A Hybrid Movie Recommendation System

Võ Kiều Hoa[1],Nguyễn Thị Thắm[2]

KHDL2018 - Trường đại học Công Nghệ Thông Tin - ĐHQG TP.HCM
Email: {18520767[1],18521384[2] }@gm.uit.edu.vn

Abstract. Hệ khuyến nghị là hệ thống phân tích khối dữ liệu người dùng và đưa ra dự đoán, gợi ý đề xuất được cho là phù hợp với sở thích của người dùng tại thời điểm bất kỳ trên các ứng dụng và nền tảng trực tuyến giúp tiết kiệm thời gian tiềm kiếm, truy cập nội dung dễ dàng đồng thời giúp nâng cao trải nghiệm khách hàng. Trong đồ án này, chúng tôi thực hiện xây dựng hệ khuyến nghị nhằm gợi ý xếp hạng cho người dùng đối với những bộ phim nhất định và gợi ý các bộ phim cho người dùng. Chúng tôi sử dụng tập dữ liệu MovieLens phiên bản 100,000 lượt đánh giá với ba phương pháp tiếp cận là lọc dựa trên nội dung - content-based filtering (CBF), lọc dựa trên cộng tác -collaborative filtering (CF) và phương pháp kết hợp - hybrid.

Keywords: Recommendation system

1 Introduction

Hệ thống khuyến nghị (Recommender Systems) được ứng dụng rất nhiều trong phân tích dữ liệu người dùng và dự đoán, gợi ý, đưa ra nội dung đề xuất cho người dùng. Hiện nay hệ khuyến nghị đang được ứng dụng trong rất nhiều lĩnh vực khác nhau giúp hỗ trợ quá trình ra quyết định cho người dùng, ví dụ như mua hàng hóa nào, nghe nhạc gì hoặc đọc tin tức gì. Một số hệ thống khuyến nghị tiêu biểu phải kể đến là hệ thống của Amazon, Netflix và Youtube. Ở Việt nam ngày nay hệ khuyến nghị cũng được áp dụng nhiều vào lĩnh vực thương mại điện tử như Shopee, Tiki, Lazada. Như vậy, hệ thống khuyến nghị hỗ trợ người dùng ra quyết định và gợi ý sản phẩm đến người dùng, giúp tiết kiệm thời gian, tăng tốc độ tìm kiếm và giúp người dùng truy cập tới nội dung họ quan tâm một cách dễ dàng hơn. Chính vì tính ứng dụng cao nên hệ khuyến nghị ngày càng được sử dụng rộng rãi, không những thế, các công trình nghiên cứu về hệ khuyến nghị cũng ngày một tăng lên nhằm cải thiện hiệu suất hệ thống.

Đồ án chúng tôi thực hiện hai bài toán là dự đoán xếp hạng của người dùng đối với bộ phim và gợi ý phim cho người dùng. Ở bài toán dự đoán xếp hạng phim chúng tôi dự đoán xếp hạng cho mỗi người dùng với mỗi bộ phim mới(phim mà người dùng chưa đánh giá). Đầu vào là movieId(ID của bộ phim) và userId(ID của người dùng), đầu ra là rating(đánh giá của người dùng). Hình 1 minh họa bài toán. Đối với bài toán gợi ý phim cho người dùng đầu vào là ID của người dùng(userId), đầu ra của hệ thống gợi ý thông tin những bộ phim mới(movieId, title, genres) cho người dùng đó. Hình 2 minh họa bài toán. Chúng tôi sử dụng bộ dữ liệu MovieLens phiên bản 100,000 (tháng 09 năm 2018) cùng với ba phương pháp thử nghiệm là content-based filtering (CBF), lọc dựa trên cộng tác -collaborativefiltering (CF) và phương pháp kết hợp - hybrid.

Báo cáo này bao gồm các nội dung như sau chúng tôi giới thiệu dữ liệu sử dụng ở mục 2, tiếp đến là phương pháp tiếp cận ở mục 3, kết quả thực nghiệm và đánh giá mục 4 và cuối cùng là kết luận và hướng phát triển ở mục 5.

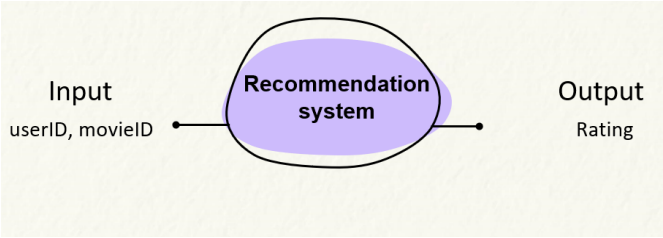

Fig. 1. Bài toán gợi ý đánh giá phim (Rating recommendation).

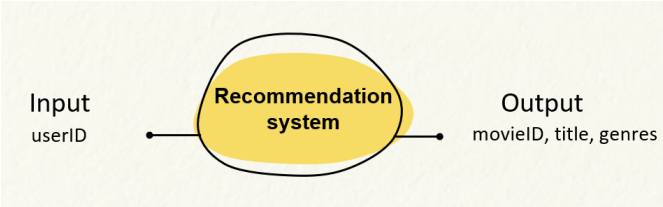

Fig. 2. Bài toán gợi ý phim (Movie recommendation).

2 Dữ liệu

Chúng tôi sử dụng bộ dữ liệu MovieLens từ GroupLens. Hiện nay có rất nhiều bộ dữ liệu có sẵn dùng cho nghiên cứu hệ khuyến nghị. Trong số đó, bộ dữ liệu MovieLens là một trong những bộ dữ liệu phổ biến nhất. Bộ dữ liệu MovieLens ở GroupLen có nhiều phiên bản khác nhau. Ở đây, chúng tôi sử dụng phiên bản MovieLens 100K (tháng 09 năm 2018).
Bộ dữ liệu gồm 100836 lượt xếp hạng(rating) của 610 người dùng(user) và 3683 lượt gắn thẻ(tag) trên 9742 bộ phim(movie). Sau khi phân tích, thăm dò dữ liệu chúng tôi chia thành 2 tập train, test với tỉ lệ 9:1.

Hình bên dưới mô phỏng phân phối xếp hạng của người dùng theo bậc từ 0.5 đến 5.0. Ta thấy dữ liệu phân bố ở các mức rating không đồng đều, tập trung nhiều nhất ở 3.0, 4.0 và 5.0.

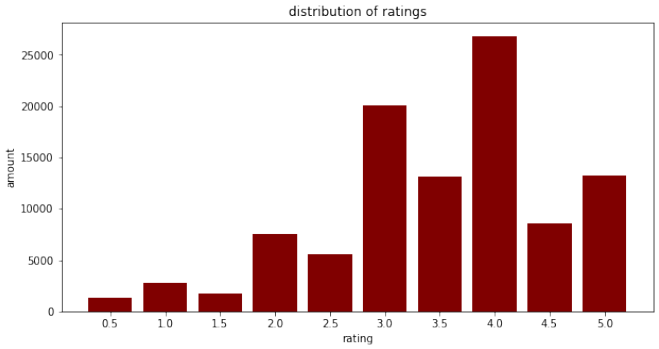

Fig. 3. Phân phối dữ liệu rating

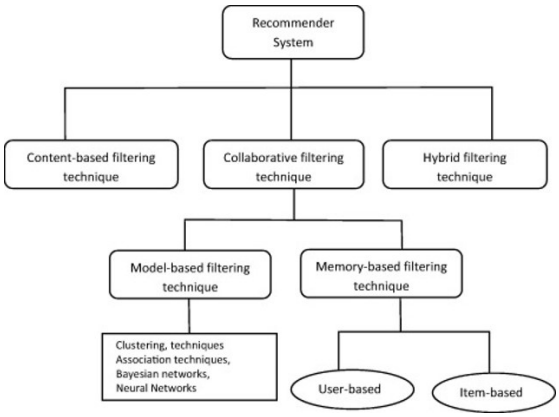

Fig. 4. Tổng quan phương pháp cho hệ khuyến nghị

3 Hướng tiếp cận

3.1 Content-based Filtering

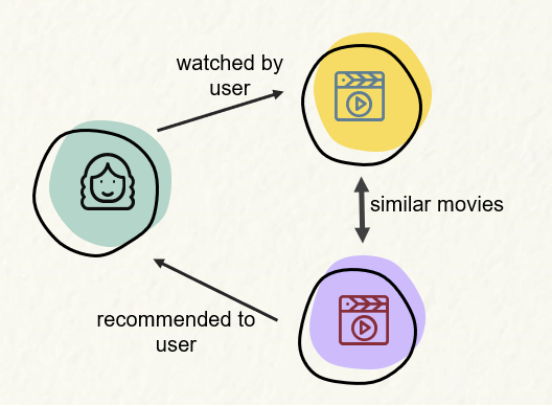

Fig. 5. Minh họa lọc dựa trên nội dung.

Content-based-Filtering dựa trên những thông tin có trong hồ sơ của người dùng hoặc dựa vào nội dung/thuộc tính của những bộ phim tương tự như những bộ phim mà người dùng đã lựa chọn trong quá khứ để đưa ra những gợi ý, đề xuất. Chúng tôi sử dụng hai thông tin có trong lịch sử xem của người dùng để thực hiện phương pháp này. Thứ nhất là thể loại(genres) của những bộ phim đã được đánh giá bởi mỗi người dùng, thứ hai là các thẻ(tag) đã được gán bởi mỗi người dùng cho các bộ phim.

Ví dụ một người được đánh giá là thích phim hành động, tâm lý(từng xem rất nhiều phim hành động, tâm lý), hệ thống sẽ đề xuất phim cùng thể loại cho người đó. Cách tiếp cận này yêu cầu việc sắp xếp các phim vào từng nhóm hoặc phân tích các đặc trưng của từng phim. Độ tương đồng giữa các bộ phim được tính bởi công thức cosine như sau, trong đó A và B là hai vector đặc trưng trích xuất từ tags của hai bộ phim cần tính độ tương đồng:

$$similarity = \cos(\theta) = \frac{A * B}{|A| * |B|}$$

3.2 Collaborative Filtering

Collaborative Filtering (Lọc cộng tác) là phương pháp phân tích dữ liệu người dùng để tìm ra mối tương quan giữa các đối tượng người dùng. Cụ thể ý tưởng của phương pháp lọc cộng tác trong bài toán này là dự đoán mức độ yêu thích của một người dùng đối với một bộ phim dựa trên các người dùng khác có đặc điểm gần giống với người dùng đang xét. Việc xác định độ "giống nhau" giữa các người dùng có thể dựa vào đánh giá của các người dùng này với các bộ phim đã xem và đánh giá trước đó mà hệ thống đã biết trong quá khứ.

Ví dụ hai người H và T đều thích phim hoạt hình (tức là cả hai đều đánh giá từ 4 đến 5 sao). Dựa vào lịch sử xem đã có trong hệ thống, ta thấy T từng đánh giá 5 sao cho bộ phim "Doraemon", nhiều khả năng H cũng thích phim này, từ đó hệ thống sẽ đề xuất "Doraemon" cho H.

Lọc cộng tác có hai hướng tiếp cận chính là memory-based và model-based. Trong đồ án này chúng tôi thực hiện trên cả hai hướng trên. Ở memory-based có user-based và item-based được minh họa ở hình 6 (bên trái là user-based, bên phải là item-based). Cụ thể hơn, thứ nhất user-based là phương pháp xác định mức độ quan tâm của mỗi người dùng tới một bộ phim dựa trên mức độ quan tâm của người dùng tương tự. Thứ hai là thay vì xác định người dùng tương tự, hệ thống sẽ xác định các bộ phim tương tự. Từ đó hệ thống gợi ý những bộ phim gần giống với những bộ phim mà người dùng có mức độ quan tâm cao.

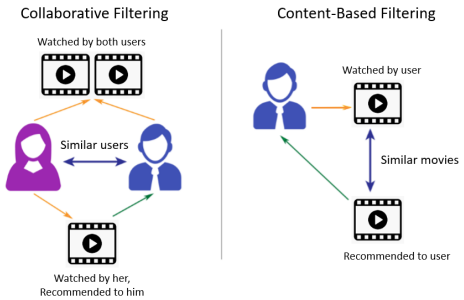

Fig. 6. Minh họa user-based và item-based

Độ tương đồng của hai bộ phim hay hai người dùng được tính bằng công thức cosine như sau, với A, B lần lượt là hai vector đặc trưng của hai bộ phim hay hai người dùng cần tính độ tương đồng:

$$similarity = \cos(\theta) = \frac{A * B}{|A| * |B|}$$

3.3 Hybrid recommendation

Trong bài toán cho hệ khuyến nghị, lọc cộng tác là phương pháp được lự chọn hàng đầu do thế mạnh là chúng hoàn toàn độc lập với sự biểu diễn của các đối tượng đang được gợi ý, tuy nhiên chúng còn tồn tại hạn chế là 'cold start' đối với những người dùng mới hay nói cách khác lọc cộng tác yêu cầu một lượng lớn dữ liệu hiện có của người dùng để đưa ra các đề xuất chính xác, nhưng người dùng mới còn ít dữ liệu nên hạn chế về độ chính xác. Trong khi đó lọc dựa theo nội dung, có thể hiểu được người dùng dựa trên những thông tin đã tồn tại của họ. Dẫn đến cần giải pháp để khắc phục điều này là phương pháp kết hợp(hybrid).

Phương pháp hybrid hay còn gọi là phương pháp lai là phương pháp kết hợp mô cả mô

hình lọc theo nội dung và mô hình lọc cộng tác để giảm thiểu những hạn chế còn tồn tại ở mỗi phương pháp, từ đó đưa ra được những gợi ý tối ưu nhất cho người dùng.

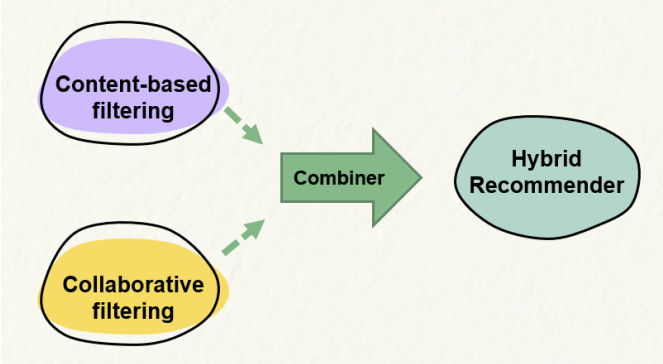

Fig. 7. Hệ thống gợi ý kết hợp giữa lọc theo nội dung và lọc cộng tác.

4 Thực nghiệm

4.1 Content-based Filtering

Đầu tiên, với lọc dựa trên nội dung của thể loại phim, chúng tôi sử dụng tệp dữ liệu movies.csv: chứa thông tin của bộ phim bao gồm các thuộc tính: id phim(movieId), tên phim(title), thể loại(genres). Ở đây chúng tôi chỉ sử dụng hai thuộc tính movieId, genres. Một bộ phim có thể có nhiều thể loại được ngăn cách bởi "|". Chúng tôi kết hợp với dữ liệu ở tệp train.csv(được chia từ ratings.csv), chỉ sử dụng những thuộc tính rating, movieId, userId. Sau đó, tôi mở rộng phim theo thể loại và gom nhóm các bộ phim(movieId) theo thể loại(genres) cho mỗi người dùng(user) và tính trung bình rating mỗi nhóm. Bảng 1, 2 lần lượt minh họa dữ liệu trước và sau khi xử lý.

	userId	genres_explode	ratings_avg
0	471	Fantasy	4.625000
1	471	Animation	3.928571
2	471	Thriller	3.562500
3	471	Romance	3.916667
4	471	Drama	4.050000
5	471	Adventure	3.650000
6	471	Musical	3.500000
7	471	Crime	4.000000

Table 1. Mở rộng dữ liệu theo thể loại

Với lọc dựa trên loại nội dung các thẻ(tags), sử dụng tập dữ liệu tags.csv gom tất cả các tag theo movieId, các tag sẽ được tiền xử lý: token, loại bỏ stopword, sử dụng IDF và Word2Vec để tạo vecto đặc trưng (bảng 3). Sau đó tính độ tương đồng của các vector này bằng công thức

	userId	movieId	genres	rating_predict
0	471	7147	Drama|Fantasy|Romance	4.197222
1	471	6539	Action|Adventure|Comedy|Fantasy	3.993750
2	471	68157	Action|Drama|War	3.875000

Table 2. Gom nhóm phim theo thể loại cho mỗi người dùng

movieId	tag	word_vec
471	hula hoop	[0.0,0.0,0.0,0.0,0.0,...
1088	music	[0.00130013225134...
1580	aliens	[0.0045067868219...
1645	lawyers	[-7.2882772656157...
1959	Africa	[0.0,0.0,0.0,0.0,0.0,...
2122	Stephen King	[5.62643224839121...

Table 3. Chuyển tag thành vector

	movieId	score	title
0	2145	1.0	Pretty in Pink (1986)
1	3071	1.0	Stand and Deliver (1988)
2	1380	1.0	Grease (1978)
3	126548	1.0	The DUFF (2015)
4	2410	1.0	Rocky III (1982)
5	3210	1.0	Fast Times at Ridgemont High (1982)
6	30707	1.0	Million Dollar Baby (2004)

Table 4. Tính độ tương đồng

cosine(mục 4.2) ở bảng 4 từ đó có thể đề xuất những bộ phim tương đồng nhất với những bộ phim mà người dùng quan tâm.

Cuối cùng, sau khi tính được trung bình rating cho từng nhóm phim của từng người dùng bằng 2 phương pháp trên. Ta tiến hành dự đoán rating cho tập test.

4.2 Collaborative Filtering

Hướng tiếp cận thứ hai trong đồ án này là lọc cộng tác. Ở lọc cộng tác, chúng tôi sử dụng cả phương pháp model-based và memory-based(user-based và item-based).

item-based (Memory-based) Đối với phương pháp thứ nhất - dựa trên các bộ phim tương đồng. Chúng tôi kết hợp dữ liệu đánh giá(ratings), phim(movies) và gắn thẻ(tags). Chuyển giá trị thuộc tính thể loại(genres) thành chuỗi những thể loại của mỗi bộ phim, cách nhau bằng khoảng trắng. Kế tiếp, chúng tôi gom nhóm các thẻ(tag) cho mỗi bộ phim và chuyển thành chuỗi gồm các thẻ cách nhau bởi khoảng trắng(bảng 4.2). Sau đó, chúng tôi kết hợp thẻ(tag) và thể loại(genres) lại với nhau thành thuộc tính document(bảng 4.2). Tiếp theo, chúng tôi sử dụng TfidfVectorizer để chuyển chuỗi kết hợp vừa tạo thành Tfidf matrix (9742x9742 tương ứng 9742 bộ phim), sau đó tính độ tương đồng cosine similarity với ma trận vừa tạo, bảng 7 minh họa độ tương đồng của một số phim.

Cuối cùng, sau khi có ma trận cosine similarity chúng ta sẽ dự đoán rating của những bộ

phim ở tập test như sau:
Dự đoán rating của user u0 cho movie m1:

- Tìm danh sách những phim mà user u0 đã từng xem.
- Tính độ tương đồng của phim m1 với các các phim đã tìm được ở danh sách trên.
- Chọn ra top 10 phim có độ tương đồng cao nhất với i1 và >=0.8.
- Dự đoán rating của user u0 cho movie m1 bằng cách lấy trung bình rating top 10 phim đã chọn ra ở trên.

movieId	list_tag	length_tag
29	Native Americans American Indians	33
1202	will ferrell funny will ferrell funny comedy well ferrell Highly quotable funny	79
2030	Mark Wahlberg Leonardo DiCaprio heroin addiction	48
2944	soundtrack sequel Poor plot deverlopment Horrid characterisation action choreography	83
1951	visually stunnig anime animation	33
1217	touching sweet sad quirky poignant philosophical mental illness friendship dark comedy...	138
287	suspence psychology gothic drama disturbing Hannibal Lector	59

Table 5. Dữ liệu sau khi biến đổi tag

document
{beat poetry}Comedy Romance Thriller
{Native Americans American Indians}Adventure Drama Western
{}Comedy
{fairy tales}Adventure Animation Children Comedy Fantasy Romance
{hit men}Action Crime Romace Thriller
{}Drama

Table 6. Thêm thuộc tính document - kết hợp giữa tag và genres

	0	1	2	3	4	5
0	1.000000	.000000	0.144490	0.058773	0.036829	0.000000
1	0.000000	1.000000	.000000	0.031937	0.000000	0.000000
2	0.144490	0.000000	1.000000	0.132378	0.254888	0.000000
3	0.058773	0.031937	0.132378	1.000000	0.263657	0.357168
4	0.036829	0.000000	0.254888	0.263657	1.000000	0.335199
5	0.000000	0.000000	0.000000	0.357168	0.335199	1.000000

Table 7. Minh họa độ tương đồng của một số phim

user-based(Memory-based) Đối với phương pháp thứ hai - dựa trên các người dùng tương tự. Chúng tôi khởi tạo ma trận dữ liệu dựa trên 3 thành phần: người dùng(user), phim(movie), đánh giá(rating). Ma trận dữ liệu sau khi tạo thành có kích thước là 160x9742(tương ứng với 160 người dùng và 9742 bộ phim). Ma trận tạo thành được kết hợp bởi 2 tập train, test (rating

trong tập test đã được thay thế bởi giá trị rỗng)(hình 8 minh họa một vài giá trị trong ma trận). Ma trận này có rất nhiều giá trị khuyết (các ô có dấu ?).

Nhiệm vụ của hệ thống là dựa vào các ô có dữ liệu trong ma trận (dữ liệu từ quá khứ), thông qua mô hình đã được xây dựng, dự đoán các ô còn trống. Để có thể sử dụng cho việc tính toán thì ma trận phải được xử lý tất cả các giá trị khuyết. Có nhiều hướng xử lý, chẳng hạn như thay giá trị khuyết bằng giá trị 0, giá trị trung bình toàn bộ đánh giá, giá trị trung bình ngưỡng 2.5 hay một giá trị ngẫu nhiên từ 0 đến 5. Tuy nhiên điều này sẽ gặp hạn chế với nhiều người dùng khó tính. Chúng tôi sử dụng giá trị trung bình cộng của mỗi người dùng (avg)(hình 9 minh họa một vài giá trị trong ma trận), thay đánh giá của mỗi người dùng bằng đánh giá của người dùng đó trừ đi avg. Sau đó, thay các giá trị khuyết bởi giá trị 0, được minh họa bởi hình 10. Mục đích của quá cách xử lý này nhằm phân loại đánh giá thành 2 loại giá trị âm và dương (để phân nhóm thích và không thích đối với mỗi bộ phim).

Bước tiếp đến chúng tôi tính ma trận cosine similarity dựa vào ma trận rating đã chuẩn hóa phía trên. Chúng tôi được ma trận thể hiện sự tương đồng giữa các bộ phim (với kích thước 160x160). Kết quả của ma trận consine similarity là một số từ -1 đến 1, giá trị càng lớn thì độ tương đồng càng cao, càng nhỏ thì càng đối lập. Hình 11 minh họa một vài giá trị trong ma trận cosine similarity.

		users						
		0	**1**	**2**	**3**	**4**	**5**	**6**
movies	**0**	5	5	2	0	1	?	?
	1	4	?	?	0	?	2	?
	2	?	4	1	?	?	1	1
	3	2	2	3	4	4	?	4
	4	2	0	4	?	?	?	5

Fig. 8. Minh họa ma trận đánh giá

Tiếp theo chúng tôi sẽ dự đoán đánh giá của một người dùng dựa trên 2 người dùng gần nhất, tương tự như phương pháp K-nearest neighbors (KNN) theo trình tự như sau:
Dự đoán rating của user u1 cho movie i1.

- Tìm danh sách tất cả các người dùng đã đánh giá cho movie i1.
- Tính độ tương đồng của u1 với tất cả các người dùng tìm được ở danh sách trên và chọn ra 2 người dùng có độ tương đồng với u1 cao nhất.
- Dự đoán rating của user u1 cho movie i1 bằng công thức:

$$r_{u_i,m_j} = \frac{s_{u_i,u_k} r_{u_k,m_j} + s_{u_i,u_h} r_{u_h,m_j}}{|s_{u_i,u_k}| + |s_{u_i u_h}|}$$

- Trong đó r_{u_k,m_j}, r_{u_h,m_j} lần lượt là rating của user u_k, u_h cho bộ phim m_j và s_{u_i,u_k}, s_{u_i,u_h} lần lượt là độ tương đồng giữa user u_i với u_k, u_h.

		users						
		0	**1**	**2**	**3**	**4**	**5**	**6**
movies	**0**	5	5	2	0	1	?	?
	1	4	?	?	0	?	2	?
	2	?	4	1	?	?	1	1
	3	2	2	3	4	4	?	4
	4	2	0	4	?	?	?	5
Trung bình		3.2	2.75	2.5	1.33	2.5	1.5	3.33

Fig. 9. Minh họa ma trận đánh giá sau khi tính trung bình phim cho mỗi người dùng

	users							
		0	**1**	**2**	**3**	**4**	**5**	**6**
movies	**0**	1.75	2.25	-0.5	-1.33	-1.5	?	0
	1	0.75	0	0	-1.33	0	0.5	0
	2	0	1.2	-1.5	0	0	-0.5	-2.33
	3	-1.25	-0.75	0.5	2.67	1.5	0	0.67
	4	-1.25	-2.75	1.5	0	0	0	1.67

Fig. 10. Minh họa ma trận đánh giá sau khi chuẩn hóa

Sau khi dự đoán cho các giá trị rating bị khuyết, chúng tôi chuẩn hóa về thang đo từ 0 đến 5 như ban đầu bằng cách cộng rating dự đoán với trung bình rating đã tính(ở hình 9).

ALS model (Model-based) Phương pháp cuối cùng là lọc dựa trên mô hình. Ở đây chúng tôi sử dụng mô hình Alternating Least Squares với bộ tham số maxIter = 25, rank=200, regParam= 0.0479 để thực nghiệm với bài toán trên.

4.3 Hybrid rcommend

Ở phương pháp Hybrid, chúng tôi thực hiện kết hợp kết quả của lọc cộng tác và lọc dựa theo nội dung. Cụ thể hơn, sau khi có kết quả dự đoán của mỗi người dùng cho mỗi bộ phim bằng lọc nội dung và lọc cộng tác, chúng tôi tính trung bình rating của người dùng cho mỗi phim bằng cách tính rating trung bình của 2 kết quả của 2 phương pháp trên trên. Trường hợp

	moviesId							
moviesId		0	1	2	3	4	5	6
	0	1	0.83	-0.58	-0.79	-0.82	0.2	-0.38
	1	0.83	1	-0.87	-0.4	-0.55	-0.23	-0.71
	2	-0.58	-0.87	1	0.27	0.32	0.47	0.96
	3	-0.79	-0.4	0.27	1	0.87	-0.29	0.18
	4	-0.82	-0.55	0.32	0.87	1	0	0.16
	5	0.2	-0.23	0.47	-0.29	0	1	0.56
	6	-0.38	-0.71	0.96	0.18	0.16	0.56	1

Fig. 11. Minh họa trận cosine similarity giữa các phim

chỉ có một phương pháp cho kết quả thì rating sau khi kết hợp sẽ bằng rating của phương pháp đó. Sau đó chọn ra top 5 phim có rating dự đoán cao nhất của mỗi người dùng để đề xuất cho người dùng đó.

4.4 Kết quả thực nghiệm

Sau khi thực nghiệm các phương pháp đã đề xuất phía trên, chúng tôi được kết quả ở bảng 4.4.

STT	Phương pháp	RMSE
1	**Trung bình mỗi nhóm movieId**	0.9643
2	ALS model	0.8765
3	Content-filtering	0.8763
4	ALS model + items-based +Content - filtering	0.8809
5	**ALS model + users-based+ Content-filtering**	0.8699
6	ALS model + Content-filtering	0.8676

Table 8. Kết quả thực nghiệm

Phân tích kết quả Phương pháp tiếp cận cho kết quả tốt nhất là ALS model + Content-filering (rmse = 0.876). Kết quả cho thấy có sự khác biệt ở các phương pháp tuy nhiên không nhiều. Về mặt dữ liệu, vì giới hạn tài nguyên nên dữ liệu sử dụng có kích thước ở mức tương đối, không đủ lớn để thể hiện rõ các phương pháp. Ở các bộ dữ liệu 1M, 2M, 25M, các thẻ(tag) có kích thước lớn hơn và nội dung đa dạng hơn.

5 Kết luận

Đồ án này chúng tôi đã sử dụng tập dữ liệu MovieLens phiên bản 100,000 đánh giá để thực nghiệm dự đoán đánh giá phim của người dùng và đề xuất, gợi ý phim cho từng người dùng.

Phương pháp cho kết quả tốt nhất là phương pháp lai(hybrid) kết hợp giữa lọc cộng tác(ALS model) và lọc dựa trên nội dung(Content-filtering) đạt RMSE = 0.8676. Hệ thống có thể đề xuất phim cho từng người dùng theo id của người dùng.

Qua quá trình thực hiện, chúng tôi thấy được mỗi phương pháp có những điểm mạnh khác nhau. Lọc công tác dễ dàng thực hiện khi đưa ra gợi ý đánh giá cho phim với mô hình ALS, phương pháp này cũng không đòi hỏi dữ liệu chi tiết về người dùng. Lọc dựa trên nội dung không đòi hỏi số lượng lớn thông tin về người dùng đã có thể học được đặc trưng khá chính xác từ đó có thể đưa ra những bộ phim tương tự hợp với người dùng. Tuy nhiên, các mặt hạn chế của các phương pháp trên vẫn còn tồn tại, mặc dù áp dụng phương pháp lai(hybrid) nhưng không thể khắc phục hoàn toàn như khi có người dùng mới chưa có bất kì thông tin nào thì hệ thống không thể đưa ra đề xuất cho họ.

Để cải thiện hệ thống gợi ý phim cho người dùng, chúng tôi sẽ nghiên cứu cách kết khác của phương pháp lai(hybrid) như thêm các khả năng dựa trên nội dung vào phương pháp cộng tác, cải thiện thêm về mặt dữ liệu sử dụng dữ liệu lớn hơn với hiện tại(100k) và khai thác thêm các thuộc tính khác của dữ liệu để góp phần nâng cao hệ thống gợi ý chuẩn xác hơn.