TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN ĐIỆN TỬ - VIỄN THÔNG

ĐỒ ÁN III

Đề tài:

Ứng dụng hệ thống gợi ý trong lĩnh vực thương mại điện tử

Sinh viên thực hiện: CHU ĐỨC HIẾU ĐIỆN TỬ 06 – K60

Giảng viên hướng dẫn: ThS. NGUYỄN THỊ KIM THOA

Hà Nội, 1-2020

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN ĐIỆN TỬ - VIỄN THÔNG

ĐỒ ÁN III

Đề tài:

Ứng dụng hệ thống gợi ý trong lĩnh vực thương mại điện tử

Sinh viên thực hiện: CHU ĐỨC HIẾU ĐIỆN TỬ 06 – K60

Giảng viên hướng dẫn: ThS. NGUYỄN THỊ KIM THOA

Hà Nội, 1-2020

LỜI NÓI ĐẦU

Trong những năm gần đây, sự phát triển của thương mại điện tử (E-Commerce) đã đem lại nhiều lợi ích to lớn cho nền kinh tế toàn cầu. Thông qua thương mại điện tử, nhiều loại hình kinh doanh mới được hình thành, trong đó có mua bán hàng trên mạng. Với hình thức này người tiêu dùng có thể tiếp cận với hàng hóa một cách dễ dàng và nhanh chóng hơn rất nhiều so với hình thức mua bán hàng truyền thống.

Hiện nay các hệ thống bán hàng trực tuyến đã tạo nhiều điều kiện thuận lợi để người mua có thể tiếp cận nhiều mặt hàng cùng một lúc. Tuy nhiên, một website thương mại thì luôn luôn mong muốn phát triển số lượng khách hàng, và muốn có nhiều khách hàng thì họ phải đa dạng hóa các loại sản phẩm để đáp ứng được nhu cầu mua sắm của nhiều loại khách hàng, do vậy số lượng sản phẩm và loại sản phẩm được trưng bày trong website ngày càng tăng và sẽ làm hạn chế khả năng giao tiếp chọn sản phẩm của khách hàng, họ phải duyệt qua nhiều liên kết, sàng lọc nhiều thông tin mới có thể tìm được sản phẩm mong muốn. Vậy làm sao hỗ trợ khách hàng trong công việc lựa chọn sản phẩm mua sắm? Cụ thể, những sản phẩm nào nên được đề xuất tiếp theo các sản phẩm đã được khách hàng đánh giá hoặc chọn trong giỏ hàng? Nên đề xuất bao nhiêu sản phẩm là tốt nhất cho khách hàng?

Để khách hàng có thể tìm và mua được một sản phẩm ưng ý thì một lời khuyên, một sự trợ giúp là rất quan trọng. Một người bán trong phương thức mua bán truyền thống là một lợi thế rất lớn. Do đó để hình thức mua bán qua mạng thực sự phát triển thì bên cạnh các lợi thế vốn có của mình việc có thêm một "người trợ giúp" là hết sức cần thiết. Hệ tư vấn được hình thành và phát triển không nằm ngoài mục đích đáp ứng những yêu cầu trên. Một hệ thống tư vấn tốt có thể đóng vai trò như người trung gian hỗ trợ khách hàng đưa ra quyết định chọn hàng. Tiện ích này đóng vai trò như một người bán hàng có khả năng thu thập thông tin về sở thích của khách hàng, sau đó tìm trong kho hàng vô tận của mình những mặt hàng thích hợp nhất với sở thích ñó. Thực chất của một hệ thống tư vấn này là quá trình hỗ trợ khách hàng đưa ra quyết định.

MỤC LỤC

DANH SÁCH HÌNH VẼ

DANH SÁCH CÁC BẢNG BIỂU

PHẦN MỞ ĐẦU

Đặt vấn đề

Ngày nay, mua sắm là nhu cầu thiết yếu của mỗi con người, và khi chúng ta mua sắm, đó chắc chắn là sản phẩm chúng ta thích hoặc bạn bè của chúng ta thích. Với lượng thông tin ngày càng tăng trên internet và số lượng người dùng tăng lên đáng kể, điều quan trọng đối với các công ty là tìm kiếm, liên kết và cung cấp cho khách hàng những thông tin liên quan theo sở thích và thị hiếu của họ. Người dùng các hệ thống thông tin, đặc biệt là các website thương mại điện tử thường gặp các vấn đề về tìm kiếm sản phẩm phù hợp với nhu cầu của họ do lượng sản phẩm lớn, thời gian có hạn. Và đó là lý do trong thời đại kỹ thuật số ngày nay, bất kỳ cửa hàng trực tuyến nào chúng ta ghé thăm cũng đều sử dụng một số loại hệ thống gợi ý.

Hướng triển khai đề tài

Đầu tiên, tác giả sẽ tìm hiểu khái niệm chung về hệ thống gợi ý, sau đó sẽ tập trung vào khảo sát các nhóm thuật toán phổ biến trong các hệ thống gợi ý hiện nay. Cuối cùng, tác giả sẽ thực hiện viết mã một số phương pháp gợi ý cơ bản và thử nghiệm trên các bộ dữ liệu thực tế, qua đó hiểu rõ ưu điểm và nhược điểm của các phương pháp này khi được áp dụng.

Tổng quan đồ án

Mục tiêu của đồ án là khảo sát lý thuyết chung về hệ thống gợi ý, sau đó xây dựng mã nguồn thuật toán dựa trên lý thuyết và sử dụng mã nguồn đó đánh giá kết quả trên dữ liệu thực tế. Đồ án cũng có một phần sử dụng mã nguồn của một bài báo khoa học để thử nghiệm lại nhằm mục đích có cái nhìn sâu sắc hơn về ứng dụng của hệ thống gợi ý.

Có các cách tiếp cận chính sau để xây dựng hệ thống gợi ý: nhóm giải thuật lọc theo nội dung (content-based filtering), nhóm giải thuật lọc cộng tác (collaborative filtering), nhóm giải thuật lai ghép (hybrid filtering) và nhóm giải thuật không cá nhân hóa (non-personalization). Các phương pháp này sẽ được giới thiệu chi tiết trong các

chương tiếp theo.

Đầu ra của các mô hình gợi ý là những nội dung được dự đoán là sẽ được người dùng yêu thích. Mức độ hiệu quả của mô hình sẽ được đánh giá khi áp dụng lên hai bộ dữ liệu thực tế là Movielens và Globo.com, dựa trên các phương pháp theo lý thuyết (RMSE, MAE, ...) và thực tế (Hit Rate, MRR, ...).

Cấu trúc đồ án

Đồ án gồm có 2 chương, đi theo hướng từ nghiên cứu lý thuyết đến áp dụng thực tế:

- **CHƯƠNG 1. TỔNG QUAN VỀ HỆ GỢI Ý (RECOMMENDER SYSTEMS)**
- **ÁP DỤNG THUẬT TOÁN GỢI Ý VỚI MỘT SỐ BỘ DỮ LIỆU THỰC TẾ**

CHƯƠNG 1. TỔNG QUAN VỀ HỆ GỢI Ý (RECOMMENDER SYSTEMS)

1.1 Giới thiệu

Hệ thống gợi ý (Recommender Systems - RS) là một dạng của hệ thống lọc thông tin (information filtering), nó được sử dụng để dự đoán sở thích (preferences) hay xếp hạng (rating) mà người dùng có thể dành cho một mục thông tin (item) nào đó mà họ chưa xem xét tới trong quá khứ (item có thể là bài hát, bộ phim, đoạn video clip, sách, bài báo,..).

Ví dụ, trong hệ thống bán hàng trực tuyến (chẳng hạn như Amazon), nhằm tối ưu hóa khả năng mua sắm của khách hàng (user), người ta quan tâm đến việc những khách hàng nào đã 'yêu thích' những sản phẩm (item) nào bằng cách dựa vào dữ liệu quá khứ của họ (dữ liệu này có thể là xếp hạng mà người dùng đã bình chọn trên sản phẩm, thời gian duyệt (browse) trên sản phẩm, số lần click chuột trên sản phẩm,..) từ đó hệ thống sẽ dự đoán được người dùng có thể thích sản phẩm nào và đưa ra những gợi ý phù hợp cho họ. Hình 1 là một ví dụ minh họa cho hệ thống gợi ý bán hàng của Amazon.

Ngoài lĩnh vực thương mại điện tử như đã thấy ở ví dụ trên, hiện tại RS cũng được ứng dụng khá thành công trong nhiều lĩnh vực khác như trong giải trí: gợi ý bài hát cho người nghe (ví dụ, hệ thống của LastFM - www.last.fm), gợi ý phim ảnh (ví dụ, hệ thống của Netflix - www.netflix.com), gợi ý các video clip (ví dụ, hệ thống của YouTube - www.youtube.com); trong giáo dục và đào tạo (gợi ý nguồn tài nguyên học tập như sách, bài báo, địa chỉ web,… cho người học).

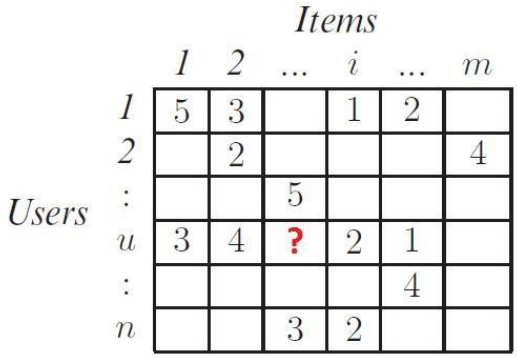
Hình 1.1 Hệ thống gợi ý sản phẩm của Amazon

Hệ thống gợi ý không chỉ đơn thuần là một dạng Hệ thống thông tin mà nó còn là cả một lĩnh vực nghiên cứu hiện đang rất được các nhà khoa học quan tâm. Kể từ năm 2007 đến nay, hàng năm đều có hội thảo chuyên về hệ thống gợi ý của ACM (ACM RecSys) cũng như các tiểu bang dành riêng cho RS trong các hội nghị lớn khác như ACM KDD, ACM CIKM,..

1.2 Hệ thống gợi ý (Recommender Systems - RS)

1.2.1 Các khái niệm chính

Trong RS, thông thường người ta quan tâm đến ba thông tin chính là người dùng (user), mục tin (item, item có thể là sản phẩm, bộ phim, bài hát, bài báo,.. tùy hệ thống), và phản hồi (feedback) của người dùng trên mục tin đó (thư ờng là các xếp hạng/đánh giá – rating biểu diễn mức độ thích/quan tâm của họ). Các thông tin này được biểu diễn thông qua một ma trận như trong Hình 2. Ở đó, mỗi dòng là một user, mỗi cột là một item, và mỗi ô là một giá trị phản hồi (ví dụ, xếp hạng) biểu diễn "mức độ thích" của user trên item tương ứng. Các ô có giá trị là những item mà các user đã xếp hạng trong quá khứ. Những ô trống là những item chưa được xếp hạng (điều đáng lưu ý là mỗi user chỉ xếp hạng cho một vài item trong quá khứ, do vậy có rất nhiều ô trống trong ma trận này – còn gọi là ma trận thưa – sparse matrix).



Hình 1.2 Ma trận biểu diễn dữ liệu trong RS (user-item-rating matrix)

Nhiệm vụ chính của RS là dựa vào các ô đã có giá trị trong ma trận trên (dữ liệu thu được từ quá khứ), thông qua mô hình đã đư ợc xây dựng, RS dự đoán các ô còn trống (của user hiện hành), sau đó sắp xếp kết quả dự đoán (ví dụ, từ cao xuống thấp) và chọn ra Top-N items theo thứ tự, từ đó gợi ý chúng cho người dùng.

1.2.2 Thông tin phản hồi từ người dùng và hai dạng bài toán chính trong RS

Trong RS, giá trị phản hồi (feedback) r_{ui} của mỗi người dùng trên mục tin sẽ được ghi nhận lại để làm cơ sở cho việc dự đoán các giá trị kế tiếp. Tùy theo hệ thống mà giá trị này sẽ có ý nghĩa khác nhau, ví dụ nó có thể dùng để đo độ "phù hợp" hay "mức độ thích" (thường là các đánh giá trên các sản phẩm) trong các hệ thống thương mại điện tử hay "năng lực/kết quả thực hiện" của người dùng trong các hệ thống e-learning.

Giá trị r_{ui} có thể được xác định một cách tường minh (explicit feedbacks) như thông qua việc đánh giá/xếp hạng (ví dụ, rating từ ★ đến ★★★★★; hay like (1) và dislike (0),…) mà người dùng *u* đã bình chọn cho item *i*; hoặc r_{ui} có thể được xác định một cách không tường minh (implicit feedbacks) thông qua số lần click chuột, thời gian mà *u* đã duyệt/xem *i*,…

Có 2 dạng bài toán chính trong RS là *dự đoán xếp hạng (rating prediction)* của các hệ thống có phản hồi tường minh như đã trình bày ở trên và *dự đoán mục thông tin* (item prediction/recommendation) là việc xác định xác suất mà người dùng thích mục tin tương ứng.

1.3 Các kỹ thuật chính trong RS

Hiện tại, trong RS có rất nhiều giải thuật được đề xuất, tuy nhiên có thể gom chúng vào trong các nhóm chính: nhóm giải thuật lọc theo nội dung (content-based filtering), nhóm giải thuật lọc cộng tác (collaborative filtering), nhóm giải thuật lai ghép (hybrid filtering) và nhóm giải thuật không cá nhân hóa (non-personalization).

1.3.1 Lọc cộng tác

Một cách tiếp cận để thiết kế các hệ thống recommender được sử dụng rộng rãi là lọc cộng tác. Các phương pháp lọc cộng tác dựa trên việc thu thập và phân tích một lượng lớn thông tin về hành vi, hoạt động hoặc sở thích của người dùng và dự đoán những gì người dùng sẽ thích dựa trên sự tương đồng của họ với người dùng khác. Một lợi thế quan trọng của phương pháp lọc cộng tác là nó không dựa vào nội dung phân tích máy và do đó nó có khả năng đề xuất chính xác các mục phức tạp như phim mà không yêu cầu "hiểu biết" về mục đó. Nhiều thuật toán đã được sử dụng để đo lường sự giống nhau của người dùng hoặc sự tương đồng về mặt hàng trong các hệ thống giới thiệu. Ví dụ, cách tiếp cận hàng xóm gần nhất (k-nearest neighbor) và Pearson Correlation được Allen triển khai lần đầu tiên.

Lọc cộng tác dựa trên giả định rằng những người đã đồng ý trong quá khứ sẽ đồng ý trong tương lai và rằng họ sẽ thích các loại mặt hàng tương tự như họ thích trong quá khứ.

Khi xây dựng mô hình từ hành vi của người dùng, sự phân biệt thường được thực hiện giữa các hình thức thu thập dữ liệu rõ ràng và tiềm ẩn.

Ví dụ về thu thập dữ liệu rõ ràng bao gồm:

- Yêu cầu người dùng xếp hạng một mục trên thang trượt.
- Yêu cầu người dùng tìm kiếm.
- Yêu cầu người dùng xếp hạng một bộ sưu tập các mục từ yêu thích đến ít yêu thích nhất.
- Trình bày hai mục cho một người dùng và yêu cầu anh ta / cô ấy chọn một trong số chúng tốt hơn.
- Yêu cầu người dùng tạo danh sách các mục mà anh / cô ấy thích.

Ví dụ về thu thập dữ liệu ngầm bao gồm:

- Quan sát các mục mà người dùng xem trong cửa hàng trực tuyến.

- Phân tích thời gian xem mục / người dùng.
- Lưu giữ một bản ghi các mục mà người dùng mua trực tuyến.
- Lấy danh sách các mục mà người dùng đã nghe hoặc xem trên máy tính của họ.
- Phân tích mạng xã hội của người dùng và khám phá những lượt thích và không thích tương tự.

Hệ thống recommender so sánh dữ liệu đã thu thập với dữ liệu tương tự và khác nhau được thu thập từ những người khác và tính toán danh sách các mục được đề xuất cho người dùng. Một số ví dụ thương mại và phi thương mại được liệt kê trong bài viết về các hệ thống lọc cộng tác .

Một trong những ví dụ nổi tiếng nhất về lọc cộng tác là lọc cộng tác theo từng mục (những người mua x cũng mua y), một thuật toán được phổ biến rộng rãi bởi hệ thống gợi ý của Amazon.com. Các ví dụ khác bao gồm:

- Như đã đề cập chi tiết ở trên, Last.fm đề xuất âm nhạc dựa trên so sánh thói quen nghe của những người dùng tương tự, trong khi Readgeek so sánh xếp hạng sách cho các đề xuất.
- Facebook , MySpace , LinkedIn và các mạng xã hội khác sử dụng tính năng lọc cộng tác để giới thiệu bạn bè, nhóm và các kết nối xã hội khác (bằng cách kiểm tra mạng kết nối giữa người dùng và bạn bè của họ). Twitter sử dụng nhiều tín hiệu và tính toán trong bộ nhớ để giới thiệu cho người dùng của họ rằng họ nên "theo dõi".

Các phương pháp lọc cộng tác thường gặp phải ba vấn đề: Cold Start, khả năng mở rộng và sự thưa thớt (sparsity).

- Cold Start: Các hệ thống này thường yêu cầu một lượng lớn dữ liệu hiện có của người dùng để đưa ra các đề xuất chính xác.
- Khả năng mở rộng: Trong nhiều môi trường mà các hệ thống này đưa ra các khuyến nghị, có hàng triệu người dùng và sản phẩm. Do đó, một lượng lớn công suất tính toán thường là cần thiết để tính toán các gợi ý.
- Sparsity: Số lượng các mặt hàng được bán trên các trang web thương mại điện tử lớn là cực kỳ lớn. Những người dùng tích cực nhất sẽ chỉ đánh giá một tập con nhỏ của cơ sở dữ liệu tổng thể. Do đó, ngay cả những mặt hàng phổ biến nhất cũng có rất ít xếp hạng.

Một loại thuật toán lọc cộng tác cụ thể sử dụng hệ số ma trận hóa (matrix factorization), kỹ thuật xấp xỉ ma trận cấp thấp (low-rank matrix approximation).

Các phương pháp lọc cộng tác được phân loại là bộ lọc cộng tác dựa trên bộ nhớ và dựa trên mô hình. Một ví dụ nổi tiếng về các phương pháp dựa trên bộ nhớ là thuật toán dựa trên người dùng và các phương pháp dựa trên mô hình là Kernel-Mapping Recommender.

1.3.2 Lọc dựa trên nội dung

Một cách tiếp cận phổ biến khác khi thiết kế hệ thống recommender là lọc nội dung . Phương pháp lọc dựa trên nội dung dựa trên mô tả về mặt hàng và hồ sơ về các tùy chọn của người dùng.

Trong hệ thống gợi ý dựa trên nội dung, từ khóa được sử dụng để mô tả các mục và hồ sơ người dùng được xây dựng để chỉ ra loại mục mà người dùng này thích. Nói cách khác, các thuật toán này cố gắng đề xuất các mục tương tự với các mục mà người dùng đã thích trong quá khứ (hoặc đang kiểm tra trong hiện tại). Cụ thể, các mục đề cử khác nhau được so sánh với các mục được đánh giá trước đây bởi người dùng và các mục phù hợp nhất được đề xuất. Cách tiếp cận này có nguồn gốc từ việc thu thập thông tin và nghiên cứu lọc thông tin .

Để tóm tắt các tính năng của các mục trong hệ thống, một thuật toán trình bày mục được áp dụng. Một thuật toán được sử dụng rộng rãi là **biểu diễn tf – idf** (còn được gọi là biểu diễn không gian vectơ).

Để tạo hồ sơ người dùng, hệ thống chủ yếu tập trung vào hai loại thông tin:

1. Một mô hình ưu tiên của người dùng.
2. Lịch sử tương tác của người dùng với hệ thống gợi ý.

Về cơ bản, các phương thức này sử dụng một hồ sơ mặt hàng (ví dụ, một tập hợp các thuộc tính và tính năng rời rạc) mô tả mục trong hệ thống. Hệ thống tạo hồ sơ dựa trên nội dung của người dùng dựa trên vectơ trọng số của các đối tượng địa lý. Trọng số biểu thị tầm quan trọng của từng tính năng đối với người dùng và có thể được tính từ các vectơ nội dung được xếp hạng riêng lẻ bằng nhiều kỹ thuật. Các phương pháp đơn giản sử dụng các giá trị trung bình của vector hạng mục trong khi các phương pháp phức tạp khác sử dụng các kỹ thuật máy học như Bayesian Classifiers , phân tích cụm , cây quyết định và mạng thần kinh nhân tạo (artificial neural networks) để ước tính xác suất người dùng sẽ thích mục đó.

Phản hồi trực tiếp từ người dùng, thường dưới dạng nút thích hoặc không thích , có thể được sử dụng để gán trọng số cao hơn hoặc thấp hơn về tầm quan trọng của các thuộc tính nhất định (sử dụng phân loại Rocchio hoặc các kỹ thuật tương tự khác).

Một vấn đề quan trọng với lọc dựa trên nội dung là liệu hệ thống có thể tìm hiểu các tùy chọn của người dùng từ hành động của người dùng liên quan đến một nguồn nội dung hay không và sử dụng chúng trên các loại nội dung khác. Khi hệ thống bị hạn chế đề xuất nội dung cùng loại với người dùng đang sử dụng, giá trị từ hệ thống đề xuất thấp hơn đáng kể so với các loại nội dung khác từ các dịch vụ khác có thể được đề xuất. Ví dụ: giới thiệu các bài viết tin tức dựa trên việc duyệt tin tức hữu ích nhưng sẽ hữu ích hơn nhiều khi bạn có thể đề xuất âm nhạc, video, sản phẩm, cuộc thảo luận, v.v. từ các dịch vụ khác nhau dựa trên duyệt tin tức.

Pandora Radio là một ví dụ về hệ thống giới thiệu dựa trên nội dung phát nhạc có các đặc điểm tương tự như một bài hát do người dùng cung cấp làm hạt giống ban đầu. Ngoài ra còn có một số lượng lớn các hệ thống gợi ý dựa trên nội dung nhằm cung cấp các đề xuất phim, một vài ví dụ như Rotten Tomatoes , Internet Movie Database , Jinni , Rovi Corporation và Jaman . Các hệ thống gợi ý giới thiệu tài liệu liên quan nhằm mục đích cung cấp các đề xuất tài liệu cho các nhà nghiên cứu. Các chuyên gia y tế công cộng đã nghiên cứu các hệ thống gợi ý để cá nhân hóa giáo dục sức khỏe và các chiến lược phòng ngừa.

1.3.3 Hệ thống gợi ý lai (Hybrid recommender systems)

Nghiên cứu gần đây đã chứng minh rằng một phương pháp lai, kết hợp lọc cộng tác và lọc dựa trên nội dung có thể hiệu quả hơn trong một số trường hợp. Các phương pháp lai có thể được thực hiện theo nhiều cách:

- Bằng cách đưa ra các dự đoán dựa trên nội dung và dựa trên lọc cộng tác riêng biệt và sau đó kết hợp chúng.
- Bằng cách thêm các khả năng dựa trên nội dung vào phương pháp cộng tác (và ngược lại).
- Bằng cách thống nhất các phương pháp tiếp cận thành một mô hình.

Một số nghiên cứu thực nghiệm so sánh hiệu suất của phương pháp lai với các phương pháp cộng tác thuần túy và chứng minh rằng các phương pháp lai có thể cung cấp các khuyến nghị chính xác hơn các phương pháp thuần túy. Những phương pháp này

cũng có thể được sử dụng để khắc phục một số vấn đề thường gặp trong hệ thống gợi ý như Cold Start và vấn đề thưa thớt.

Netflix là một ví dụ tốt về việc sử dụng các hệ thống hybrid recommender. Trang web đưa ra các đề xuất bằng cách so sánh thói quen xem và tìm kiếm của những người dùng tương tự (ví dụ: lọc cộng tác) cũng như bằng cách cung cấp những bộ phim có chung đặc điểm với những bộ phim mà người dùng đánh giá cao (lọc dựa trên nội dung).

Một loạt các kỹ thuật đã được đề xuất làm cơ sở cho các hệ thống gợi ý: các kỹ thuật hợp tác (collaborative), dựa trên nội dung (content-based), dựa trên kiến thức (knowledge-based) và nhân khẩu học (demographic techniques). Mỗi kỹ thuật này đều có những thiếu sót, như vấn đề Cold Start cho các hệ thống cộng tác và dựa trên nội dung (phải làm gì với người dùng mới với ít xếp hạng) và tắc nghẽn kỹ thuật tri thức (knowledge engineering bottleneck) trong các phương pháp dựa trên tri thức . Một hệ thống gợi ý lai là một hệ thống trong đó kết hợp nhiều kỹ thuật với nhau để đạt được một số sức mạnh tổng hợp giữa chúng.

- Cộng tác – Collaborative: Hệ thống tạo đề xuất chỉ sử dụng thông tin về hồ sơ xếp hạng cho những người dùng hoặc mục khác nhau. Các hệ thống cộng tác định vị "người dùng/mục" ngang hàng với lịch sử xếp hạng tương tự như người dùng hoặc mục hiện tại và tạo đề xuất sử dụng vùng lân cận này. Các thuật toán dựa trên người dùng và dựa trên hàng gần nhất có thể được kết hợp để giải quyết vấn đề Cold Start và cải thiện kết quả đề xuất.
- Dựa trên nội dung – Content-based: Hệ thống tạo đề xuất từ hai nguồn: các tính năng liên quan đến sản phẩm và xếp hạng mà người dùng đã cung cấp cho họ. Đề xuất dựa trên nội dung coi đề xuất là sự cố phân loại người dùng cụ thể và tìm hiểu trình phân loại cho lượt thích và không thích của người dùng dựa trên các tính năng của sản phẩm.
- Nhân khẩu học – demographic techniques: Trình giới thiệu nhân khẩu học cung cấp các đề xuất dựa trên hồ sơ nhân khẩu học của người dùng. Sản phẩm được đề xuất có thể được sản xuất cho các mục nhân khẩu học khác nhau, bằng cách kết hợp xếp hạng của người dùng trong các mục đó.
- Dựa trên tri thức – knowledge-based: Trình giới thiệu dựa trên kiến thức gợi ý các sản phẩm dựa trên các suy luận về nhu cầu và sở thích của người dùng. Kiến thức này đôi khi sẽ chứa kiến thức chức năng rõ ràng về cách các tính năng sản phẩm nhất định đáp ứng nhu cầu của người dùng.

Thuật ngữ Hybrid recommender systems được sử dụng ở đây để mô tả bất kỳ hệ thống recommender nào kết hợp nhiều kỹ thuật đề xuất với nhau để tạo dữ liệu đầu ra của nó.

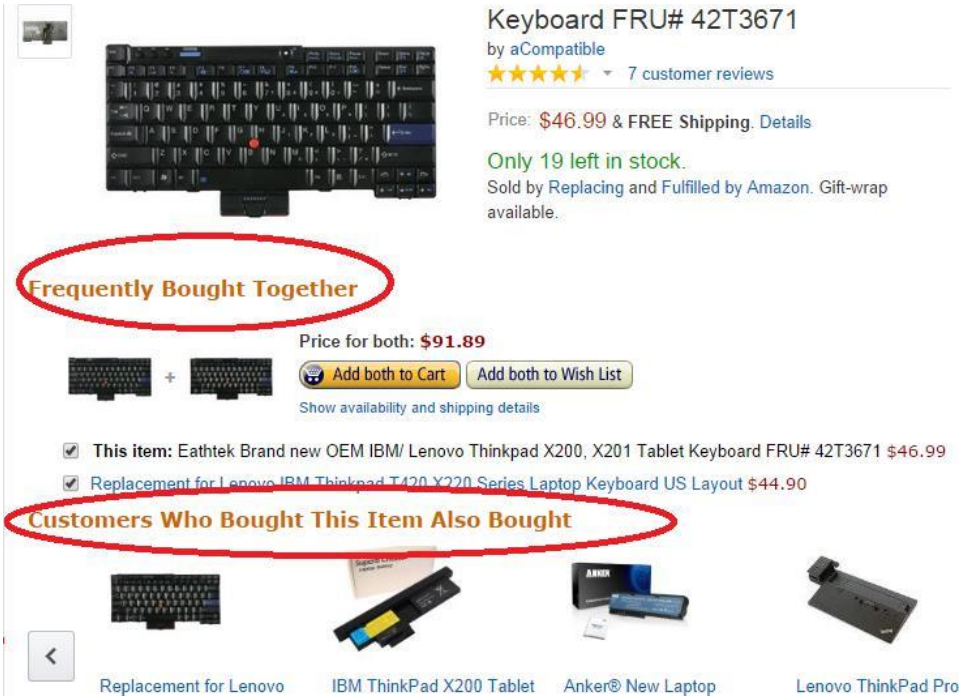
Có bảy kỹ thuật lai cơ bản (hybridization techniques):

- Có trọng số (Weighted): Điểm số của các thành phần đề xuất khác nhau được kết hợp theo số lượng.
- Chuyển đổi (Switching): Hệ thống chọn giữa các thành phần đề xuất và áp dụng hệ thống được chọn.
- Hỗn hợp (Mixed): Các khuyến nghị từ những người giới thiệu khác nhau được trình bày cùng nhau để đưa ra đề xuất.
- Kết hợp tính năng (Feature Combination): Các tính năng được lấy từ các nguồn tri thức khác nhau được kết hợp với nhau và được đưa ra cho một thuật toán gợi ý duy nhất.
- Tính năng tăng cường (Feature Augmentation): Một kỹ thuật gợi ý được sử dụng để tính toán một tính năng hoặc tập hợp các tính năng, sau đó là một phần của đầu vào cho kỹ thuật tiếp theo.
- Cascade: Các khuyến nghị được ưu tiên nghiêm ngặt, với những ưu tiên thấp hơn phá vỡ các mối quan hệ trong việc tính điểm của những người cao hơn.
- Cấp độ meta (Meta-level): Một kỹ thuật đề xuất được áp dụng và tạo ra một số loại mô hình, sau đó là đầu vào được sử dụng bởi kỹ thuật tiếp theo.

1.3.4 Các kỹ thuật không cá nhân hóa

Trong nhóm kỹ thuật này, do chúng khá đơn gi ản, dễ cài đ ặt nên nên thường được các website/hệ thống tích hợp vào, gồm cả các website thương mại, website tin tức, hay giải trí. Chẳng hạn như trong các hệ thống bán hàng trực tuyến, người ta thường gợi ý các sản phẩm được xem/mua/bình luận/.. nhiều nhất; gợi ý các sản phẩm mới nhất; gợi ý các sản phẩm cùng loại/ cùng nhà sản xuất/..; gợi ý các sản phẩm được mua/chọn cùng nhau. Một ví dụ khá điển hình là thông qua luật kết hợp (như Apriori), Amazon đã áp dụng khá thành công để tìm ra các sản phẩm hay được mua cùng nhau như minh họa trong Hình 4.

Tuy vậy, bất lợi của các phương pháp này là không cá nhân hóa cho từng người dùng, nghĩa là tất cả các user đều được gợi ý giống nhau khi chọn cùng sản phẩm.



Hình 1.3 Gợi ý sản phẩm thường được mua cùng nhau

1.4 Deep learning trong hệ thống khuyến nghị:

Deep Learning (DL) là một chủ đề nóng trong cộng đồng học máy. Sự phổ biến của việc áp dụng học sâu vào hệ thống khuyến nghị là tương đối chậm, vì chủ đề này chỉ trở nên phổ biến trong năm 2016, với hội thảo Deep Learning for recommender Systems tại ACM RecSys 2016.

Mạng nơ-ron hồi quy (RNN) có một số thuộc tính làm cho chúng trở nên phù hợp để mô hình hóa chuỗi các phiên truy cập của người dùng. Đặc biệt, chúng có khả năng kết hợp đầu vào từ các sự kiện xảy ra trong quá khứ, cho phép dự đoán tốt hơn ý định của người dùng.

1.5 Hệ thống gợi ý tin tức:

Các cổng tin tức phổ biến, như Google News, Yahoo! News, The New York Times, Washington Post, cùng với nhiều cổng thông tin khác đã thu hút được sự chú ý ngày càng tăng từ một lượng lớn độc giả trên internet. Các hệ thống khuyến nghị tin tức trực tuyến

đã được các nhà nghiên cứu đề cập đến trong những năm qua, bằng cách sử dụng nhiều phương pháp khác nhau: lọc dựa trên nội dung, lọc cộng tác và phương pháp lai kết hợp.

Một số thách thức đối với hệ khuyến nghị tin tức có thể kể đến:

- Hồ sơ người dùng thưa thớt - phần lớn độc giả là ẩn danh và họ thực sự chỉ đọc một vài câu chuyện từ toàn bộ kho lưu trữ. Điều này dẫn đến mức độ thưa thớt cực cao trong ma trận bài viết - người dùng, vì người dùng thường theo dõi rất ít thông tin về hành vi trong quá khứ của họ, nếu có.
- Số lượng bài viết tăng nhanh - hàng trăm bài viết mới được thêm vào hàng ngày trong các cổng tin tức (ví dụ: hơn 300 bài trên trang The New York Times). Điều này làm nghiêm trọng vấn đề cold-start, vì đối với các bài viết mới, ta không có nhiều tương tác trong quá khứ để có thể dựa vào đó và đề xuất chúng. Đối với các công cụ tổng hợp tin tức, các vấn đề về khả năng mở rộng có thể phát sinh, vì một khối lượng lớn các bài báo sẽ làm quá tải web trong khoảng thời gian giới hạn.
- Thời gian sống của bài viết - giá trị thông tin phân rã theo thời gian. Điều này đặc biệt đúng trong lĩnh vực tin tức, vì hầu hết người dùng quan tâm đến thông tin mới. Vì vậy, mỗi bài viết sẽ có thời hạn sử dụng ngắn. Thị hiếu của người dùng liên tục thay đổi - chủ đề tin tức được quan tâm không ổn định như trong lĩnh vực giải trí. Một số sở thích của người dùng thay đổi theo thời gian, trong khi một số chủ đề khác vẫn ổn định. Mối quan tâm hiện tại của người dùng trong một phiên truy cập có thể bị ảnh hưởng bởi bối cảnh của phiên đó (ví dụ: địa điểm, thời gian truy cập) hoặc bởi các bối cảnh chung (ví dụ: tin nóng hoặc các sự kiện quan trọng).

CHƯƠNG 2. ÁP DỤNG THUẬT TOÁN GỢI Ý VỚI MỘT SỐ BỘ DỮ LIỆU THỰC TẾ

2.1 Xây dựng thuật toán gợi ý phim:

2.1.1 Bộ dữ liệu Movielens:

Movielens 10M là bộ dữ liệu được sử dụng rộng rãi trong các bài báo nghiên cứu khoa học, bao gồm 10,000,054 xếp hạng của 71,567 người dùng dành cho 10,681 bộ phim. Bộ dữ liệu được chia thành nhiều tập tin nhỏ:

- user.dat: chứa thông tin về người dùng. Người dùng Movielens được chọn ngẫu nhiên để đưa vào. Id của họ đã được ẩn danh. Họ cũng được chọn riêng để đưa vào tập dữ liệu xếp hạng và gắn thẻ, ngụ ý rằng id người dùng có thể xuất hiện trong một tập dữ liệu nhưng không xuất hiện trong tập còn lại.

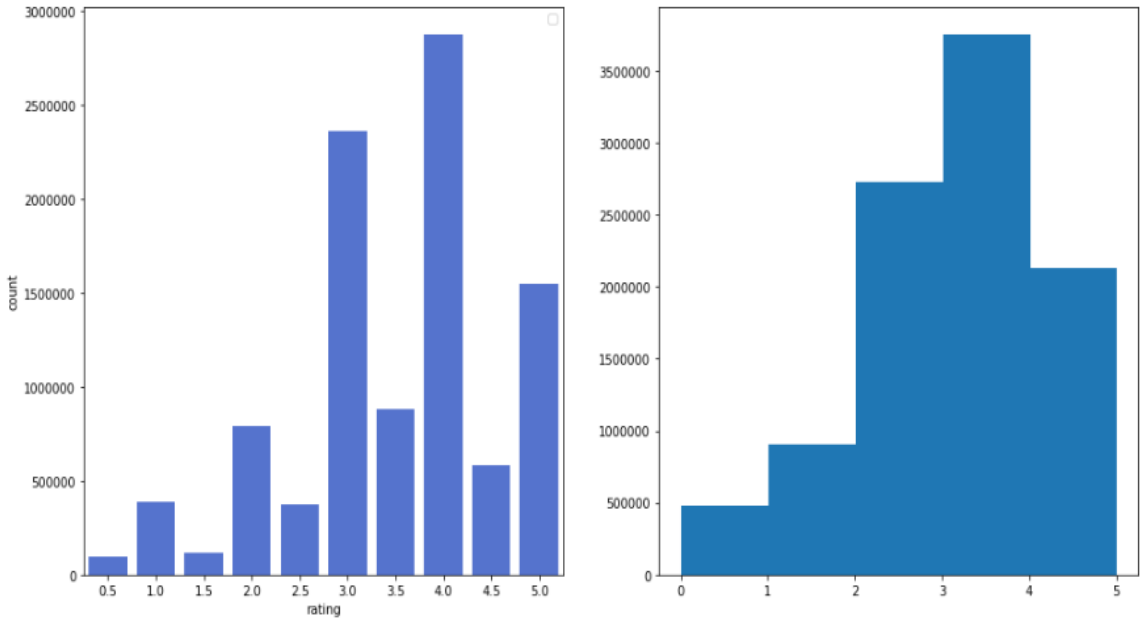
- ratings.dat: Tất cả các xếp hạng được chứa trong tập tin xếp hạng.dat. Mỗi dòng của tệp này đại diện cho một xếp hạng của một phim bởi một người dùng và có định dạng sau: UserID :: MovieID :: Xếp hạng :: Timestamp. Xếp hạng được thực hiện theo thang điểm 5 sao, bao gồm cả các giá trị nửa sao (0.5, 1.5, ...). Timestamp biểu thị số giây kể từ nửa đêm theo Giờ phối hợp quốc tế (UTC) ngày 1 tháng 1 năm 1970.

- movies.dat: Thông tin phim được chứa trong tập tin movies.dat. Mỗi dòng của tệp này đại diện cho một bộ phim và có định dạng sau: MovieID :: Title :: Thể loại. MovieID là id MovieLens thực sự. Tiêu đề phim, theo chính sách, phải được nhập chính xác cho những phim được tìm thấy trong IMDB, bao gồm cả năm phát hành. Tuy nhiên, chúng được nhập thủ công, do đó lỗi và sự không nhất quán có thể tồn tại. Thể loại phim được biểu diễn dưới dạng một danh sách, được chọn từ các mục sau: Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, Western.

- tags.dat: Tất cả các thẻ được chứa trong tập tin tags.dat. Mỗi dòng của tệp này đại diện cho một thẻ được áp dụng cho một phim bởi một người dùng và có định dạng sau: UserID :: MovieID :: Tag :: Timestamp. Thẻ là siêu dữ liệu do người dùng tạo ra, mỗi thẻ thường là một từ đơn hoặc cụm từ ngắn. Ý nghĩa, giá trị và mục đích của một thẻ cụ thể được xác định bởi mỗi người dùng. Timestamp biểu thị số giây kể từ nửa đêm Giờ phối hợp quốc tế (UTC) ngày 1 tháng 1 năm 1970.

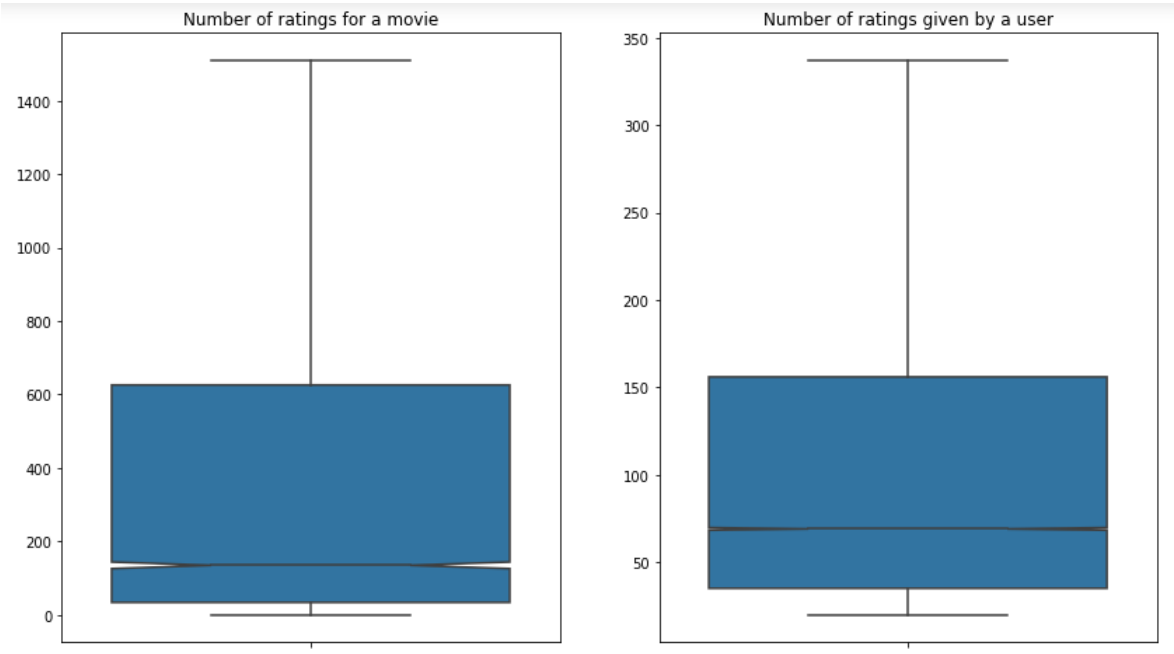
2.1.2 Phân tích thống kê cơ bản:

Có tổng cộng 10,000,054 xếp hạng của 71,567 người dùng dành cho 10,681 bộ phim. Dữ liệu xếp hạng là rất thưa thớt, độ thưa thớt của ma trận user-item là 98.7%. Xếp hạng của người dùng chủ yếu nhận giá trị trong khoảng từ 3.0 đến 4.0.



Hình 2.1: Phân bố điểm xếp hạng của người dùng

Mỗi người dùng xếp hạng ít nhất 20 bộ phim. Tuy nhiên, có khoảng 1700 bộ phim nhận được ít hơn 20 xếp hạng.



Hình 2.2: Phân bố số lượng xếp hạng của mỗi người dùng và mỗi bộ phim

2.1.3. Kỹ thuật gợi ý lai ghép (Hybrid Recommender systems):

Hầu hết các hệ thống gợi ý hiện nay sử dụng phương pháp kết hợp, kết hợp lọc cộng tác, lọc dựa trên nội dung và các phương pháp khác. Không có lý do tại sao một số kỹ thuật khác nhau cùng loại không thể được lai ghép. Phương pháp lai có thể được thực hiện theo nhiều cách: bằng cách đưa ra các dự đoán dựa trên lọc nội dung và lọc cộng tác riêng biệt và sau đó kết hợp chúng; bằng cách thêm các tính chất dựa trên lọc nội dung vào cách tiếp cận dựa trên lọc cộng tác (và ngược lại); hoặc bằng cách thống nhất các phương pháp tiếp cận thành một mô hình hoàn chỉnh. Một số nghiên cứu so sánh thực nghiệm hiệu suất của phương pháp lai ghép với các phương pháp lọc cộng tác và lọc nội dung thuần túy đã chứng minh rằng các phương pháp lai có thể cung cấp các gợi ý chính xác hơn so với các phương pháp thuần túy. Các phương pháp này cũng có thể được sử dụng để khắc phục một số vấn đề phổ biến trong các hệ thống gợi ý như cold-start problem và sự thưa thớt dữ liệu, cũng như vấn đề knowledge engineering bottleneck trong các phương pháp dựa trên tri thức.

Netflix là một ví dụ điển hình về việc sử dụng các hệ thống đề xuất lai. Trang web đưa ra gợi ý bằng cách so sánh thói quen xem và tìm kiếm của những người dùng tương tự (nghĩa là lọc cộng tác) cũng như bằng cách cung cấp các phim có chung đặc điểm với các phim mà người dùng đánh giá cao (lọc dựa trên nội dung).

Một số kỹ thuật lai bao gồm:

- Có trọng số (Weighted) : Kết hợp số điểm của các thành phần gợi ý khác nhau.

- Chuyển đổi (Switching) : Lựa chọn giữa các thành phần gợi ý và áp dụng một trong những thành phần được chọn.

- Hỗn hợp (Mixed) : Các gợi ý từ các thành phần khác nhau được kết hợp cùng nhau để đưa ra gợi ý cuối cùng.

- Kết hợp đặc trưng (Feature combination) : Các tính chất đặc trưng của dữ liệu xuất phát từ các nguồn tri thức khác nhau được kết hợp với nhau và được cung cấp cho một thuật toán gợi ý duy nhất.

- Mở rộng đặc trưng (Feature augmentation) : Tính toán một hoặc tập hợp các đặc trưng của dữ liệu, sau đó sử dụng chúng như là một phần của đầu vào cho kỹ thuật tiếp theo.

- Nối tầng (Cascade) : Các thành phần gợi ý được sử dụng với các mức độ ưu tiên khác nhau.

- Cấp độ meta (Meta-level) : Một kỹ thuật gợi ý được áp dụng và tạo ra một số loại

mô hình, sau đó là đầu vào được sử dụng bởi kỹ thuật tiếp theo.

2.1.4 Thử nghiệm kỹ thuật lai ghép với bộ dữ liệu Movielens:

➢ **Ý tưởng:**

- Ý tưởng chính đằng sau Matrix Factorization cho Recommendation Systems là tồn tại các *latent features* (tính chất ẩn) mô tả sự liên quan giữa các *items* và *users*. Ví dụ với hệ thống gợi ý các bộ phim, tính chất ẩn có thể là *hình sự, chính trị, hành động, hài*, …; cũng có thể là một sự kết hợp nào đó của các thể loại này; hoặc cũng có thể là bất cứ điều gì mà chúng ta không thực sự cần đặt tên. Mỗi *item* sẽ mang tính chất ẩn ở một mức độ nào đó, hệ số càng cao tương ứng với việc mang tính chất đó càng cao. Tương tự, mỗi *user* cũng sẽ có xu hướng thích những tính chất ẩn nào đó và được mô tả bởi các hệ số trong vector biểu diễn. Hệ số cao tương ứng với việc *user* thích các bộ phim có tính chất ẩn đó. Điều này nghĩa là *item* mang các tính chất ẩn mà *user* thích, vậy thì nên gợi ý *item* này cho *user* đó.

- Dựa vào ý tưởng trên, ta có thể sử dụng thuật toán lọc cộng tác Matrix Factorization để học những tính chất ẩn (latent features) của các bộ phim, sau đó kết hợp các đặc trưng này với các tính chất sẵn có là thể loại và năm phát hành thành một bộ dữ liệu training mới. Thuật toán lọc nội dung (Content based filtering) sẽ được thực hiện trên bộ dữ liệu này. Việc đánh giá mô hình thuật toán được thực hiện trên các tập dữ liệu validation và test như bình thường.

➢ **Xây dựng item profile:**

Dữ liệu phim ban đầu có dạng như sau:

```
In [3]: movielens.head(10)
```

Out[3]:

	movieid	title	genres
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	2	Jumanji (1995)	Adventure|Children|Fantasy
2	3	Grumpier Old Men (1995)	Comedy|Romance
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	5	Father of the Bride Part II (1995)	Comedy
5	6	Heat (1995)	Action|Crime|Thriller
6	7	Sabrina (1995)	Comedy|Romance
7	8	Tom and Huck (1995)	Adventure|Children
8	9	Sudden Death (1995)	Action
9	10	GoldenEye (1995)	Action|Adventure|Thriller

Hình 2.3: Dữ liệu phim gốc

Ta cần làm 2 việc: trích xuất thông tin về năm phát hành từ trường ‘title’ và biến đổi dữ liệu trong trường ‘genres’ (thể loại) thành dạng số.

```
In [15]: movielens
```

Out[15]:

	movieid	title	year	(no genres listed)	Action	Adventure	Animation	Children	Comedy	Crime	...	Film-Noir	Horror	IMAX	Musical	Mystery	Romanc
0	1	Toy Story	0.860215	0	0	1	1	1	1	0	...	0	0	0	0	0	
1	2	Jumanji	0.860215	0	0	1	0	1	0	0	...	0	0	0	0	0	
2	3	Grumpier Old Men	0.860215	0	0	0	0	0	1	0	...	0	0	0	0	0	
3	4	Waiting to Exhale	0.860215	0	0	0	0	0	1	0	...	0	0	0	0	0	
4	5	Father of the Bride Part II	0.860215	0	0	0	0	0	1	0	...	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	

Hình 2.4: Dữ liệu sau khi tiền xử lý và chuẩn hóa

Trường ‘year’ là dữ liệu về năm phát hành bộ phim, đã được chuẩn hóa để giá trị nằm trong khoảng [0, 1]. Các trường ‘Action’, ‘Adventure’, ‘Mystery’, ... biểu thị thể loại của bộ phim. Các trường này sẽ có giá trị bằng 1 nếu bộ phim thuộc thể loại tương ứng và có giá trị bằng 0 trong trường hợp ngược lại.

Các tính chất ẩn của bộ phim (mà ta chưa biết rõ đó là gì) đã được suy ra từ mô hình Matrix Factorization có dạng như sau (tổng cộng 10 tính chất):

```
In [17]: latent_features = pd.read_csv('item_features.csv', index_col=0)
         latent_features.head()
```

Out[17]:

	0	1	2	3	4	5	6	7	8	9
0	-0.063262	0.051241	0.337060	0.459566	-0.142059	0.042321	0.142999	0.226889	0.029189	0.511252
1	-0.258458	-0.213175	-0.066041	-0.009825	-0.380848	-0.376066	-0.042113	-0.263620	-0.149817	-0.045909
2	-0.436517	-0.128984	0.120946	-0.505279	-0.197796	-0.105685	0.107190	-0.615069	-0.221788	-0.044816
3	-0.513603	-0.159172	-0.336230	-0.601875	-0.162694	-0.560031	0.116927	-0.146628	-0.265206	-0.331519
4	-0.416224	-0.138634	-0.249614	-0.305365	-0.473933	-0.319636	0.278289	-0.337422	-0.245865	-0.289678

Hình 2.5: Tính chất ẩn của các bộ phim dưới dạng ma trận

Các giá trị của ma trận trong hình trên có giá trị phần lớn thuộc khoảng [-1, 1], điều này lý giải cho sự cần thiết của việc chuẩn hóa dữ liệu ‘year’ như đã làm trong phần trước.

Kết hợp 2 phần dữ liệu đã xử lý, ta có được 1 bộ dữ liệu huấn luyện hoàn chỉnh:

```
In [24]: movielens.head()
```

Out[24]:

	movieid	title	year	(no genres listed)	Action	Adventure	Animation	Children	Comedy	Crime	...	0	1	2	3	4	
0	1	Toy Story	0.860215	0	0	1	1	1	1	0	...	-0.063262	0.051241	0.337060	0.459566	-0.142059	0
1	2	Jumanji	0.860215	0	0	1	0	1	0	0	...	-0.258458	-0.213175	-0.066041	-0.009825	-0.380848	-0
2	3	Grumpier Old Men	0.860215	0	0	0	0	0	1	0	...	-0.436517	-0.128984	0.120946	-0.505279	-0.197796	-0
3	4	Waiting to Exhale	0.860215	0	0	0	0	0	1	0	...	-0.513603	-0.159172	-0.336230	-0.601875	-0.162694	-0
4	5	Father of the Bride Part II	0.860215	0	0	0	0	0	1	0	...	-0.416224	-0.138634	-0.249614	-0.305365	-0.473933	-0

5 rows × 33 columns

```
In [25]: movielens.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 10603 entries, 0 to 10680
Data columns (total 33 columns):
movieid              10603 non-null int64
title                10603 non-null object
year                 10603 non-null float64
```

Hình 2.6: Bộ dữ liệu huấn luyện mới

Bộ dữ liệu này có 10603 dòng tương ứng 10603 bộ phim khác nhau, với 33 cột tương đương 33 tính chất của mỗi bộ phim. Cột 'movieid' và 'title' sẽ được loại bỏ khi sử dụng để huấn luyện mô hình lọc dựa trên nội dung trong phần sau.

➢ **Huấn luyện và đánh giá mô hình:**

Thuật toán Content-based filtering: từ thông tin mô tả của item, biểu diễn item dưới dạng vec-tơ thuộc tính. Sau đó dùng các vec-tơ này để học mô hình của mỗi user, là ma trận trọng số của user với mỗi item.

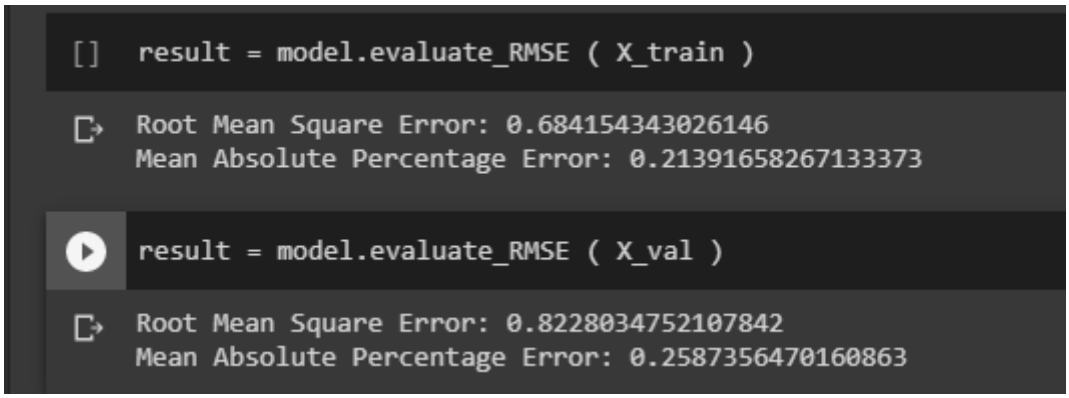
Như vậy, thuật toán content-based gồm 2 bước:

- Bước 1: Biểu diễn items dưới dạng vec-tơ thuộc tính – item profile
- Bước 2: Học mô hình của mỗi user

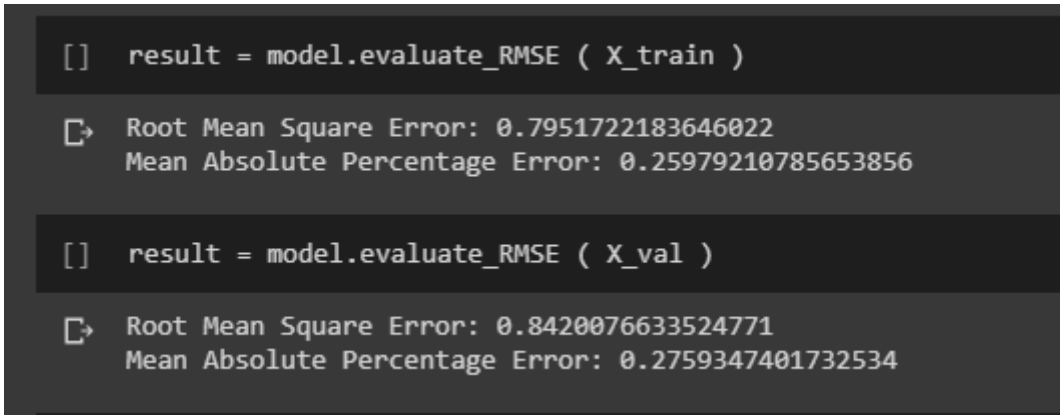
Bước 1 đã được thực hiện ở phần trước. Với bước 2, thuật toán Ridge Regression được sử dụng để học mô hình cho mỗi user.

Tập dữ liệu xếp hạng được chia thành 3 phần: training set, validation set và test set theo tỉ lệ 6:2:2. Thuật toán Content-based filtering xây dựng mô hình khi áp dụng trên tập dữ liệu training với 6,000,000 xếp hạng, sau đó mô hình được đánh giá trên tập validation và test, mỗi tập có 2,000,000 xếp hạng. Quá trình huấn luyện và đánh giá mô hình được thực hiện trên Google Colab.

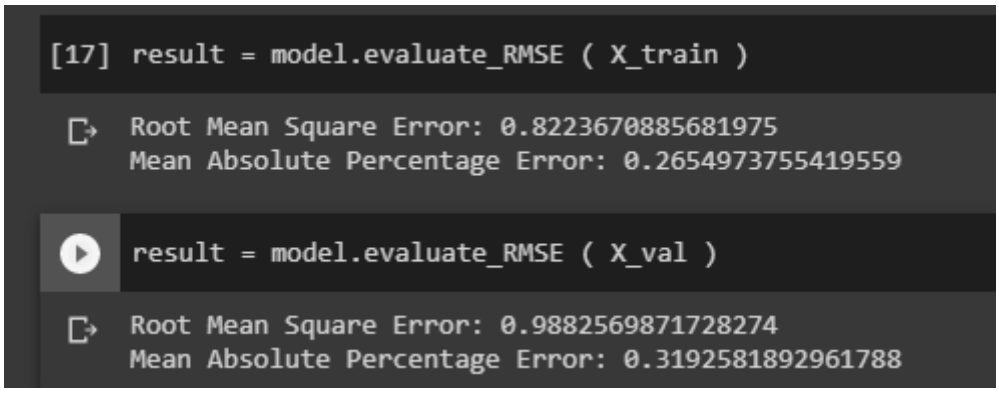
Kết quả như sau:



Hình 2.7: Kết quả khi sử dụng kỹ thuật Hybrid filtering



Hình 2.8: Kết quả khi sử dụng kỹ thuật Matrix factorization



Hình 2.9: Kết quả khi chỉ sử dụng Content-based filtering

- **Bảng so sánh kết quả đánh giá các mô hình trên tập test:**

Bảng 2.1: Bảng so sánh kết quả đánh giá các mô hình

	Root Mean Square Error	Mean Absolute Percentage Error
Content-based filtering	0.988	31.92%
Collaborative filtering (Matrix factorization)	0.842	27.59%
Hybrid filtering	0.822	25.87%

➢ **Nhận xét:**

- Thuật toán Matrix factorization và Hybrid filtering cho kết quả tốt hơn nhiều so với thuật toán Content-based filtering.

- Kết quả trên tập test khi dùng thuật toán Hybrid filtering tốt hơn một chút so với Matrix factorization.

- Mặt khác, thuật toán Hybrid filtering lại cho kết quả tốt hơn nhiều so với Matrix factorization khi đánh giá trên tập train. Do đó, kiểu hệ thống gợi ý kết hợp như trong bài vẫn bị overfitting khá nhiều.

=> Cần tìm hiểu thêm các cách kết hợp khác để cải thiện kết quả.

2.2 Hệ thống gợi ý tin tức dựa trên phiên sử dụng mạng nơ-ron sâu (News Session-Based Recommendations using Deep Neural Networks):

Các hệ thống giới thiệu tin tức có nhiệm vụ cá nhân hóa trải nghiệm của người dùng và giúp họ khám phá các bài viết có liên quan từ một không gian tìm kiếm rộng lớn và luôn biến động. Do đó, gợi ý tin tức là một lĩnh vực đầy thách thức đối với các hệ thống khuyến nghị, do hồ sơ người dùng thưa thớt, số lượng tin tức tăng nhanh và sự thay đổi sở thích nhanh chóng của người dùng.

Một số kết quả đầy hứa hẹn đã đạt được gần đây bằng cách sử dụng các kỹ thuật Deep Learning trên hệ thống gợi ý, đặc biệt cho việc trích xuất các đặc trưng của bài viết và đưa ra các đề xuất dựa trên phiên (session-based) với mạng nơ-ron hồi quy (Recurrent Neural Networks).

Bài báo "**News Session-Based Recommendations using Deep Neural Networks**" [1] đề xuất mô hình CHAMELEON - một kiến trúc học tập sâu cho các hệ thống giới thiệu tin tức. Kiến trúc này bao gồm hai mô-đun: mô-đun đầu tiên chịu trách nhiệm học các biểu diễn dưới dạng số của các bài viết, dựa trên nội dung văn bản và siêu dữ liệu của chúng (tác giả, thể loại,...) và mô-đun thứ hai nhằm cung cấp các đề xuất dựa trên phiên sử dụng Mạng nơ-ron hồi quy.

Nhiệm vụ của mô hình này là dự đoán mục tiếp theo cho các phiên truy cập của người dùng: "bài viết tiếp theo mà người dùng có khả năng đọc trong phiên là gì?"

Các thông tin về ngữ cảnh phiên truy cập của người dùng được mô hình tận dụng để cung cấp thông tin bổ sung để giải quyết vấn đề cold-start trong khuyến nghị tin tức, khi mà chưa có nhiều dữ liệu lịch sử truy cập của người dùng. Cả đặc trưng của bài viết và

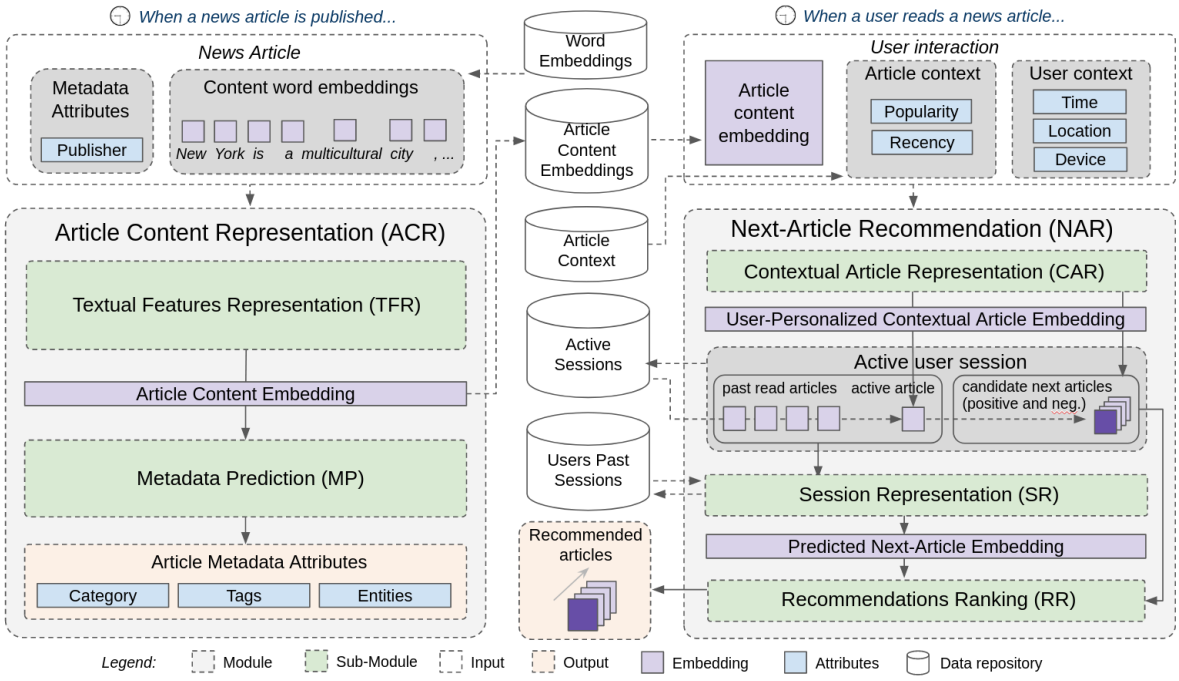
hành vi của người dùng đều được hợp nhất để thực hiện mô hình khuyến nghị theo cách tiếp cận đề xuất kết hợp (Hybrid recommendation systems).

Các thử nghiệm với nhiều phương pháp đề xuất dựa trên phiên đã được thực hiện và việc sử dụng kiến trúc CHAMELEON đã mang đến sự cải thiện đáng kể về độ chính xác và các tham số đánh giá khác (10% đối với tham số Hit Rate và 13% đối với tham số MRR) so với các phương pháp được dùng để so sánh.

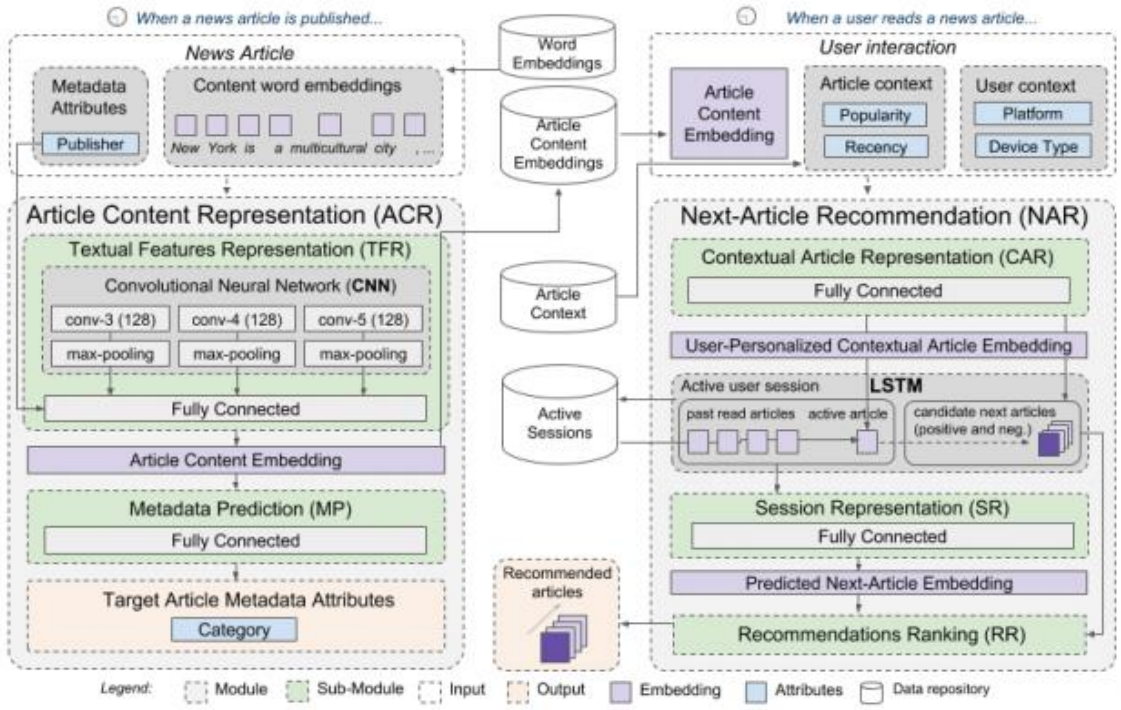
2.2.1 Giải pháp:

Bài báo đề xuất CHAMELEON – một kiến trúc meta học tập sâu cho các hệ thống giới thiệu tin tức. Kiến trúc meta là kiến trúc tham chiếu tập hợp các quyết định liên quan

đến một chiến lược kiến trúc chung. Nó có thể được khởi tạo như các kiến trúc khác nhau với các đặc điểm tương tự để hoàn thành một nhiệm vụ chung, trong trường hợp này là hệ khuyến nghị tin tức.



Hình 2.10: Kiến trúc Chameleon (1)



Hình 2.11: Kiến trúc Chameleon (2)

Như đã được mô tả trong Hình 1, CHAMELEON bao gồm hai mô-đun, với vòng đời độc lập để đào tạo (training) và suy luận (learning): Mô-đun Article Content Representation (ACR) và mô-đun Next Article Recommendation (NAR).

2.2.2 Article Content Representation (ACR)

Mô-đun ACR chịu trách nhiệm trích xuất các đặc trưng từ văn bản bài viết và siêu dữ liệu, sau đó tìm ra **một biểu diễn phân tán (embeddings) cho từng bối cảnh bài viết tin tức.** Các đầu vào cho mô-đun ACR là (1) thuộc tính siêu dữ liệu của bài viết (ví dụ: nhà xuất bản) và (2) nội dung văn bản của bài viết, được biểu diễn dưới dạng một chuỗi các từ nhúng (word embeddings).

Một phương pháp phổ biến trong Xử lý ngôn ngữ tự nhiên (NLP) là training trước các từ nhúng bằng cách sử dụng các phương thức như Word2Vec và GloVe trong một kho văn bản lớn hơn (ví dụ: Wikipedia). Trong phần khởi tạo của mô-đun phụ Textual Features Representation (TFR) từ mô đun ACR, các CNN 1D đã được sử dụng để trích xuất các đặc trưng từ các nội dung văn bản.

Các đặc trưng của văn bản và các đầu vào siêu dữ liệu được kết hợp bằng cách sử dụng một chuỗi các lớp nơ-ron được kết nối đầy đủ (Fully connected) để tạo ra **các biểu diễn cho nội dung bài viết.**

Các biểu diễn cho nội dung bài viết sau khi được huấn luyện sẽ được lưu trữ trong một kho lưu trữ, để sau này được sử dụng bởi mô-đun NAR.

2.2.3 Next-Article Recommendation (NAR)

Mô-đun NAR chịu trách nhiệm cung cấp các đề xuất tin tức cho các phiên hoạt động. Do mức độ thưa thớt của người dùng và sự thay đổi sở thích liên tục của họ, mô hình này chỉ sử dụng thông tin theo ngữ cảnh dựa trên phiên hoạt đọng, bỏ qua các phiên hoạt động trong quá khứ của người dùng.

Các đầu vào cho mô-đun NAR là: (1) Biểu diễn nội dung bài viết đã được huấn luyện trước của bài viết vừa được xem bởi người dùng; (2) các thuộc tính theo ngữ cảnh của bài viết (mức độ phổ biến và những lần truy cập gần đây); và (3) bối cảnh của người dùng (ví dụ: thời gian, địa điểm và thiết bị truy cập). Các đầu vào này được kết hợp bởi các lớp nơ-ron được kết nối đầy đủ để tạo ra **một biểu diễn bài viết theo ngữ cảnh được cá nhân hóa bởi người dùng (User-Personalized Contextual Article Embedding).** Có thể có các cách biểu diễn khác nhau cho cùng một bài viết, tùy thuộc vào bối cảnh người dùng và bối cảnh bài viết hiện tại (mức độ phổ biến và lần truy cập gần đây).

Mô-đun NAR sử dụng một loại mô hình RNN – Long-Short Term Memory (LSTM) - để mô hình hóa chuỗi bài viết mà người dùng đọc trong các phiên của họ, được thể hiện bằng các biểu diễn bài viết theo ngữ cảnh được cá nhân hóa của họ. Đối với mỗi bài viết trong chuỗi, RNN đưa ra một biểu diễn bài viết theo ngữ cảnh – biểu diễn của một nội dung tin tức mà được dự đoán sẽ được đọc tiếp theo bởi người dùng trong phiên hoạt động.

Trong hầu hết các kiến trúc học tập sâu được đề xuất cho hệ gợi ý, mạng nơ-ron sẽ có đầu ra là một vectơ có số chiều là số lượng vật phẩm (item) có sẵn. Cách tiếp cận như vậy là hiệu quả đối với các lĩnh vực mà số vật phẩm là ổn định, như phim và sách. Mặc dù, trong hoàn cảnh thay đổi liên tục của các hệ khuyến nghị tin tức, hàng ngàn trong số các bài viết được thêm vào và loại bỏ hàng ngày, cách tiếp cận như vậy có thể yêu cầu huấn luyện lại toàn bộ mạng nơ-ron, một cách thường xuyên ngay khi các bài viết mới được xuất bản.

Vì lý do này, thay vì sử dụng hàm mất mát là softmax cross entropy, mô-đun NAR được huấn luyện để tối đa hóa sự tương đồng giữa biểu diễn bài viết theo ngữ cảnh được dự đoán và biểu diễn bài viết theo ngữ cảnh tương ứng với bài viết tiếp theo mà người dùng thực sự đọc trong phiên của mình (positive sample), trong khi giảm thiểu sự tương

đồng của nó với các negative samples (các bài viết không được người dùng đọc trong phiên). Với chiến lược này, một bài viết mới được xuất bản có thể được đề xuất ngay lập tức, ngay khi biểu diễn ngữ cảnh theo nội dung của nó (Article Content Embeddings) được huấn luyện và thêm vào kho lưu trữ.

2.2.4 Thử nghiệm và đánh giá:

- **Dữ liệu:**

Thử nghiệm và đánh giá mô hình được thực hiện trên một bộ dữ liệu độc quyền đã được Globo.com cung cấp. Globo.com là cổng thông tin phổ biến nhất ở Brazil, với hơn 80 triệu người dùng và 100.000 nội dung mới mỗi tháng. Mẫu dữ liệu chứa các tương tác của người dùng từ ngày 1 đến 16 tháng 10 năm 2017, bao gồm hơn 3 triệu lượt tương tác (click), được phân bố trong 1,2 triệu phiên hoạt động từ 330.000 người dùng đối với hơn 50.000 bài báo khác nhau trong khoảng thời gian đó.

Trong bộ dữ liệu Globo.com, một phiên hoạt động biểu thị một chuỗi các lần nhấp của người dùng với không quá 30 phút giữa các tương tác. Để huấn luyện mô-đun NAR, các chuỗi tương tác của người dùng được nhóm theo phiên và được sắp xếp theo thời gian xảy ra. Các phiên chỉ có 1 tương tác (không có tác dụng cho việc dự đoán lần nhấp chuột tiếp theo) và với hơn 20 tương tác (người dùng đặc biệt - outliers hoặc cũng có thể là bot) đã bị loại bỏ.

- **Tham số đánh giá:**

- Top-N recommender systems: Các hệ thống giới thiệu Top-N có ở khắp mọi nơi từ các trang web mua sắm trực tuyến đến các cổng video. Hệ thống cung cấp cho người dùng một danh sách được xếp hạng gồm N mặt hàng mà họ có thể sẽ quan tâm, để khuyến khích lượt xem và mua hàng.

- Hit Rate (HR): nếu người dùng tương tác với một trong các sản phẩm được đề xuất, chúng ta xem xét nó là một "hit". Lấy tổng số "hit" chia cho tổng số lần nhấp chuột của các người dùng, ta được tham số Hit Rate.

- Mean Reciprocal Rank (MRR): Thứ hạng đối ứng trung bình là một thước đo thống kê để đánh giá bất kỳ quy trình nào tạo ra danh sách các câu trả lời có thể có cho một mẫu truy vấn, được sắp xếp theo xác suất chính xác. Thứ hạng đối ứng của một phản hồi truy vấn là nghịch đảo nhân của thứ hạng của câu trả lời đúng đầu tiên: 1 cho vị trí thứ nhất, 1⁄2 cho vị trí thứ hai, 1⁄3 cho vị trí thứ ba, v.v. Xếp hạng đối ứng trung bình là trung bình của các cấp kết quả đối ứng cho một mẫu truy vấn Q:

$$\mathrm{MRR} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{\mathrm{rank}_i}.$$

Ví dụ:

Query	Proposed Results	Correct response	Rank	Reciprocal rank
cat	catten, cati, **cats**	cats	3	1/3
tori	torii, **tori**, toruses	tori	2	1/2
virus	**viruses**, virii, viri	viruses	1	1

Cho ba mẫu dữ liệu trên, chúng ta có thể tính thứ hạng đối ứng trung bình là: MRR = (1/3 + 1/2 + 1) / 3 = 11/18 hoặc khoảng 0,61.

Đối với một hệ thống Top-N recommender systems, ta có các ký hiệu tham số tương ứng:

+ HR@N: kiểm tra xem mục đã được nhấp vào của người dùng có hiện diện trong N mục được xếp hạng hàng đầu không.

+ MRR@N: tương tự như trên.

Trong thử nghiệm này, **N được chọn bằng 5**, tức là gợi ý danh sách gồm 5 bài viết cho người dùng.

➢ **Các phương pháp dùng để đối chiếu kết quả (baseline methods):**

Đối với thử nghiệm này, một số mô hình thuật toán khuyến nghị dựa trên phiên sẽ được sử dụng để so sánh.

GRU4Rec - Kiến trúc bán nơ-ron sử dụng RNN cho các đề xuất dựa trên phiên.

Co-occurrent - Đề xuất các bài viết thường được xem cùng với bài viết vừa đọc, trong các phiên của người dùng khác. Thuật toán này là phiên bản đơn giản hóa của kỹ thuật quy tắc kết hợp (Association Rule), với kích thước quy tắc tối đa là hai bài viết cùng được đọc.

Sequential Rules (SR) - Một phiên bản tốt hơn của các quy tắc kết hợp, xem xét chuỗi các mục được nhấp trong phiên. Một quy tắc được tạo ra khi một mục q xuất hiện sau một mục p trong phiên, ngay cả khi các mục khác được xem giữa p và q.

Item-kNN - Trả về k mục tương tự với bài viết đã đọc gần đây nhất, sử dụng độ đo tương tự Cosin.

Vector Multiplication Session-Based kNN (V-SkNN) - So sánh toàn bộ phiên hoạt động với các phiên trước đây và tìm các mục có thể được đề xuất.

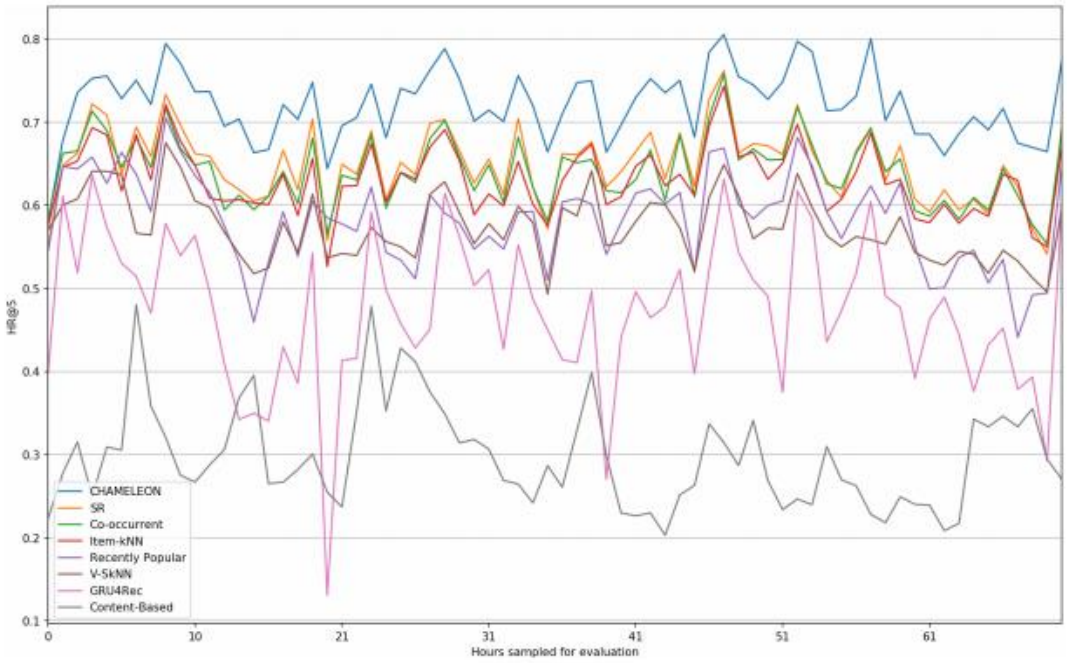
Recently Popular - Đề xuất các bài viết được xem nhiều nhất từ N lần nhấp vào gần đây nhất.

Content-Based - Đối với mỗi bài viết đọc bởi người sử dụng, khuyến cáo bài viết tương tự dựa trên sự tương đồng giữa các vectơ A Content Embeddings, từ N lần nhấp chuột gần đây nhất.

- **Phương pháp thực hiện quá trình thử nghiệm: huấn luyện và đánh giá liên tục mỗi năm giờ đồng hồ, trong 15 ngày (từ ngày 1 đến 15 tháng 10 năm 2017).**
- **Kết quả**

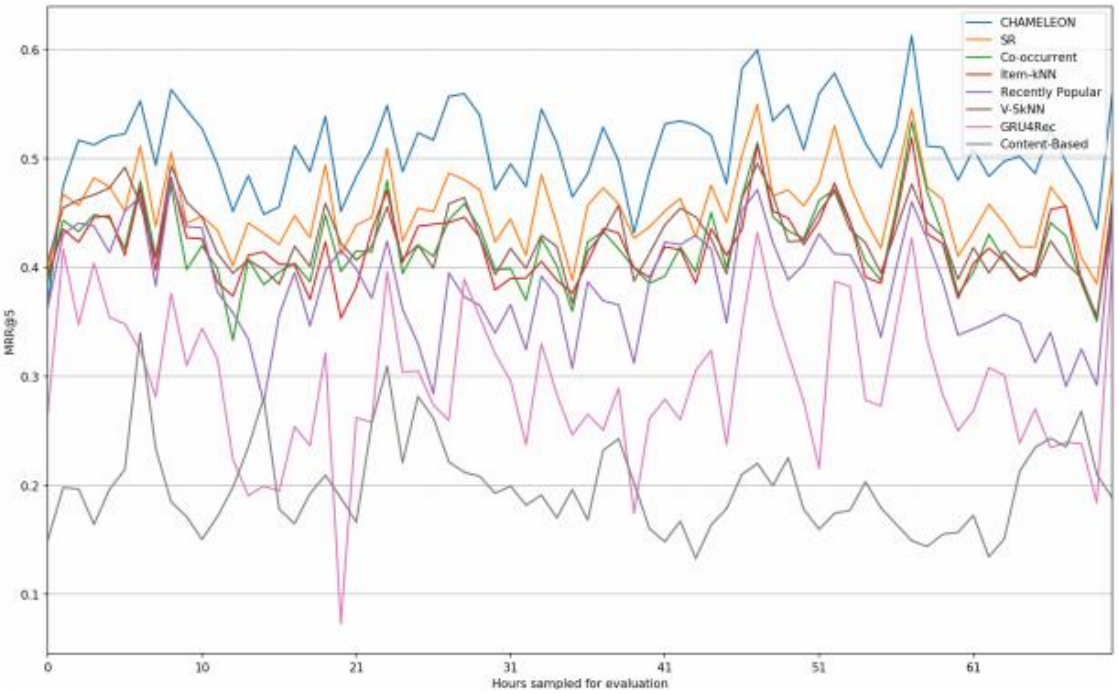
*** Kết quả trong bài báo:**

- **HR@5:**



Hình 2.12: HR@5 trung bình: 0.72

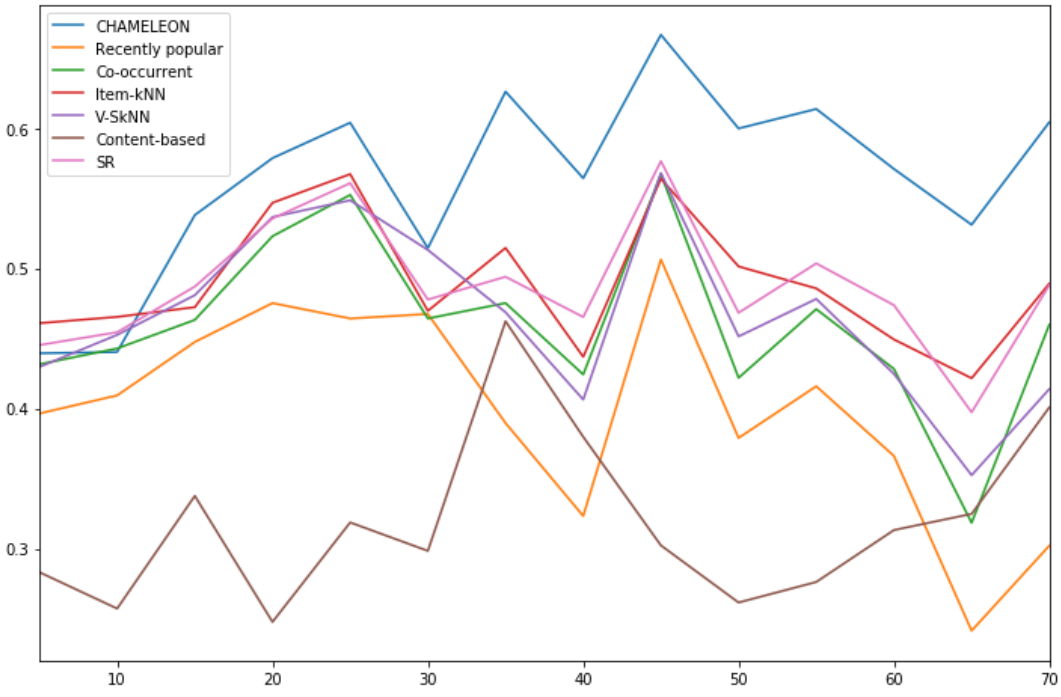
- **MRR@5:**



Hình 2.13: MRR@5 trung bình: 0.51

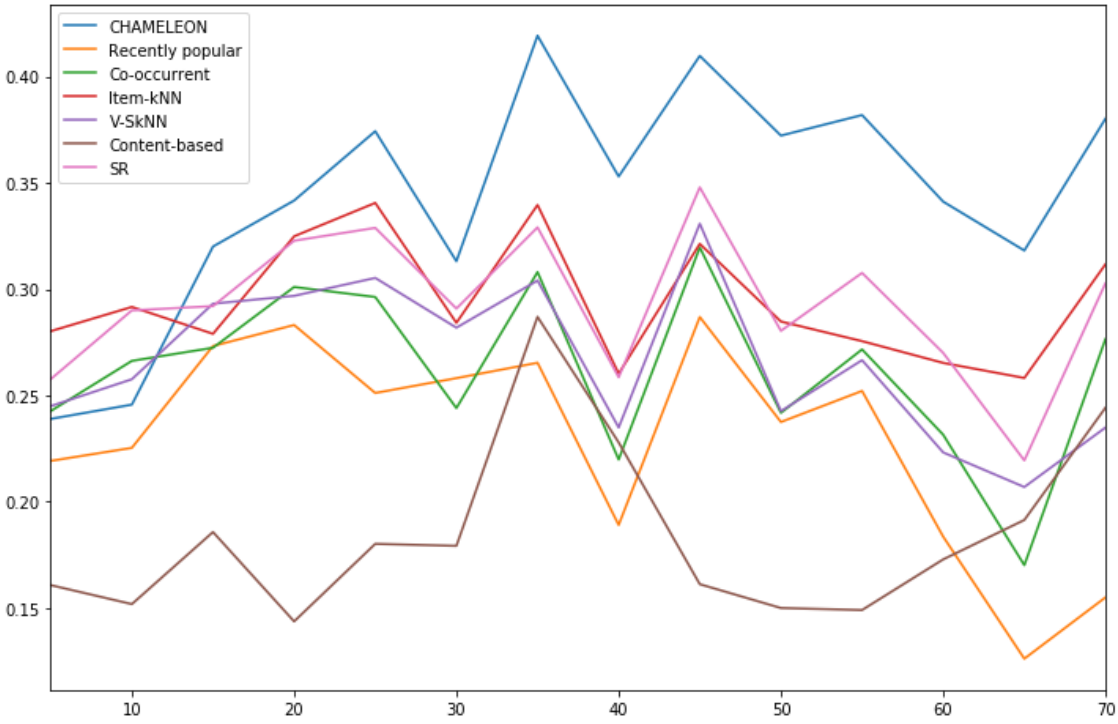
*** Kết quả khi thực hiện lại thuật toán:**

- **HR@5:**



Hình 2.14: HR@5 trung bình: 0.58

- **MRR@5:**



Hình 2.15: MRR@5 trung bình: 0.35

➢ **Nhận xét:**

- Khi thực hiện lại thử nghiệm, mô hình Chameleon vẫn cho kết quả tốt hơn so với các mô hình thuật toán khác: HR@5 trung bình > 11% và MRR@5 trung bình > 8% khi so sánh với hình tốt nhất còn lại là SR.

- Dạng đồ thị kết quả của lần thực hiện trong bài báo và lần thực hiện lại là tương đương nhau.

- Tuy nhiên, kết quả này vẫn kém một chút so với kết quả của mô hình Chameleon trong bài báo: HR@5 trung bình là 0.58 < 0.72, và MRR@5 trung bình là 0.35 < 0.51.

=> Cần phải tối ưu các tham số được thiết lập ban đầu (Hyperparameter Tuning) để thu được kết quả tốt hơn.

KẾT LUẬN

Thông qua việc thực hiện đề tài **"Ứng dụng hệ thống gợi ý trong lĩnh vực thương mại điện tử"**, em đã tính lũy được rất nhiều kiến thức thực tế cũng như lý thuyết về chuyên ngành Điện tử – Viễn thông và lĩnh vực Công nghệ thông tin, cô giáo hướng dẫn đã tạo cho chúng em niềm say mê học tập, tìm tòi những kiến thức mới. Cô còn giúp em hoàn thiện các kỹ năng mềm như kỹ năng thuyết trình, làm việc nhóm, làm việc trong các môi trường chuyên nghiệp Power Point, các phần mềm lập trình, triển khai Machine learning: Python, Pandas, Sublime Text, Colab Notebooks, ...

Do vốn kiến thức còn hạn hẹp nên việc thực hiện ý tưởng còn nhiều hạn chế. Nếu còn có gì sai sót, em mong cô giúp đỡ và tạo điều kiện để em có thể hoàn thành một cách tốt nhất ý tưởng này.

Em xin chân thành cảm ơn!

TÀI LIỆU THAM KHẢO

[1] https://github.com/gabrielspmoreira/chameleon_recsys

[2] https://machinelearningcoban.com/

[3] https://vi.wikipedia.org/