Phương pháp dự đoán theo nội dung cũng dễ dàng cài đặt dựa trên mô hình đồ thị bằng cách xem xét tất cả các đường đi thông qua đỉnh đặc trưng nội dung ( $u_5-p_2-c_1-p_1$ ,  $u_5-p_5-c_1-p_1$  và  $u_5-p_6-c_1-p_1$ ). Những sản phẩm nào có nhiều đường đi nhất thông qua đỉnh đặc trưng nội dung sẽ được phân bổ cho người dùng hiện thời.

### ***3.3.3.3. Phương pháp lọc kết hợp đơn giản***

Phương pháp lọc kết hợp đơn giản (Ký hiệu là SimpleHybrid) được thực hiện bằng cách kết hợp phương pháp lọc nội dung như đã trình bày trong Mục 3.4.3.2 và lọc cộng tác trong Mục 3.4.3.1. Những sản phẩm nào có số đường đi nhiều nhất đến nó sẽ được dùng để phân bổ cho người dùng hiện thời. Phương pháp này dễ dàng được thực hiện bằng cách tổng hợp số đường đi độ dài 3 từ đỉnh người dùng đến đỉnh sản phẩm theo từng phương pháp riêng biệt nhau, sau đó cộng kết quả để tìm những sản phẩm có nhiều đường đi nhất để phân bổ cho người dùng.

### ***3.3.3.4. Phương pháp kết hợp đề xuất***

Như đã trình bày ở trên, phương pháp dự đoán đề xuất dựa trên việc ước lượng mức độ quan trọng các đặc trưng nội dung cho mỗi người dùng. Để thực hiện điều này trên đồ thị kết hợp, ta xem xét và thực hiện tính toán mức độ đóng góp vào kết quả dự đoán cho hai loại đường đi: *đường đi thông qua đỉnh nội dung* (*đường đi loại 1*) và *đường đi thông qua đỉnh sản phẩm* (*đường đi loại 2*).



*Đường đi loại 1* luôn có độ dài 2 đi từ đỉnh người dùng  $u_i \in U$  thông qua các cạnh nối đỉnh nội dung  $c_k \in C$  đến đỉnh sản phẩm  $p_j \in P$ . Những đường đi này phản ánh sự tương tự của người dùng sản phẩm đối với các đặc trưng nội dung. Trong ví dụ Hình 3.8, đường đi này có dạng  $u_1-c_1-p_2, u_1-c_1-p_4$ . Điều này là hoàn toàn tự nhiên đối với người dùng  $u_1$  vì  $u_1$  thích hợp với việc phân bổ các sản phẩm có nội dung  $c_1$  và  $p_2, p_5$  là hai sản phẩm có đặc trưng nội dung  $c_1$ . Cách làm này giống như các phương pháp lọc theo nội dung. Tuy nhiên, điểm khác biệt quan trọng của mô hình này và lọc nội dung ở chỗ việc so sánh nội dung dựa trên cơ sở đánh giá của người dùng. Ngoài các đường đi độ dài 2, phương pháp không mở rộng thêm độ dài đường đi loại này. Trọng số mỗi đường đi này được cho là 1.

*Đường đi loại 2* bao gồm các đường đi từ đỉnh người dùng đến đỉnh sản phẩm chưa được người dùng đánh giá thông qua các đỉnh sản phẩm và đỉnh người dùng trung gian. Độ dài những đường đi này không vượt quá  $L$ . Chẳng hạn các đường đi có dạng  $u_1-p_4-u_3-p_2, u_1-p_4-u_3-p_2-u_3-p_7$ . Vì chúng ta quan tâm đến những liên kết giữa đỉnh người dùng và đỉnh sản phẩm nên độ dài các đường đi này luôn là một số lẻ. Những đường đi độ dài lẻ có thể thông qua các cạnh có trọng số âm hoặc các cạnh có trọng số dương đều được xem xét đến trong quá trình dự đoán. Các đường đi loại này bao gồm:

- *Tất cả các đường đi từ đỉnh người dùng đến đỉnh sản phẩm thông qua các cạnh trung gian đều có trọng số dương.* Ví dụ các đường đi  $u_1-p_4-u_2-p_2$ , và  $u_2-p_2-u_3-p_3$  (Hình 3.8). Những đường đi loại này được xem là quan trọng và sẽ được đánh trọng số cao. Những đường đi càng dài sẽ ít được chú ý hơn bằng cách nhân với một thừa số  $\alpha$  ( $0 \leq \alpha \leq 1$ ) để giảm trọng số. Trọng số các đường đi này được tính toán trên đồ thị  $G^+$  như đã trình bày trong Mục 3.2.2.2.
- *Tất cả các đường đi từ đỉnh người dùng đến đỉnh sản phẩm thông qua các cạnh trung gian đều có trọng số âm.* Ví dụ các đường đi  $u_1-p_3-u_2-p_5$ , và  $u_1-p_6-u_4-p_1$  trong Hình 3.8. Những đường đi loại này cũng được xem là

quan trọng và sẽ được đánh trọng số cao. Những đường đi càng dài sẽ ít được chú ý hơn bằng cách nhân nó với một thừa số  $\alpha$  ( $0 \leq \alpha \leq 1$ ) để giảm trọng số. Trọng số các đường đi này được tính toán trên đồ thị  $G$  như đã trình bày trong Mục 3.2.2.3.

- *Những đường đi qua hai đỉnh trung gian và kết thúc tại cùng một đỉnh nhưng trái dấu*, điều đó có nghĩa cả hai người dùng có đánh giá khác nhau về sản phẩm này. Đối với những đường đi này, chúng ta không cần xem xét đến vì hai người dùng không tương đồng với nhau về sở thích, ví dụ các đường đi  $u_1-p_3-u_3-p_4$  có trọng số  $(u_1, p_3) = -1, (p_3, u_3) = 1$ .
- *Những đường thông qua hai đỉnh liên tục nhau đều có trọng số âm*, ví dụ đường  $u_1-p_6-u_4-p_5$ , trong đó liên kết  $u_1-p_6$  và  $p_6-u_4$  đều có trọng số âm. Điều này có nghĩa hai người dùng đều tương tự với  $p_6$  (đều là không thích hợp). Tuy nhiên, trong thử nghiệm các đường đi loại này cho lại kết quả dự đoán không cao. Do vậy, ta không cần xem xét đến những đường đi này.

Để xác định mức độ đóng góp của mỗi loại đường đi vào kết quả dự đoán, ta sử dụng tham số  $\lambda$  ( $0 \leq \lambda \leq 1$ ) điều chỉnh mức độ ưu tiên cho từng loại. Gọi  $Y_1^2$  là trọng số các đường đi loại 1 có độ dài 2 từ đỉnh người dùng bất kỳ đến các đỉnh sản phẩm thông các đỉnh nội dung có dạng  $u_i-c_j-p_r$ . Gọi  $X_\alpha^L$  là trọng số các đường đi loại 2 từ đỉnh người dùng bất kỳ  $u_i$  đến các đỉnh sản phẩm  $p_r$  thông qua các đỉnh sản phẩm có dạng  $u_i-p_j-u_k-p_r$ . Khi đó, mức độ đóng góp của mỗi loại đường đi  $W$  được xác định theo công thức (3.14). Thuật toán dự đoán trên đồ thị kết hợp được thể hiện trong Hình 3.9.

$$W = \lambda X_\alpha^L + (1 - \lambda) Y_1^2 \quad (3.14)$$

Trong công thức (3.14), nếu ta ưu tiên cho lọc cộng tác thì  $\lambda$  được lấy gần với 0. Nếu ưu tiên cho lọc nội dung thì  $\lambda$  được lấy gần với 1. Nếu  $\lambda = 0$  thì phương pháp dự đoán trở lại đúng mô hình lọc cộng tác dựa trên tất cả đánh giá như đã trình bày trong Mục 3.2.2.4. Nếu  $\lambda = 1$  thì phương pháp dự đoán hoàn

toàn dựa trên nội dung. Nếu lấy  $\lambda = 0.5$  thì mức độ ưu tiên cho lọc cộng tác và lọc nội dung là như nhau.

**Đầu vào:**

- Ma trận biểu diễn các cạnh Người dùng - Nội dung;
- Ma trận  $X^+$ ,  $X^-$  biểu diễn đồ thị  $G^+$ ,  $G^-$ ;
- $L$  là độ dài đường đi trên đồ thị  $G^+$ ,  $G^-$ ;
- $\alpha$ ,  $\lambda$  là các hằng số ( $0 < \alpha, \lambda \leq 1$ );
- $K$  là số sản phẩm cần tư vấn.

**Đầu ra:**

- $K$  sản phẩm có trọng số cao nhất chưa được người dùng đánh giá

**Các bước thực hiện:**

**Bước 1:** Xác định trọng số các đường đi loại 1 là  $Y_1^2$ .

**Bước 2:** Xác định trọng số các đường đi loại 2 là  $X_\alpha^L$ :

- a. Tìm  $(X^+)_\alpha^L$  là trọng số các đường đi trên đồ thị  $G^+$  theo thuật toán được trình bày trong Mục 3.2.2.2.
- b. Tìm  $(X^-)_\alpha^L$  là trọng số các đường đi trên đồ thị  $G^-$  theo thuật toán được trình bày trong Mục 3.2.2.3.
- c. Kết hợp trọng số  $(X^+)_\alpha^L$  và  $(X^-)_\alpha^L$ :

$$X_\alpha^L = (X^+)_\alpha^L + (X^-)_\alpha^L.$$

**Bước 3:** Hợp nhất trọng số của hai loại đường đi theo công thức (3.14)

ta nhận được:  $T = \lambda.X_\alpha^L + (1-\lambda)Y_1^2$

**Bước 4:** Sắp xếp các sản phẩm theo thứ tự giảm dần của trọng số.

**Bước 5:** Chọn  $K$  sản phẩm có trọng số cao nhất tư vấn cho người dùng hiện thời.

**Hình 3.9.** Thuật toán dự đoán trên đồ thị kết hợp

**Mệnh đề 3.4.** Độ phức tạp thuật toán trên đồ thị kết hợp là  $O(L \cdot |U|^{2.376} + (|U|+|P|+|C|)^{2.376})$ . Trong đó,  $|U|$  là số lượng người dùng,  $|P|$  là số lượng sản phẩm,  $|C|$  là số lượng các đặc trưng nội dung.

**Chứng minh.** Để tính toán  $Y_1^2$  tại bước 1 của thuật toán, ta cần thực hiện lấy lũy thừa bậc 2 của ma trận vuông có cấp  $(|U|+|P|+|C|)$ , sau đó chiếu xuống thành phần  $(|U|P)$ . Độ phức tạp tính toán của toàn bộ bước 1 là  $O(|U|+|P|+|C|)^{2.376}$ .

Theo Mệnh đề 3.3, độ phức tạp của bước 2 là  $O(|U|)^{2.376}$ . Như vậy, độ phức tạp tính toán của bước 1 và bước 2 là  $O(L \cdot |U|^{2.376} + (|U|+|P|+|C|)^{2.376})$ .

**Ví dụ minh họa:**

Giả sử ta cần tư vấn  $K=2$  sản phẩm cho người dùng  $u_5$  trong hệ gồm 5 người dùng và 7 sản phẩm với ma trận  $R, X, Y$  được cho trong Bảng 3.6, Bảng 3.7, Bảng 3.8 theo thứ tự. Chọn  $\alpha = 0.5, L=5, \lambda = 0.72$ . Khi đó các bước thực hiện của thuật toán trên đồ thị kết hợp cho lại kết quả như sau:

Tại bước 1 của thuật toán ta tính được trọng số các đường đi loại 1 là:

$$Y_1^2 = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Thực hiện tuần tự (a), (b), (c) trong bước 2 của thuật toán với  $\alpha=0.5, L=5$ , ta có:

Tại bước (2.a) của thuật toán, trọng số các đường đi độ dài  $L=5, \alpha = 0.5$  trên đồ thị  $G^+$  là:

$$(X^+)^s_{0.5} = \begin{bmatrix} 0.15625 & 0.18750 & 0.03125 & 0.28125 & 0.03125 & 0.03125 & 0.03125 \\ 0.12500 & 0.59375 & 0.18750 & 0.34375 & 0.25000 & 0.18750 & 0.25000 \\ 0.03125 & 0.81250 & 0.37500 & 0.21875 & 0.43750 & 0.25000 & 0.56250 \\ 0.00000 & 0.43750 & 0.18750 & 0.06250 & 0.37500 & 0.18750 & 0.37500 \\ 0.03125 & 0.81250 & 0.25000 & 0.21875 & 0.56250 & 0.37500 & 0.43750 \end{bmatrix}$$

Tại bước (2.b) của thuật toán, trọng số các đường đi độ dài L=5,  $\alpha = 0.5$  trên đồ thị  $G$  là:

$$(X^-)_{0.5}^5 = \begin{bmatrix} -0.12500 & -0.00000 & -0.31250 & -0.00000 & -0.12500 & -0.31250 & -0.00000 \\ -0.03125 & -0.00000 & -0.28125 & -0.00000 & -0.15625 & -0.15625 & -0.00000 \\ -0.00000 & -0.00000 & -0.00000 & -0.12500 & -0.00000 & -0.00000 & -0.00000 \\ -0.15625 & -0.00000 & -0.15625 & -0.00000 & -0.03125 & -0.28125 & -0.00000 \\ -0.00000 & -0.00000 & -0.00000 & -0.12500 & -0.00000 & -0.00000 & -0.00000 \end{bmatrix}$$

Tại bước (2.c) của thuật toán, trọng số các đường đi độ dài L=5,  $\alpha = 0.5$  trên tất cả các đánh giá là:

$$X_{0.5}^5 = \begin{bmatrix} +0.03125 & +0.18750 & -0.28125 & +0.28125 & -0.09375 & +0.00000 & +0.03125 \\ +0.09375 & +0.59375 & -0.09375 & +0.34375 & +0.09375 & +0.03125 & +0.25000 \\ +0.03125 & +0.81250 & +0.37500 & +0.09375 & +0.43750 & +0.25000 & +0.56250 \\ +0.15625 & +0.43750 & +0.03125 & +0.06250 & +0.34375 & -0.09375 & +0.37500 \\ +0.03125 & +0.81250 & +0.25000 & +0.09375 & +0.56250 & +0.37500 & +0.43750 \end{bmatrix}$$

Kết hợp trọng số giữa lọc cộng tác và lọc nội dung theo bước 3 của thuật toán với  $\lambda=0.7$ :  $W = \gamma X_{0.5}^L + (1-\gamma)Y_1^2 = 0.7 * X_{0.5}^L + 0.3 * Y_1^2$ .

$$(0.7) * X_{0.5}^5 = \begin{bmatrix} +0.02188 & +0.13125 & -0.19688 & +0.19688 & -0.06563 & +0.00000 & +0.02188 \\ +0.06563 & +0.42563 & -0.06563 & +0.24063 & +0.06563 & +0.02188 & +0.17500 \\ +0.02188 & +0.56875 & +0.26250 & +0.06563 & +0.30625 & +0.17500 & +0.39375 \\ +0.10938 & +0.30625 & +0.02188 & +0.04375 & +0.34063 & -0.06563 & +0.26250 \\ +0.02186 & +0.56875 & +0.17500 & +0.06563 & +0.39375 & +0.26250 & +0.30625 \end{bmatrix}$$

$$(1-\gamma)Y_1^2 = \begin{bmatrix} 0.30000 & 0.30000 & 0.00000 & 0.30000 & 0.30000 & 0.30000 & 0.0000 \\ 0.30000 & 0.30000 & 0.00000 & 0.30000 & 0.30000 & 0.30000 & 0.0000 \\ 0.00000 & 0.00000 & 0.30000 & 0.30000 & 0.30000 & 0.00000 & 0.30000 \\ 0.00000 & 0.00000 & 0.30000 & 0.30000 & 0.30000 & 0.00000 & 0.30000 \\ 0.30000 & 0.30000 & 0.00000 & 0.30000 & 0.30000 & 0.30000 & 0.00000 \end{bmatrix}$$

$$W = \begin{bmatrix} +0.32188 & +0.43125 & -0.19688 & +0.49688 & +0.23437 & +0.30000 & +0.02188 \\ +0.36563 & +0.72563 & -0.06563 & +0.54063 & +0.36563 & +0.32188 & +0.17500 \\ +0.02188 & +0.56875 & +0.56250 & +0.36563 & +0.60625 & +0.17500 & +0.69375 \\ +0.10938 & +0.30625 & +0.32188 & +0.34375 & +0.64063 & +0.23437 & +0.56250 \\ +0.32186 & +0.86875 & +0.17500 & +0.36563 & +0.69375 & +0.56250 & +0.30625 \end{bmatrix}$$

Tại bước 4 của thuật toán ta sắp xếp được:  $p_1, p_7, p_3$ .

Tại bước 5 của thuật toán, chọn  $p_1, p_7$  tư vấn cho người dùng  $u_5$ .

### 3.3.4. Thuật toán lan truyền mạng

Một trong những khó khăn khi thực hiện dự đoán các sản phẩm cho người dùng hiện thời dựa trên đồ thị ở trên là chúng ta phải thực hiện nhiều phép nhân ma trận để tính toán  $(X^+)_\alpha^L$  và  $(X^-)_\alpha^L$ . Thuật toán nhân ma trận tốt nhất hiện nay có độ phức tạp tính toán là  $O(N^{2.376})$  [44], trong đó  $N$  là cấp của ma trận. Điều này khó có thể thực hiện được khi số lượng người dùng và sản phẩm lớn. Chính vì vậy, ta có thể sử dụng thuật toán lan truyền mạng để tránh các phép nhân ma trận có kích cỡ lớn trong các thuật toán trình bày ở trên.

Ý tưởng đầu tiên của thuật toán lan truyền mạng xuất phát từ lĩnh vực tâm lý học được sử dụng rộng rãi trong trí tuệ nhân tạo áp dụng cho các mạng ngữ nghĩa [60, 61, 119]. Hiện nay, thuật toán mạng truyền được Google áp dụng thành công trong các máy tìm kiếm được gọi là Google PageRank. Đặc biệt, thuật toán đóng vai trò quan trọng trong tin sinh học giải quyết các bài toán phân loại Protein [60]. Huang [119] áp dụng hiệu quả thuật toán mạng lan truyền cho lọc cộng tác dựa trên biểu diễn đồ thị hai phía biểu diễn các đánh giá “*thích hợp*”. Thuật toán có thể được mô tả như sau.

Gọi  $u_a$  là người dùng hiện thời cần được phân bổ các sản phẩm. Để đơn giản trong trình bày thuật toán, ta ký hiệu  $N$  là tập các đỉnh  $u_a$  có thể đi qua đến các đỉnh sản phẩm, đỉnh  $n_i \in N$  được xem là một đỉnh người dùng. Gọi  $e_{ij}$  là trọng số liên kết giữa đỉnh  $n_i$  và  $n_j$ . Giá trị  $e_{aj}$  là trọng số liên kết giữa đỉnh người dùng hiện thời  $u_a$  và  $n_j$ .

Đối với các đường đi loại 1, ma trận  $(e_{ij})$  được chuẩn hóa từ ma trận  $X^+$ ,  $X$  như đã được trình bày ở trên. Đối với các đường đi loại 2, ma trận  $(e_{ij})$  được chuẩn hóa từ ma trận biểu diễn đồ thị chỉ có các cạnh nối giữa đỉnh người dùng với đỉnh nội dung. Gọi  $a_i(t)$  là trọng số các đường đi giữa  $u_a$  và đỉnh  $n_i \in N$  khi duyệt các đường đi độ dài  $L$ . Thuật toán hội tụ đến tập đỉnh khi thực hiện đúng  $L$  bước lặp được thể hiện trong Hình 3.10.

**Đầu vào:**

- Ma trận  $(e_{ij})$  được chuẩn hóa từ ma trận tương ứng cho mỗi loại đường đi;
- $\alpha$  là hằng số điều chỉnh trọng số đường đi ( $0 \leq \alpha \leq 1$ );
- $K$  là số sản phẩm cần tư vấn.

**Đầu ra:**

- $K$  sản phẩm có trọng số cao nhất chưa được người dùng đánh giá.

**Các bước thực hiện:**

1. Thiết lập  $a_i(0) = 0$  cho tất cả  $n_i \in N$ ,  $a_d(0) = 1$
2. for  $t = 0, 1, 2, \dots, L$
3.     for  $n_i \in N$  do
4.          $a_i(t) = e_{ai}$  ;
5.     for  $n_j \in N$  do
6.         if  $e_{ij} > 0$  or  $t = L$  then
7.              $a_i(t) \leftarrow a_i(t) + \alpha \cdot e_{ji} \cdot a_j(t-1)$ ;
8.         endfor
9.     endfor
10. endfor
11. return  $(a_i(L))$ : là trọng số các đường đi độ dài  $L$

**Hình 3.10. Thuật toán lan truyền mạng**

Trong thuật toán trên,  $\alpha \in [0, 1]$  là hằng số được dùng để giảm trọng số các đường đi có độ dài lớn. Trong kiểm nghiệm, ta lấy  $\alpha = 1$  để tính toán trọng số các đường đi loại 1 và  $\alpha = 0.5$  cho các đường đi loại 2.

Độ phức tạp thuật toán lan truyền mạng là  $O(N.S)$ , trong đó  $N$  là số lượng người dùng,  $S$  là số lượng trung bình các phần tử khác 0 của  $(e_{ij})$  [2, 80].

### 3.4. THỬ NGHIỆM VÀ KẾT QUẢ

Mô hình đề xuất được thử nghiệm trên bộ dữ liệu MovieLens. Sai số dự đoán được ước lượng thông qua độ chính xác (*precision*), độ nhạy (*recall*) và *F-Measure* theo thủ tục được mô tả dưới đây.

#### 3.4.1. Dữ liệu thử nghiệm

Hầu hết các kết quả nghiên cứu về lọc thông tin trước năm 2004 đều được kiểm nghiệm trên hai tập dữ liệu EachMovie và MovieLens [11, 12, 14, 18, 20, 21, 25, 27, 29, 30, 32, 41, 42, 47, 48, 55, 56, 95, 96]. Hiện nay, hãng HP đã ngừng cung cấp bộ EachMovie vì vậy các kết quả nghiên cứu chủ yếu được kiểm nghiệm trên bộ dữ liệu MovieLens [8, 9, 10, 15, 16, 17, 24, 28, 38, 39, 43, 59, 86, 105]. Trong điều kiện hiện tại, bộ dữ liệu EachMovie được chúng tôi thu thập được không có các đặc trưng nội dung phim nên không thể tiến hành kiểm nghiệm được cho mô hình lọc kết hợp đề xuất. Chính vì vậy, mô hình đề xuất được tiến hành thử nghiệm trên hai tập dữ liệu của bộ dữ liệu MovieLens.

Tập dữ liệu MovieLens thứ nhất (MovieLens1) gồm 1682 người dùng, 942 phim với trên 100000 đánh giá. Tập dữ liệu MovieLens thứ hai (MovieLens2) gồm 6040 người dùng, 3900 phim với trên 1000000 đánh giá như đã được mô tả trong Chương 2 ([www.grouplens.org/node/12](http://www.grouplens.org/node/12)).

Chọn ngẫu nhiên trong tập MovieLens1 500 người dùng làm dữ liệu huấn luyện, chọn ngẫu nhiên trong số còn lại 150 người dùng làm dữ liệu kiểm tra. Chọn ngẫu nhiên trong tập MovieLens2 1000 người dùng làm dữ liệu huấn luyện, chọn ngẫu nhiên trong số còn lại 320 người dùng làm dữ liệu kiểm tra. Hai mức đánh giá cao nhất (4, 5) được biến đổi thành “*thích*” (+1), các mức còn lại biến đổi thành “*không thích*” (-1) [29]. Các đặc trưng nội dung của phim được chọn là thể loại và đạo diễn. Các tập dữ liệu này cũng được Grouplens cung cấp kèm theo các tập dữ liệu tương ứng.



### 3.4.2. Phương pháp thử nghiệm

Phương pháp đánh giá sai số phân loại dựa trên độ chính xác  $P$  (*Precision*) và độ nhạy  $R$  (*Recall*) được Billsus và Basu đề xuất năm 1998 cho các hệ thống lọc văn bản và được xem như phương pháp tiêu chuẩn cho các hệ thống lọc theo nội dung và lọc kết hợp [20, 26]. Phương pháp được tiến hành như sau.

Trước tiên toàn bộ sản phẩm trong tập dữ liệu kiểm tra được chia thành hai lớp: Lớp các sản phẩm phân bổ thích hợp và lớp các sản phẩm phân bổ không thích hợp. Gọi  $N$  là tổng số các đánh giá người dùng trong tập dữ liệu kiểm tra, trong đó  $N_r$  là số các sản phẩm người dùng đã đánh giá thích hợp,  $N_{rs}$  là số các sản phẩm phương pháp lọc dự đoán chính xác, khi đó độ chính xác  $P$  được tính theo công thức (3.15), độ nhạy  $R$  được tính toán theo công thức (3.16), và độ đo  $F$  (*F-Measure*) được tính theo công thức (3.17). Giá trị  $P$ ,  $R$ ,  $F\_Measure$  càng lớn độ, chính xác của phương pháp càng cao.

$$P = \frac{N_{rs}}{N_r} \quad (3.15)$$

$$R = \frac{N_{rs}}{N} \quad (3.16)$$

$$F\_Measure = \frac{2 \times P \times R}{(P + R)} \quad (3.17)$$

### 3.4.3. So sánh và đánh giá dựa vào Precision, Recall và F-measure

Mô hình lọc cộng tác kết hợp với lọc nội dung dựa trên đồ thị (ký hiệu là CombinedGraph). Độ chính xác, độ nhạy và *F-Measure* được tính toán dựa trên danh sách đầu tiên của 10, 20 và 50 sản phẩm dùng để tư vấn. Các giá trị ngưỡng lần lượt được chọn là:  $\gamma = 20$  và  $\alpha = 0.5$ ,  $\lambda = 0.8$ .

Kết quả kiểm nghiệm của mô hình đề xuất được lấy trung bình từ 10 lần kiểm nghiệm ngẫu nhiên cùng với kết quả của các phương pháp:

- Phương pháp lọc cộng tác dựa trên người dùng sử dụng thuật toán *KNN* và độ tương quan Pearson (ký hiệu là UserBased) [52]. Đây là phương pháp lọc cộng tác thông dụng nhất và thường được sử dụng khi so sánh.

- Phương pháp lọc cộng tác trên đồ thị  $G^+$  (Ký hiệu là 3Hop) như đã trình bày trong Mục 3.2.2.2. Đây là một trong những phương pháp có độ chính xác tốt nhất hiện nay.
- Phương pháp lọc theo nội dung (ký hiệu là ContentBased) dựa trên mô hình đồ thị như đã trình bày trong Mục 3.3.3.2.
- Phương pháp lọc kết hợp đơn giản (Ký hiệu là SimpleHybrid) như đã trình bày trong Mục 3.3.3.3.

**Bảng 3.9.** Giá trị Precision, Recall, F-Measure kiểm nghiệm trên tập MovieLens1

Phương pháp	Độ đo	Số sản phẩm dùng để tư vấn		
		10	20	50
UserBased	Độ nhạy	0.001	0.031	0.078
	Độ chính xác	0.003	0.041	0.054
	F-Measure	0.123	0.028	0.054
ContentBased	Độ nhạy	0.018	0.026	0.046
	Độ chính xác	0.038	0.032	0.026
	F-Measure	0.020	0.024	0.028
3Hop	Độ nhạy	0.138	0.207	0.361
	Độ chính xác	0.331	0.286	0.214
	F-Measure	0.152	0.190	0.222
SimpleHybrid	Độ nhạy	0.098	0.144	0.259
	Độ chính xác	0.211	0.174	0.144
	F-Measure	0.105	0.123	0.152
CombinedGraph	Độ nhạy	0.142	0.215	0.366
	Độ chính xác	0.339	0.291	0.215
	F-Measure	0.157	0.195	0.224

**Bảng 3.10.** Giá trị Precision, Recall, F-Measure kiểm nghiệm trên tập MovieLens2

Phương pháp	Độ đo	Số sản phẩm dùng để tư vấn		
		10	20	50
UserBased	Độ nhạy	0.007	0.021	0.069
	Độ chính xác	0.015	0.025	0.034
	F-Measure	0.009	0.023	0.045
ContentBased	Độ nhạy	0.009	0.017	0.037
	Độ chính xác	0.022	0.020	0.018
	F-Measure	0.013	0.018	0.024
3Hop	Độ nhạy	0.155	0.222	0.377
	Độ chính xác	0.284	0.225	0.164
	F-Measure	0.200	0.223	0.228
SimpleHybrid	Độ nhạy	0.117	0.162	0.279
	Độ chính xác	0.186	0.148	0.118
	F-Measure	0.144	0.155	0.166
CombinedGraph	Độ nhạy	0.165	0.234	0.381
	Độ chính xác	0.292	0.240	0.175
	F-Measure	0.211	0.237	0.240

Kết quả kiểm nghiệm cho thấy mô hình đề xuất cho lại kết quả độ chính xác, độ nhạy và F-Measure đều lớn hơn so với các phương pháp còn lại. Điều đó chứng tỏ việc xác định mức độ ưa thích của người dùng đối với những đặc trưng nội dung sản phẩm có ý nghĩa đặc biệt quan trọng để nâng cao chất lượng dự đoán cho các hệ thống tư vấn.

#### 3.4.4. Phân tích kết quả

Để thấy rõ sự nổi trội của mô hình, chúng tôi lấy giá trị trung bình F-Measure sau 10 lần kiểm nghiệm ngẫu nhiên của 150 người dùng trong tập dữ liệu kiểm tra của MovieLens1 và 320 người dùng trong tập dữ liệu kiểm tra của

MovieLens2 để tiến hành một paired t-test [58]. Các tham số thống kê so sánh mức độ nổi trội thống kê giữa CombinedGraph và các phương pháp còn lại bao gồm: Số bậc tự do của paired t-test ( $DF$ ), trung bình độ lệch giữa CombinedGraph và phương pháp so sánh ( $Mean$ ), độ lệch chuẩn ( $SD$ ) giữa CombinedGraph và phương pháp so sánh, lỗi chuẩn trung bình ( $SE$ ) của CombinedGraph và phương pháp so sánh,  $t$  và  $p$  là giá trị nổi trội thống kê ( $SS$ ) của kiểm nghiệm của pair t-test giữa CombinedGraph và phương pháp so sánh. Giá trị  $p < 0.05$  chứng tỏ phương pháp CombinedGraph cho lại giá trị F-Measure lớn hơn phương pháp so sánh ít nhất 5% trên tổng số lần lần quan sát.

Kết quả kiểm nghiệm các tham số thống kê giữa CombinedGraph và các phương pháp ứng với trường hợp sử dụng  $K= 10, 20, 50$  sản phẩm cần tư vấn của tập MovieLens1 được thể hiện trong Bảng 3.11, Bảng 3.12, Bảng 3.13 theo thứ tự. Kết quả kiểm nghiệm các tham số thống kê giữa CombinedGraph và các phương pháp ứng với trường hợp sử dụng  $K= 10, 20, 50$  sản phẩm cần tư vấn của tập MovieLens2 được thể hiện trong Bảng 3.14, Bảng 3.15, Bảng 3.16 theo thứ tự. Giá trị  $p$  ( $p$ -value) tính toán được đều nhỏ hơn 0.05 trong tất cả các trường hợp  $K=10, 20, 50$  trên hai tập dữ liệu. Chính vì vậy, ta có thể khẳng định phương pháp đề xuất thực hiện tốt hơn so với UserBased, ContentBased, 3Hop, và SimpleHybrid.

**Bảng 3.11.** Kết quả kiểm nghiệm paired t-test với  $K=10$  sản phẩm cần tư vấn trên tập MovieLens1

<b>Phương pháp so sánh</b>	<b>DF</b>	<b>Mean</b>	<b>SD</b>	<b>SE</b>	<b>t-value</b>	<b>p-value</b>
UserBased	149	0.034	0.198	0.016	2.104	0.037
ContentBased	149	0.137	0.772	0.063	2.175	0.031
3Hop	149	0.005	0.027	0.002	2.291	0.023
SimpleHybrid	149	0.052	0.266	0.022	2.391	0.018

**Bảng 3.12.** Kết quả kiểm nghiệm paired t-test với K=20 sản phẩm cần tư vấn trên tập MovileLens1

<b>Phương pháp so sánh</b>	<b>DF</b>	<b>Mean</b>	<b>SD</b>	<b>SE</b>	<b>t-value</b>	<b>p-value</b>
UserBased	149	0.167	0.815	0.067	2.511	0.013
ContentBased	149	0.171	0.868	0.071	2.413	0.012
3Hop	149	0.005	0.031	0.003	1.993	0.048
SimpleHybrid	149	0.072	0.366	0.030	2.411	0.018

**Bảng 3.13.** Kết quả kiểm nghiệm paired t-test với K=50 sản phẩm cần tư vấn trên tập MovieLens1

<b>Phương pháp so sánh</b>	<b>DF</b>	<b>Mean</b>	<b>SD</b>	<b>SE</b>	<b>t-value</b>	<b>p-value</b>
UserBased	149	0.170	0.821	0.067	2.536	0.012
ContentBased	149	0.196	0.916	0.075	2.622	0.010
3Hop	149	0.002	0.012	0.001	2.019	0.045
SimpleHybrid	149	0.072	0.381	0.031	2.317	0.022

**Bảng 3.14.** Kết quả kiểm nghiệm paired t-test với K=10 sản phẩm cần tư vấn trên tập MovileLens2

<b>Phương pháp so sánh</b>	<b>DF</b>	<b>Mean</b>	<b>SD</b>	<b>SE</b>	<b>t-value</b>	<b>p-value</b>
UserBased	319	0.202	1.823	0.102	2.078	0.039
ContentBased	319	0.198	1.784	0.100	1.985	0.048
3Hop	319	0.011	0.076	0.004	2.574	0.011
SimpleHybrid	319	0.067	0.534	0.030	2.243	0.026

**Bảng 3.15.** Kết quả kiểm nghiệm paired t-test với K=20 sản phẩm cần tư vấn trên tập MovileLens2

Phương pháp so sánh	DF	Mean	SD	SE	t-value	p-value
UserBased	319	0.214	1.750	0.098	2.188	0.029
ContentBased	319	0.219	1.768	0.099	2.216	0.027
3Hop	319	0.014	0.103	0.006	2.437	0.015
SimpleHybrid	319	0.082	0.654	0.037	2.243	0.027

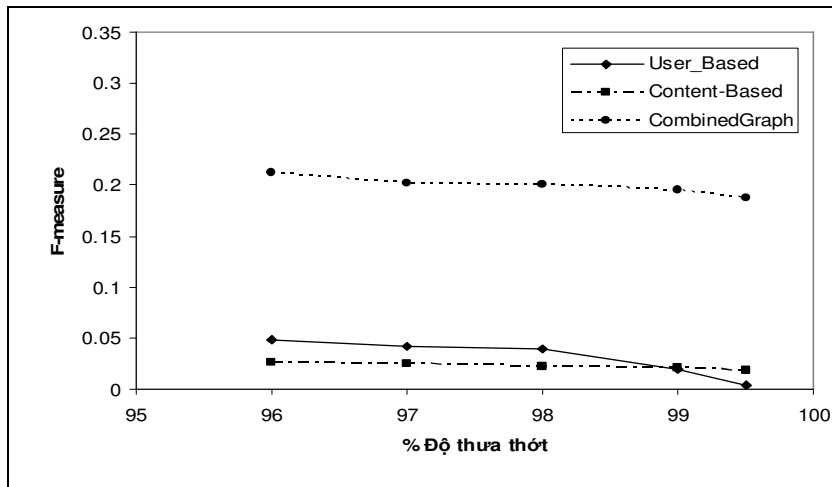
**Bảng 3.16.** Kết quả kiểm nghiệm paired t-test với K=50 sản phẩm cần tư vấn trên tập MovileLens2

Phương pháp so sánh	DF	Mean	SD	SE	t-value	p-value
UserBased	319	0.222	1.567	0.088	2.534	0.012
ContentBased	319	0.216	1.313	0.073	1.983	0.004
3Hop	319	0.012	0.095	0.005	2.251	0.025
SimpleHybrid	319	0.074	0.523	0.029	2.531	0.013

### 3.4.5. Trường hợp dữ liệu thưa

Một trong những khó khăn của các hệ tư vấn là vấn đề ít dữ liệu đánh giá. Các phương pháp User-Based và Content-Based cho lại kết quả tư vấn thấp vì hầu hết liên kết giữa đỉnh người dùng và đỉnh sản phẩm không được thiết lập. Trái lại, trong trường hợp này mô hình đề xuất phát huy hiệu quả bằng cách xem xét các mối liên kết giữa đỉnh người dùng và đỉnh nội dung. Chính vì vậy, mô hình ít bị ảnh hưởng khi dữ liệu đánh giá thưa thớt. Để kiểm tra điều này, ta lấy ngẫu nhiên 400 người dùng làm dữ liệu huấn luyện, 100 người dùng làm dữ liệu kiểm tra trong tập MovieLens1. 25% đánh giá của mỗi người dùng trong tập dữ liệu kiểm tra được ẩn đi để thực hiện dự đoán, sau đó loại bỏ ngẫu nhiên các phần tử của ma trận đánh giá  $R$  để làm tăng mức độ dữ liệu thưa. Giá trị F-

Measure dùng cho 50 sản phẩm đầu tiên để tư vấn của CombinedGraph được thể hiện trong Hình 3.11 ổn định hơn so với User-Based và Content-Based.



Hình 3.11. Giá trị F-Measure ở các mức độ thưa thớt dữ liệu.

### 3.5. KẾT LUẬN

Chương này trình bày một mô hình trực quan, đơn giản và hiệu quả kết hợp giữa lọc cộng tác và lọc nội dung. Mô hình cho phép biểu diễn tất cả các đối tượng tham gia quá trình lọc bằng đồ thị, bao gồm: Người dùng, đánh giá người dùng đối với sản phẩm, sản phẩm và nội dung sản phẩm.

Đối với lọc cộng tác, mô hình biểu diễn tất cả các đánh giá người dùng trên một đồ thị hai phía. Dựa trên biểu diễn này, quá trình phân bổ sản phẩm thích hợp cho mỗi người dùng được thực hiện trên đồ thị chỉ bao gồm các biểu diễn “*thích*”, quá trình lược bỏ thông tin không thích hợp được thực hiện trên đồ thị chỉ bao gồm những biểu diễn “*không thích*”. Phương pháp dự đoán trên tất cả các đánh giá, cho phép ta giảm thiểu các lỗi có thể xảy ra trong quá trình dự đoán và phân bổ thông tin (Một sản phẩm người dùng “*không thích*” có thể có mặt trong danh sách các sản phẩm cần tư vấn. Một sản phẩm người dùng “*thích*” có thể có mặt trong danh sách các sản phẩm cần loại bỏ).

Đối với lọc nội dung, mô hình xây dựng phương pháp trích chọn đặc trưng không dựa vào nội dung sản phẩm như các cách tiếp cận trước đây, mà thực hiện trích chọn đặc trưng nội dung dựa vào đánh giá người dùng. Trên cơ sở ước lượng mức độ quan trọng của các đặc trưng nội dung cho mỗi người dùng, mô hình thiết lập liên kết trực tiếp giữa người dùng với các đặc trưng đó, đồng thời lược bỏ những đặc trưng không quan trọng hoặc không ảnh hưởng đến thói quen sử dụng sản phẩm của mỗi người dùng. Bằng cách làm này, mô hình cá nhân hóa được ảnh hưởng của các đặc trưng nội dung cho mỗi người dùng.

Liên kết giữa người dùng với những đặc trưng nội dung sản phẩm quan trọng đối với người dùng được thiết lập tạo nên mối liên hệ giữa lọc cộng tác và lọc nội dung. Đây cũng là điểm mới khác biệt quan trọng của mô hình đề xuất so với các mô hình trước đây. Trong đó, lọc cộng tác được thực hiện bằng cách lan truyền tủa có trọng số trên các cạnh biểu diễn đánh giá người dùng đối với sản phẩm, lọc nội dung được thực hiện bằng cách lan truyền có trọng số trên các cạnh người dùng và đặc trưng nội dung sản phẩm.

Phương pháp dự đoán được đưa về bài toán tìm kiếm trên đồ thị cho phép ta sử dụng biểu diễn đồ thị bằng ma trận thưa để giảm thiểu không gian biểu diễn dữ liệu, đồng thời có thể sử dụng các thuật toán hiệu quả trên đồ thị. Kết quả kiểm nghiệm trên bộ dữ liệu MovieLens cho thấy, mô hình cho lại kết quả tốt hơn các phương pháp lọc cộng tác dựa trên độ tương quan và lọc theo nội dung thuần túy.



## KẾT LUẬN

Lọc cộng tác và lọc nội dung là hai phương pháp tiếp cận chính được áp dụng cho các hệ thống lọc thông tin. Lọc nội dung thực hiện tốt trên các đối tượng dữ liệu được biểu diễn dưới dạng các đặc trưng nội dung nhưng lại khó thực hiện trên các dạng thông tin đa phương tiện. Lọc cộng tác có thể lọc được mọi loại thông tin nhưng gặp phải khó khăn khi người dùng dữ liệu đánh giá thưa thớt, một người dùng mới chưa có đánh giá nào về sản phẩm, một sản phẩm mới chưa được người dùng nào đánh giá. Dựa vào những nghiên cứu cơ bản này, luận án tập trung giải quyết vào hai vấn đề chính còn tồn tại của lọc thông tin cho các hệ tư vấn, đó là vấn đề dữ liệu thưa của lọc cộng tác và vấn đề kết hợp hiệu quả giữa lọc cộng tác và lọc nội dung.

Đối với *vấn đề dữ liệu thưa của lọc cộng tác*, luận án đề xuất sử dụng phương phân loại bằng kỹ thuật Boosting dựa trên gốc quyết định đã được áp dụng thành công trong nhiều lĩnh vực khác nhau của học máy [3, 81]. Trên cơ sở áp dụng phương pháp Boosting, luận án đề xuất phương pháp MC-Boost hạn chế ảnh hưởng của vấn đề dữ liệu thưa trong lọc cộng tác bằng kỹ thuật học đa nhiệm. Kết quả kiểm nghiệm trên các bộ dữ liệu về phim cho thấy, trong trường hợp dữ liệu tương đối đầy đủ phương pháp Boosting và MC-Boost đều cho lại kết quả tốt hơn so với phương pháp lọc theo độ tương quan Pearson. Trong trường hợp dữ liệu thưa, phương pháp MC-Boost cho lại kết quả tốt hơn so với phương pháp Boosting cho từng bài toán phân loại.

Đối với *vấn đề kết hợp giữa lọc cộng tác và lọc nội dung*, luận án đề xuất một mô hình đồ thị biểu diễn tất cả các đối tượng tham gia hệ thống lọc, bao gồm: Người dùng, đánh giá người dùng, sản phẩm và nội dung sản phẩm [2, 80]. Để phát huy tính hiệu quả của lọc cộng tác, mô hình biểu diễn tất cả các đánh giá người dùng bằng một đồ thị hai phía. Việc biểu diễn quan hệ Người dùng- Sản phẩm như một đồ thị hai phía cho phép ta giảm thiểu không gian biểu

diễn dữ liệu vì ma trận đánh giá  $R$  có rất ít dữ liệu đánh giá. Dựa trên biểu diễn đồ thị này, hệ thống tư vấn có thể được triển khai dễ dàng theo tất cả các khía cạnh: Phân bổ thông tin thích hợp hoặc gỡ bỏ thông tin không thích hợp cho mỗi người dùng.

Để kết hợp hiệu quả giữa lọc cộng tác và lọc nội dung, mô hình xây dựng phương pháp trích chọn đặc trưng nội dung sản phẩm dựa vào đánh giá người dùng. Trên cơ sở trích chọn những đặc trưng nội dung sản phẩm quan trọng cho mỗi người dùng, mô hình thiết lập liên kết giữa người dùng với các đặc trưng đó, đồng thời lược bỏ những đặc trưng không quan trọng hoặc không ảnh hưởng đến thói quen sử dụng sản phẩm của người dùng (Mục 3.3.2). Bằng cách làm này, mô hình cá nhân hóa được ảnh hưởng của các đặc trưng nội dung đối với mỗi người dùng.

Phương pháp dự đoán của mô hình được xem xét như một bài toán tìm kiếm trên đồ thị bằng thuật toán lan truyền mạng. Đóng góp vào kết quả dự đoán cho mỗi loại đường đi (Đường đi thông qua đỉnh nội dung sản phẩm, đường đi thông qua các cạnh đánh giá) được điều chỉnh linh hoạt, mềm dẻo cho từng ứng dụng cụ thể thông qua các hằng số khử nhiễu. Kết quả kiểm nghiệm trên bộ dữ liệu MovieLens cho thấy, mô hình cho lại kết quả tốt hơn các phương pháp lọc cộng tác dựa trên độ tương quan và lọc theo nội dung thuần túy. Đặc biệt, mô hình thực hiện tốt trong trường hợp dữ liệu đánh giá thưa thớt.

Tóm lại, đóng góp chính của luận án đó là:

Thứ nhất, luận án đề xuất sử dụng phương pháp Boosting dựa trên gốc quyết định (GentleBoost) cho lọc cộng tác trong trường hợp có tương đối đầy đủ dữ liệu. Trong trường hợp dữ liệu thưa, luận án đề xuất phương hạn chế vấn đề này bằng pháp học đa nhiệm (MC-Boost).

Thứ hai, luận án đề xuất một phương pháp biểu diễn đơn giản và hiệu quả chung cho lọc cộng tác và lọc nội dung trên mô hình đồ thị. Mô hình cho phép tận dụng hiệu quả các mối liên hệ gián tiếp của lọc cộng tác vào quá trình tư vấn.

Thứ ba, luận án đề xuất một phương pháp trích chọn đặc trưng nội dung dựa vào thói quen sử dụng sản phẩm của người dùng. Tiếp cận theo phương pháp này, mô hình khắc phục được hạn chế trong trích chọn đặc trưng của các phương pháp lọc nội dung.

Cuối cùng, phương pháp lọc kết hợp đề xuất được sử dụng để xây dựng hệ tư vấn lựa chọn phim (được trình bày trong Phụ lục 1). Hệ thống phản ánh đầy đủ các chức năng cơ bản của một hệ thống lọc thông tin, bao gồm thành phần phân tích thông tin, thành phần mô hình người dùng, thành phần học và thành phần lọc. Hệ thống cho lại kết quả tư vấn tốt trên bộ dữ liệu MovieLens gồm 3900 phim và 6040 người dùng.

## DANH MỤC CÁC CÔNG TRÌNH CÔNG BỐ

[1] Nguyen Duy Phuong, Le Quang Thang, Tu Minh Phuong (2008), “A Graph-Based for Combining Collaborative and Content-Based Filtering”, *PRICAI 2008*: 859-869.

[2] Nguyen Duy Phuong, Tu Minh Phuong (2008), “Collaborative Filtering by Multi-Task Learning”, *RIVF 2008*: 227-232.

[3] Nguyễn Duy Phương, Từ Minh Phương (2009), “Lọc cộng tác và lọc theo nội dung dựa trên mô hình đồ thị”, *Chuyên san các công trình nghiên cứu, phát triển và ứng dụng Công nghệ thông tin và Truyền thông*, Tập V-1 số 1, trang: 4-12.

[4] Nguyễn Duy Phương, Từ Minh Phương (2008), “Một thuật toán lọc cộng tác cho trường hợp ít dữ liệu”, *Tạp chí Tin học và Điều khiển học*, tập 24, trang: 62-74.

[5] Nguyễn Duy Phương, Phạm Văn Cường, Từ Minh Phương (2008), “Một số giải pháp lọc thư rác tiếng Việt”, *Chuyên san các công trình nghiên cứu, phát triển và ứng dụng Công nghệ thông tin và Truyền thông*, số 19, trang: 102-112.

### TÀI LIỆU THAM KHẢO (TIẾNG VIỆT):

- [1] Đinh Mạnh Tường (2002), “Trí tuệ nhân tạo”. *Nhà xuất bản KHKT Hà Nội*.
- [2] Nguyễn Duy Phương, Từ Minh Phương (2009), ”Lọc cộng tác và lọc theo nội dung dựa trên mô hình đồ thị”, *Chuyên san các công trình nghiên cứu, phát triển và ứng dụng Công nghệ thông tin và Truyền thông*, Tập V-1 số 1, trang: 4-12.
- [3] Nguyễn Duy Phương, Từ Minh Phương (2008), “Một thuật toán lọc cộng tác cho trường hợp ít dữ liệu”, *Tạp chí Tin học và Điều khiển học*, tập 24, trang: 62-74.
- [4] Nguyễn Duy Phương, Phạm Văn Cường, Từ Minh Phương (2008), “Một số giải pháp lọc thư rác tiếng Việt”, *Chuyên san các công trình nghiên cứu, phát triển và ứng dụng Công nghệ thông tin và Truyền thông*, số 19, trang: 102-112.
- [5] Nguyễn Duy Phương, Lê Quang Thắng, Từ Minh Phương (2008), “Kết hợp lọc cộng tác và lọc theo nội dung sử dụng đồ thị”, *Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông*, trang: 155-166.
- [6] B.N.Lan, L.Đ.Long, L.T. Dũng, P.H. Nguyễn (2005), “Phương pháp Bayesian trong lọc thư rác tiếng Việt”, *Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông*, Hải Phòng, trang : 69-73.

### TÀI LIỆU THAM KHẢO (TIẾNG ANH):

- [7] A. Ansari, S. Essegaiar, R. Kohli (2000), “Internet Recommendations Systems”. *J. Marketing Research*, pp. 363-375.
- [8] A. Gunawardana, C. Meek (2009), “A unified approach to building hybrid recommender systems. *Microsoft Research* , RecSys 2009: 117-124.
- [9] A. Gunawardana, C. Meek (2008), “Tied boltzmann machines for cold start recommendations. *Microsoft Research*”, RecSys 2008: 19-26.
- [10] A Lazanas, N. Karacapilidis (2010), “On the integration of hybrid recommendation techniques into an agent-based transportation transactions management platform”, *International Journal of Information and Decision Sciences 2010*, Vol. 2, No.2 pp. 170 - 187.
- [11] A. Nakamura, N. Abe (1998), “Collaborative Filtering Using Weighted Majority Prediction Algorithms”, *Proc. 15th Int'l Conf. Machine Learning*.
- [12] A. Popescul, L.H. Ungar, D.M. Pennock, and S. Lawrence (2001), “Probabilistic Models for Unified Collaborative and Content-Based

Recommendation in Sparse-Data Environments”, *Proc. 17th Conf. Uncertainty in Artificial Intelligence*.

[13] A. Torralba, K.P. Murphy, and W. T. Freeman (2007), “Sharing Visual Features for Multiclass and Multiview Object Detection”. *IEEE Trans. On Pattern Analysis And Machine Intelligence*, vol. 29, N<sup>o</sup>. 5.

[14] A.I. Schein, A. Popescul, L.H. Ungar, and D.M. Pennock (2002), “Methods and Metrics for Cold-Start Recommendations”. *Proc. 25th Ann. Int’l ACM SIGIR Conf.*

[15] A. Umyarov, Alexander Tuzhilin: Leveraging aggregate ratings for better recommendations. *RecSys 2007*: 161-164.

[16] A. Umyarov, A. Tuzhilin: Improving rating estimation in recommender systems using aggregation- and variance-based hierarchical models. *RecSys 2009*: 37-44.

[17] A. Umyarov, Alexander Tuzhilin: Improving Collaborative Filtering Recommendations Using External Data. *ICDM 2008*: 618-627.

[18] B. Marlin (2003), “Modeling User Rating Profiles for Collaborative Filtering”, *Proc. 17th Ann. Conf. Neural Information Processing Systems (NIPS ’03)*.

[19] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa (2002), “Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization,” *Data Mining and Knowledge Discovery*, vol. 6, N<sup>o</sup>. 1, pp. 61-82.

[20] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl (2000), “Application of Dimensionality Reduction in Recommender Systems—A Case Study”, *Proc. ACM WebKDD Workshop*.

[21] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl (2001), “Item-Based Collaborative Filtering Recommendation Algorithms”, *Proc. 10th Int’l WWW Conf.*

[22] B.N. Miller, I. Albert, S.K. Lam, J.A. Konstan, and J. Riedl (2003), “MovieLens Unplugged: Experiences with an Occasionally Connected Recommender System”, *Proc. Int’l Conf. Intelligent User Interfaces*.

[23] C. Basu, H. Hirsh, and W. Cohen (1998), “Recommendation as Classification: Using Social and Content-Based Information in Recommendation”, *Recommender Systems. Papers from 1998 Workshop*, Technical Report WS-98-08, AAAI Press 1998.

[24] C. Desrosiers, G. Karypis (2008), “Solving the Sparsity Problem: Collaborative Filtering via Indirect Similarities”, *Department of Computer Science and Engineering University of Minnesota (Technical Report)*.

- [25] C. Dellarocas (2003), “The Digitization of Word of Mouth: Promise and Challenges of Online Feedback Mechanisms”, *Management Science*, vol. 49, N<sup>o</sup>. 10, pp. 1407-1424.
- [26] C.C. Peddy and D. Armentrout (2003), “Building Solutions with Microsoft Commerce Server 2002”, *Microsoft Press*.
- [27] C.C. Aggarwal, J.L. Wolf, K.L. Wu, and P.S. Yu (1999), “Horting Hatches an Egg: A New Graph-Theoretic Approach to Collaborative Filtering”, *Proc. Fifth ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining*.
- [28] D. Anand, K. Bharadwaj (2010), “Enhancing Accuracy of Recommender System through Adaptive Similarity Measures Based on Hybrid Features”, *Intelligent Information and Database Systems*, pp: 1-10.
- [29] D. Billsus and M. Pazzani (1998), “Learning Collaborative Information Filters”, *Proc. Int’l Conf. Machine Learning*.
- [30] D. Billsus and M. Pazzani (2000), “User Modeling for Adaptive News Access”, *User Modeling and User-Adapted Interaction*, vol. 10, N<sup>o</sup>. 2-3, pp. 147-180.
- [31] D. DeCoste (2006), “Collaborative prediction using ensembles of maximum margin matrix factorizations,” in *Proceedings of the 23rd International Conference on Machine Learning (ICML ’06)*, pp. 249–256, Pittsburgh, Pa, USA.
- [32] D. Goldberg, D. Nichols, B.M. Oki, D. Terry (1992), “Using Collaborative Filtering to Weave an Information Tapestry,” *Comm.ACM*, vol. 35, N<sup>o</sup>. 12, pp. 61-70.
- [33] D. Nikovski, V. Kulev (2006), “Induction of compact decision trees for personalized recommendation”, in *Proceedings of the ACM Symposium on Applied Computing*, vol. 1, pp. 575–581, Dijon, France.
- [34] D. Pavlov and D. Pennock (2002), “A Maximum Entropy Approach to Collaborative Filtering in Dynamic, Sparse, High-Dimensional Domains”, *Proc. 16th Ann. Conf. Neural Information Processing Systems (NIPS ’02)*.
- [35] G. Adomavicius and A. Tuzhilin (2001), “Multidimensional Recommender Systems: A Data Warehousing Approach”, *Proc. Second Int’l Workshop Electronic Commerce (WELCOM ’01)*.
- [36] G. Adomavicius, A. Tuzhilin (2005), “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*, vol. 17, N<sup>o</sup>. 6, 2005.
- [37] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin (2005), “Incorporating Contextual Information in Recommender Systems Using a Multidimensional Approach”, *ACM Trans. Information Systems*, vol. 23, N<sup>o</sup>. 1.

- [38] G. Adomavicius, A. Tuzhilin, S. Berkovsky, E. William De Luca, A. Said, “Context-awareness in recommender systems: research workshop and movie recommendation challenge. RecSys 2010: 385-386.
- [39] G. Adomavicius, A. Tuzhilin: Context-aware recommender systems. RecSys 2008: 335-336.
- [40] G. Linden, B. Smith, and J. York (2003), “Amazon.com Recommendations: Item-to-Item Collaborative Filtering”, *IEEE Internet Computing*.
- [41] G. Shani, R. Brafman, and D. Heckerman (2002), “An MDP-Based Recommender System”, *Proc. 18th Conf. Uncertainty in Artificial Intelligence*.
- [42] G. Somlo and A. Howe (2001), “Adaptive Lightweight Text Filtering”, *Proc. Fourth Int’l Symp. Intelligent Data Analysis*.
- [43] G. Takács, I. Pilászy, B. Németh, D. Tikk (2008), “Investigation of various matrix factorization methods for large recommender systems”, in *Proceedings of the IEEE International Conference on Data Mining Workshops (ICDM '08)*, pp. 553–562, Pisa, Italy.
- [44] G.H. Golub and C.F. van Loan (2002), “Matrix Computations”. Johns Hopkins University Press, Baltimore, MD, second edition.
- [45] G.-R. Xue, C. Lin, Q. Yang, W. Xi, H.-J. Zeng, Y. Yu, and Z. Chen (2005), “Scalable collaborative filtering using cluster-based smoothing”. *In Proc. of SIGIR*.
- [46] I. Soboroff and C. Nicholas (1999), “Combining Content and Collaboration in Text Filtering” *Proc. Int’l Joint Conf. Artificial Intelligence Workshop: Machine Learning for Information Filtering*.
- [47] J. Balisico, T. Hofmann (2004), “Unifying collaborative and content-based filtering”, *In Proceedings of Int’l. Conf. on Machine learning (ICML-2004)*.
- [48] J. Baxter (2000), “A Model for Inductive Bias Learning”, *J. of Artificial Intelligence Research*.
- [49] P. J. Denning (1982), “Electronic junk”, *Communications of the ACM*, vol 25, pp. 163-165.
- [50] J. Friedman, T. Hastie and R. Tibshirani. *Additive logistic regression: a statistical view of boosting*. The Annals of Statistics, 38(2):337-374, April, 2000.
- [51] J. Li and O.R. Zainane (2004), “Combining Usage, Content, and Structure Data to Improve Web Site Recommendation”, *Proc. Fifth Int’l Conf. Electronic Commerce and Web Technologies (EC-Web ’04)*, pp. 305-315.
- [52] J. S. Breese, D. Heckerman, and C. Kadie (1998), “Empirical analysis of Predictive Algorithms for Collaborative Filtering”, *In Proc. of 14th Conf. on Uncertainty in Artificial Intelligence*, pp. 43-52.



- [53] J.A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordon, and J. Riedl (1997), "GroupLens: Applying Collaborative Filtering to Usenet News", *Comm. ACM*, vol. 40, N<sup>o</sup>. 3, pp. 77-87, 1997.
- [54] J.B. Schafer, J.A. Konstan, and J. Riedl (2001), "E-Commerce Recommendation Applications," *Data Mining and Knowledge Discovery*, vol. 5, pp. 115-153.
- [55] J.L. Herlocker, J.A. Konstan, and J. Riedl (2000), "Explaining Collaborative Filtering Recommendations", *Proc. ACM Conf. Computer Supported Cooperative Work*.
- [56] J.L. Herlocker, J.A. Konstan, L.G. Terveen, and J.T. Riedl (2004), "Evaluating Collaborative Filtering Recommender Systems", *ACM Trans. Information Systems*, vol. 22, N<sup>o</sup>. 1, pp. 5-53.
- [57] J.Wang, A.P de Vries, M.J.T Reinders (2006), "Unifying user-based and item-based collaborative filtering approaches by similarity fusion", *Proc. of SIGIR'06*.
- [58] J. Roderick, A. Little, B. Donald. *Statistical analysis with missing data*. John Wiley & Sons, Inc., 1987.
- [59] J. Wang, A. P. de Vries, M. J. T. Reinders (2008), "Unified relevance models for rating prediction in collaborative filtering," *ACM Transactions on Information Systems*, vol. 26, N<sup>o</sup>. 3, pp. 1-42.
- [60] J.Weston, A. Elisseeff, D. Zhou, C.S Leslie, and W.S.Noble: *Protein ranking: From local to global structure in the protein similarity network*. Proceedings of National Academy of Science Vol. 101(17). pp: 6559-6563. (2004).
- [61] K. Crammer, and Y. Singer (2002), "Pranking with ranking", *Advances in Neural Information Processing Systems*, Vol. 14, pp. 641-647.
- [62] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins (2001), "Eigentaste: A Constant Time Collaborative Filtering Algorithm", *Information Retrieval J.*, vol. 4, N<sup>o</sup>. 2, pp. 133-151.
- [63] K. Yu, A. Schwaighofer, V. Tresp, X. Xu, and H.-P. Kriegel (2004), "Probabilistic Memory-Based Collaborative Filtering", *IEEE Trans. Knowledge and Data Eng.*, vol. 16, N<sup>o</sup>. 1, pp. 56-69.
- [64] K. Yu, X. Xu, J. Tao, M. Ester, and H.-P. Kriegel (2002), "Instance Selection Techniques for Memory-Based Collaborative Filtering", *Proc. Second SIAM Int'l Conf. Data Mining (SDM '02)*.
- [65] L. Getoor and M. Sahami (1999), "Using Probabilistic Relational Models for Collaborative Filtering", *Proc. Workshop Web Usage Analysis and User Profiling (WEBKDD '99)*.

- [66] L. Si and R. Jin (2003), “Flexible Mixture Model for Collaborative Filtering”, *Proc. 20th Int’l Conf. Machine Learning*.
- [67] L. Terveen, W. Hill, B. Amento, D. McDonald, and J. Creter (1997), “PHOAKS: A System for Sharing Recommendations”, *Comm. ACM*, vol. 40, N<sup>o</sup>. 3, pp. 59-62.
- [68] L.H. Ungar and D.P. Foster (1998), “Clustering Methods for Collaborative Filtering”, *Proc. Recommender Systems*, Papers from 1998 Workshop, Technical Report WS-98-08 1998.
- [69] M. Balabanovic and Y. Shoham (1997), “Fab: Content-Based, Collaborative Recommendation”, *Comm. ACM*, vol. 40, N<sup>o</sup>. 3, pp. 66-72.
- [70] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin (1999), “Combining Content-Based and Collaborative Filters in an Online Newspaper”, *Proc. ACM SIGIR ’99 Workshop Recommender Systems: Algorithms and Evaluation*.
- [71] M. Condliff, D. Lewis, D. Madigan, and C. Posse (1999), “Bayesian Mixed-Effects Models for Recommender Systems”, *Proc. ACM SIGIR ’99 Workshop Recommender Systems: Algorithms and Evaluation*.
- [72] M. Deshpande and G. Karypis (2004), “Item-Based Top-N Recommendation Algorithms”, *ACM Trans. Information Systems*, vol. 22, N<sup>o</sup>. 1, pp. 143-177.
- [73] M. Pazzani and D. Billsus (1997), “Learning and Revising User Profiles: The Identification of Interesting Web Sites”, *Machine Learning*, vol. 27, pp. 313-331.
- [74] M. Pazzani (1999), “A Framework for Collaborative, Content-Based, and Demographic Filtering”, *Artificial Intelligence Rev.*, pp. 393-408.
- [75] N.J. Belkin and B. Croft (1992), “Information Filtering and Information Retrieval” *Comm. ACM*, vol. 35, N<sup>o</sup>. 12, pp. 29-37.
- [76] N. Good, J.B. Schafer, J.A. Konstan, A. Borchers, B. Sarwar, J.L. Herlocker, and J. Riedl (1999), “Combining Collaborative Filtering with Personal Agents for Better Recommendations”, *Proc. Conf. Am. Assoc. Artificial Intelligence (AAAI-99)*, pp. 439-446.
- [77] N. Littlestone, M. Warmuth (1994), “The Weighted Majority Algorithm”, *Information and Computation*, vol. 108, N<sup>o</sup>. 2, pp. 212-261.
- [78] N. Ramakrishnan, B.J. Keller, B.J. Mirza, A.Y. Grama, and G.Karypis (2001), “Privacy Risks in Recommender Systems”, *IEEE Internet Computing*, vol. 5, N<sup>o</sup>. 6, pp. 54-62.

- [79] N. Srebro, T. Jaakola (2003), “Weighted low-rank approximations”, *In Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)*.
- [80] Nguyen Duy Phuong, Le Quang Thang, Tu Minh Phuong (2008), “A Graph-Based for Combining Collaborative and Content-Based Filtering”. *PRICAI 2008*: 859-869.
- [81] Nguyen Duy Phuong, Tu Minh Phuong (2008), “Collaborative Filtering by Multi-Task Learning”. *RIVF 2008*: 227-232.
- [82] P. Melville, R.J. Mooney, and R. Nagarajan (2002), “Content-Boosted Collaborative Filtering for Improved Recommendations”, *Proc. 18th Nat’l Conf. Artificial Intelligence*.
- [83] P. Resnick, N. Iakovou, M. Sushak, P. Bergstrom, and J. Riedl (1994), “GroupLens: An Open Architecture for Collaborative Filtering of Netnews”, *In Proceedings of ACM*, pp. 175-186.
- [84] R. Baeza-Yates and B. Ribeiro-Neto (1999), “Modern Information Retrieval”. Addison-Wesley.
- [85] R. Burke (2000), “Knowledge-Based Recommender Systems”, *Encyclopedia of Library and Information Systems*, A. Kent, ed., vol. 69, Supplement 32, Marcel Dekker.
- [86] R. Bell, Y. Koren (2007), “Improved neighborhood-based collaborative filtering”, in *Proceedings of KDD Cup and Workshop*.
- [87] R. Caruana (1997), “Multi-task learning”, *Machine Learning*, 28, pp. 41–75.
- [88] R. Jin, L. Si, and C. Zhai (2003), “Preference-Based Graphic Models for Collaborative Filtering”, *Proc. 19th Conf. Uncertainty in Artificial Intelligence (UAI 2003)*.
- [89] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins (2001), “Recommendation Systems: A Probabilistic Analysis”, *J. Computer and System Sciences*, vol. 63, N<sup>o</sup>. 1, pp. 42-61.
- [90] R. Schaback and H. Wendland (2001), “Characterization and Construction of Radial Basis Functions”, *Multivariate Approximation and Applications*, N. Dyn, D. Leviatan, D. Levin, and A. Pinkus, eds., Cambridge Univ. Press, 2001.
- [91] R. Schapire (2001), “The Boosting Approach to Machine Learning: An Overview”, *Proc. MSRI Workshop Nonlinear Estimation and Classification*.
- [92] R.J. Mooney and L. Roy (1999), “Content-Based Book Recommending Using Learning for Text Categorization”, *Proc. ACM SIGIR ’99 Workshop Recommender Systems: Algorithms and Evaluation*.

- [93] S.E. Middleton, N.R. Shadbolt, and D.C. de Roure (2004), “Ontological User Profiling in Recommender Systems”, *ACM Trans. Information Systems*, vol. 22, N<sup>o</sup>. 1, pp. 54-88.
- [94] S. M. McNee, J. Riedl, J. A. Konstan (2006), “Accurate is not always good: how accuracy metrics have hurt recommender systems,” in *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '06)*.
- [95] T. Hofmann (2003), “Collaborative Filtering via Gaussian Probabilistic Latent Semantic Analysis”, *Proc. 26th Ann. Int’l ACM SIGIR Conf.*
- [96] T. Hofmann (2004), “Latent Semantic Models for Collaborative Filtering”, *ACM Trans. Information Systems*, vol. 22, N<sup>o</sup>. 1, pp. 89-115.
- [97] T. Mitchell (1997), “Machine Learning”, 1<sup>ed</sup>. *McGraw Hill*.
- [98] T. Tran and R. Cohen (2000), “Hybrid Recommender Systems for Electronic Commerce”, *Proc. Knowledge-Based Electronic Markets*, Papers from the AAAI Workshop, Technical Report WS-00-04, AAAI Press.
- [99] U. Hanani, B. Shapira, P. Shoval (2001), “Information Filtering: Overview of Issues, Research and Systems”, *User Modeling and User-Adapted Interaction*, vol 11, N<sup>o</sup>.3, pp.203-209.
- [100] U. Shardanand and P. Maes (1995), “Social Information Filtering: Algorithms for Automating ‘Word of Mouth’”, *Proc. Conf. Human Factors in Computing Systems*.
- [101] W. Wade (2003), “A Grocery Cart that Holds Bread, Butter, and Preferences”, *New York Times*.
- [102] W.W. Cohen, R.E. Schapire, and Y. Singer (1999), “Learning to Order Things”, *J. Artificial Intelligence Research*, vol. 10, pp. 243-270, 1999.
- [103] X. Su and T. M. Khoshgoftaar (2006), “Collaborative filtering for multi-class data using belief nets algorithms”, in *Proceedings of the International Conference on Tools with Artificial Intelligence (ICTAI '06)*, pp. 497–504.
- [104] X. Su, R. Greiner, T. M. Khoshgoftaar, X. Zhu (2007), “Hybrid collaborative filtering algorithms using a mixture of experts” in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI '07)*, pp. 645–649, Silicon Valley, Calif, USA.
- [105] X. Su, T. M. Khoshgoftaar, R. Greiner (2008), “A mixture imputation-boosted collaborative filter”, in *Proceedings of the 21th International Florida Artificial Intelligence Research Society Conference (FLAIRS '08)*, pp. 312–317, Coconut Grove, Fla, USA.
- [106] X. Su, T. M. Khoshgoftaar, X. Zhu, R. Greiner (2008), “Imputation-boosted collaborative filtering using machine learning classifiers,” in

*Proceedings of the 23rd Annual ACM Symposium on Applied Computing (SAC '08)*, pp. 949–950, Ceará Fortaleza, Brazil.

[107] X. Su, T. M. Khoshgoftaar (2009), “A Survey of Collaborative Filtering Techniques”. *Advances in Artificial Intelligence*, vol 2009, pp.1-20.

[108] Y. Koren (2008), “Tutorial on recent progress in collaborative filtering”, in *Proceedings of the the 2nd ACM Conference on Recommender Systems*.

[109] Y. Koren (2008), “Factorization meets the neighborhood: a multifaceted collaborative filtering model” in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '08)*, pp. 426–434, Las Vegas, Nev, USA.

[110] Y. Freund and R. Schapire (1996), “Experiments with a new boosting algorithm”. In *Machine Learning: Proceedings of the Thirteenth International Conference*, pp.148-156.

[111] Y. Freund, R. Iyer, R.E. Schapire, and Y. Singer (1998), “An Efficient Boosting Algorithm for Combining Preferences”, *Proc. 15th Int'l Conf. Machine Learning*.

[112] Y. Zhang and J. Callan (2001), “Maximum Likelihood Estimation for Filtering Thresholds”, *Proc. 24th Ann. Int'l ACM SIGIR Conf.*

[113] Y. Zhang, J. Callan, and T. Minka (2002), “Novelty and Redundancy Detection in Adaptive Filtering”, *Proc. 25th Ann. Int'l ACM SIGIR Conf*, pp. 81-88.

[114] Y.-H. Chien and E.I. George (1999), “A Bayesian Model for Collaborative Filtering”, *Proc. Seventh Int'l Workshop Artificial Intelligence and Statistics*.

[115] Y. Park, A. Tuzhilin: The long tail of recommender systems and how to leverage it. *RecSys 2008*: 11-18.

[116] Z. Huang, D. Zeng, H. Chen (2007), “Analyzing Consumer-product Graphs: Empirical Findings and Applications in Recommender Systems”, *Management Science*, 53(7), 1146-1164.

[117] Z. Huang, D. Zeng, H. Chen (2007), “A Comparison of Collaborative-Filtering Recommendation Algorithms for E-commerce”, *IEEE Intelligent Systems*, 22(5): 68-78.

[118] Z. Huang, D. Zeng (2005), “Why Does Collaborative Filtering Work? Recommendation Model Validation and Selection by Analyzing Random Bipartite Graphs”, *The Fifteenth Annual Workshop on Information Technologies and Systems (WITS 2005)*, Best Paper Nominee.

[119] Z. Huang, H. Chen, D. Zeng (2004), “Applying Associative Retrieval Techniques to Alleviate the Sparsity Problem in Collaborative Filtering”, *ACM Transactions on Information Systems*, vol. 22(1) pp. 116–142

- [120] Z. Huang, W. Chung, H. Chen (2004), “ A Graph Model for E-Commerce Recommender Systems”, *Journal of The American Society for Information and Technology (JASIST)*, 55(3):259–274.
- [121] Z. Huang, W. Chung, T. Ong, , And H. Chen (2002), “A graph-based recommender system for digital library”. *In Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries* (Portland, Ore.). ACM, New York, 65–73.
- [122] Z. Huang (2005), “Graph-based Analysis for E-commerce Recommendation”, PhD Thesis, The University of Arizona (ACM SIGMIS Best Dissertation Award 2005).

## **PHỤ LỤC 1**

### **XÂY DỰNG HỆ THỐNG TƯ VẤN LỰA CHỌN PHIM**

#### **DỰA TRÊN MÔ HÌNH ĐỒ THỊ KẾT HỢP**

Hệ thống tư vấn lựa chọn phim (*Film Recommendation System*) được xây dựng dựa vào mô hình đồ thị kết hợp đã được trình bày trong Chương 3. Hệ thống cho phép người dùng xem phim, tra cứu nội dung phim, đánh giá phim, tìm kiếm nội dung phim, tư vấn phim và một số chức năng cập nhật thông tin về phim và thông tin người dùng. Toàn bộ hệ thống được xây dựng dựa trên công nghệ JSP với giao diện Web thân thiện, đẹp và dễ sử dụng.

#### **1.1. KIẾN TRÚC TỔNG QUÁT CỦA HỆ THỐNG**

Kiến trúc của hệ thống được thiết kế theo mô hình ba tầng: Tầng trình bày, tầng logic và tầng dữ liệu. Ngoài ra, để có thể tương tác giữa tầng trình bày và tầng logic, hệ thống sử dụng khối điều khiển để quản lý các luồng thực thi công việc. Nhiệm vụ chi tiết của mỗi tầng được mô tả trong Hình 1.

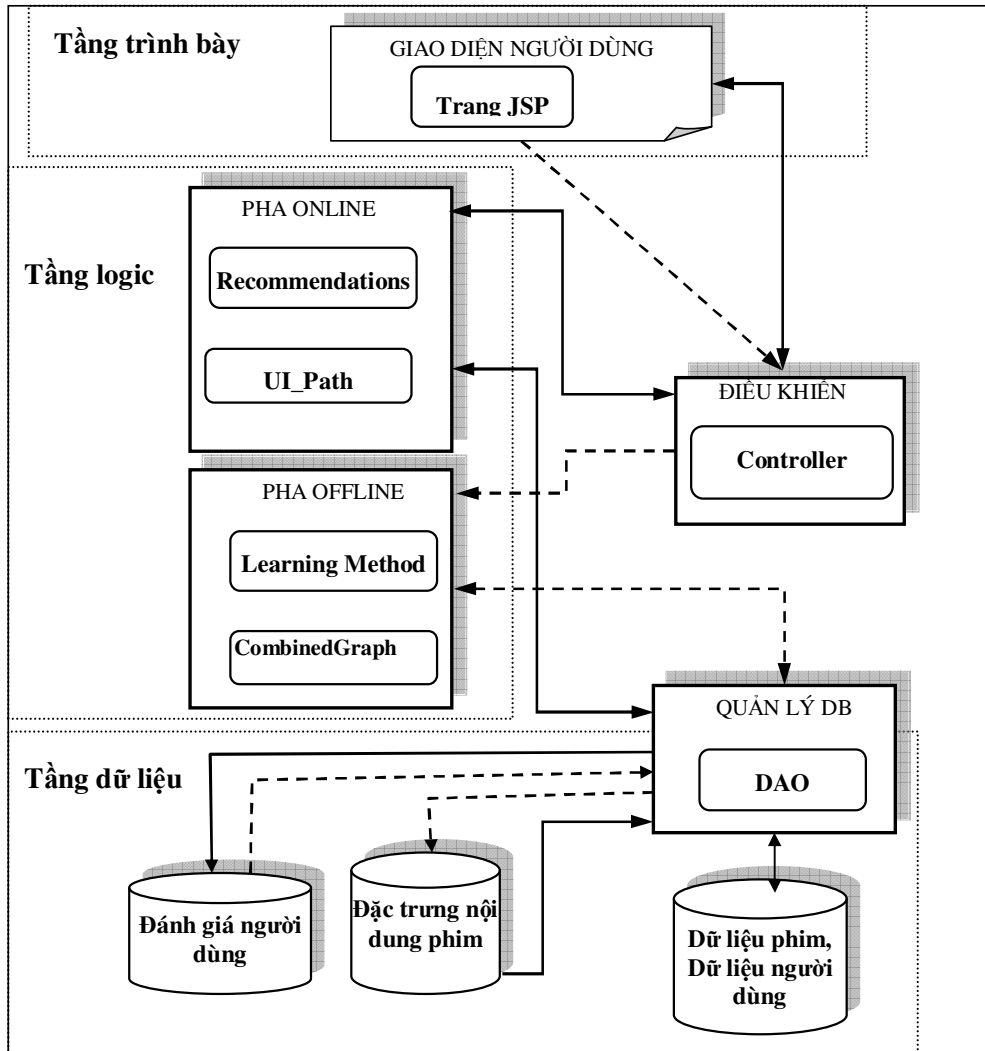
##### **1.1.1. Tầng trình bày**

Là tầng trên cùng có nhiệm vụ giao tiếp trực tiếp với người dùng. Tầng này được xây dựng dưới dạng một website. Nhiệm vụ của tầng là cung cấp giao diện cho người dùng, gửi yêu cầu tư vấn, tìm kiếm, đánh giá cho các tầng bên dưới thông qua khối điều khiển.

##### **1.1.2. Tầng Logic**

Đây là tầng xử lý những công việc quan trọng nhất của hệ thống, bao gồm các nhiệm vụ học, lọc và tạo nên tư vấn cho người dùng. Kết quả các chức năng trong tầng này giao tiếp thông qua khối điều khiển và chuyển đến tầng giao diện để sinh ra tư vấn cho người dùng hiện thời. Khi được kích hoạt, các chức năng sẽ lấy dữ liệu cần thiết, thực hiện tính toán và hiển thị kết quả. Trong tầng Logic, hai chức năng chính được thiết lập đó là chức năng học được thực hiện

offline (“Pha Offline”) và chức năng dự đoán được thực hiện online (“Pha Online”).



**Hình 1.** Kiến trúc hệ thống.

- Chức năng học có nhiệm vụ xây dựng mô hình dựa trên dữ liệu đánh giá phim của người dùng, kết hợp với các đặc trưng nội dung của phim được xây dựng theo mô hình đồ thị đã trình bày trong Chương 3. Trong chức năng này, mô đồ thị kết hợp được xây dựng trong khối “CombinedGraph”, sau đó thực hiện tính toán theo thuật toán mạng lan truyền cho mỗi người dùng để lưu lại kết quả trong khối “Đặc trưng nội dung phim”.



- Chức năng “Tur vản”: Khi có yêu cầu tư vản từ một người dùng được gửi từ khối điều khiển, chức năng sẽ sử dụng dữ liệu được xây dựng từ pha học offline để tìm trọng số đường đi độ dài L cao nhất (UI\_Path) và chọn ra Top-N phim có trọng số cao nhất để tư vản cho khách hàng (Top – N).

### 1.1.3. Tầng dữ liệu

Có nhiệm vụ quản trị cơ sở dữ liệu của hệ thống bao gồm: dữ liệu về người dùng, nội dung phim, đánh giá người dùng cho các phim, các đặc trưng nội dung phim và kết quả tính toán theo mô hình mạng lan truyền nhận được từ chức năng học.

## 1.2. CÁC CHỨC NĂNG CHÍNH CỦA HỆ THỐNG

Hệ thống được thiết kế và thực hiện trên Hệ điều hành Windows XP sử dụng ngôn ngữ lập trình Java trong môi trường PHP. Hệ thống sử dụng tập dữ liệu MovieLens, được thiết kế trên MySql. Toàn bộ hệ thống được thiết kế như một Website với những chức năng chính như sau:

**Đăng nhập:** Trang chủ của hệ thống được định danh bằng một địa chỉ Website, người dùng có thể đăng nhập thông qua địa chỉ Web để truy nhập vào trang chủ của hệ thống. Nếu người dùng đã có tài khoản trong hệ thống, người dùng sẽ nhận được danh sách các phim hệ thống tư vản cho họ. Nếu là người dùng mới đăng ký tài khoản lần đầu vào hệ thống, hệ thống sẽ yêu cầu người dùng đánh giá tối thiểu 20 phim trước khi thực hiện cung cấp dịch vụ tư vản phim. Sau khi đăng nhập, người dùng được phép sử dụng các dịch vụ hệ thống FRM cung cấp, như: Tìm kiếm thông tin về phim, xem danh sách phim, danh sách phim mới, xem phim, tóm tắt nội dung phim, các vai đóng trong phim, những bình luận về phim, đánh giá cho phim, đánh giá nhân vật của tháng, phim của tháng ...

**Tìm kiếm:** Chức năng này cung cấp công cụ tìm kiếm thông tin về phim. Người dùng có thể đưa ra tùy chọn của mình thông qua các lựa chọn: Từ khóa, diễn viên, đạo diễn, thể loại phim.

**Danh sách phim:** Chức năng này cung cấp danh sách các phim theo thể loại, nước sản xuất, hãng phim, công ty phát hành, đạo diễn hoặc diễn viên dưới dạng một tùy chọn do người dùng chỉ định. Người dùng có thể đưa ra lựa chọn của mình như thể loại phim (Hành động, tình cảm, viễn tưởng, hoạt hình...), theo nước sản xuất (Anh, Mỹ, Ca-na-đa, Trung Quốc...) hoặc theo hãng sản xuất, công ty phát hành, diễn viên đóng ... Những thông tin này cung cấp cho người dùng nhằm hỗ trợ họ có lựa chọn chính xác phim để xem, đánh giá và bình phẩm.

**Phim mới:** Chức năng này cung cấp danh sách các phim mới đang được trình chiếu cùng với những giới thiệu tóm tắt về phim, đánh giá về phim do những người dùng khác của hệ thống đã đánh giá, những bình phẩm và bình chọn của khán giả cho phim.

**Xem phim:** Đưa ra màn hình trình chiếu phim Online, người dùng có thể xem, đánh giá, bình phẩm và đánh giá cho phim.

**Tóm tắt nội dung phim:** Cung cấp thông tin nội dung của phim như tóm tắt phim, diễn viên đóng phim, người viết kịch bản, hãng phát hành phim.

**Bình chọn của người dùng:** Hệ thống sẽ tự động đưa ra bình chọn của người dùng cho các phim. Những phim hay của tháng, đạo diễn, diễn viên phim của tháng được hệ thống cung cấp cho người dùng.

**Thông tin điện ảnh:** Cung cấp những thông tin về điện ảnh mới nhất cho người dùng.

**Thông tin bên lề:** Cung cấp những bình luận của người dùng, nhà phê bình, các chuyên gia trong lĩnh vực điện ảnh cho mỗi phim.

**Giới thiệu diễn viên:** Giới thiệu thông tin về các diễn viên trong phim, diễn viên nổi tiếng. Những thông tin này hỗ trợ người dùng trong việc tìm kiếm những phim mà mình ưa thích thông qua diễn viên.

## 1.2.1. Giao diện trang chủ của hệ thống

**Phim! Của tôi**  
Hệ thống tư vấn lựa chọn phim cho bạn

TRANG CHỦ DANH SÁCH PHIM PHIM MỚI HỎI ĐÁP LIÊN HỆ ĐĂNG NHẬP TÌM KIẾM

**Star Trek**  
Được xem như cuộc phiêu lưu vĩ đại nhất mọi thời đại. Xoay quanh một nhóm phi hành gia...

**Đại chiến Robot**  
Trái Đất rơi vào tình trạng hỗn loạn bởi cuộc chiến giữa hai chủng robot tới từ một hành tinh

**Kẻ Hủy Diệt 4**  
Sau khi xóa sổ hầu hết nhân loại trong hàng trăm vụ nổ hạt nhân, Skynet và binh đoàn Hủy

**Người Sói**  
Bộ phim kể về nguồn gốc của Logan - Wolverine. Logan đã cùng người anh em kết

**Phim đang chiếu**

**Buổi đêm ở viện bảo tàng 2**  
Một viên thuốc bí ẩn từ thời AI Cập cổ đại biến các bức tượng và nhân vật trong tranh của bảo tàng Lịch sử tự nhiên Mỹ thành sinh vật sống vào ban đêm. Chúng chồm nhau chí chết sau khi mặt trời lặn, rồi quay trở về tư thế ban đầu khi bình minh tới. Larry Da...

**H2 Halloween 2**  
Tại rạp 09-05-2009  
Giới thiệu: Bệnh viện tâm thần chẳng thể giữ nổi Michael Myers, hắn đã trốn thoát và tìm về quê nhà của mình, thị trấn Haddonfield và đó cũng là lúc những người dân của cái thị trấn nhỏ bé, yên ả này, lần lượt, lần lượt, từng người, từng người một được "diện kiến" một lần và mãi mãi dưới cái lốt của Michael "Halloween" ...

**Vút Bay**  
Tại rạp 09-05-2009  
Walt Disney Pictures và Pixar Animation Studios lại một lần nữa đưa người xem đến thế giới

**Danh sách** **Đánh giá**

- Trang chủ
- Danh sách phim
- Phim mới
- Hỏi đáp
- Liên hệ
- Đăng nhập
- Tìm kiếm

**Phim cuối tuần** **Phim trong tuần**

- Star Trek
- Người nhện
- Bạn gái ác quỷ
- Theo dõi
- 17 lần lặp lại
- Không khí ngày mới
- Người lính
- Quái vật và Alien
- Phòng thủ
- Người sói
- Trận chiến


Hình 2. Giao diện trang chủ của hệ thống

## 1.2.2. Mô tả chi tiết phim

# Phim!Của tôi


Hệ thống tư vấn lựa chọn phim cho bạn

TRANG CHỦ DANH SÁCH PHIM PHIM MỚI HỎI ĐÁP LIÊN HỆ ĐĂNG NHẬP TÌM KIẾM



### Buổi đêm ở viện bảo tàng 2

Xem kết quả: ●●●●○ / 656  
Bình thường ● ● ● ● ● Tuyệt vời **Bỏ phiếu**



Một viên thuốc bí ẩn từ thời Ai Cập cổ đại biến các bức tượng và nhân vật trong tranh của bảo tàng Lịch sử tự nhiên Mỹ t vào ban đêm. Chúng chống nhau chí chết sau khi mặt trời lặn, rồi quay trở về tư thế ban đầu khi bình minh tới.

Larry Daley (Ben Stiller) là anh chàng rất lộn độn trong công việc. Quá chán ngán trước ông chồng hờn dẫu, cô vợ dầm từ đó, tình hình của Larry càng trở nên bi đát hơn. Anh chẳng giữ được việc nào quá một tuần. Không có khả năng tài chính giành được quyền nuôi cậu con trai Nick (Jake Cherry). Không biết sau lần nộp đơn xin việc lần thứ mười mấy, Larry được nhiên nhận vào làm nhân viên gác đêm.

Theo lời khuyên của Cecil, Larry lao vào tìm hiểu nguồn cơn của những sự kiện bất thường để có thể đối phó tốt hơn tro Anh gặp cô sinh viên Rebecca (Carla Gugino) khi cô tới bảo tàng để tìm tư liệu về nhân vật huyền thoại Sacagewea của ng bức bối khi Larry nói rằng anh có thể giới thiệu cô với Sacagewea bằng xương bằng thịt để nói chuyện trực tiếp. Mặc dù vậ thực hiện ý định của mình. Kế hoạch đổ vỡ khi một con khỉ có tên Dexter lấy trộm chùm chìa khóa và để cho một người tiê qua cửa sổ. Hầu quả là sáng hôm sau, người tiền sử đã biến thành đất.

Đạo diễn: Shawn Levy  
Kịch bản: Robert Ben Garant và Thomas Lennon  
Ngày khởi chiếu: 22 May 2009 (Tại Mỹ)

Thể loại: Hành động | Phiêu lưu mạo hiểm | Hải kịch | Gia đình | Giả tưởng Tất cả  
Từ khóa: Bảo tàng | Bảo vệ | 1940s | Thủy thú  
Giải thưởng : 1 trong 3 đề cử.

#### Diễn viên

	Ben Stiller	... Larry Daley
	Amy Adams	... Amelia Earhart
	Owen Wilson	... Jedediah Smith
	Hank Azaria	... Kahmunrah / The Thinker / Abe Lincoln
	Robin Williams	... Teddy Roosevelt
	Christopher Guest	... Ivan the Terrible
	Alain Chabat	... Napoleon Bonaparte
	Steve Coogan	... Octavius
	Ricky Gervais	... Dr. McPhee
	Bill Hader	... General George Armstrong Custer
	Jon Bernthal	... Al Capone
	Patrick Gallagher	... Attila the Hun
	Jake Cherry	... Nicky Daley
	Rami Malek	... Ahkmenrah
	Mizuo Peck	... Sacajawea

#### Một số thông tin khác:

Thời lượng: 105 phút  
Quốc gia: Mỹ | Canada  
Ngôn ngữ: Tiếng anh  
Phim màu: Color  
Tỉ lệ nền: 2.35 : 1  
Hòa âm: Dolby Digital | DTS  
Công ty sản xuất: Twentieth Century-Fox Film Corporation more

Hình 3. Mô tả chi tiết phim

### 1.2.3. Giao diện tìm kiếm thông tin về phim

Hệ thống tư vấn lựa chọn phim cho bạn

TRANG CHỦ DANH SÁCH PHIM PHIM MỚI HỎI ĐÁP LIÊN HỆ ĐĂNG NHẬP TÌM KIẾM

## Tìm kiếm

Tìm theo từ khóa:

Tất cả các từ  Từ bất kỳ  Tìm chính xác cụm từ

Thứ tự:

Chỉ tìm kiếm:  Diễn viên  Đạo diễn  Nội dung  Thể loại  Các thông tin khác  Bài viết

Tìm theo từ khóa **nhân vật**

Tổng cộng có 5 kết quả được tìm thấy.

Hiển thị #

- Leonardo DiCaprio quá béo với cảnh hành động**  
(FAQS/TEMPLATE SETTINGS)  
... với tờ Radar (Anh), Leo đang chịu áp lực nặng nề khi có một cảnh giao đấu quay vào cuối năm nay, và anh phải có vẻ học hác. **Nhân vật** của anh là Jacob Hastley, **nhân viên** văn phòng dính vào một vụ tổng tiền ...
- Buổi đêm ở viện bảo tàng 2**  
(YOU MOVIES/TAI RAP)  
Một viên thuốc bí ẩn từ thời Ai Cập cổ đại kích hoạt các bức tượng và **nhân vật** trong tranh của bảo tàng Lịch sử tự nhiên Mỹ thành sinh **vật** sống vào ban đêm. Chúng choáng nhau chí chết sau khi mặt trời ...
- Kẻ Hủy Diệt 4**  
(YOU MOVIES/TAI RAP)  
... 2018, khi cuộc chiến của **nhân** loại với những cỗ máy hủy diệt do hệ thống máy tính thông minh siêu việt tên Skynet làm chủ. Sau khi âm mưu xóa sổ **nhân** loại trong hàng trăm vụ nổ hạt **nhân**, Skynet và binh ...
- Vút Bay**  
(YOU MOVIES/TAI RAP)  
... nhà thám hiểm tự phong 8 tuổi Russell. Cuộc hành trình đưa họ tới một thế giới đã mất đầy những **nhân vật** quái đản, kỳ lạ và bất ngờ. Một cuộc hành trình vui nhộn, đầy xúc cảm, sẽ đưa trí tưởng tượng của ...
- Thiên Thần & Ác Quỷ**  
(YOU MOVIES/TAI RAP)  
... gái của ông và cũng là một nhà khoa học xinh đẹp của tổ chức CERN để giải mã những ký hiệu để lại trên ngực nạn **nhân**. Những manh mối đã dẫn dắt họ tìm kiếm khắp nơi trong Tòa thánh, để dần phát hiện ra ...

Phim!Của tôi được xây dựng bởi Duy Phương RSS | CSS Valid | XHTML Valid | Top

**Hình 4.** Giao diện tìm kiếm thông tin về phim.

## 1.2.4. Hiện thị phim theo thể loại

# Phim!Của tôi

Hệ thống tư vấn lựa chọn phim cho bạn

TRANG CHỦ
DANH SÁCH PHIM
PHIM MỚI
HỒI ĐÁP
LIÊN HỆ
ĐĂNG NHẬP
TÌM KIẾM

**Phim khoa học**

**Những phim được đánh giá cao.**

Vị trí	Đánh giá trung bình	Tiêu đề phim
1.	4.4	Star Wars: Episode V - The Empire Strikes Back (1980)
2.	4.4	Star Wars (1977)
3.	4.3	The Matrix (1999)
4.	4.3	WALL·E (2008)
5.	4.3	Alien (1979)
6.	4.3	Terminator 2: Judgment Day (1991)
7.	4.2	Aliens (1986)
8.	4.2	District 9 (2009)
9.	4.2	Metropolis (1927)
10.	4.2	The Prestige (2006)
11.	4.2	2001: A Space Odyssey (1968)
12.	4.1	Back to the Future (1985)
13.	4.1	Star Wars: Episode VI - Return of the Jedi (1983)
14.	4.1	Blade Runner (1982)
15.	4.1	Star Trek (2009)
16.	4.0	Donnie Darko (2001)
17.	4.0	Ivan Vasilevich menyayet professiyu (1973)
18.	4.0	The Thing (1982)
19.	4.0	The Terminator (1984)
20.	4.0	V for Vendetta (2005)
21.	4.0	Twelve Monkeys (1995)
22.	4.1	Frankenstein (1931)
23.	4.1	Children of Men (2006)
24.	4.0	Bride of Frankenstein (1935)
25.	4.0	The Day the Earth Stood Still (1951)
26.	4.0	Kaze no tani no Naushika (1984)
27.	4.0	Kin-Dza-Dza (1986)
28.	4.0	Invasion of the Body Snatchers (1956)
29.	4.0	Planet of the Apes (1968)
30.	4.0	Stalker (1979)
31.	4.0	Moon (2009)
32.	4.0	The Man from Earth (2007)
33.	4.0	Tanin no kao (1966)
34.	4.0	Night of the Living Dead (1968)
35.	4.0	Brazil (1985)
36.	4.0	Young Frankenstein (1974)
37.	4.0	Dawn of the Dead (1978)
38.	4.0	Solyaris (1972)
39.	3.9	The Truman Show (1998)
40.	3.9	Toki o kakeru shōjo (2006)
41.	3.9	Evangelion shin gekijōban: Jo (2007)
42.	3.9	Iron Man (2008)
43.	3.9	Seksmisja (1984)
44.	3.9	Batoru rowaiaru (2000)
45.	3.9	E.T.: The Extra-Terrestrial (1982)
46.	3.9	Serenity (2005)
47.	3.9	Grindhouse (2007)
48.	3.9	The Iron Giant (1999)
49.	3.8	Kōkaku kiddōtai: Stand Alone Complex Solid State Society (2006)
50.	3.8	Shin seiki Evangelion Gekijō-ban: Air/Magokoro wo, kimi ni (1997)

**Những phim tồi nhất**

Vị trí	Đánh giá trung bình	Tiêu đề phim
1.	1.2	The Last Gateway (2007)
2.	1.3	Ultra Warrior (1990)
3.	1.3	Terror at Tate Manor (2002)
4.	1.4	Bigfoot (1970)
5.	1.4	Universal Soldiers (2007)
6.	1.4	Evil Behind You (2006)
7.	1.4	Zaat (1975)
8.	1.5	Monster a-Go Go (1965)
9.	1.5	Flesh Feast (1970)
10.	1.5	Awaken the Dead (2007)

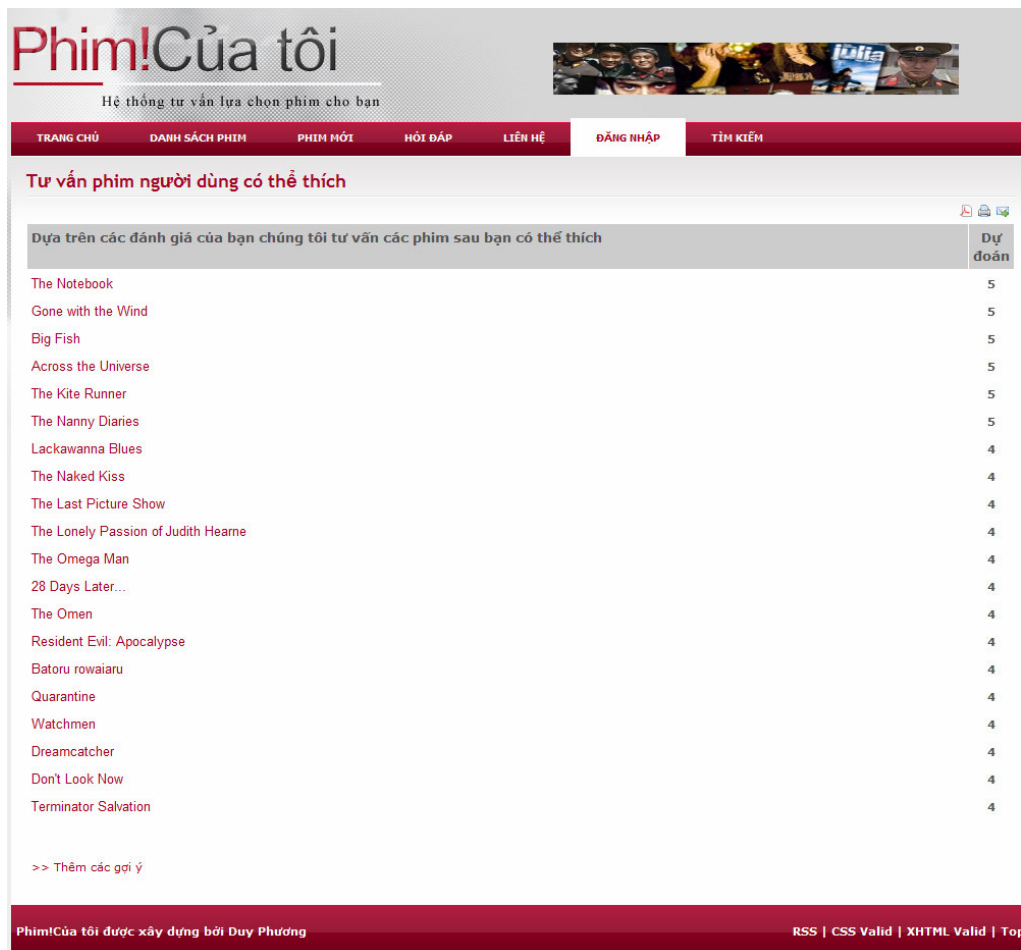
Việc đưa ra thống kê đánh giá cho từng phim được thực hiện như sau:  
Đánh giá = Tổng số các đánh giá / Số đánh giá

>> Thêm thông tin về các phim khác

PhimCủa tôi được xây dựng bởi Duy Phương
RSS | CSS Valid | XHTML Valid | Top

Hình 5. Danh sách phim theo thể loại.

## 1.2.5. Tư vấn phim cho người dùng



The screenshot shows the 'Phim!Của tôi' website interface. At the top, there is a navigation bar with links: TRANG CHỦ, DANH SÁCH PHIM, PHIM MỚI, HỎI ĐÁP, LIÊN HỆ, ĐĂNG NHẬP, and TÌM KIẾM. Below the navigation bar, the main content area is titled 'Tư vấn phim người dùng có thể thích'. Underneath this title, there is a sub-header: 'Dựa trên các đánh giá của bạn chúng tôi tư vấn các phim sau bạn có thể thích'. To the right of this sub-header is a 'Dự đoán' (Prediction) button. The main content is a table listing movies and their predicted ratings:

Movie Title	Dự đoán
The Notebook	5
Gone with the Wind	5
Big Fish	5
Across the Universe	5
The Kite Runner	5
The Nanny Diaries	5
Lackawanna Blues	4
The Naked Kiss	4
The Last Picture Show	4
The Lonely Passion of Judith Hearne	4
The Omega Man	4
28 Days Later...	4
The Omen	4
Resident Evil: Apocalypse	4
Batoru rowaiaru	4
Quarantine	4
Watchmen	4
Dreamcatcher	4
Don't Look Now	4
Terminator Salvation	4

At the bottom of the table, there is a link: '>> Thêm các gợi ý'. The footer of the page contains the text: 'Phim!Của tôi được xây dựng bởi Duy Phương' and 'RSS | CSS Valid | XHTML Valid | Top'.

*Hình 6. Kết quả tư vấn cho mỗi người dùng sau khi đăng nhập.*

## 1.3. KẾT LUẬN

Hệ thống tư vấn lựa chọn phim được xây dựng dựa vào mô hình đồ thị kết hợp đề xuất đã mô tả đầy đủ các chức năng chính của một hệ thống lọc thông tin, bao gồm: chức năng học, chức năng lọc, chức năng phân tích dữ liệu, chức năng người dùng. Ứng dụng cho lại kết quả tư vấn tốt ngay cả trong trường hợp dữ liệu đánh giá người dùng thưa thớt.