BỘ GIÁO DỤC VÀ ĐÀO TẠO **BỘ NÔNG NGHIỆP VÀ PTNT**

TRƯỜNG ĐẠI HỌC THỦY LỢI

Nhóm 11:

Nguyễn Hà Phương Uyên - 65KTPM

Nguyễn Thị Phương Thảo - 65KTPM

XÂY DỰNG MÔ HÌNH GỢI Ý PHIM DỰA TRÊN HỌC MÁY LAI

BÀI TẬP LỚN MÔN HỌC HỌC MÁY

HÀ NỘI, NĂM 2025

BỘ GIÁO DỤC VÀ ĐÀO TẠO **BỘ NÔNG NGHIỆP VÀ PTNT**

TRƯỜNG ĐẠI HỌC THỦY LỢI

Nhóm 11:

Nguyễn Hà Phương Uyên - 65KTPM

Nguyễn Thị Phương Thảo - 65KTPM

XÂY DỰNG MÔ HÌNH GỢI Ý PHIM DỰA TRÊN HỌC MÁY LAI

Lớp môn học: 65TTNT

GIẢNG VIÊN HƯỚNG DẪN: Tạ Quang Chiểu

HÀ NỘI, NĂM 2025

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

----------★----------

NHIỆM VỤ BÀI TẬP LỚN
MÔN HỌC HỌC MÁY

Tên nhóm: Nhóm 11

Môn học: Học máy

Khoa: Công nghệ thông tin

Lớp môn học: 65TTNT

Thông tin thành viên nhóm:

Tên thành viên	Mã sinh viên	Lớp quản lý
Nguyễn Thị Phương Thảo	2351170632	65KTPM
Nguyễn Hà Phương Uyên	2351170621	65KTPM

Tên đề tài: Xây dựng mô hình gợi ý phim dựa trên học máy lai

Tóm tắt:

Trong nghiên cứu này, chúng tôi đề xuất một mô hình gợi ý phim lai (Hybrid Recommendation System) kết hợp giữa Collaborative Filtering (CF) và Content-Based Filtering (CBF) theo hai cơ chế cascade và weighted hybrid nhằm nâng cao độ chính xác của quá trình gợi ý.

Người dùng được phân cụm bằng thuật toán K-Means, giúp giảm không gian tìm kiếm khi áp dụng KNN-based Collaborative Filtering. Mức độ tương đồng giữa người dùng được tính bằng Adjusted Cosine Similarity, từ đó xác định nhóm người dùng lân cận

và dự đoán điểm đánh giá cho các phim chưa xem thông qua Weighted Average Prediction.
Đồng thời, mô hình Content-Based Filtering được xây dựng dựa trên hồ sơ thể loại (genre profile) của từng người dùng, tính toán điểm nội dung (Content Score) cho mỗi phim bằng cách trung bình hóa điểm thể loại tương ứng. Cuối cùng, hai thành phần CF và CBF được kết hợp bằng phương pháp Weighted Hybrid.

Kết quả thực nghiệm cho thấy mô hình lai này đạt được sự cân bằng giữa khả năng học từ dữ liệu hành vi người dùng và khả năng hiểu ngữ nghĩa nội dung phim, giúp cải thiện chất lượng gợi ý so với các phương pháp đơn lẻ.

Nhiệm vụ từng thành viên:

Tên thành viên	**Nhiệm vụ**
Nguyễn Thị Phương Thảo	● Thu thập dữ liệu, Tiền xử lý dữ liệu ● Áp dụng thuật toán K-Means để phân cụm người dùng ● Tính toán độ tương đồng giữa người dùng ● Lựa chọn tập người dùng lân cận (KNN) ● Slide thuyết trình và tổng hợp nội dung báo cáo liên quan đến các bước trên.
Nguyễn Hà Phương Uyên	● Dự đoán điểm đánh giá cho phim trong CF ● Content-Based Filtering (CBF) dựa trên thể loại phim. ● Kết hợp giữa CF và CBF thành hybrid model. ● Hyper Tuning siêu tham số ● Đánh giá mô hình ● Slide thuyết trình và tổng hợp nội dung báo cáo liên quan đến các bước trên.

1. NỘI DUNG CÁC PHẦN THUYẾT MINH VÀ TÍNH TOÁN: Tỷ lệ %

Nội dung các phần	Tỷ lệ %
Chương 1: Tổng quan về bài toán	10%
Chương 2: Cơ sở lý thuyết mô hình học máy lai trong gợi ý phim	20%
Chương 3: Xây dựng mô hình gợi ý phim	40%
Chương 4: Đánh giá mô hình gợi ý phim dựa trên học máy lai.	30%

2. GIÁO VIÊN HƯỚNG DẪN TỪNG PHẦN

Phần	Họ và tên giáo viên hướng dẫn
Chương 1: Tổng quan về bài toán	Tạ Quang Chiểu
Chương 2: Cơ sở lý thuyết mô hình học máy lai trong gợi ý phim	Tạ Quang Chiểu
Chương 3: Xây dựng mô hình gợi ý phim	Tạ Quang Chiểu
Chương 4: Đánh giá mô hình gợi ý phim dựa trên học máy lai.	Tạ Quang Chiểu

LỜI CAM ĐOAN

Nhóm tác giả xin cam đoan đây là bài tập lớn của bản thân nhóm tác giả. Các kết quả trong bài tập lớn này là trung thực, và không sao chép từ bất kỳ một nguồn nào và dưới bất kỳ hình thức nào.Việc tham khảo các nguồn tài liệu (nếu có) đã được thực hiện trích dẫn và ghi nguồn tài liệu tham khảo đúng quy định.

NHÓM TÁC GIẢ BTL

Chữ ký Chữ ký

Nguyễn Hà Phương Uyên Nguyễn Thị Phương Thảo

LỜI CẢM ƠN

Kính thưa quý thầy cô,

Lời đầu tiên, nhóm chúng em xin được gửi lời cảm ơn chân thành đến thầy giáo Tạ Quang Chiểu, giảng viên trực tiếp giảng dạy và hướng dẫn chúng em trong suốt quá trình học tập môn học máy.

Để hoàn thành bài tập lớn này, chúng em đã nhận được sự chỉ bảo tận tình, những góp ý chuyên môn sâu sắc và sự động viên kịp thời từ thầy. Thầy không chỉ giúp chúng em củng cố nền tảng kiến thức mà còn định hướng rõ ràng phương pháp thực hiện đề tài, giúp chúng em vượt qua những khó khăn trong quá trình vận dụng lý thuyết vào giải quyết vấn đề. Chúng em thật sự trân trọng sự đồng hành và những bài học quý giá mà thầy đã dành cho chúng em.

Dù đã nỗ lực hết mình, song do hạn chế về thời gian và kinh nghiệm, bài báo cáo không thể tránh khỏi những thiếu sót. Chúng em kính mong nhận được những ý kiến đóng góp quý báu và sự chỉ bảo thêm từ thầy để bài làm được hoàn thiện hơn, cũng như rút kinh nghiệm cho các dự án học tập trong tương lai.

Chúng em xin chân thành cảm ơn!

MỤC LỤC

DANH MỤC CÁC HÌNH ẢNH

DANH MỤC BẢNG BIỂU

DANH MỤC CÁC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ

Từ viết tắt	Thuật ngữ	Giải thích
RS	Recommendation System	Hệ thống Gợi ý
CF	Collaborative Filtering	Lọc Cộng tác
CBF	Content-Based Filtering	Lọc Dựa trên Nội dung
UCF	User-based CF	Lọc Cộng tác Dạng Người dùng
KNN	K-Nearest Neighbors	Thuật toán K láng giềng gần nhất
RMSE	Root Mean Square Error	Sai số căn bậc hai trung bình
MAE	Mean Absolute Error	Sai số tuyệt đối trung bình

MỞ ĐẦU

Trong những năm gần đây, cùng với sự phát triển mạnh mẽ của các nền tảng giải trí trực tuyến như Netflix, YouTube, Disney+ hay các trang xem phim trực tuyến, lượng dữ liệu người dùng phát sinh mỗi ngày ngày càng khổng lồ. Mỗi cá nhân đều có sở thích, hành vi và thói quen xem phim khác nhau, dẫn đến nhu cầu được cá nhân hoá trải nghiệm ngày càng cao. Tuy nhiên, sự đa dạng của hàng ngàn bộ phim với nhiều thể loại, quốc gia và năm phát hành đã khiến người dùng gặp khó khăn trong việc lựa chọn nội dung phù hợp, hiện tượng thường được gọi là "quá tải thông tin" (information overload).

Để giải quyết vấn đề này, hệ thống gợi ý (Recommendation System) đã được phát triển và ứng dụng rộng rãi trong các nền tảng số nhằm tự động phân tích hành vi người dùng và đề xuất các nội dung phù hợp nhất. Hệ thống gợi ý đóng vai trò quan trọng trong việc nâng cao trải nghiệm người dùng, tăng mức độ tương tác, đồng thời giúp doanh nghiệp tối ưu hóa hiển thị và hiệu quả kinh doanh.

Tuy nhiên, các phương pháp gợi ý truyền thống như lọc cộng tác (Collaborative Filtering – CF) và lọc dựa trên nội dung (Content-Based Filtering – CBF) vẫn còn tồn tại những hạn chế nhất định. CF thường gặp vấn đề cold-start khi xuất hiện người dùng hoặc phim mới, cũng như data sparsity khi ma trận đánh giá quá thưa thớt. Trong khi đó, CBF lại dễ dẫn đến quá chuyên môn hóa (over-specialization), chỉ gợi ý những phim tương tự với các phim người dùng từng xem, làm giảm khả năng khám phá nội dung mới.

Trước những hạn chế trên, hệ thống gợi ý lai (Hybrid Recommender System) ra đời như một hướng tiếp cận hiệu quả nhằm kết hợp ưu điểm và khắc phục nhược điểm của các phương pháp đơn lẻ. Bằng cách phối hợp giữa lọc cộng tác và lọc dựa trên nội dung, mô hình lai có khả năng nâng cao độ chính xác, tăng khả năng đa dạng hóa và cải thiện tính cá nhân hóa trong kết quả gợi ý.

Trong báo cáo này, nhóm tác giả tập trung xây dựng mô hình gợi ý phim dựa trên học máy lai (Hybrid Machine Learning Model) kết hợp giữa User-based Collaborative Filtering (UCF) và Content-Based Filtering (CBF) theo kiến trúc Cascade được đề xuất bởi Burke (2002). Ngoài ra, mô hình còn tích hợp thuật toán K-Means để phân cụm người dùng theo đặc trưng lai và KNN-based CF để tìm nhóm lân cận trong từng cụm. Kết quả gợi ý được kết hợp tuyến tính giữa điểm dự đoán từ CF và điểm nội dung từ CBF, nhằm đạt hiệu năng tối ưu trong việc cá nhân hóa đề xuất phim.

Hệ thống được đánh giá bằng các thước đo chuẩn trong học máy gồm RMSE (Root Mean Square Error) và MAE (Mean Absolute Error) cho dự đoán rating. Bố cục của bài tập lớn gồm 4 chương chính:

- Chương 1: Tổng quan về bài toán xây dựng mô hình gợi ý phim dựa trên học máy lai.
- Chương 2: Cơ sở lý thuyết về hệ thống gợi ý và các phương pháp học máy liên quan.
- Chương 3: Thiết kế và xây dựng mô hình gợi ý phim dựa trên học máy lai.
- Chương 4: Đánh giá mô hình gợi ý phim dựa trên học máy lai.

CHƯƠNG 1. TỔNG QUAN VỀ BÀI TOÁN XÂY DỰNG MÔ HÌNH GỢI Ý PHIM DỰA TRÊN HỌC MÁY LAI

Tóm tắt chương 1:

Chương này trình bày tổng quan về bài toán xây dựng hệ thống gợi ý phim lai (Hybrid Recommender Systems), thuộc kiến trúc Cascade theo phân loại của Burke (2002) [3], trong đó lọc cộng tác dạng người dùng (User-based Collaborative Filtering – UCF) là giai đoạn chính, được bổ trợ bởi lọc dựa trên nội dung (Content-Based Filtering – CBF) ở giai đoạn tiền xử lý và lọc lại top phim.

Mục tiêu là khắc phục cold-start, sparsity và tăng độ chính xác gợi ý — các vấn đề cốt lõi được chỉ ra trong khảo sát hệ thống của Çano & Morisio (2017) [2]. Chương cũng giới thiệu dữ liệu đầu vào, phương pháp đề xuất, các kỹ thuật học máy được áp dụng và thước đo đánh giá hiệu suất.

1.1 Mô tả bài toán

Trong kỷ nguyên số, lượng dữ liệu trên các nền tảng giải trí trực tuyến tăng trưởng theo cấp số nhân. Ngành công nghiệp điện ảnh – với sự đa dạng về thể loại, quốc gia, phong cách – đang đối mặt với hiện tượng "quá tải thông tin" (information overload). Người dùng thường lúng túng khi lựa chọn giữa hàng ngàn bộ phim, dẫn đến giảm trải nghiệm và tương tác.

Hệ thống gợi ý (Recommendation Systems – RS) ra đời nhằm cá nhân hóa nội dung, giúp người dùng khám phá phim phù hợp và hỗ trợ nền tảng tối ưu hóa hiển thị. Tuy nhiên, các phương pháp truyền thống tồn tại hạn chế cố hữu:

- Lọc cộng tác (Collaborative Filtering – CF) dựa trên sự tương đồng về hành vi giữa người dùng hoặc vật phẩm. Dù mang lại độ chính xác cao khi dữ liệu phong phú, CF gặp phải vấn đề khởi đầu lạnh (cold start) khi xuất hiện người dùng hoặc vật phẩm mới, và tính thưa thớt dữ liệu (data sparsity) khi ma trận đánh giá user–item có tỷ lệ giá trị trống lớn.
- Lọc dựa trên nội dung (Content-Based – CB) lại phụ thuộc vào mô tả đặc trưng của vật phẩm, gợi ý các phim tương tự với những gì người dùng đã yêu thích.

Phương pháp này khắc phục được cold start cho vật phẩm nhưng lại dễ rơi vào tình trạng quá chuyên môn hóa (over-specialization), hạn chế khả năng gợi ý những phim mang yếu tố khám phá mới.

Theo Çano & Morisio (2017), hệ thống gợi ý lai (Hybrid RS) là xu hướng tất yếu để kết hợp ưu điểm và giảm thiểu nhược điểm của cả hai phương pháp trên [2].

Dữ liệu đầu vào của mô hình bao gồm:

- User Ratings: Điểm đánh giá (1–5) từ người dùng đối với phim
- Movie Data: Thông tin đặc trưng: thể loại, năm phát hành, đánh giá trung bình

Nguồn dữ liệu: MovieLens Latest-Small (~100.000 ratings, 610 users, 9.742 phim) — được sử dụng trong 72% các nghiên cứu về RS [2].

Mô hình đề xuất thuộc lớp Hybrid Recommender Systems theo phân loại của Burke (2002) [3], cụ thể là kiến trúc Cascade (CF → CB), trong đó lọc cộng tác dạng người dùng (User-based Collaborative Filtering – UCF) là giai đoạn đầu tiên để dự đoán điểm đánh giá, sau đó lọc dựa trên nội dung (Content-Based Filtering – CBF) được sử dụng ở giai đoạn thứ hai nhằm lọc và sắp xếp lại top phim, từ đó tăng tính cá nhân hóa và khả năng khám phá.

Cụ thể, mô hình áp dụng phương pháp lọc cộng tác (CF) kết hợp với lọc dựa trên nội dung (CBF), đồng thời tích hợp thuật toán K-Means để phân cụm người dùng theo đặc trưng lai (hybrid features) và KNN-based CF để xác định nhóm lân cận (neighborhood) trong cùng cụm. Độ tương đồng giữa người dùng được tính bằng Adjusted Cosine Similarity — loại bỏ bias của phim — trong khi điểm dự đoán được ước lượng bằng Weighted Average Prediction.

Để tăng tính cá nhân hóa, mô hình xây dựng hồ sơ thể loại người dùng (User Content Profile) dựa trên lịch sử đánh giá, từ đó tính Content Score cho từng phim. Điểm lai cuối cùng (Final Hybrid Score) được hình thành bằng kết hợp tuyến tính có trọng số giữa điểm dự đoán từ UCF (CF Score) và Content Score.

Hiệu quả của mô hình được đánh giá nghiêm ngặt với các thước đo chuẩn trong học máy: RMSE (Root Mean Square Error) và MAE (Mean Absolute Error) đảm bảo tính định lượng cao và khả năng so sánh với các mô hình khác.

1.2 Tình hình nghiên cứu quốc tế và trong nước hiện nay

1.2.1 Nghiên cứu quốc tế

Lĩnh vực hệ thống gợi ý đã trải qua nhiều giai đoạn phát triển, từ các phương pháp thống kê cổ điển đến các mô hình học sâu hiện đại.

Trong giai đoạn đầu, hai hướng tiếp cận chủ đạo là lọc cộng tác (CF) và lọc dựa trên nội dung (CB) chiếm ưu thế. Cột mốc quan trọng của lĩnh vực là cuộc thi Netflix Prize (2006–2009), nơi các mô hình phân rã ma trận (Matrix Factorization – MF) như Singular Value Decomposition (SVD) chứng minh khả năng phát hiện các đặc trưng ẩn (latent features) hiệu quả, giúp cải thiện độ chính xác trong dự đoán đánh giá của người dùng.

Bước sang thập kỷ gần đây, xu hướng nghiên cứu quốc tế tập trung vào hai hướng chính:

- Hệ thống gợi ý lai (Hybrid Systems):
  Các nghiên cứu tìm cách tích hợp nhiều phương pháp – như CF + CB, MF + CB hoặc CF + Deep Learning – nhằm tận dụng ưu thế và giảm thiểu nhược điểm của từng kỹ thuật. Các phương pháp lai có thể được thực hiện theo nhiều cơ chế như lai trọng số (weighted hybrid), lai xếp tầng (stacked hybrid) hoặc lai đặc trưng (feature-level hybrid). Nhiều công trình đã chứng minh các mô hình này đạt độ chính xác và độ đa dạng cao hơn đáng kể so với các mô hình đơn lẻ.
- Ứng dụng học sâu (Deep Learning):
  Sự xuất hiện của các mô hình học sâu đã mở ra hướng nghiên cứu mới cho hệ thống gợi ý. Các kiến trúc như MLP (Multi-Layer Perceptron), CNN (Convolutional Neural Network) và RNN/LSTM (Recurrent Neural Network) được ứng dụng để xử lý dữ liệu đa phương thức (hình ảnh, văn bản, chuỗi hành vi). Các mô hình tiêu biểu như Neural Collaborative Filtering (NCF) hay Wide

& Deep Learning của Google đã chứng minh khả năng mô hình hóa các mối quan hệ phi tuyến phức tạp giữa người dùng và vật phẩm, vượt xa giới hạn của MF truyền thống.

1.2.2 Nghiên cứu trong nước

Tại Việt Nam, việc ứng dụng hệ thống gợi ý đang dần phổ biến trong các nền tảng thương mại điện tử (Tiki, Shopee), giải trí (FPT Play, Zing MP3, VTV Go) và tin tức (Báo Mới, Zing News). Ở mức nghiên cứu học thuật, các công trình trong nước chủ yếu dừng lại ở các mô hình cơ bản như User-based CF, Item-based CF, SVD, hoặc các mô hình CB đơn giản dựa trên mô tả thể loại.

Tuy nhiên, các mô hình gợi ý lai phức tạp, đặc biệt là những mô hình kết hợp giữa lọc cộng tác với phân cụm (Clustering) hoặc học sâu, vẫn còn khá mới mẻ. Các hướng tiếp cận này đòi hỏi năng lực tính toán và lượng dữ liệu lớn, điều mà phần lớn các đồ án nghiên cứu trong nước chưa có điều kiện triển khai đầy đủ.

Dự án này hướng đến việc lai hóa mô hình theo hướng thực tiễn, bằng cách sử dụng K-Means để tiền phân cụm người dùng, giảm chi phí tính toán cho bước tìm láng giềng trong KNN-based CF, sau đó lai hóa kết quả với mô hình lọc nội dung theo thể loại phim. Hướng tiếp cận này vừa mang tính khả thi trong điều kiện dữ liệu hạn chế, vừa có giá trị ứng dụng cao đối với các hệ thống gợi ý trong môi trường Việt Nam.

1.3 Các yếu tố ảnh hưởng tới gợi ý phim

Lịch sử đánh giá của người dùng (User Rating History)

Đây là yếu tố đầu vào quan trọng nhất, phản ánh sở thích hiển nhiên (explicit preferences) của người dùng. Mỗi điểm đánh giá (rating) là một biểu hiện định lượng của cảm xúc và nhận thức cá nhân về bộ phim, đóng vai trò trung tâm trong việc xây dựng hồ sơ người dùng.

Hồ sơ sở thích nội dung (User Content Profile)

Thay vì chỉ dựa vào từng bộ phim riêng lẻ, hồ sơ này tổng hợp xu hướng nội dung mà người dùng ưa thích (ví dụ: thể loại, quốc gia, đạo diễn). Trong dự án này, hồ sơ được định lượng thông qua điểm đánh giá trung bình theo từng thể loại, giúp hệ thống xác định khuynh hướng thưởng thức tổng quát của từng cá nhân.

Độ tương đồng của nhóm lân cận (Neighborhood Similarity)

Đây là nền tảng của phương pháp Collaborative Filtering, dựa trên giả định rằng "những người có hành vi tương tự sẽ có sở thích tương tự". Độ tương đồng càng cao giữa các người dùng trong nhóm lân cận, độ tin cậy của gợi ý càng lớn.

Thiên kiến đánh giá cá nhân (Individual Rating Bias)

Mỗi người dùng có xu hướng cho điểm khác nhau: có người dễ dãi (thường chấm cao), có người khắt khe (thường chấm thấp). Mô hình cần chuẩn hóa hoặc hiệu chỉnh thiên kiến này để tránh hiểu sai mức độ yêu thích. Ví dụ, điểm "3 sao" của người khó tính có thể tương đương với "4 sao" từ người dễ tính.

1.3. Các vấn đề trong Hệ thống Gợi ý Lai

Trong các nghiên cứu về hệ thống gợi ý lai, một số vấn đề nổi bật thường xuất hiện[2]:

- **Cold-start**: Vấn đề gợi ý cho người dùng hoặc mục mới do thiếu thông tin. Để giải quyết, một số nghiên cứu kết hợp Collaborative Filtering (CF) với Content-Based Filtering (CBF) hoặc sử dụng mô hình xác suất để sinh các điểm đánh giá giả.
- **Data sparsity (Ma trận thưa)**: Người dùng thường đánh giá rất ít mục, dẫn đến ma trận user-item thưa thớt, làm giảm chất lượng gợi ý. Giải pháp bao gồm phân rã ma trận, kết hợp CF với Naive Bayes hoặc khai thác các đánh giá từ người dùng tương tự.
- **Accuracy (Độ chính xác)**: Độ chính xác dự đoán sở thích người dùng vẫn là thách thức, đặc biệt trong tình huống ma trận thưa. Một số nghiên cứu cải thiện

bằng cách kết hợp CF-CBF hoặc xây dựng hồ sơ sở thích dài hạn dựa trên lịch sử hành vi.

- **Scalability (Khả năng mở rộng)**: Hệ thống cần xử lý số lượng người dùng và mục lớn. Các giải pháp bao gồm nén dữ liệu, tối ưu thuật toán CF-CBF và lựa chọn lân cận thích hợp.
- **Diversity (Đa dạng)**: Tăng sự đa dạng của các mục gợi ý để tránh thiên lệch theo độ phổ biến. Ví dụ, sử dụng mô hình K-Furthest Neighbors hoặc xác định "experts" để tạo gợi ý mới lạ nhưng phù hợp.
- **Các vấn đề khác**: Bao gồm cá nhân hóa, bảo mật dữ liệu, giảm nhiễu, tích hợp nguồn dữ liệu, tính mới và khả năng thích ứng với sở thích người dùng.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT MÔ HÌNH GỢI Ý PHIM DỰA TRÊN HỌC MÁY LAI

Tóm tắt chương 2:

Chương trình bày cơ sở lý thuyết về học máy và các lý thuyết, mô hình, kỹ thuật được sử dụng trong gợi ý phim, gồm: hệ thống gợi ý, lọc cộng tác, lọc theo nội dung, hệ thống lai, k-mean, Adjusted Cosine Similarity,..... Mỗi mô hình được giải thích chi tiết về nguyên lý hoạt động, ưu nhược điểm và tính phù hợp với bài toán

2.1 Giới thiệu về mô hình học máy

Học máy (Machine Learning - ML) là một lĩnh vực của trí tuệ nhân tạo (AI), tập trung vào việc phát triển các thuật toán và mô hình cho phép máy tính "học" trực tiếp từ dữ liệu.

Các phương pháp:

- Học có giám sát (Supervised Learning)
- Học không giám sát (Unsupervised Learning)
- Học tăng cường (Reinforcement Learning)

Trong lĩnh vực gợi ý phim, học máy giữ vai trò trung tâm trong việc hiểu và dự đoán sở thích người dùng dựa trên dữ liệu hành vi, đánh giá, hoặc đặc trưng nội dung của phim. Các mô hình học máy giúp phát hiện mối tương quan tiềm ẩn giữa người dùng và các bộ phim, từ đó đề xuất những lựa chọn có khả năng phù hợp cao mà người dùng chưa từng xem.

Nhờ vào nền tảng học máy, hệ thống gợi ý không chỉ dừng lại ở việc khớp dữ liệu đơn thuần, mà còn học hỏi liên tục từ hành vi người dùng, thích ứng với xu hướng mới, và cá nhân hóa trải nghiệm xem phim theo thời gian. Đây chính là điểm khác biệt cốt lõi giữa hệ thống gợi ý truyền thống và mô hình học lai hiện đại mà đề tài hướng tới.

2.2 Hệ thống gợi ý (Recommendation System)

2.2.1 Cơ sở lý thuyết

Hệ thống gợi ý (Recommendation System) là một ứng dụng của trí tuệ nhân tạo và học máy, nhằm dự đoán sở thích hoặc nhu cầu của người dùng để đưa ra các đề xuất phù hợp. Cơ chế hoạt động của hệ thống dựa trên việc phân tích dữ liệu lịch sử tương tác giữa người dùng và các đối tượng (item) như sản phẩm, phim, bài hát, hoặc tài liệu.

Về cơ bản, hệ thống gợi ý được chia thành ba nhóm chính:

- Lọc cộng tác (Collaborative Filtering): Dựa trên hành vi và đánh giá của người dùng tương tự để gợi ý.
- Lọc theo nội dung (Content-Based Filtering): Dựa vào đặc trưng mô tả của item và sở thích cá nhân của người dùng.
- Hệ lai (Hybrid System): Kết hợp hai phương pháp trên để cải thiện độ chính xác và khả năng bao phủ

2.2.2 Nhận xét và đánh giá

Ưu điểm:

- Cải thiện đáng kể trải nghiệm người dùng nhờ khả năng cá nhân hóa nội dung.
- Tăng mức độ tương tác, thời gian sử dụng và doanh thu cho các nền tảng trực tuyến.
- Có thể tự động học hỏi và thích ứng theo thời gian khi dữ liệu người dùng thay đổi.
- Ứng dụng linh hoạt trong nhiều lĩnh vực: thương mại điện tử, giải trí, giáo dục, y tế, v.v.

Nhược điểm:

- Hiệu quả phụ thuộc lớn vào chất lượng và quy mô dữ liệu đầu vào. Dữ liệu thưa (sparse data) hoặc thiếu đa dạng có thể làm giảm độ chính xác.
- Gặp vấn đề cold-start, khi xuất hiện người dùng hoặc item mới chưa có dữ liệu tương tác.
- Có thể gây ra thiên lệch gợi ý (recommendation bias), chỉ tập trung vào các item phổ biến và làm giảm tính đa dạng.
- Đòi hỏi tài nguyên tính toán và lưu trữ lớn khi mở rộng quy mô hệ thống.

2.3 Lọc theo nội dung (Content-Based Filtering)

2.3.1 Cơ sở lý thuyết

Lọc theo nội dung (Content-Based Filtering – CBF) là phương pháp gợi ý dựa trên việc **so sánh đặc trưng của các vật phẩm (items)** với **hồ sơ sở thích của người dùng (user profile)**. Ý tưởng chính là "người dùng có xu hướng thích các vật phẩm có nội dung tương tự với những vật phẩm họ từng thích trong quá khứ"**[4]**.

Mỗi vật phẩm i (ví dụ: phim) được biểu diễn bởi một vector đặc trưng $\mathbf{i} = (f_1, f_2, \ldots, f_n)$, trong đó các đặc trưng có thể bao gồm thể loại, diễn viên, đạo diễn, hoặc từ khóa. Các đặc trưng thường được **mã hóa bằng TF-IDF (Term Frequency – Inverse Document Frequency)** để phản ánh tầm quan trọng tương đối:

$$[TF\text{-}IDF_{t,i} = TF_{t,i} \times \log \frac{N}{n_t}]$$

Trong đó:
$TF_{t,i}$: tần suất xuất hiện của đặc trưng t trong phim i,
N: tổng số phim trong tập dữ liệu,
n_t: số phim có chứa đặc trưng t.

Sau khi biểu diễn được vector đặc trưng cho từng phim, hồ sơ người dùng (User Profile) được hình thành bằng cách tính trung bình trọng số của các vector phim mà người dùng đã đánh giá, theo công thức:

$$[\mathbf{u} = \frac{1}{|I_u|} \sum_{i \in I_u} r_{u,i} \cdot \mathbf{i}]$$

Trong đó:
I_u: tập các phim mà người dùng u đã đánh giá,
$r_{u,i}$: điểm đánh giá của người dùng u cho phim i.

Khi đã có hồ sơ người dùng và vector đặc trưng của từng phim, độ tương đồng giữa người dùng u và phim i được tính bằng độ đo Cosine Similarity:

$$[sim(u,i) = \frac{\mathbf{u} \cdot \mathbf{i}}{\|\mathbf{u}\| \, \|\mathbf{i}\|}]$$

Giá trị $sim(u,i) \in [0,1]$, càng gần 1 thì mức độ tương đồng càng cao, nghĩa là phim càng phù hợp với sở thích của người dùng.

2.3.2 Nhận xét và đánh giá

Ưu điểm:

- Không phụ thuộc vào dữ liệu của người dùng khác, phù hợp khi hệ thống còn ít người dùng (giải quyết phần nào vấn đề cold-start ở phía user).
- Dễ giải thích kết quả gợi ý, vì có thể chỉ ra rằng item được gợi ý tương tự về nội dung với các item người dùng từng thích.
- Hiệu quả trong các miền dữ liệu có mô tả phong phú và có cấu trúc rõ ràng (như phim, sách, bài hát, sản phẩm).

Nhược điểm:

- Phụ thuộc mạnh vào chất lượng và độ chi tiết của dữ liệu mô tả nội dung; nếu dữ liệu thiếu hoặc mơ hồ, hiệu quả giảm đáng kể.
- Khó gợi ý các item khác biệt hoàn toàn so với lịch sử người dùng (hiệu ứng "thiếu đa dạng" – overspecialization).
- Không khai thác được xu hướng cộng đồng hoặc mối quan hệ giữa người dùng, do chỉ dựa trên đặc trưng nội dung.

2.4 Lọc cộng tác (Collaborative Filtering)

2.4.1 Cơ sở lý thuyết

Collaborative Filtering (lọc cộng tác) là phương pháp gợi ý dựa trên giả định rằng người dùng có hành vi hoặc sở thích tương tự trong quá khứ sẽ có xu hướng thích các

đối tượng giống nhau trong tương lai. Phương pháp này không yêu cầu thông tin chi tiết về nội dung của sản phẩm mà chỉ dựa vào ma trận tương tác người dùng – đối tượng (user-item matrix), trong đó các ô biểu diễn mức độ đánh giá hoặc tương tác.
Có hai hướng tiếp cận chính:

- User-Based Collaborative Filtering: Gợi ý cho người dùng dựa trên các người dùng có hành vi tương tự.
- Item-Based Collaborative Filtering: Gợi ý các đối tượng có đặc điểm tương đồng với những đối tượng mà người dùng đã tương tác.

Độ tương đồng thường được đo bằng Cosine Similarity, Pearson Correlation hoặc Adjusted Cosine Similarity. Từ đó, hệ thống dự đoán điểm đánh giá cho các đối tượng chưa được người dùng xem và lựa chọn các đối tượng có giá trị dự đoán cao nhất để gợi ý.

Phương pháp User-Based thường được ưa chuộng hơn trong các hệ thống quy mô nhỏ hoặc trung bình, nơi dữ liệu người dùng phong phú và có sự tương đồng đáng kể giữa các nhóm người.

2.4.2 Nhận xét và đánh giá

Ưu điểm:

- Không phụ thuộc vào dữ liệu mô tả nội dung, phù hợp với các hệ thống mà thông tin về đối tượng không đầy đủ hoặc khó trích xuất.
- Có khả năng phát hiện các mối quan hệ tiềm ẩn giữa người dùng và đối tượng mà không cần hiểu rõ đặc trưng của sản phẩm.
- Dễ mở rộng và áp dụng cho nhiều lĩnh vực như thương mại điện tử, xem phim, nghe nhạc, học trực tuyến.
- User-Based có khả năng thích ứng nhanh với hành vi người dùng mới và mang tính cá nhân hóa cao.
- Item-Based ổn định hơn khi hệ thống có lượng người dùng lớn và ít thay đổi theo thời gian.

Nhược điểm:

- Dữ liệu tương tác thường thưa (sparse), khiến việc tính toán độ tương đồng kém chính xác.
- Gặp vấn đề cold-start khi có người dùng hoặc đối tượng mới chưa có dữ liệu đánh giá.
- Độ phức tạp tính toán cao khi số lượng người dùng và đối tượng lớn, đòi hỏi tối ưu hóa và giảm chiều dữ liệu.
- Dễ bị ảnh hưởng bởi nhiễu dữ liệu hoặc đánh giá không trung thực, làm giảm độ tin cậy của gợi ý.
- User-Based yêu cầu cập nhật thường xuyên do sở thích người dùng thay đổi nhanh; trong khi Item-Based tuy ổn định hơn nhưng khó phản ánh kịp thời xu hướng mới.

Trong dự án này, nhóm chúng tôi sử dụng phương pháp User-Based Collaborative Filtering vì phù hợp với các hệ thống có quy mô người dùng vừa phải và dữ liệu đánh giá phong phú. Việc gợi ý dựa trên những người có sở thích tương tự giúp tăng tính cá nhân hóa, phản ánh xu hướng và cảm nhận của người dùng thực tế tốt hơn so với cách dựa trên item. Trong các ứng dụng như gợi ý phim hoặc nhạc, nơi cảm xúc và sở thích mang tính chủ quan cao, User-Based Collaborative Filtering thường mang lại trải nghiệm tự nhiên và linh hoạt hơn.

2.4.3. User-Based Collaborative Filtering (UCF)

User-Based Collaborative Filtering (UCF) là một kỹ thuật phổ biến trong hệ thống gợi ý dựa trên việc tìm kiếm sự tương đồng giữa người dùng. Ý tưởng chính là nếu hai người dùng có lịch sử đánh giá tương tự nhau đối với một tập phim (hoặc sản phẩm) nào đó, thì họ có khả năng thích các mục mà người kia thích nhưng bản thân chưa trải nghiệm.

Các bước cơ bản của UCF:

1. Xây dựng ma trận người dùng – sản phẩm: Ma trận R có các phần tử $r_{u,i}$ là điểm đánh giá của người dùng u cho sản phẩm i.
2. Tính độ tương đồng giữa các người dùng: Các độ đo phổ biến là Cosine Similarity, Pearson Correlation, hoặc Adjusted Cosine Similarity.
3. Dự đoán điểm đánh giá chưa biết: Điểm dự đoán $\hat{r}_{u,i}$ cho người dùng u và sản phẩm i được tính bằng trung bình trọng số của các đánh giá từ các người dùng tương tự:

$$[\hat{r}_{u,i} = \frac{\sum_{v \in N(u)} \text{sim}(u,v) \cdot r_{v,i}}{\sum_{v \in N(u)} |\text{sim}(u,v)|}]$$

Trong đó:

- $N(u)$ là tập các hàng xóm gần nhất của người dùng u.
- $r_{v,i}$ là điểm đánh giá của user v cho item i.
- $\text{sim}(u,v)$ là độ tương đồng giữa user u và v.

Ưu điểm:

- Dễ triển khai, trực quan, không cần thông tin về nội dung sản phẩm.
- Hiệu quả cao nếu dữ liệu đánh giá đầy đủ và người dùng có lịch sử tương đồng

Nhược điểm:

- Khó khăn khi dữ liệu thưa (sparse data).
- Không hiệu quả với số lượng người dùng lớn do tính toán tương đồng nhiều.
- Không gợi ý tốt cho sản phẩm mới chưa có đánh giá (cold-start problem).

2.5 Hệ thống lai (Hybrid System)

2.5.1 Cơ sở lý thuyết

Hệ thống gợi ý lai (Hybrid Recommendation System) là mô hình kết hợp nhiều phương pháp gợi ý khác nhau nhằm tận dụng ưu điểm và khắc phục hạn chế của từng phương pháp riêng lẻ, như Content-Based Filtering (CBF) và Collaborative Filtering (CF). Mục tiêu của hệ thống lai là tăng độ chính xác, tính đa dạng và khả năng bao phủ trong quá trình gợi ý.

Các loại mô hình lai phổ biến gồm:

1. Weighted Hybrid:

Các mô hình gợi ý thành phần (ví dụ: CBF và CF) được huấn luyện song song, sau đó kết hợp kết quả theo trọng số. Mỗi phương pháp được gán một hệ số w_i phản ánh mức độ tin cậy hoặc hiệu quả của nó.

$$\text{Score}(u, i) = w_1 \cdot \text{SCBF}(u, i) + w_2 \cdot \text{SCF}(u, i)$$

Trong đó $\text{SCBF}(u, i)$ và $\text{SCF}(u, i)$ lần lượt là điểm gợi ý từ hai mô hình, và w_1, w_2 được điều chỉnh để tối ưu hiệu năng.

Phương pháp này linh hoạt, dễ điều chỉnh theo dữ liệu và cho phép tận dụng đồng thời thông tin về nội dung và hành vi người dùng.

2. Cascade Hybrid:

Hệ thống gợi ý hoạt động theo chuỗi tầng (cascade), trong đó kết quả của mô hình đầu tiên được sử dụng làm đầu vào lọc hoặc tinh chỉnh cho mô hình tiếp theo. Ví dụ, mô hình Collaborative Filtering có thể được dùng để tạo danh sách gợi ý sơ bộ (candidate list), sau đó Content-Based Filtering sẽ tái xếp hạng (re-rank) các item dựa trên mức độ phù hợp chi tiết với sở thích cá nhân của người dùng.

Cấu trúc này giúp giảm nhiễu, tăng tốc xử lý, đồng thời tạo ra các gợi ý chính xác và có ý nghĩa ngữ cảnh hơn.

2.5.2 Nhận xét và đánh giá

Ưu điểm:

- Tăng độ chính xác và khả năng bao phủ nhờ kết hợp nhiều nguồn thông tin.
- Giảm thiểu hạn chế của từng phương pháp riêng lẻ, đặc biệt là vấn đề cold-start và dữ liệu thưa.
- Dễ tùy biến theo đặc thù dữ liệu và yêu cầu của hệ thống.

- Với Weighted Hybrid, hệ thống có thể học hoặc điều chỉnh trọng số động, phù hợp với người dùng hoặc nhóm người dùng khác nhau.
- Với Cascade Hybrid, kết quả gợi ý được lọc dần qua nhiều tầng, giúp loại bỏ các item không phù hợp và tăng chất lượng danh sách gợi ý cuối cùng.

Nhược điểm:

- Cấu trúc phức tạp, đòi hỏi nhiều tài nguyên tính toán và xử lý dữ liệu.
- Khó xác định trọng số hoặc thứ tự ưu tiên giữa các mô hình, đặc biệt khi dữ liệu đa dạng và không đồng nhất.
- Cần quá trình hiệu chỉnh (tuning) và đánh giá định kỳ để duy trì hiệu năng ổn định.

Trong phạm vi dự án này, hệ thống được xây dựng dựa trên sự kết hợp giữa hai phương pháp lai – Cascade Hybrid và Weighted Hybrid. Cách tiếp cận này tận dụng khả năng lọc tầng của Cascade để giảm nhiễu và tập trung vào các item tiềm năng, đồng thời áp dụng Weighted Hybrid để tính điểm tổng hợp từ hai hướng tiếp cận Content-Based và Collaborative Filtering. Việc kết hợp này giúp hệ thống nâng cao độ chính xác, tăng khả năng cá nhân hóa, và đảm bảo hiệu quả gợi ý ổn định ngay cả khi dữ liệu người dùng không đầy đủ hoặc phân tán. Cụ thể đã được chúng tôi trình bày ở chương 3.

2.6. Mô hình K-mean

2.6.1 Cơ sở lý thuyết

Thuật toán K-Means là một phương pháp phân cụm phổ biến trong học máy không giám sát, được sử dụng để chia tập dữ liệu thành K cụm sao cho các điểm dữ liệu trong cùng một cụm có độ tương đồng cao nhất.

Quy trình của K-Means bao gồm các bước chính sau:

- Khởi tạo ngẫu nhiên K tâm cụm ban đầu $\{\mu_1, \mu_2, ..., \mu_K\}$.
- Gán mỗi điểm dữ liệu x_i vào cụm có tâm gần nhất:

$$C_i = \arg\min_k ||x_i - \mu_k||^2.$$

- Cập nhật lại tâm cụm bằng cách tính trung bình các điểm thuộc cụm đó:

$$\mu_k = \frac{1}{|C_k|} \sum_{x_i \in C_k} x_i.$$

- Lặp lại hai bước trên cho đến khi các tâm cụm hội tụ hoặc thay đổi không đáng kể.

Trong hệ thống gợi ý, K-Means thường được sử dụng để phân cụm người dùng hoặc phim dựa trên đặc trưng hành vi hoặc nội dung. Điều này giúp giảm độ phức tạp tính toán trong quá trình tìm kiếm hàng xóm tương tự.

2.6.2 Nhận xét và đánh giá

Ưu điểm

- Đơn giản, dễ triển khai và hiệu quả với dữ liệu lớn.
- Giúp chia nhỏ không gian tìm kiếm, tối ưu thời gian cho các mô hình gợi ý dựa trên tương đồng.
- Có thể kết hợp với CF hoặc CBF để tạo mô hình lai (hybrid recommender).

Nhược điểm

- Phải xác định trước số cụm K.
- Dễ bị ảnh hưởng bởi việc khởi tạo tâm cụm ban đầu.
- Không đảm bảo hội tụ đến nghiệm tối ưu toàn cục.

2.6.3. Các kỹ thuật tìm K

2.6.3.1. Elbow Method

Elbow Method là kỹ thuật trực quan phổ biến để chọn K. Ý tưởng cơ bản là tính toán tổng bình phương khoảng cách (Within-Cluster Sum of Squares, WCSS) giữa các điểm dữ liệu và tâm cụm tương ứng với nhiều giá trị K. Khi vẽ đồ thị WCSS theo K, đường cong sẽ giảm dần. Điểm mà sau đó WCSS giảm chậm lại tạo thành "cú khuỷu" (elbow) được chọn làm K tối ưu.

2.6.3.2. Silhouette Method

Silhouette Method đánh giá chất lượng cụm bằng **Silhouette Score**. Điểm số này đo lường mức độ gần gũi của các điểm trong cùng cụm so với cụm khác. Công thức Silhouette Score cho một điểm dữ liệu i là:

$$[s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}]$$

Trong đó:

- $a(i)$ là khoảng cách trung bình từ điểm i đến các điểm khác trong cùng cụm.
- $b(i)$ là khoảng cách trung bình từ điểm i đến các điểm trong cụm gần nhất không chứa i.

Silhouette Score nằm trong khoảng $[-1, 1]$. Điểm càng gần 1 nghĩa là điểm được phân cụm tốt. Giá trị trung bình của tất cả điểm dữ liệu được sử dụng để chọn K tối ưu: K cho Silhouette Score cao nhất được coi là phù hợp.

2.7 Adjusted Cosine Similarity

2.7.1 Cơ sở lý thuyết

Adjusted Cosine Similarity là một biến thể của Cosine Similarity được sử dụng trong Collaborative Filtering để đo độ tương đồng giữa các sản phẩm (items) thay vì người dùng. Điểm khác biệt là nó điều chỉnh trung bình điểm đánh giá của người dùng, nhằm loại bỏ sự lệch do xu hướng đánh giá của từng người.

Công thức:

$$\text{Sim}(i, j) = \frac{\sum_{u \in U}(r_{u,i} - \bar{r}_u)(r_{u,j} - \bar{r}_u)}{\sqrt{\sum_{u \in U}(r_{u,i} - \bar{r}_u)^2}\sqrt{\sum_{u \in U}(r_{u,j} - \bar{r}_u)^2}}$$

$r_{u,i}$i: điểm đánh giá của người dùng u với item i

$\bar{r}_u$: điểm trung bình của người dùng u

U: tập người dùng đã đánh giá cả i và j

2.7.2 Nhận xét và đánh giá

Ưu điểm:

- Loại bỏ bias cá nhân của người dùng.
- Thường chính xác hơn Cosine truyền thống trong các bài toán dự đoán rating.

Nhược điểm:

- Tính toán phức tạp hơn khi dữ liệu lớn.
- Chỉ áp dụng được khi các item được nhiều người đánh giá chung.

2.8 Mô hình KNN-based CF

2.8.1 Cơ sở lý thuyết

KNN-based CF là phương pháp mở rộng khái niệm "user-based" hoặc "item-based" CF. Nhấn mạnh việc chọn K hàng xóm gần nhất để tính toán, giúp giảm chi phí tính toán và tăng độ ổn định.

Công thức tương tự UCF nhưng tập trung vào K neighbors:

$$[\hat{r}_{u,i} = \frac{\sum_{v \in K(u)} \text{sim}(u,v) \cdot r_{v,i}}{\sum_{v \in K(u)} |\text{sim}(u,v)|}]$$

2.8.2 Nhận xét và đánh giá

Ưu điểm

- Giảm chi phí tính toán so với User-Based CF gốc bằng cách chỉ sử dụng K hàng xóm gần nhất.
- Có thể áp dụng linh hoạt cho User-Based và Item-Based.
- Dễ hiểu, trực quan, dễ triển khai.

Nhược điểm

- Vẫn gặp vấn đề với dữ liệu sparse khi đánh giá hiếm.
- Hiệu quả phụ thuộc vào giá trị K và phương pháp đo độ tương đồng.

- Đối với tập dữ liệu lớn, việc tính độ tương đồng ban đầu vẫn tốn chi phí.

2.9 Weighted Average Prediction

2.9.1 Cơ sở lý thuyết

Weighted Average Prediction là một phương pháp mở rộng trong lọc cộng tác, trong đó điểm đánh giá dự đoán của người dùng u với sản phẩm i được tính bằng trung bình có trọng số của các điểm đánh giá từ những người dùng tương tự. Công thức tổng quát như sau:

$$\hat{r}_{u,i} = \bar{r}_u + \frac{\sum_{v \in N(u)} \text{sim}(u, v) \cdot (r_{v,i} - \bar{r}_v)}{\sum_{v \in N(u)} |\text{sim}(u, v)|}$$

Trong đó:

$N(u)$: tập các hàng xóm gần nhất của người dùng u.

$\text{sim}(u, v)$: độ tương đồng giữa người dùng u và v .

$r_{v,i}$: điểm đánh giá thực tế của người dùng v cho sản phẩm i.

$\bar{r}_u, \bar{r}_v$: điểm đánh giá trung bình của người dùng u và v.

$\hat{r}_{u,i}$: điểm đánh giá dự đoán.

Phương pháp này giúp cân nhắc sự khác biệt về thang điểm đánh giá giữa các người dùng, làm cho dự đoán chính xác hơn so với phương pháp trung bình đơn giản.

2.9.2 Nhận xét và đánh giá

Ưu điểm

- Giảm bias do sự khác nhau trong thang điểm của các người dùng.
- Dự đoán chính xác hơn so với phương pháp trung bình không trọng số.
- Dễ triển khai và giải thích.

Nhược điểm

- Phụ thuộc vào độ chính xác của phép đo độ tương đồng.
- Khi dữ liệu sparse, tập $N(u)$ nhỏ, kết quả dự đoán có thể không ổn định.
- Tính toán trọng số phức tạp hơn trung bình đơn giản, đặc biệt với tập dữ liệu lớn.

2.4 Các độ đo đánh giá

Nhóm độ đo này tập trung vào việc đo lường sự sai khác giữa điểm đánh giá dự đoán $\hat{r}_{u,i}$ *và điểm đánh giá thực tế* $r_{u,i}$.

2.4.1. Sai số Căn bậc hai Trung bình (Root Mean Square Error - RMSE)

RMSE[5] là một thước đo chuẩn để đánh giá sai số của mô hình trong việc dự đoán dữ liệu định lượng. Nó được định nghĩa là căn bậc hai của trung bình các bình phương sai số giữa giá trị dự đoán và giá trị thực tế.

Trong bối cảnh hệ thống gợi ý, RMSE đo lường sự khác biệt giữa các điểm đánh giá dự đoán bởi mô hình và các điểm đánh giá thực tế mà người dùng đã cho. Công thức tính RMSE được viết như sau:

$$RMSE = \sqrt{\frac{1}{|T|} \sum_{(u,i) \in T} (r_{u,i} - \hat{r}_{u,i})^2}$$

Trong đó, $\mathbb{T}$ là tập hợp các cặp người dùng-vật phẩm trong tập kiểm thử, và $|\mathbb{T}|$ là kích thước của tập này.

2.4.2 Sai số tuyệt đối trung bình (Mean Absolute Error - MAE)

MAE[5] là một độ đo được sử dụng để đánh giá độ chính xác của mô hình dự đoán. MAE tính trung bình độ lớn của sai số giữa giá trị dự đoán và giá trị thực tế, không quan tâm đến hướng sai số.

Công thức tính MAE được viết như sau:

$$\text{MAE} = \frac{1}{|\mathbb{T}|} \sum_{(u,i) \in \mathbb{T}} |r_{u,i} - \hat{r}_{u,i}|$$

Cả RMSE và MAE đều được sử dụng rộng rãi trong các mô hình Collaborative Filtering (CF), đặc biệt là User-Based CF, Item-Based CF, và Weighted Average Prediction, nhằm đánh giá khả năng dự đoán rating của mô hình.

CHƯƠNG 3. XÂY DỰNG MÔ HÌNH GỢI Ý PHIM DỰA TRÊN HỌC MÁY LAI

Tóm tắt chương 3:

Chương trình bày quá trình xây dựng mô hình gợi ý phim dựa trên học máy lai, kết hợp Collaborative Filtering và Content-based Filtering. Dữ liệu MovieLens được xử lý, chuẩn hóa và mã hóa thể loại phim để tạo profile người dùng và phim. Mô hình CF tính độ tương đồng giữa người dùng, sử dụng KNN trong cụm để dự đoán rating, trong khi phần Content-based đánh giá mức độ phù hợp của thể loại phim. Điểm dự đoán cuối cùng là sự kết hợp trọng số giữa hai thành phần, cho phép cá nhân hóa gợi ý và nâng cao độ chính xác.

3.1. Mô hình tổng quát

Các bước chính trong mô hình tổng quát bao gồm thu thập dữ liệu, xử lý dữ liệu, chia hai tập train và test, xây dựng mô hình, đánh giá mô hình, ứng dụng mô hình, và giải thích kết quả đầu ra. Mỗi bước được thiết kế dựa trên tham khảo quy trình của Furtado & Singh (2020)[6] và kết hợp với cách xử lý theo bài báo của YiTong Dou[7] nhằm đảm bảo tính chính xác, minh bạch và khả năng ứng dụng thực tế của mô hình học máy trong lĩnh vực gợi ý phim.

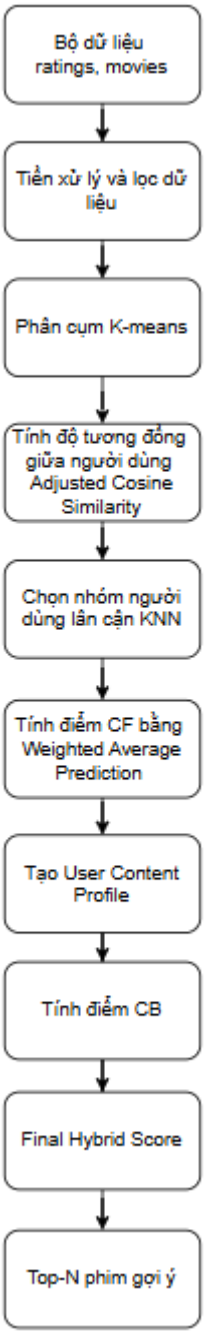
Bước đầu tiên là thu thập dữ liệu, trong đó dữ liệu được lấy từ các nguồn phim, bao gồm các đặc trưng như: movieId, title, genres, và các lượt đánh giá của người dùng với các thuộc tính userId, movieId, rating. Sau đó, dữ liệu được đưa vào bước xử lý dữ liệu, bao gồm các thao tác làm sạch, chuẩn hóa, và mã hóa (one-hot encoding thể loại phim) để đảm bảo dữ liệu nhất quán và phù hợp với các mô hình học máy.

Khi dữ liệu đã sẵn sàng, hệ thống thực hiện bước chia tập train và test, cho phép đánh giá mô hình trên dữ liệu chưa từng thấy và giảm nguy cơ overfitting. Tiếp theo là xây dựng mô hình, nơi mô hình gợi ý phim hybrid được triển khai bao gồm hai thành phần chính: Collaborative Filtering dựa trên user (User-based CF) và Content-based Filtering dựa trên thông tin thể loại phim.

Trong quá trình xây dựng Collaborative Filtering, các bước bao gồm: tạo ma trận user-movie, chuẩn hóa để giảm bias (z-score per user), phân cụm người dùng bằng K-Means, tính toán độ tương đồng giữa các user trong cùng cụm bằng Adjusted Cosine hoặc Pearson, và lựa chọn top-K neighbors để dự đoán rating thông qua phương pháp trung bình có trọng số (weighted average). Song song đó, Content-based Filtering dựa trên profile thể loại phim mà người dùng đã đánh giá để dự đoán sở thích. Hai thành phần này được kết hợp theo trọng số α để tạo ra điểm dự đoán cuối cùng (hybrid score).

Sau khi mô hình được xây dựng, tập test được sử dụng để đánh giá hiệu năng dự đoán rating, thông qua các chỉ số như RMSE, MAE. Khi mô hình được chọn đã đạt hiệu năng tối ưu, bước tiếp theo là ứng dụng mô hình để gợi ý phim mới cho người dùng dựa trên lịch sử đánh giá và sở thích thể loại.

Để đảm bảo tính minh bạch và khả năng giải thích, hệ thống còn có thể phân tích các thành phần ảnh hưởng đến điểm dự đoán, ví dụ như trọng số CF vs Content, hay mức độ tương đồng với các neighbors, giúp người dùng và nhà quản trị hiểu được lý do gợi ý và từ đó cải thiện chất lượng dịch vụ. (Hình 3.1)



Hình 3.1: Sơ đồ mô hình tổng quát

3.2. Mô tả dữ liệu

3.2.1 Thông tin dữ liệu

Dự án sử dụng bộ dữ liệu MovieLens Latest-Small — một trong những bộ dữ liệu chuẩn (benchmark dataset) được sử dụng rộng rãi trong nghiên cứu hệ thống gợi ý phim, do GroupLens Research Lab (Đại học Minnesota) công bố. Bộ dữ liệu

MovieLens-latest-small có quy mô vừa phải, phù hợp cho việc thử nghiệm mô hình và kiểm chứng thuật toán với độ phức tạp tính toán hợp lý.

Nguồn gốc bộ dữ liệu: MovieLens | GroupLens

Bộ dữ liệu bao gồm bốn tệp chính với nội dung như sau:

Tệp dữ liệu	Nội dung	Số lượng mẫu
ratings.csv	Lịch sử đánh giá phim của người dùng, bao gồm ID người dùng, ID phim, điểm đánh giá và thời gian đánh giá	100.836 lượt đánh giá
movies.csv	Thông tin mô tả phim bao gồm mã phim, tiêu đề phim và danh sách thể loại	9.742 phim
tags.csv	Các thẻ mô tả phim được gán thủ công bởi người dùng	3.686 thẻ
links.csv	Liên kết ID phim với cơ sở dữ liệu IMDb và TMDB phục vụ mục đích tra cứu và mở rộng dữ liệu	9.742 dòng

Bảng 3.1: Các tệp trong nguồn bộ dữ liệu

Trong đó, dự án sử dụng tệp dữ liệu ratings.csv và movies.csv để xây dựng User-based Collaborative Filtering (UCF) và Content-Based Filtering (CBF) dựa trên thể loại phim.

Các thành phần dữ liệu chính bao gồm:

UserID: Mã định danh người dùng (được ẩn danh nhằm bảo mật thông tin)

MovieID: Mã định danh phim

Rating: Điểm đánh giá theo thang từ **0.5 đến 5.0** với bước nhảy **0.5**

Timestamp: Thời gian đánh giá (Unix time)

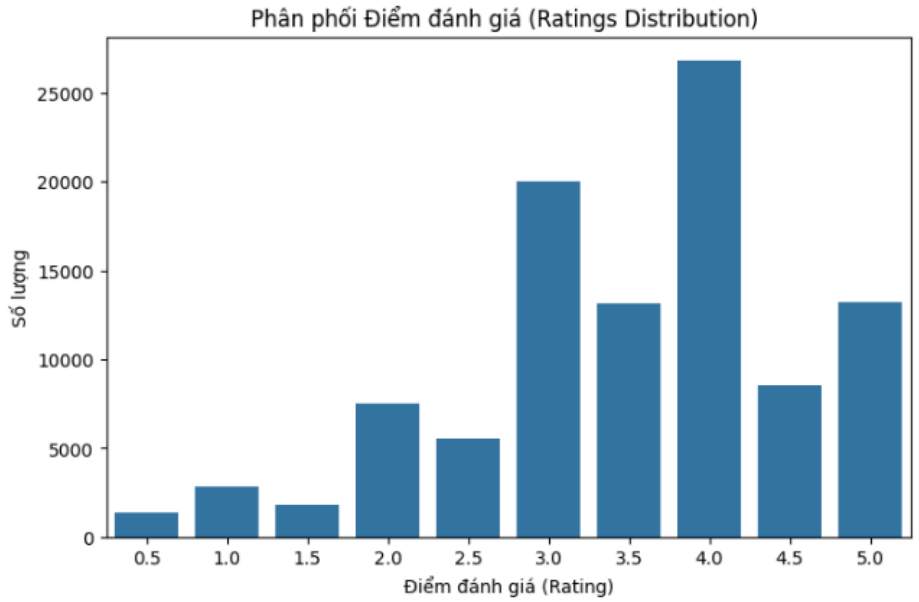
Genres: Danh sách thể loại phim (ví dụ: *Drama|Comedy|Romance*)

Cả hai tệp ratings.csv và movies.csv đều không có giá trị thiếu (missing values) nên có thể thấy dữ liệu sạch về mặt cấu trúc.

3.2.2. Trực quan hóa dữ liệu

3.2.2.1. Phân bố điểm đánh giá

Phân tích phân phối điểm đánh giá cho thấy dữ liệu có phân bố lệch phải nhẹ với điểm trung bình 3.501 và trung vị 3.5, phản ánh rõ xu hướng đánh giá tích cực của người dùng. Hơn 50% tổng số đánh giá nằm trong khoảng từ 3.5 trở lên, trong đó rating 4.0 chiếm tỷ lệ cao nhất với 26.6% (26.846 lượt), vượt trội so với các mức điểm khác. Điều này phù hợp với hành vi thực tế trong các nền tảng trực tuyến: người dùng thường chỉ đánh giá khi hài lòng và ít có xu hướng dành thời gian cho phim không ưng ý, dẫn đến sự thiếu hụt đánh giá trung bình hoặc tiêu cực. Đặc biệt, không tồn tại rating 0.0 hoặc giá trị âm, đồng thời rating thấp nhất là 0.5, chứng tỏ dữ liệu hoàn toàn sạch về mặt nhiễu tiêu cực và đáng tin cậy cho việc huấn luyện mô hình.



Hình 3.2: Biểu đồ bar phân phối điểm đánh giá

Tuy nhiên, mỗi người dùng sở hữu thang điểm chủ quan riêng biệt – người nghiêm khắc có thể coi 4.0 là mức trung bình, trong khi người dễ tính lại xem 3.0 là đủ tốt – dẫn đến hiện tượng bias cá nhân (user bias). Nếu không xử lý, hiện tượng này sẽ làm sai lệch độ tương đồng khi tính Adjusted Cosine Similarity, từ đó giảm độ chính xác dự đoán.

3.2.2.2. Phân tích độ thưa thớt (Sparsity)

1. Công thức tính độ thưa thớt

Độ thưa thớt của ma trận đánh giá được định nghĩa là tỷ lệ phần trăm các ô trống trong ma trận User-Item. Đây là một độ đo quan trọng phản ánh mức độ thưa thớt của dữ liệu tương tác.

$$\text{Sparsity} = 1 - \frac{|R|}{|U| \times |I|}$$

Trong đó:

- $|R|$: Số lượng rating đã được cung cấp (số phần tử khác 0)
- $|U|$: Số lượng người dùng (Users)
- $|I|$: Số lượng sản phẩm/mục (Items)

Công thức này tương đương với:

$$\text{Sparsity}(\%) = \left(1 - \frac{|R|}{|U| \times |I|}\right) \times 100$$

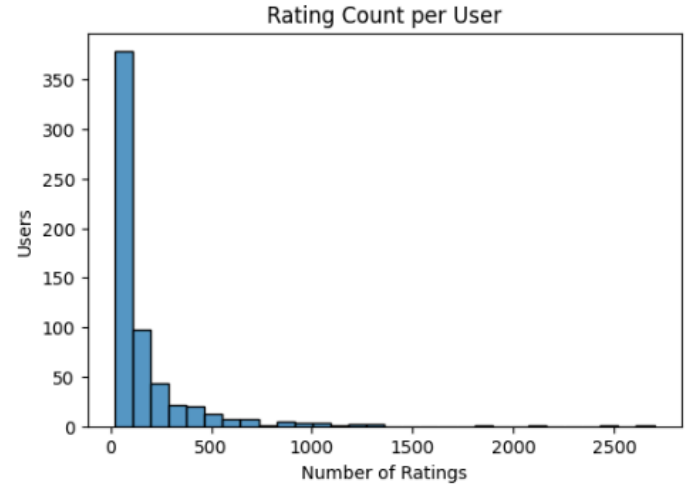
2. Tính độ thưa thớt trên dữ liệu gốc

$$\text{Sparsity} = \left(1 - \frac{100836}{610 \times 9724}\right) \times 100 \approx 98.30\%$$

Kết quả cho thấy độ thưa thớt (sparsity) của ma trận đánh giá đạt khoảng 98.30%, nghĩa là chỉ có khoảng 1.7% giá trị trong ma trận User–Item được gán rating. Điều này phản ánh đặc trưng điển hình của các hệ thống gợi ý trong thực tế, nơi phần lớn người dùng chỉ đánh giá một lượng rất nhỏ mục phim so với tổng số mục có sẵn. Mức độ thưa thớt cao tạo ra thách thức đáng kể cho mô hình gợi ý, đặc biệt đối với các phương pháp Collaborative Filtering truyền thống, do thiếu thông tin để suy luận sở thích một cách chính xác. Vì vậy, việc áp dụng các kỹ thuật xử lý dữ liệu phù hợp, kết hợp thêm thông tin nội dung phim hoặc hồ sơ người dùng, hay sử dụng các phương pháp học máy lai (Hybrid Recommender Systems) trở nên cần thiết nhằm cải thiện hiệu quả dự đoán và giảm thiểu vấn đề cold-start và dữ liệu thưa thớt.

3.2.2.3. Phân phối số lượng đánh giá theo người dùng và phim

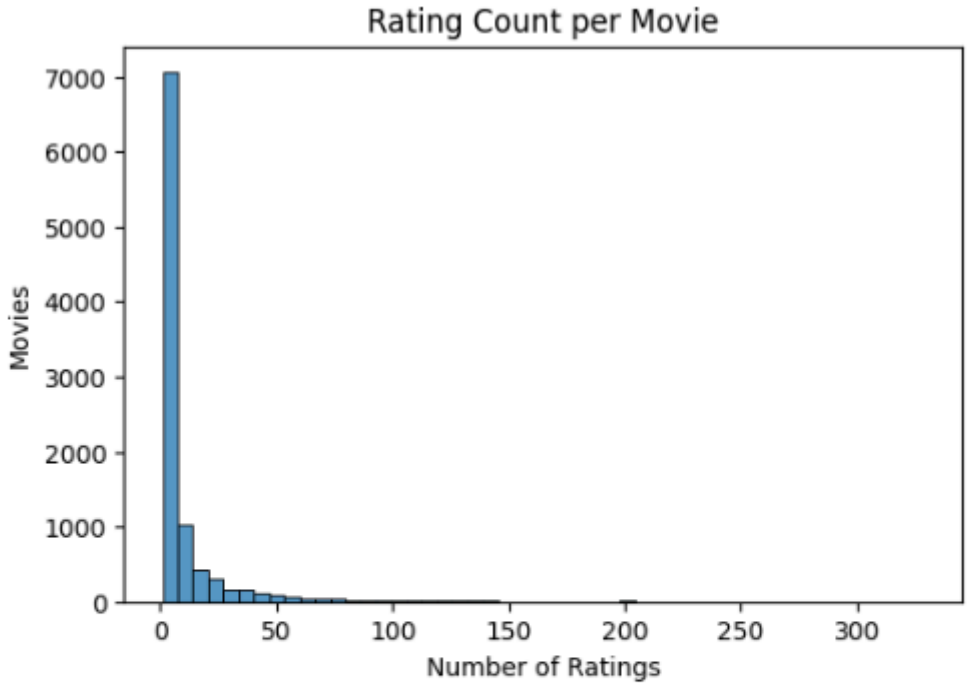
Phân tích phân bố số lượng đánh giá theo người dùng và theo phim cho thấy cả hai đều tuân theo quy luật long-tail – một đặc trưng phổ biến trong hệ thống gợi ý, phản ánh sự chênh lệch lớn giữa nhóm thiểu số tích cực và phần lớn thụ động.




```
User activity summary:
 count     610.000000
mean      165.304918
std       269.480584
min        20.000000
25%        35.000000
50%        70.500000
75%       168.000000
max      2698.000000
dtype: float64
```

Hình 3.3: Phân phối số lượng đánh giá theo người dùng

Cụ thể, phân bố theo người dùng (Hình 3.3) có trung bình 165.3 đánh giá/người dùng, nhưng trung vị chỉ đạt 70, với 44.8% người dùng (273/610) có dưới 50 đánh giá và một người dùng duy nhất (userId 414) đóng góp tới 2.696 lượt – chiếm 2.67% tổng dữ liệu.




```
Movie popularity summary:
 count    9724.000000
mean       10.369807
std        22.401005
min         1.000000
25%         1.000000
50%         3.000000
75%         9.000000
max       329.000000
dtype: float64
```

Hình 3.4: Phân phối số lượng đánh giá theo phim

Tương tự, phân bố theo phim (Hình 3.4) còn khắc nghiệt hơn: trung bình 10.4 đánh giá/phim, trung vị chỉ 3, và 93.9% phim (9.152/9.742) có dưới 30 đánh giá.

Hiện tượng này đặt ra ba vấn đề cốt lõi:

- Thứ nhất, người dùng và phim có ít đánh giá → thiếu dữ liệu để học hành vi và đặc trưng, dẫn đến cold-start nghiêm trọng và độ tương đồng thấp trong User-based CF.

- Thứ hai, hiệu ứng power user/blockbuster → một vài thực thể chiếm tỷ trọng quá lớn, dễ làm lệch cụm K-Means và ưu tiên gợi ý phim phổ biến, giảm tính đa dạng.
- Thứ ba, sparsity cao (98.3%) → ít phim chung giữa các user, khiến Adjusted Cosine Similarity không đáng tin cậy.

3.3. Tiền xử lý

Trước khi đưa vào mô hình học máy, dữ liệu cần được xử lý để đảm bảo tính nhất quán, đầy đủ và phù hợp với yêu cầu của các mô hình. Các bước tiền xử lý được thực hiện như sau:

3.3.1. Loại bỏ các biến dư thừa

Mục tiêu của bước này là tinh gọn dữ liệu, loại bỏ các cột không đóng góp giá trị phân tích hoặc không cần thiết cho quá trình huấn luyện mô hình:

- Cột timestamp (rating.csv): chỉ thời điểm người dùng đánh giá phim, không mang ý nghĩa trực tiếp trong việc xác định sở thích cốt lõi của người dùng hoặc đặc điểm của vật phẩm (phim) trong mô hình CF và CBF cơ bản. Do đó, cột này được **loại bỏ** để đơn giản hóa tập dữ liệu và giảm thiểu kích thước bộ nhớ.

3.3.2. Chuẩn bị dữ liệu cho CF

Tinh Gọn Dữ Liệu và Lọc Độ Thưa Thớt (Sparsity Reduction)

Mục tiêu của bước tiền xử lý này là giảm nhiễu phát sinh từ các người dùng và vật phẩm có số lượng tương tác quá thấp, vốn thường tạo ra tín hiệu không ổn định và suy giảm chất lượng mô hình gợi ý dựa trên cộng tác. Để đạt được điều này, hệ thống áp dụng chiến lược lọc tuần tự hai chiều với các ngưỡng tối thiểu được định nghĩa rõ ràng: mỗi người dùng phải có tối thiểu 30 đánh giá (min_user_ratings = 30) và mỗi bộ phim phải có tối thiểu 20 đánh giá (min_movie_ratings = 20).

Quy trình lọc tuần tự

Chiến lược lọc được triển khai theo ba pha như sau:

1. Lọc người dùng lần thứ nhất

   Hệ thống loại bỏ các người dùng có số lượng đánh giá nhỏ hơn 30. Bước này bảo đảm rằng tập dữ liệu ban đầu chỉ bao gồm các người dùng có hành vi tương tác đủ rõ ràng, tạo điều kiện cho mô hình học được sở thích cá nhân một cách đáng tin cậy.

2. Lọc phim

   Tiếp theo, các bộ phim có ít hơn 20 lượt đánh giá bị loại bỏ. Những phim hiếm khi được đánh giá thường không cung cấp tín hiệu ổn định về thị hiếu người xem, dẫn đến vấn đề cold-item và cản trở quá trình xây dựng mô hình dự đoán chính xác.

3. Lọc người dùng lần thứ hai (tái lọc)

   Sau khi loại bỏ các phim ít phổ biến, số lượng đánh giá của một số người dùng có thể giảm xuống dưới ngưỡng 30. Do đó, hệ thống tiến hành lọc lại để chỉ giữ lại những người dùng vẫn thỏa điều kiện tối thiểu về số lượt tương tác.

Kết quả sau lọc

Thống kê	**Giá trị**
Số người dùng còn lại	464
Số bộ phim còn lại	1.281
Tổng số rating còn lại	64.526

Bảng 3.2: Thống kê kết quả sau khi lọc dữ liệu

Kết quả trên cho thấy phần lớn các tương tác có ý nghĩa vẫn được giữ nguyên, đồng thời các phần tử kém thông tin đã được loại bỏ hiệu quả. Việc thu gọn tập dữ liệu theo cách có chọn lọc này giúp giảm độ thưa thớt, nâng cao độ tin cậy của tín hiệu, và tạo điều kiện cho mô hình học sâu hơn vào các mẫu hành vi thực chất, từ đó cải thiện hiệu suất gợi ý.

Mã hóa Thể loại (Genre One-Hot Encoding):

Để bổ sung thông tin ngữ nghĩa cho mô hình gợi ý, một bước xây dựng đặc trưng nội dung dựa trên thể loại phim được thực hiện. Trước hết, danh sách thể loại được biểu diễn dưới dạng chuỗi ký tự với dấu phân tách "|" trong trường genres được chuyển đổi thành hệ thống biến giả (one-hot encoding). Mỗi thể loại được ánh xạ thành một biến nhị phân và nhận giá trị 1 nếu bộ phim thuộc thể loại đó, hoặc 0 nếu không. Kết quả thu được là một ma trận đặc trưng thể loại gắn liền với từng bộ phim thông qua khóa định danh movieId. Ma trận này được kết hợp với thông tin phim ban đầu để hình thành bảng movies_genres, dùng làm nguồn dữ liệu đặc trưng nội dung.

Tiếp theo, dữ liệu đánh giá của người dùng được hợp nhất với bảng đặc trưng thể loại thông qua thao tác nối bảng (merge) dựa trên khóa movieId. Bảng kết quả ratings_with_genres thể hiện đầy đủ cả tương tác của người dùng và đặc trưng nội dung của các bộ phim đã xem. Bước này cho phép mô hình học được xu hướng tiêu thụ nội dung của người dùng theo từng nhóm thể loại, từ đó tạo điều kiện xây dựng hồ sơ sở thích cá nhân chính xác hơn.

Việc tích hợp thông tin thể loại vào dữ liệu tương tác mang ý nghĩa quan trọng trong bối cảnh hệ gợi ý kết hợp. Đặc trưng nội dung giúp bổ sung tín hiệu ngữ nghĩa, giảm tác động của độ thưa dữ liệu, đồng thời hỗ trợ khắc phục vấn đề cold-start đối với các bộ phim mới hoặc người dùng có lịch sử tương tác hạn chế. Qua đó, mô hình có thể nâng cao chất lượng dự đoán và khả năng cá nhân hóa khuyến nghị trong giai đoạn huấn luyện và triển khai thực nghiệm.

Xây dựng User-Genre Profile:

Để mô hình gợi ý có khả năng hiểu rõ sở thích của từng người dùng, bước tiếp theo trong tiền xử lý là xây dựng hồ sơ sở thích người dùng theo thể loại phim. Mỗi người dùng được biểu diễn bởi một vector đặc trưng, trong đó các giá trị phản ánh mức độ yêu thích từng thể loại dựa trên các lượt đánh giá trước đó.

Quy trình thực hiện như sau:

1. Nhóm dữ liệu theo người dùng: Tất cả lượt đánh giá của mỗi người dùng được gom lại, cho phép phân tích sở thích cá nhân trên toàn bộ phim mà họ đã xem.
2. Tính trung bình có trọng số theo thể loại:

   Với mỗi người dùng, hệ thống tính giá trị trung bình của các cột thể loại, sử dụng điểm số đánh giá làm trọng số. Cách làm này đảm bảo rằng các phim được đánh giá cao sẽ đóng góp nhiều hơn vào hồ sơ sở thích, phản ánh chính xác ưu tiên cá nhân của người dùng. Ví dụ, nếu một người dùng đánh giá cao các phim hành động và khoa học viễn tưởng, vector sở thích của họ sẽ có giá trị cao ở các cột Action và Sci-Fi.
3. Đảm bảo đủ người dùng cuối cùng:

   Các vector được lập chỉ mục theo danh sách final_active_users, đồng thời các giá trị thiếu được điền bằng 0. Điều này đảm bảo tất cả người dùng trong tập dữ liệu cuối cùng đều có hồ sơ sở thích đầy đủ, sẵn sàng cho các bước xây dựng mô hình tiếp theo.

Kết quả của bước này là DataFrame user_genre_pref, trong đó:

- Hàng (index): người dùng (userId)
- Cột (columns): các thể loại phim
- Giá trị (values): mức độ ưa thích từng thể loại (trung bình trọng số theo rating)

Hồ sơ sở thích người dùng này là nền tảng quan trọng cho các mô hình gợi ý lai (hybrid recommender system), giúp kết hợp tín hiệu từ lịch sử đánh giá với đặc trưng nội dung của phim. Việc sử dụng trọng số từ điểm đánh giá không chỉ phản ánh chính xác sở thích thực tế mà còn giảm ảnh hưởng của các tương tác ít quan trọng, từ đó nâng cao hiệu quả cá nhân hóa và độ chính xác của mô hình.

Chuẩn hóa Điểm Đánh giá bằng Phương pháp Z-score

Sau khi hoàn tất quy trình lọc dữ liệu và xây dựng đặc trưng nội dung, hệ thống tiến hành khởi tạo và chuẩn hóa ma trận đánh giá nhằm chuẩn bị cho giai đoạn mô hình hóa gợi ý dựa trên cộng tác. Ma trận đánh giá được ký hiệu là R, trong đó mỗi hàng tương ứng với một người dùng, mỗi cột ứng với một bộ phim, và phần tử $R_{u,i}$ biểu

diễn mức đánh giá của người dùng $$u$$ đối với bộ phim $$i$$. Đối với các trường hợp người dùng chưa đánh giá một bộ phim cụ thể, giá trị 0 được sử dụng để duy trì cấu trúc ma trận đầy đủ và thuận tiện cho các phép biến đổi ma trận tiếp theo.

Trong thực tế, dữ liệu đánh giá thường tồn tại hiện tượng thiên lệch cá nhân (user rating bias). Một số người dùng có xu hướng đánh giá cao hơn mặt bằng chung, trong khi những người khác lại tương đối nghiêm khắc. Nếu không xử lý, sự khác biệt này có thể làm sai lệch phép đo tương đồng giữa người dùng hoặc giữa các bộ phim, từ đó ảnh hưởng tiêu cực đến hiệu năng của các thuật toán lọc cộng tác.

Để khắc phục vấn đề này, ma trận R được chuẩn hóa theo Z-score trên từng hàng, tức theo từng người dùng. Phép chuẩn hóa này thực hiện bằng cách trừ đi giá trị trung bình đánh giá của người dùng và chia cho độ lệch chuẩn tương ứng, giúp đưa dữ liệu về cùng thang phân phối chuẩn hóa. Nhờ vậy, hệ thống giảm thiểu tác động của phạm vi chấm điểm khác nhau và thói quen đánh giá mang tính chủ quan, cho phép mô hình tập trung vào cấu trúc ưu tiên tương đối giữa các vật phẩm thay vì giá trị tuyệt đối của điểm số.

Quá trình chuẩn hóa được biểu diễn bằng công thức:

$$R'_{u,i} = \frac{R_{u,i} - \mu_u}{\sigma_u + \varepsilon}$$

Trong đó μ_u và σ_u lần lượt là trung bình và độ lệch chuẩn của các đánh giá do người dùng u thực hiện. Kết quả thu được là ma trận $R_{\text{normalized}}$, đóng vai trò đầu vào quan trọng trong các thuật toán lọc cộng tác dựa trên độ tương đồng, tiêu biểu như cosine similarity và Pearson correlation. Ma trận này đồng thời được sử dụng trong tiến trình kết hợp với đặc trưng nội dung để xây dựng mô hình gợi ý lai của hệ thống.

Hợp nhất Đặc trưng (Feature Concatenation)

Ma trận Đặc trưng Lai (H) được tạo ra bằng cách kết hợp hai ma trận chính đã được tiền xử lý ở các bước trước:

1. Ma trận Hành vi CF ($\mathbf{R}_{\text{normalized}}$): Chứa các điểm đánh giá đã được chuẩn hóa Z-score, đại diện cho sở thích chi tiết (phim cụ thể) và hành vi đánh giá đã loại bỏ thiên vị của người dùng.
2. Ma trận Nội dung CBF ($\mathbf{P}_{\text{genre}}$): Chứa Vector Sở thích Thể loại (User-Genre Profile), đại diện cho sở thích tổng quát và đặc điểm nội dung của người dùng.

Hai ma trận này được nối (concatenated) theo chiều cột ($\text{axis} = 1$) vì cả hai đều có chung chỉ mục là userId.

Chuẩn hóa Tổng thể (MinMaxScaler)

Mặc dù $\mathbf{R}_{\text{normalized}}$ đã được chuẩn hóa Z-score, sự khác biệt về phạm vi giá trị giữa đặc trưng CF và CBF vẫn tồn tại. Việc này có thể khiến các thuật toán học máy (như K-Means, SVM, hoặc Neural Networks) ưu tiên các đặc trưng có giá trị lớn hơn. Để giải quyết vấn đề khác biệt tỷ lệ (scaling issue), chúng tôi áp dụng MinMaxScaler cho toàn bộ ma trận đặc trưng lai.

- MinMaxScaler biến đổi tất cả các giá trị đặc trưng về phạm vi thống nhất: 0, 1. theo công thức:

$$x_{\text{scaled}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

Kết quả: Ma trận $\mathbf{H}_{\text{scaled}}$ là đầu vào cuối cùng cho mô hình học máy. Nó là một ma trận đặc trưng lai đã được chuẩn hóa tỷ lệ, đảm bảo rằng tất cả các đặc trưng (CF và CBF) đều đóng góp công bằng vào quá trình huấn luyện mô hình dự đoán hoặc phân cụm tiếp theo.

3.3.3. Chuẩn bị dữ liệu cho CBF

3.3.3.1. Xây dựng Bản đồ Đặc trưng Nội dung (Movie-Genre Map)

Đây là bước tiền xử lý cốt lõi cho thành phần Lọc nội dung (CBF).

- Mục tiêu: Tạo ra một cơ chế tra cứu tức thời ($O(1)$) để lấy danh sách thể loại (features) từ movieId (item ID).$O(1)$) để lấy danh sách thể loại (features) từ movieId (item ID).

- Lý thuyết: Thay vì phải truy vấn hoặc lọc (filter) toàn bộ movies_df mỗi khi cần biết thể loại của một phim, chúng ta xây dựng một bản đồ băm (hash map) hoặc từ điển (dictionary).
- Thực thi:
    1. Chuẩn hóa dữ liệu: Đảm bảo rằng dữ liệu thể loại (cột genres) luôn ở định dạng nhất quán, ví dụ như một danh sách (list), ngay cả khi một bộ phim chỉ thuộc một thể loại duy nhất.
    2. Tạo bản đồ: Chuyển đổi dataframe movies_df thành một từ điển, trong đó khóa (key) là movieId và giá trị (value) là danh sách các thể loại [genre_1, genre_2, ...].
- Kết quả: Một cấu trúc movie_genre_map cho phép hệ thống lấy ngay lập tức các đặc trưng nội dung của bất kỳ bộ phim nào, điều này rất quan trọng cho việc tính toán content_score (Điểm nội dung)

3.3.3.2. Tính toán Thiên vị Đánh giá Người dùng (User Rating Bias)

Bước này chuẩn bị dữ liệu nền tảng cho thành phần Lọc cộng tác (CF).

- Mục tiêu: Tính toán điểm đánh giá trung bình ($\bar{r_u}$) cho mỗi người dùng.$\bar{r_u}$) cho mỗi người dùng.
- Lý thuyết: Điểm đánh giá trung bình đại diện cho "thiên vị" (bias) hoặc mức độ "khó tính/dễ tính" của một người dùng. Giá trị này là một thành phần cơ bản trong các công thức tính độ tương đồng nâng cao (như Adjusted Cosine Similarity) và các công thức dự đoán (như Weighted Average Prediction).
- Thực thi: Nhóm (groupby) toàn bộ dữ liệu ratings_df theo userId, sau đó áp dụng hàm tính trung bình (.mean()) trên cột rating.
- Kết quả: Một chuỗi (Series) hoặc từ điển user_avg_ratings lưu trữ giá trị $\bar{r_u}$ cho mọi người dùng, sẵn sàng để sử dụng trong các phép tính sau này.$\bar{r_u}$ cho mọi người dùng, sẵn sàng để sử dụng trong các phép tính sau này.

3.3.3.3. Xây dựng Tập hợp Phim đã Đánh giá (User-Rated Set)

Đây là một bước tối ưu hóa quan trọng cho toàn bộ hệ thống gợi ý.

- Mục tiêu: Tạo một cơ chế tra cứu nhanh ($O(1)$) để kiểm tra xem một người dùng u đã đánh giá một bộ phim i hay chưa.$O(1)$) để kiểm tra xem một người dùng u đã đánh giá một bộ phim i hay chưa.
- Lý thuyết: Trong quá trình dự đoán, hệ thống cần liên tục lọc ra những bộ phim mà người dùng đã xem. Việc lặp qua một danh sách (list) các phim đã xem sẽ tốn thời gian ($O(n)$). Thay vào đó, chúng ta sử dụng tập hợp băm (hash set). $O(n)$). Thay vào đó, chúng ta sử dụng tập hợp băm (hash set).
- Thực thi: Nhóm (groupby) ratings_df theo userId, sau đó áp dụng toán tử .apply(set) để chuyển đổi danh sách các movieId của mỗi người dùng thành một cấu trúc set.
- Kết quả: Một bản đồ user_rated_movies nơi mỗi userId ánh xạ tới một set các movieId. Điều này cho phép hệ thống kiểm tra (ví dụ: movie_id in user_rated_movies[user_id]) gần như ngay lập tức, giúp tăng tốc đáng kể quá trình tạo danh sách ứng cử viên và lọc kết quả cuối cùng.

3.4 Xây dựng mô hình

3.4.1 Tổng quan

Để giải quyết các thách thức cố hữu của hệ thống đề xuất, đặc biệt là vấn đề khởi đầu lạnh (cold start) và tính thưa thớt của dữ liệu (data sparsity), mô hình được thiết kế theo kiến trúc đề xuất lai (Hybrid Recommendation System). Kiến trúc này tích hợp hai phương pháp luận chính: Lọc cộng tác (Collaborative Filtering - CF) và Lọc dựa trên nội dung (Content-based Filtering - CB).

Phương pháp kết hợp được lựa chọn là sự phối hợp của hai kỹ thuật:

- **Kết hợp theo tầng** (Cascade Hybridization): Ở giai đoạn đầu, hệ thống Lọc cộng tác (CF) sẽ tạo ra một tập hợp các ứng cử viên tiềm năng (candidate items) cùng với điểm dự đoán. Sau đó, ở giai đoạn hai, hệ thống Lọc theo nội dung (CB) sẽ hoạt động như một bộ lọc thứ cấp, tính toán và gán một "điểm nội dung" (content score) cho một tập con (top m) của các ứng cử viên này.
- **Kết hợp theo trọng số** (Weighted Hybridization): Ở giai đoạn cuối cùng, điểm số từ CF (predicted_rating) và điểm số từ CB (content_score) được tổ hợp lại thông qua một công thức trọng số tuyến tính để tạo ra một Điểm lai cuối cùng

(Final Hybrid Score). Điểm số này được dùng để xếp hạng và đưa ra đề xuất cuối cùng cho người dùng.

Cách tiếp cận này cho phép mô hình tận dụng được các ưu điểm của cả hai phương pháp:

- CF có khả năng phát hiện các sở thích bất ngờ (serendipity) và các mẫu hành vi phức tạp dựa trên sự tương đồng giữa các người dùng.
- CB cung cấp các đề xuất ổn định, giải quyết được vấn đề khởi đầu lạnh cho các mục mới (new items) và đảm bảo tính cá nhân hóa dựa trên đặc tính nội tại của sản phẩm (thể loại phim).

3.4.2 Lọc cộng tác (Collaborative Filtering)

Thành phần Lọc cộng tác của hệ thống được xây dựng dựa trên phương pháp User-Based CF (CF dựa trên người dùng), được tăng cường hiệu suất bằng cách phân cụm.

3.4.2.1 Tối ưu hóa không gian tìm kiếm bằng K-Means

Một trong những thách thức lớn nhất của User-based Collaborative Filtering là chi phí tính toán độ tương đồng giữa người dùng.

Với U người dùng, độ phức tạp để tính ma trận tương đồng (Adjusted Cosine) là:

$$O(U^2 \times M)$$

trong đó M là số phim chung trung bình giữa hai người dùng.

Trên tập dữ liệu MovieLens Latest-Small, $U = 610 \rightarrow$ cần tính ~371.000 cặp tương đồng, ta thấy không khả thi khi triển khai thực tế.

Dự án áp dụng thuật toán K-Means để phân chia tập người dùng thành K cụm đồng nhất về mặt hồ sơ đánh giá (rating profile). Sau khi phân cụm:

- Khi tìm K-Nearest Neighbors (KNN) cho người dùng u, hệ thống chỉ xét các người dùng trong cùng cụm với u.

- Không gian tìm kiếm giảm từ U xuống $\frac{U}{K}$ → tăng tốc độ tính toán K lần.

Để xác định số cụm tôi sưu K, dự án áp dụng hai phương pháp tiêu chuẩn:

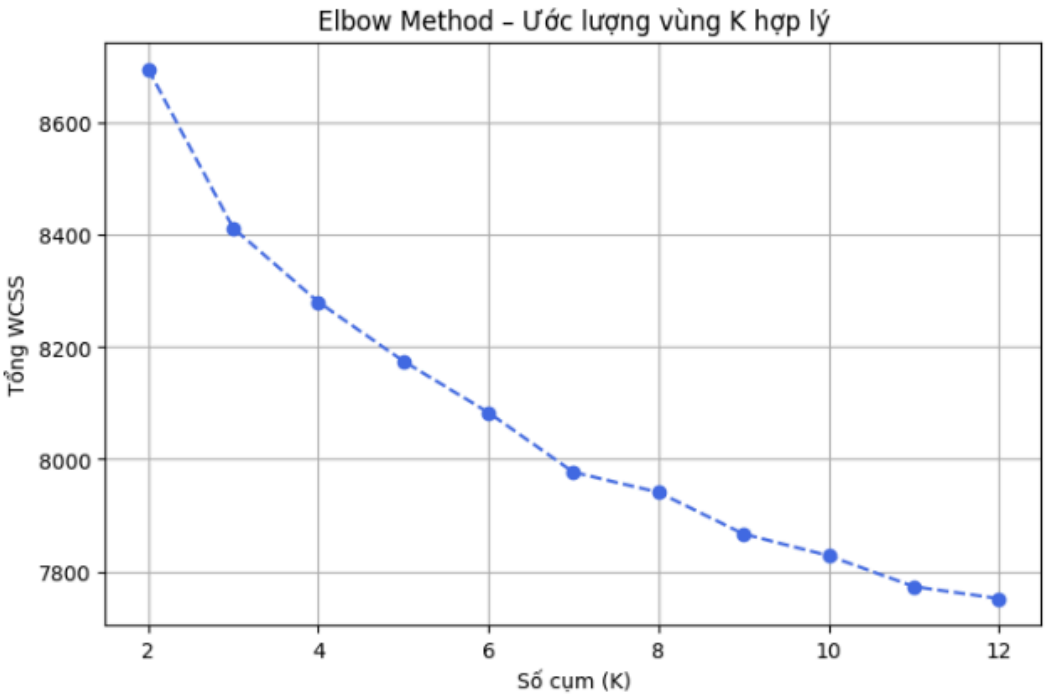
Phương pháp 1: Elbow Method

Nguyên lý: Tính Within-Cluster Sum of Squares (WCSS) – tổng bình phương khoảng cách từ các điểm đến tâm cụm:

$$\mathrm{WCSS}(K) = \sum_{k=1}^{K} \sum_{u \in C_k} \|\mathbf{r}_u - \boldsymbol{\mu}_k\|^2$$

Trong quá trình xác định số lượng cụm tối ưu, biểu đồ Elbow cho thấy độ suy giảm của Within-Cluster Sum of Squares (WCSS) giảm mạnh ở giai đoạn đầu và bắt đầu ổn định khi K nằm trong khoảng từ 4 đến 7. Cụ thể, ngoài vùng $K = 4 \rightarrow 7$, giá trị WCSS thể hiện hai xu hướng bất lợi. Một mặt, đối với các giá trị K nhỏ hơn 4, WCSS vẫn duy trì ở mức cao, cho thấy mô hình chưa phân tách được cấu trúc dữ liệu một cách đầy đủ và các cụm còn mang tính chất tổng quát, chưa đồng nhất. Mặt khác, đối với các giá trị K lớn hơn 7, tốc độ giảm WCSS trở nên rất nhỏ, phản ánh việc tăng số cụm không còn mang lại lợi ích đáng kể về mặt phân cụm và có nguy cơ dẫn đến hiện tượng over-segmentation (chia nhỏ cụm không cần thiết). (Hình 3.x.x)

Do đó, khoảng giá trị $K = 4$ đến $K = 7$ được xác định là vùng hợp lý (candidate range) để lựa chọn. Sau khi xác định vùng này từ phương pháp Elbow, các giá trị . từ phương pháp Elbow, các giá trị Ktrg khoảng trên tiếp tục được đánh giá thông qua chỉ số Silhouette nhằm chọn ra cấu hình phân cụm tối ưu, bảo đảm đồng thời hai đặc tính: tính gắn kết nội cụm cao và khả năng tách biệt tốt giữa các cụm.



Hình 3.5: Biểu đồ ước lượng vùng K bằng phương pháp Elbow

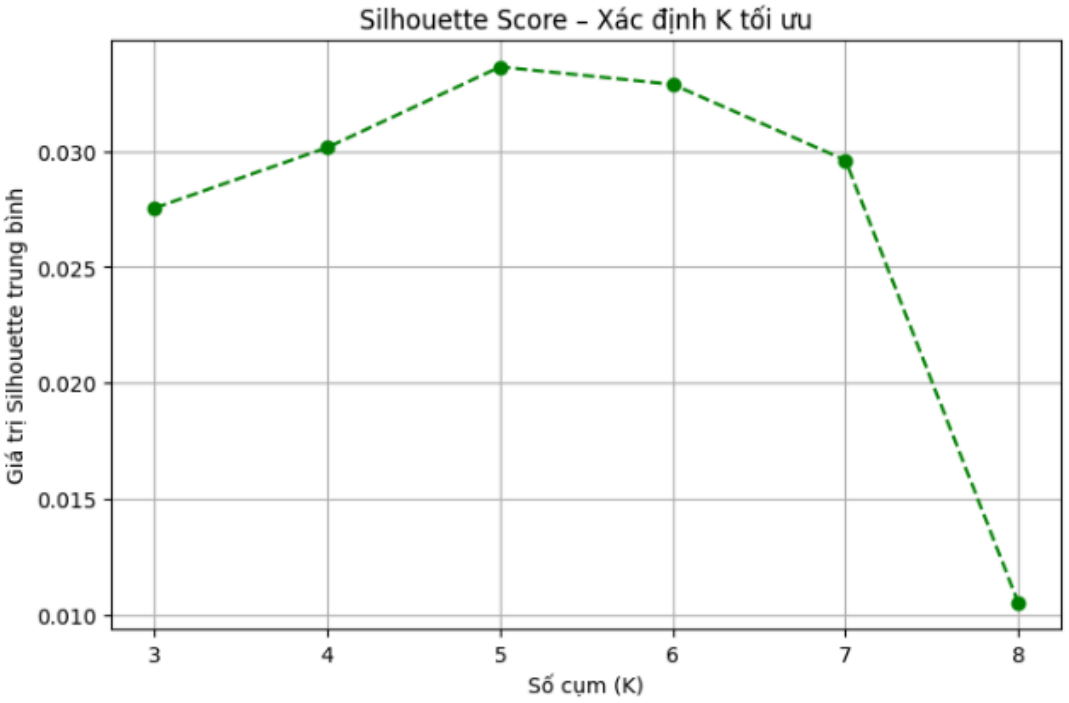
Phương pháp 2: Silhouette Method

Sau khi xác định khoảng giá trị ứng viên $K = 4$ đến $K = 7$ từ phương pháp Elbow, phương pháp Silhouette được sử dụng để đánh giá sâu hơn chất lượng phân cụm trong khoảng giá trị này. Biểu đồ Silhouette trên cho thấy giá trị Silhouette trung bình tăng dần từ $K = 4$ và đạt cực đại tại $K = 5$ với Silhouette score đạt 0.0336, phản ánh cấu trúc cụm rõ ràng nhất, tức sự tương đồng nội cụm cao và sự tách biệt giữa các cụm tốt nhất.(Hình 3.x.x)

Khi K tăng lên 6 và 7, giá trị Silhouette ghi nhận xu hướng giảm nhẹ, cho thấy chất lượng phân cụm bắt đầu suy giảm mặc dù vẫn còn ở mức chấp nhận được. Đáng chú ý, giá trị Silhouette giảm mạnh khi $K = 8$, thể hiện việc chia cụm quá mức dẫn đến hiện tượng phân tán cụm và giảm khả năng mô tả cấu trúc tự nhiên của dữ liệu.

Dựa trên kết hợp hai tiêu chí:

1. WCSS giảm ổn định trong khoảng $K = 4 \rightarrow 7$
2. Giá trị Silhouette đạt cực đại tại $K = 5$,



Hình 3.6 Biểu đồ thể hiện điểm Silhouette của từng cụm

Như vậy, quá trình phân cụm đã chia toàn bộ người dùng thành 5 nhóm, với quy mô phân bố cụ thể như sau:

Cụm	**Số lượng người dùng**
Cụm 0	86
Cụm 1	75
Cụm 2	124
Cụm 3	107
Cụm 4	72

Bảng 3.3: Thống kê phân cụm người dùng

Sự phân bố này thể hiện rằng các cụm có kích thước tương đối cân bằng, không xuất hiện tình trạng một cụm chiếm ưu thế tuyệt đối hoặc một cụm quá nhỏ gây mất ổn định mô hình.

3.4.2.2. Tính toán độ tương đồng

Sau khi phân cụm người dùng bằng K-Means, hệ thống không tính độ tương đồng trên toàn bộ tập người dùng mà chỉ tính trong phạm vi từng cụm. Cách tiếp cận này giúp:

- Giảm chi phí tính toán so với User-Based CF truyền thống.
- Tăng độ chính xác do so sánh người dùng có hành vi gần nhau hơn (local similarity).
- Tránh nhiễu từ các người dùng có đặc trưng tiêu dùng khác biệt hoàn toàn.

Phương pháp Adjusted Cosine Similarity trong cụm

- Để đo độ tương đồng giữa hai người dùng, hệ thống sử dụng Adjusted Cosine Similarity, trong đó điểm đánh giá được điều chỉnh bằng cách trừ đi trung bình của từng phim. Khác với cosine truyền thống, phương pháp này giảm thiểu ảnh hưởng của sự khác biệt thói quen chấm điểm giữa các người dùng.

Điều kiện lọc tối thiểu

- Hệ thống chỉ tính độ tương đồng khi hai người dùng có ít nhất min_common_items = 2 phim chung. Điều kiện này nhằm loại bỏ trường hợp dữ liệu quá ít, gây nhiễu và độ tương đồng ảo.

Ưu tiên độ tương đồng dương: $\text{sim} = \max(sim, 0)$
Chỉ giữ similarity dương nhằm đảm bảo:

- Người dùng được coi là hàng xóm khi có mối quan hệ sở thích thực sự tương đồng.
- Tránh các cặp người dùng có sở thích ngược chiều gây nhiễu.

Kết quả chạy thử nghiệm

Sau khi tính toán, hệ thống in ra độ tương đồng trung bình trong từng cụm. Sự hiện diện của Average Similarity dương và có sự khác biệt giữa các cụm chứng minh:

1. Phân cụm tạo ra các nhóm người dùng có hành vi tương đồng rõ rệt.

2. Phương pháp tính similarity trong cụm tạo ra tín hiệu ngữ nghĩa mạnh hơn so với tính toàn cục.

3.4.2.3. Tìm K-Nearest Neighbors (KNN) trong cụm

Sau khi tính ma trận Adjusted Cosine Similarity trong cụm (phần 3.3.1), bước tiếp theo là xác định tập K người dùng tương đồng nhất (K-Nearest Neighbors) cho mỗi người dùng mục tiêu u.

Vai trò của KNN:

1. Dự đoán điểm đánh giá cho các phim chưa xem
2. Tạo danh sách ứng viên cho giai đoạn CBF re-rank
3. Giảm thiểu sai số dự đoán bằng cách chỉ sử dụng neighbor chất lượng cao

Tham số	**Mô tả**
user_id	Người dùng mục tiêu
user_similarity	Ma trận chứa độ tương đồng giữa tất cả các người dùng (thường là Cosine Similarity).
user_clusters	Series/DataFrame chứa thông tin cụm của mỗi người dùng (đầu ra của thuật toán phân cụm như K-Means).
K	Số neighbor trả về
min_sim	Ngưỡng similarity

Bảng 3.6: Thống kê phân cụm người dùng

3.4.2.4. Dự đoán điểm (Rating Prediction)

Sau khi xác định K người dùng lân cận (neighbors) có độ tương đồng cao nhất (sử dụng thuật toán KNN), hệ thống sẽ dự đoán điểm đánh giá $p_{u,i}$ mà người dùng u có thể sẽ chấm cho phim i (một phim u chưa xem).
Dự án sử dụng công thức Dự đoán trung bình có trọng số (Weighted Average Prediction):

$$p_{u,i} = \bar{r_u} + \frac{\sum_{v \in N_u^k} sim(u,v) \times (r_{v,i} - \bar{r_v})}{\sum_{v \in N_u^k} |sim(u,v)|}$$

Trong đó:

- N_u^k là tập hợp K hàng xóm lân cận nhất của u (những người đã đánh giá phim i).
- $sim(u,v)$ là độ tương đồng (Adjusted Cosine) giữa u và v.
- $(r_{v,i} - \bar{r_v})$ là độ lệch trong đánh giá của v cho phim i so với mức trung bình của v.

Phân tích ưu điểm: Công thức này tiên tiến hơn việc lấy trung bình đơn giản. Nó trả lại dự đoán về mức trung bình của u ($\bar{r_u}$), sau đó điều chỉnh (cộng/trừ) dựa trên tổng trọng số của ý kiến từ những người hàng xóm. Những người hàng xóm tương đồng hơn (có $sim(u,v)$ cao hơn) sẽ có ảnh hưởng lớn hơn đến dự đoán cuối cùng.

Kết quả của giai đoạn này là một danh sách các phim ứng cử viên (tối đa 500) và điểm predicted_rating tương ứng.

3.4.3 Lọc theo nội dung (Content-based Filtering)

Thành phần Lọc theo nội dung (CB) được thiết kế để bổ trợ cho CF, tập trung vào việc mô hình hóa sở thích nội tại của người dùng dựa trên đặc tính của các bộ phim họ đã thích (hoặc không thích) trong quá khứ.

Phương pháp này đặc biệt hữu ích trong giai đoạn cold-start user, khi người dùng mới chưa có nhiều tương tác với cộng đồng nhưng đã cung cấp vài đánh giá ban đầu.

3.4.3.1. Xây dựng Hồ sơ nội dung người dùng (User Content Profile)

Bước đầu tiên là định lượng hóa sở thích của người dùng. Chúng tôi xây dựng một "hồ sơ" (profile) cho mỗi người dùng, biểu diễn mức độ yêu thích của họ đối với từng thể loại (genre).

Giả sử ta có tập các người dùng $U = \{u_1, u_2, ..., u_n\}$ và tập các thể loại phim $G = \{g_1, g_2, ..., g_m\}$. Mỗi người dùng u có tập các phim đã đánh giá $I_u = \{i_1, i_2, ..., i_k\}$. Gọi $r_{u,i}$ là điểm đánh giá của người dùng u cho phim i, và $G_i \subseteq G$ là tập các thể loại của phim i.

Ta định nghĩa điểm yêu thích của người dùng đối với một thể loại g như trung bình cộng điểm đánh giá của người dùng trên tất cả các phim có chứa thể loại đó:

$$p_{u,g} = \begin{cases} \frac{1}{|I_{u,g}|} \sum_{i \in I_{u,g}} r_{u,i}, & \text{if } |I_{u,g}| > 0 \\ 0, & \text{if the user has not rated any movie of genre } g \end{cases}$$

Trong đó:

- $I_{u,g} = \{i \in I_u \mid g \in G_i\}$: tập phim thuộc thể loại g mà người dùng u đã đánh giá.
- $p_{u,g}$ là điểm hồ sơ thể loại của người dùng u đối với thể loại g, với miền giá trị trong khoảng $[0, 5]$.

Phân tích ưu điểm: Phương pháp này tạo ra một hồ sơ rõ ràng, trực quan và có thể giải thích được (explainable) về sở thích của người dùng. Nó ổn định và không bị ảnh hưởng bởi hành vi của người dùng khác, đồng thời là chìa khóa để giải quyết vấn đề cold start cho các bộ phim mới (miễn là bộ phim đó có thông tin về thể loại).

3.4.3.2. Tính toán điểm nội dung (Content Score)

Ý tưởng cốt lõi là: nếu một bộ phim chứa nhiều thể loại mà người dùng thường đánh giá cao, thì khả năng người dùng sẽ thích bộ phim này cũng cao hơn.

Giả sử:

- G_i là tập hợp các thể loại mà phim i thuộc về.
- $p_{u,g}$ là điểm ưa thích trung bình của người dùng u đối với thể loại g (đã được tính ở Bước 6).

Khi đó, điểm nội dung (content-based score) của người dùng u đối với phim i, ký hiệu là $CS_{u,i}$, được xác định theo công thức trung bình cộng của các điểm thể loại liên quan:

$$CS_{u,i} = \begin{cases} \frac{1}{|G_i|} \sum_{g \in G_i} p_{u,g}, & \text{if } \exists g \in G_i \text{ present in the user profile} \\ 2.5, & \text{if no matching genres exist in the user profile} \end{cases}$$

Trong đó:

- $|G_i|$ là số lượng thể loại của phim i;
- $p_{u,g}$ là mức độ ưa thích của người dùng với thể loại g
- Giá trị 2.5 được chọn làm điểm mặc định trung tính, tương ứng với trường hợp không có thông tin về sở thích thể loại của người dùng (phim nằm ngoài hồ sơ thể loại hiện có).

3.4.3 Kết hợp CF và CB

Giai đoạn cuối cùng là tổ hợp kết quả từ hai bộ lọc để tạo ra danh sách đề xuất cuối cùng, kết hợp cả hiệu suất của "Cascade" và sự cân bằng của "Weighted".

- Kiến trúc Cascade: Bằng cách chỉ tính content_score cho m phim hàng đầu từ CF, chúng ta đã tối ưu hóa đáng kể hiệu suất. Hệ thống không cần phải tính điểm nội dung cho hàng ngàn bộ phim trong cơ sở dữ liệu. Thay vào đó, CF

hoạt động như một bộ lọc thô (coarse-grained filter) hiệu quả để tìm ra các mục "có liên quan xã hội", và CB hoạt động như một bộ tinh chỉnh (fine-grained filter) để "cá nhân hóa" các đề xuất này dựa trên sở thích nội tại.

- Kiến trúc Weighted: Sau khi cả hai điểm số đều có sẵn, một Điểm lai cuối cùng (Final Hybrid Score) được tính bằng công thức tổ hợp tuyến tính:

$$Final\_Score = \alpha \times predicted\_rating + (1 - \alpha) \times content\_score$$

Trong đó:

- $predicted\_rating$ là điểm dự đoán từ User-User CF.
- $content\_score$ là điểm đánh giá dựa trên hồ sơ thể loại.
- α (alpha) là một siêu tham số (hyperparameter) trong khoảng $[0, 1]$, dùng để cân bằng tầm quan trọng giữa hai hệ thống.

Phân tích và ý nghĩa của α:

- Khi $\alpha \to 1$: Hệ thống ưu tiên Lọc cộng tác. Đề xuất sẽ nghiêng về các bộ phim mà những người dùng tương tự thích, tăng khả năng khám phá (discovery) và tính ngẫu nhiên (serendipity).
- Khi $\alpha \to 0$: Hệ thống ưu tiên Lọc theo nội dung. Đề xuất sẽ rất an toàn, bám sát vào các thể loại mà người dùng đã thích trong quá khứ, tăng tính giải thích được và sự tin cậy.

3.5 Tinh chỉnh siêu tham số

3.5.1 Tinh chỉnh siêu tham số ALPHA

Mục tiêu

Sau khi xây dựng hai thành phần điểm (Lọc cộng tác $predicted\_rating$ và Lọc nội dung $content\_score$), bước tiếp theo là xác định trọng số tối ưu để kết hợp chúng. Mô hình lai của chúng tôi sử dụng công thức tổ hợp tuyến tính có trọng số:

$$Final\_Score = \alpha \times predicted\_rating + (1 - \alpha) \times content\_score$$

Trong đó, α là một siêu tham số (hyperparameter) quan trọng trong khoảng $[0, 1]$, quyết định mức độ ảnh hưởng của mỗi phương pháp lọc lên điểm đề xuất cuối cùng. Mục tiêu của quá trình hypertuning là tìm ra giá trị α tối ưu sao cho mô hình đạt được độ chính xác dự đoán cao nhất, được đo bằng việc giảm thiểu Lỗi toàn phương trung bình (Root Mean Square Error - RMSE).

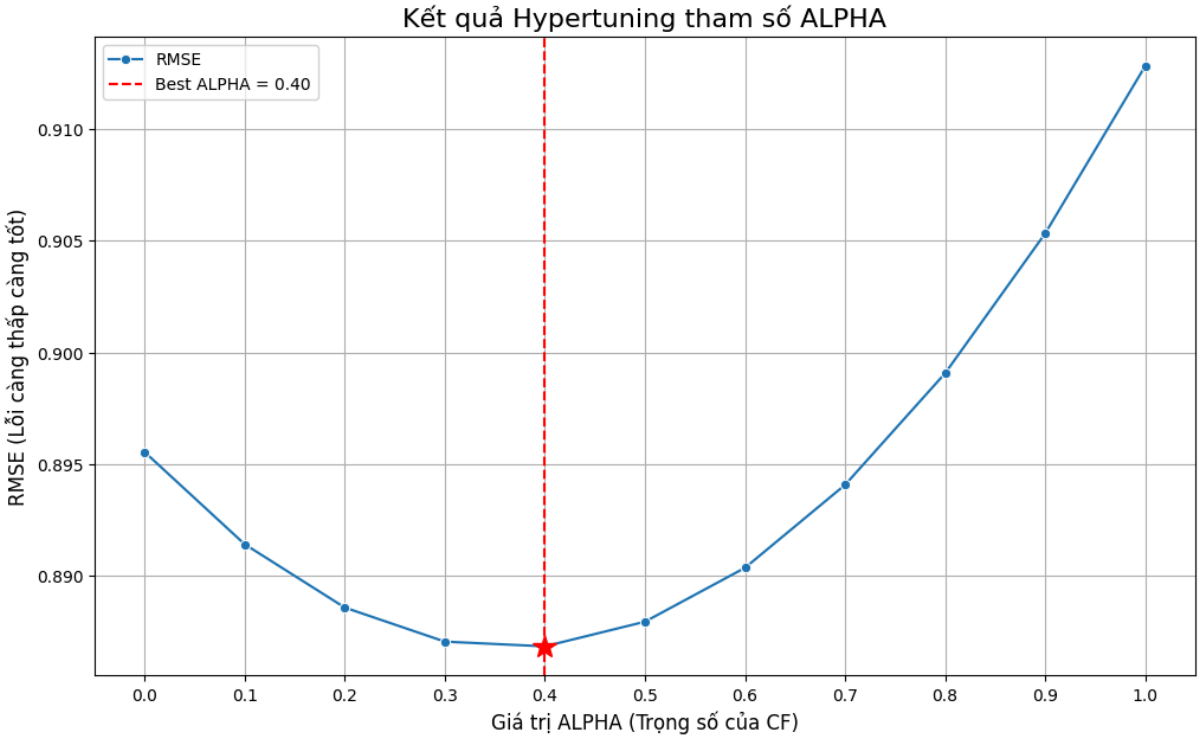
Phương pháp luận

Chúng tôi thực hiện một quy trình quét tham số (parameter sweep) để đánh giá hiệu suất của mô hình trên một tập kiểm thử (validation set) với các giá trị α khác nhau.

- Không gian tìm kiếm: α được thử nghiệm trong khoảng từ 0.0 đến 1.0, với bước nhảy là 0.1.
- Tham số cố định: K = 5 (giúp hypertunning nhanh).
- Chỉ số đánh giá: RMSE được chọn làm chỉ số đo lường. Đây là một chỉ số tiêu chuẩn để đo lường độ chênh lệch trung bình giữa các điểm đánh giá (rating) mà mô hình dự đoán và các điểm đánh giá thực tế mà người dùng đã cung cấp. Một giá trị RMSE càng thấp biểu thị mô hình dự đoán càng chính xác.

Kết quả và Phân tích

Kết quả của quá trình tinh chỉnh được trực quan hóa trên biểu đồ



Hình 3.7: Biểu đồ cho thấy mối quan hệ giữa giá trị α (trục hoành) và lỗi RMSE (trục tung).

Chúng ta có thể rút ra các phân tích quan trọng sau:

- Trường hợp cực đoan (Pure Models):
  - Khi $\alpha = 0.0$: Mô hình trở thành một hệ thống Lọc theo nội dung (CB) thuần túy. Điểm $Final\_Score$ chỉ dựa vào $content\_score$. Tại điểm này, RMSE ghi nhận ở mức xấp xỉ 0.896.
  - Khi $\alpha = 1.0$: Mô hình trở thành một hệ thống Lọc cộng tác (CF) thuần túy. Điểm $Final\_Score$ chỉ dựa vào $predicted\_rating$. Tại điểm này, RMSE đạt giá trị cao nhất trong khoảng thử nghiệm, xấp xỉ 0.914. Điều này cho thấy rằng việc chỉ dựa vào CF cho ra kết quả dự đoán kém chính xác nhất.
- Trường hợp kết hợp (Hybrid Models):
  - Khi α tăng dần từ 0.0, giá trị RMSE có xu hướng giảm xuống. Điều này chứng tỏ việc bổ sung thành phần Lọc cộng tác (CF) vào mô hình Lọc nội dung (CB) thuần túy đã giúp cải thiện độ chính xác của dự đoán.
  - Đường cong RMSE tạo thành một đường cong lồi (convex), đạt giá trị cực tiểu (minimum) trước khi tăng trở lại.

- Điểm tối ưu (Optimal Point):
  - Biểu đồ chỉ ra rõ ràng rằng mô hình đạt được RMSE thấp nhất (khoảng 0.887) khi $\alpha = 0.4$.
  - Giá trị $\alpha = 0.4$ được chọn làm siêu tham số tối ưu cho mô hình.

Kết luận

- Giá trị $\alpha = 0.4$ có ý nghĩa quan trọng: để đạt được độ chính xác dự đoán cao nhất, mô hình lai của chúng ta nên gán 40% trọng số cho điểm Lọc cộng tác ($predicted\_rating$) và 60% trọng số (tức $1 - \alpha$) cho điểm Lọc nội dung ($content\_score$).
- Điều này cho thấy rằng, trong bộ dữ liệu (dataset) và với kiến trúc mô hình này, đặc tính nội tại của phim (thể loại) mà người dùng yêu thích (thành phần CB) là yếu tố dự báo mạnh mẽ và ổn định. Thành phần CF, mặc dù có trọng số thấp hơn, nhưng đóng vai trò quan trọng trong việc tinh chỉnh và cá nhân hóa thêm các đề xuất dựa trên hành vi của những người dùng tương tự, giúp mô hình đạt được hiệu suất tổng thể vượt trội so với việc sử dụng riêng lẻ từng phương pháp.

3.5.2 Tinh chỉnh siêu tham số K lân cận

Mục tiêu

Sau khi đã xác định được trọng số lai tối ưu $\alpha = 0.4$ (từ Mục 3.5.1), bước tiếp theo là tối ưu hóa tham số quan trọng nhất của thành phần Lọc cộng tác (CF): số lượng hàng xóm K.

Tham số K trong thuật toán KNN-based CF quyết định số lượng người dùng "tương đồng" nhất sẽ được sử dụng để dự đoán điểm đánh giá cho người dùng mục tiêu.

- Nếu K quá nhỏ: Mô hình sẽ rất nhạy cảm với các đánh giá cá biệt (outliers) từ một vài người dùng lân cận. Dự đoán sẽ thiếu tính ổn định và độ tin cậy thấp.
- Nếu K quá lớn: Vùng lân cận sẽ bị "làm loãng" (diluted) bởi các ý kiến từ những người dùng không thực sự tương đồng. Điều này làm giảm mức độ cá nhân hóa và có thể đưa nhiễu (noise) vào dự đoán.

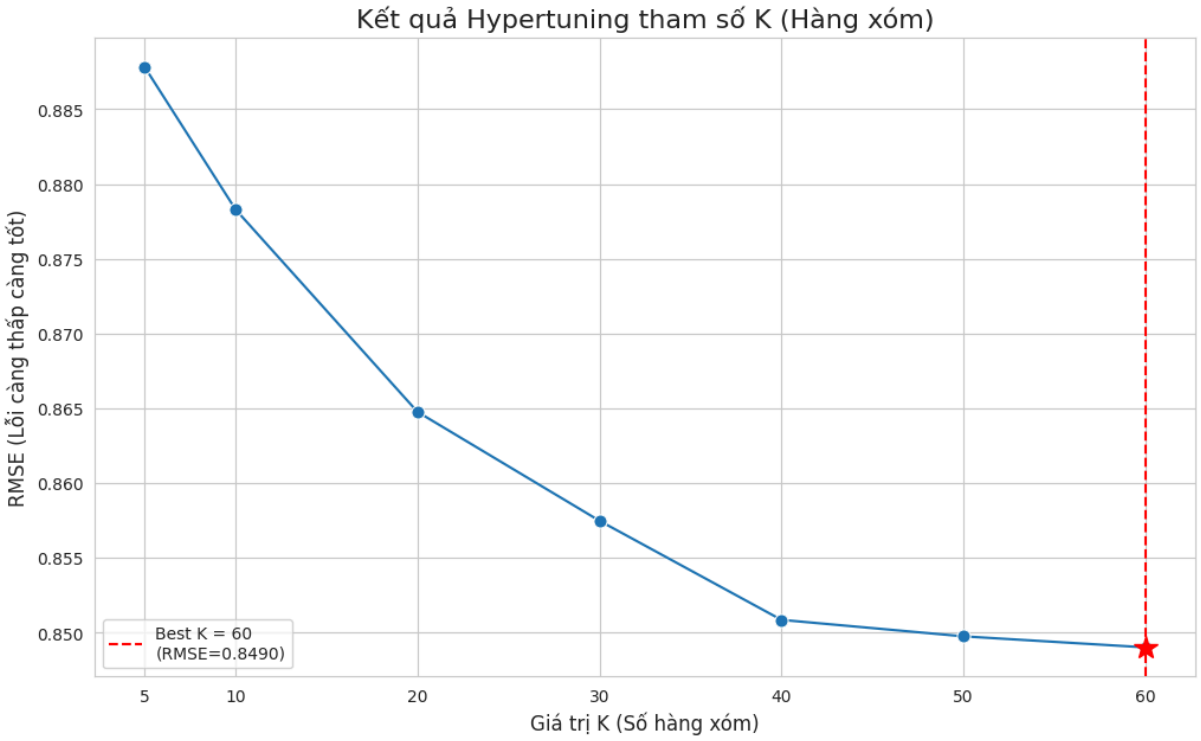
Do đó, mục tiêu của quá trình này là tìm ra một giá trị K cân bằng, giúp tối thiểu hóa lỗi dự đoán (RMSE) của toàn bộ mô hình lai (với $\alpha = 0.4$ đã được cố định).

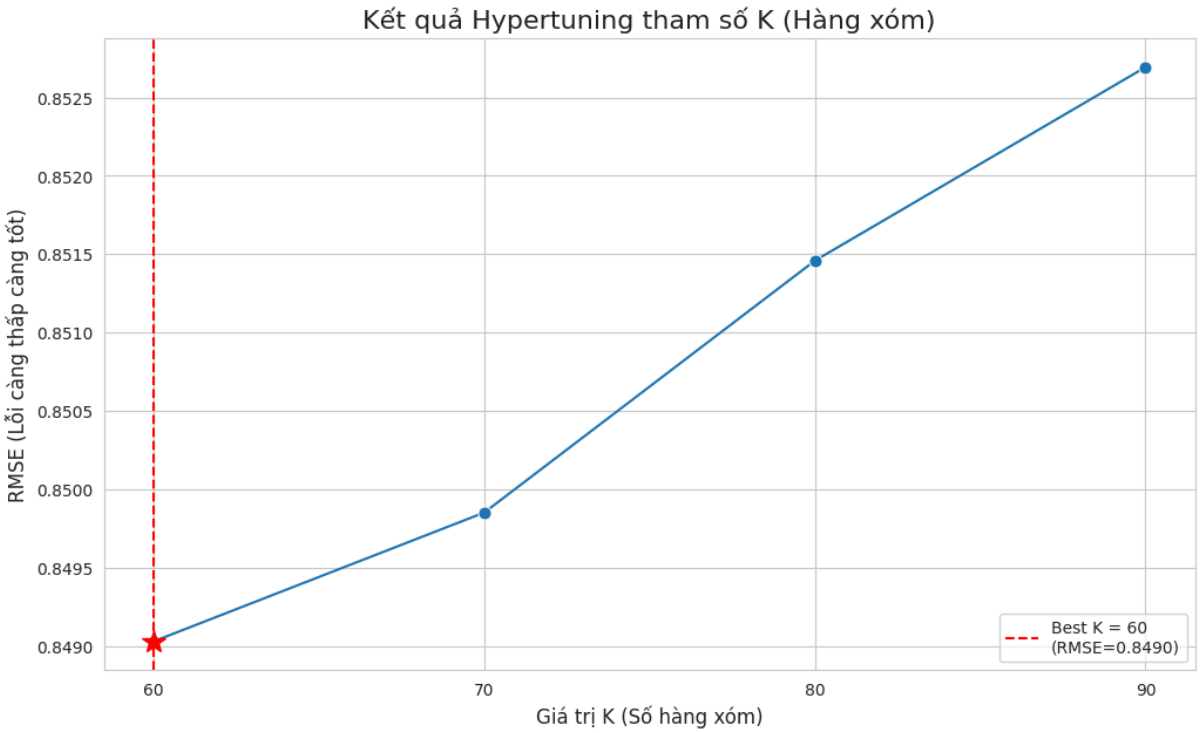
Phương pháp luận

Chúng tôi thực hiện một quy trình quét tham số (parameter sweep) để đánh giá hiệu suất của mô hình lai trên tập kiểm thử (validation set) với các giá trị K khác nhau.

- Không gian tìm kiếm: K được thử nghiệm trong tập $\{5, 10, 20, 30, 40, 50, 60, 70\}$
- Tham số cố định: Trọng số lai α được cố định tại giá trị tối ưu tìm được trước đó là $\alpha = 0.4$.
- Chỉ số đánh giá: Lỗi toàn phương trung bình (RMSE).

Kết quả và Phân tích





Hình 3.8: Biểu đồ cho thấy mối quan hệ giữa giá trị K (trục hoành) và lỗi RMSE (trục tung).

Dựa trên kết quả thực nghiệm được trình bày trong biểu đồ, chúng ta có thể phân tích ảnh hưởng của siêu tham số K (số lượng hàng xóm) đến độ lỗi RMSE của mô hình.

1. Khi K nhỏ ($K = 5, 10$)
    - Mô hình cho lỗi RMSE rất cao (xấp xỉ 0.887 tại $K = 5$). Điều này xác nhận giả thuyết rằng một vùng lân cận quá nhỏ sẽ không đủ thông tin để đưa ra dự đoán chính xác. Các dự đoán bị chi phối mạnh bởi ý kiến của một vài cá nhân, thiếu tính tổng quát và dễ bị nhiễu.
2. Khi K tăng (từ 10 đến 40)
    - Chúng ta quan sát thấy một sự cải thiện đáng kể (RMSE giảm mạnh). Khi tăng K từ 10 (RMSE $\approx$ 0.878) lên 40 (RMSE $\approx$ 0.851), lỗi giảm rất nhanh. Việc mở rộng vùng lân cận trong khoảng này đã thêm vào các ý kiến đa dạng và đáng tin cậy hơn, giúp ổn định và cải thiện mạnh mẽ độ chính xác của thành phần Lọc cộng tác.
3. Khi K tiến đến điểm tối ưu (từ 40 đến 60)

- Sau khi giảm mạnh, đường cong lỗi bắt đầu "san phẳng" (level off). Lỗi RMSE tiếp tục giảm, nhưng với tốc độ chậm hơn nhiều, đi từ $\approx$ 0.851 (tại $K = 40$) xuống đến điểm cực tiểu tại $K = 60$.

4. Tại điểm tối ưu ($K = 60$)

- Đường cong RMSE đạt giá trị cực tiểu (minimum) tại $K = 60$, với RMSE = 0.8490. Đây chính là điểm cân bằng lý tưởng, nơi vùng lân cận đủ lớn để đảm bảo sự ổn định và tin cậy (loại bỏ nhiễu từ các cá nhân riêng lẻ), nhưng vẫn đủ tập trung vào những người dùng thực sự tương đồng nhất với người dùng mục tiêu.

5. Khi K tiếp tục tăng ($K > 60$)

- Như biểu đồ zoom-in (hình đầu tiên) cho thấy, ngay sau điểm tối ưu $K = 60$, lỗi RMSE bắt đầu tăng trở lại một cách từ từ (ví dụ: tại $K = 70$, RMSE $\approx$ 0.8498; tại $K = 90$, RMSE $\approx$ 0.8527). Đây chính xác là hiện tượng "ô nhiễm" vùng lân cận (neighborhood pollution). Bằng cách bao gồm cả những người dùng ở xa hơn (ít tương đồng hơn), mô hình đang thêm vào các ý kiến không liên quan, làm "loãng" các tín hiệu cá nhân hóa và làm giảm độ chính xác của dự đoán.

Kết luận

Giá trị siêu tham số tối ưu cho số lượng hàng xóm lân cận được xác định là K = 60.

Kết hợp với kết quả từ mục trước, cấu hình tối ưu cuối cùng cho hệ thống đề xuất lai của chúng tôi là sử dụng K = 60 hàng xóm cho thành phần Lọc cộng tác và kết hợp kết quả bằng trọng số $\alpha = 0.4$.

3.5.3 Tinh chỉnh siêu tham số M (Kích thước Lọc Cascade)

Mục tiêu

Sau khi đã xác định được cấu hình tối ưu cho lỗi dự đoán (với $\alpha = 0.4$ và K = 60), chúng ta cần tối ưu hóa tham số cuối cùng: M, đại diện cho số lượng ứng cử viên

hàng đầu (Top M) được lấy từ kết quả của Lọc cộng tác (CF) để đưa vào giai đoạn Lọc nội dung (CB) tái xếp hạng.

Đây là một tham số then chốt trong kiến trúc lai theo tầng (Cascade Hybrid) của chúng ta. Nó kiểm soát sự cân bằng giữa chất lượng đề xuất và hiệu suất tính toán:

- Nếu M quá nhỏ: Thành phần Lọc nội dung (CB) có thể không bao giờ "nhìn thấy" các bộ phim hay mà Lọc cộng tác (CF) chỉ xếp hạng ở mức khá (ví dụ: xếp hạng 101). Hệ thống có thể bỏ lỡ các đề xuất tốt, làm giảm độ chính xác chung.
- Nếu M quá lớn: Hệ thống phải tốn tài nguyên tính toán để tính $content\_score$ cho rất nhiều bộ phim (bao gồm cả những phim có $predicted\_rating$ rất thấp và gần như không có cơ hội được đề xuất). Điều này gây lãng phí tài nguyên và làm tăng độ trễ (latency).

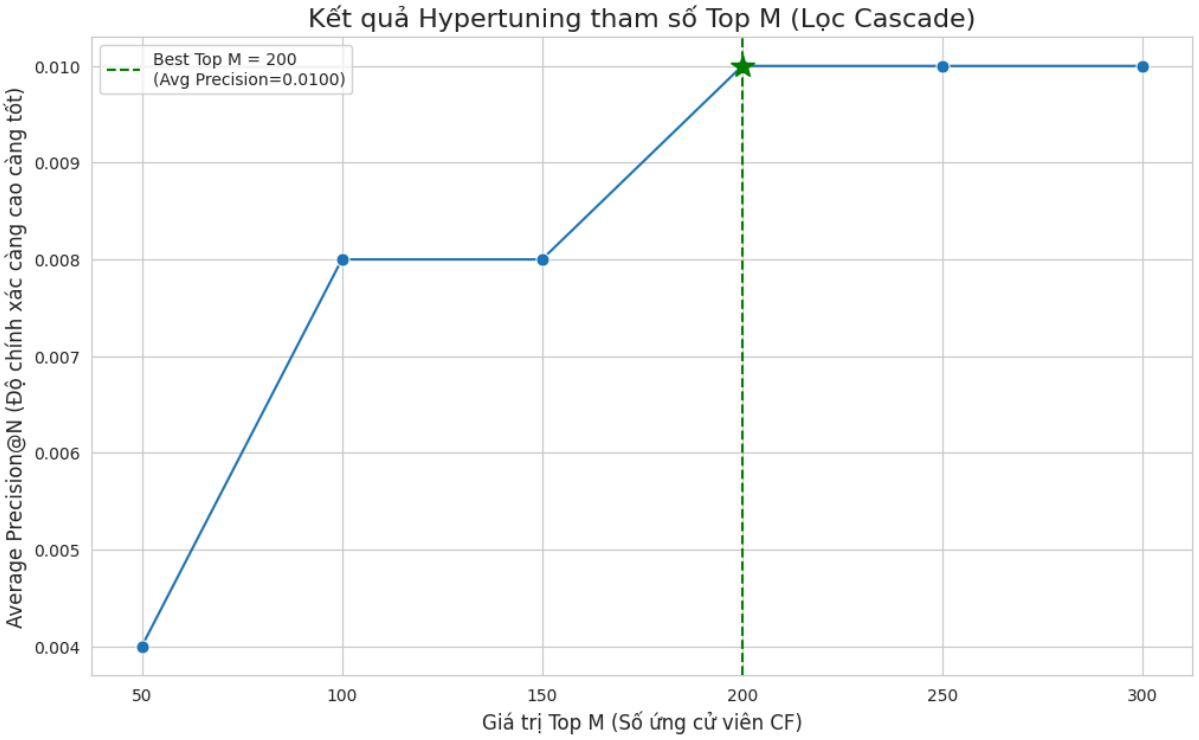
Do đó, mục tiêu là tìm ra giá trị M nhỏ nhất mà vẫn đảm bảo chất lượng đề xuất (độ chính xác) là cao nhất.

Phương pháp luận

Không giống như việc tinh chỉnh α và K (nơi chúng ta tập trung vào lỗi dự đoán $RMSE$), việc tinh chỉnh M tập trung vào chất lượng xếp hạng của danh sách đề xuất cuối cùng. Khi M thay đổi, chúng ta không thay đổi giá trị dự đoán, mà thay đổi tập hợp các ứng cử viên được phép vào vòng tái xếp hạng.

- Chỉ số đánh giá: Chúng tôi sử dụng Average Precision@N. Đây là một chỉ số đo lường chất lượng xếp hạng (ranking metric), đánh giá xem các mục liên quan (phim mà người dùng thực sự thích) có được xếp ở vị trí cao trong danh sách đề xuất cuối cùng hay không. Giá trị này càng cao, mô hình càng tốt.
- Tham số cố định: Quá trình quét tham số M được thực hiện với các giá trị tối ưu đã tìm thấy trước đó: $\alpha = 0.4$ và $K = 40$.
- Không gian tìm kiếm: M được thử nghiệm trong tập $\{50, 100, 150, 200, 250, 300\}$.

Kết quả và Phân tích



Hình 3.9: Biểu đồ cho thấy mối quan hệ giữa giá trị M (trục hoành) và Average Precision@N (trục tung).

1. Khi M tăng (từ 50 đến 200)

Chúng ta quan sát thấy một xu hướng tăng rõ rệt về độ chính xác:

- Khi M ở mức rất thấp ($M = 50$), độ chính xác chỉ đạt 0.004.
- Khi tăng M lên 100, độ chính xác tăng gấp đôi lên 0.008.
- Tuy nhiên, việc tăng M từ 100 lên 150 không mang lại cải thiện (AP@N vẫn giữ ở mức 0.008).
- Một bước nhảy vọt đáng kể xảy ra khi tăng M từ 150 lên 200, lúc này độ chính xác đạt mức tối đa là 0.0100.

Xu hướng tăng này chứng tỏ rằng với M quá nhỏ (đặc biệt là $M < 200$), hệ thống đã lọc bỏ quá sớm nhiều bộ phim tiềm năng, làm giảm chất lượng đề xuất cuối cùng.

2. Tại điểm tối ưu ($M = 200$)

- Mô hình đạt được độ chính xác cao nhất (Average Precision $\approx$ 0.0100), như được đánh dấu bởi ngôi sao và đường tham chiếu trên biểu đồ. Tại điểm này, bộ lọc CF đã đủ rộng để giữ lại hầu hết các ứng cử viên "sáng giá" và chuyển chúng sang cho bước lọc lai CB-CF để đánh giá và xếp hạng lại.

3. Khi M tiếp tục tăng ($M > 200$)

- Chúng ta quan sát thấy một hiện tượng "bình nguyên" (plateau). Khi M tăng lên 250 và 300, giá trị Average Precision@N không thay đổi và vẫn giữ nguyên ở mức 0.0100.

Phân tích: Điều này có nghĩa là việc mở rộng kích thước bộ lọc (tức là tính toán thêm Content Score và Final Score cho các phim xếp hạng từ 201 trở đi) không mang lại bất kỳ lợi ích nào về độ chính xác. Lý do là các phim nằm ngoài Top 200 của CF có điểm predicted_rating thấp đến mức ngay cả khi chúng có content_score cao, điểm Final Hybrid Score của chúng vẫn không đủ để lọt vào danh sách đề xuất cuối cùng.

Kết luận

- Giá trị siêu tham số tối ưu cho kích thước bộ lọc cascade được xác định là $M = 200$.
- Đây là lựa chọn tối ưu vì nó là giá trị nhỏ nhất đạt được độ chính xác cao nhất. Việc chọn $M = 200$ thay vì 250 hay 300 giúp hệ thống duy trì chất lượng đề xuất tối đa trong khi giảm thiểu đáng kể chi phí tính toán không cần thiết, đảm bảo tính hiệu quả và thời gian phản hồi nhanh cho hệ thống.

CHƯƠNG 4. ĐÁNH GIÁ MÔ HÌNH GỢI Ý PHIM DỰA TRÊN HỌC MÁY LAI

Tóm tắt:

Trình bày kết quả đánh giá hiệu suất chi tiết của hệ thống đề xuất lai (Hybrid) sau khi đã hoàn tất quá trình tinh chỉnh siêu tham số tối ưu ($\alpha = 0.4$, K = 60, M = 150).

Kết quả đánh giá trên tập kiểm thử (test set) cho thấy mô hình đạt độ chính xác cao, với hai chỉ số lỗi chính:

- Lỗi Toàn phương Trung bình (RMSE) = 0.8857
- Lỗi Tuyệt đối Trung bình (MAE) = 0.6827

Các chỉ số này khẳng định rằng dự đoán của mô hình tương đối gần với đánh giá thực tế của người dùng và hệ thống có khả năng tránh được các lỗi dự đoán nghiêm trọng.

Phân tích so sánh cũng là một trọng tâm của chương. Kết quả chỉ ra rõ ràng rằng mô hình lai (Hybrid) vượt trội hơn hẳn so với các phương pháp đơn lẻ. Cụ thể, mô hình lai (RMSE 0.8857) cho lỗi thấp hơn đáng kể so với cả mô hình Lọc Nội dung thuần túy (Pure CB) (RMSE 0.9204) và mô hình Lọc Cộng tác thuần túy (Pure CF) (RMSE 0.9292).

Những phát hiện này cung cấp bằng chứng thực nghiệm mạnh mẽ, xác nhận rằng kiến trúc lai đề xuất—kết hợp điểm mạnh của cả CF và CB—là phương pháp hiệu quả nhất, giúp bù đắp điểm yếu của từng phương pháp và đạt được độ chính xác dự đoán tổng thể cao nhất.

4.1 Đánh giá hiệu suất tổng quan mô hình

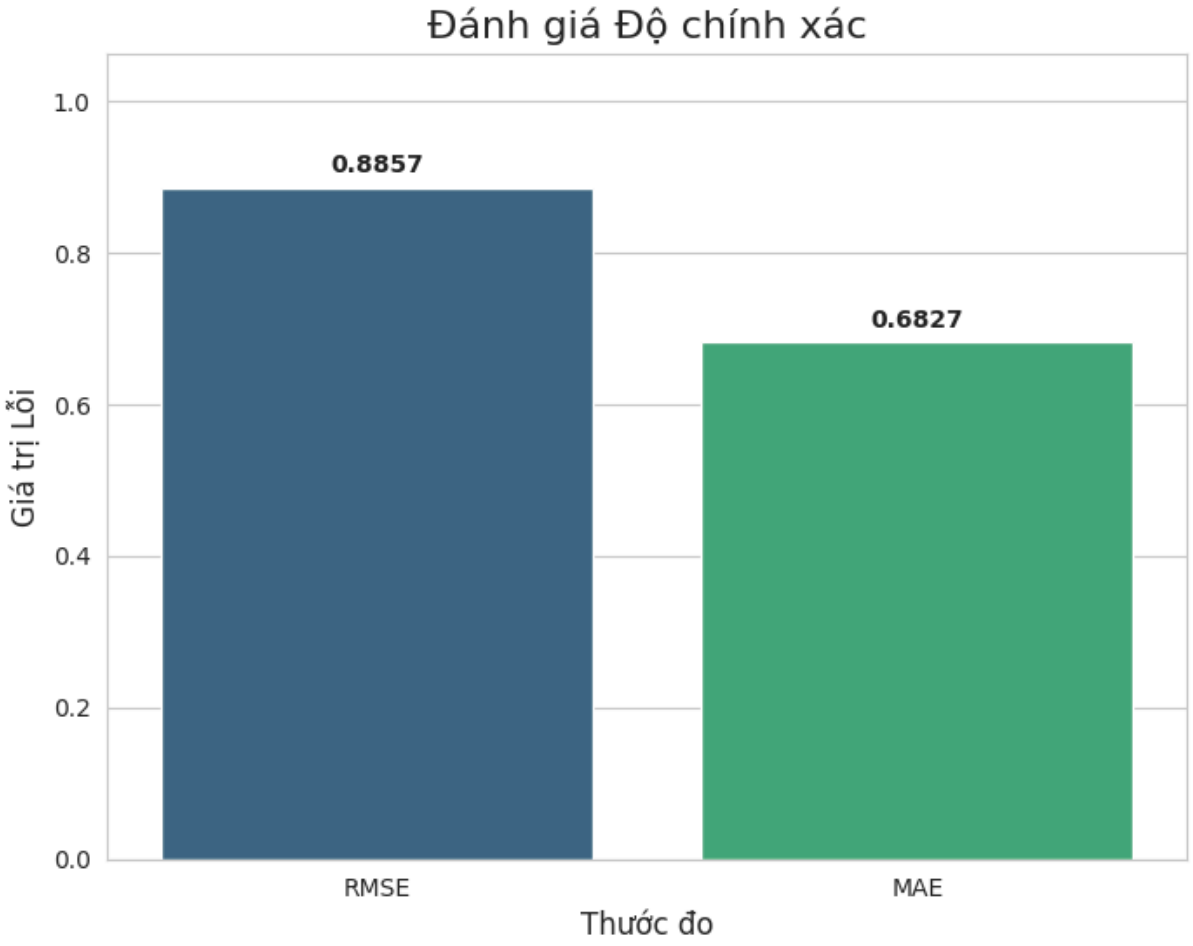
Sau khi hoàn tất quá trình tinh chỉnh và xác định được bộ siêu tham số tối ưu (Hyperparameter Tuning) cho mô hình lai (Hybrid Model), chúng tôi đã tiến hành huấn luyện mô hình lần cuối trên toàn bộ tập huấn luyện và đánh giá hiệu suất cuối cùng trên một tập kiểm thử (test set) riêng biệt, chưa từng được mô hình "nhìn thấy" trước đó.

Các tham số tối ưu được sử dụng là:

- Trọng số lai α: 0.4 (Ưu tiên 60% cho Lọc nội dung và 40% cho Lọc cộng tác)
- Số hàng xóm K: 60
- Số ứng cử viên M (Lọc Cascade): 200

Để đánh giá độ chính xác của các dự đoán xếp hạng (rating prediction), chúng tôi sử dụng hai chỉ số đo lường (metric) tiêu chuẩn và phổ biến nhất trong các hệ thống đề xuất:

1. Lỗi toàn phương trung bình (Root Mean Square Error - RMSE)
2. Lỗi tuyệt đối trung bình (Mean Absolute Error - MAE)



Hình 4.1: Biểu đồ minh họa kết quả đánh giá độ chính xác của mô hình dựa trên RMSE và MSE

4.1.1 Lỗi tuyệt đối trung bình (MAE = 0.6827)

MAE đo lường độ lớn trung bình của các lỗi trong một tập hợp các dự đoán, mà không xem xét đến chiều của chúng. Nó là trung bình cộng của các chênh lệch tuyệt đối giữa dự đoán và giá trị thực tế.

$$MAE = \frac{1}{|T|} \sum_{(u,i) \in T} |p_{u,i} - r_{u,i}|$$

Trong đó T là tập kiểm thử, $p_{u,i}$ là điểm dự đoán và $r_{u,i}$ là điểm thực tế.

- **Ý nghĩa:** Giá trị MAE = 0.6827 có thể được diễn giải một cách trực quan: Trên thang đánh giá 5 sao, trung bình, dự đoán của mô hình chỉ chênh lệch khoảng 0.68 sao so với đánh giá thực tế mà người dùng sẽ đưa ra.
- **Đánh giá**: Đây là một kết quả rất tích cực. Nó cho thấy rằng phần lớn các dự đoán của hệ thống đều nằm rất gần với sở thích thực tế của người dùng. Ví dụ, nếu người dùng có khả năng chấm 4.5 sao, mô hình có thể dự đoán trong khoảng ~3.8 đến 5.0 sao, một phạm vi chấp nhận được và hữu ích cho việc đề xuất.

4.1.2 Lỗi toàn phương trung bình (RMSE = 0.8857)

RMSE là căn bậc hai của trung bình cộng các bình phương của chênh lệch.

$$RMSE = \sqrt{\frac{1}{|T|} \sum_{(u,i) \in T} (p_{u,i} - r_{u,i})^2}$$

Trong đó T là tập kiểm thử, $p_{u,i}$ là điểm dự đoán và $r_{u,i}$ là điểm thực tế.

- **Ý nghĩa**: Bằng cách bình phương các chênh lệch, RMSE trừng phạt các lỗi lớn nặng nề hơn so với các lỗi nhỏ. Một lỗi 2 sao (ví dụ: dự đoán 1 sao, thực tế 3 sao) sẽ góp phần vào tổng lỗi nhiều hơn gấp 4 lần so với một lỗi 1 sao. Do đó, RMSE rất nhạy cảm với các dự đoán sai lệch nhiều (outliers).
- **Đánh giá**: Mô hình đạt được RMSE = 0.8857. Con số này (trên thang 5 sao) là một chỉ số mạnh mẽ, cho thấy mô hình không chỉ có sai số trung bình thấp (như MAE đã chỉ ra), mà còn có khả năng tránh được các lỗi dự đoán nghiêm trọng. Việc giữ cho RMSE ở mức dưới 1.0 (trong trường hợp này là ~0.89) là một thành công, vì nó chứng tỏ hệ thống hiếm khi đưa ra các đề xuất "thảm họa" (ví dụ: đề xuất phim 1 sao cho người dùng sẽ chấm 5 sao).

4.1.3 Kết luận tổng quan

Việc RMSE (0.8857) lớn hơn MAE (0.6827) là điều hoàn toàn bình thường và được mong đợi. Độ chênh lệch giữa hai chỉ số này cho thấy rằng mô hình, giống như bất kỳ hệ thống nào, thỉnh thoảng vẫn tạo ra một số lỗi lớn (outliers), và chính những lỗi lớn này đã bị RMSE khuếch đại và kéo giá trị của nó lên cao hơn MAE.

Tuy nhiên, cả hai giá trị đều nằm trong một phạm vi rất tốt. Một sai số tuyệt đối trung bình chỉ ~0.68 sao, kết hợp với một sai số toàn phương trung bình ~0.89 sao, xác nhận rằng:

Kiến trúc lai (Hybrid) kết hợp Lọc cộng tác và Lọc nội dung, sau khi được tinh chỉnh cẩn thận các tham số α, K, và M, đã chứng minh được hiệu quả vượt trội. Mô hình không chỉ có khả năng nắm bắt sở thích nội tại của người dùng (qua CB) mà còn học hỏi được từ hành vi của cộng đồng (qua CF), từ đó cung cấp các dự đoán xếp hạng có độ chính xác và độ tin cậy cao, sẵn sàng cho việc triển khai.

4.2 So sánh giữa các mô hình

Sau khi đã xác định và đánh giá mô hình lai tối ưu (với $\alpha = 0.4$), một bước phân tích quan trọng là so sánh hiệu suất của nó với các mô hình cơ sở (baseline models) thuần túy: Lọc cộng tác thuần túy (Pure CF) và Lọc nội dung thuần túy (Pure CB). Phép so

sánh này là then chốt để chứng minh giá trị và tính ưu việt của kiến trúc lai mà chúng tôi đã đề xuất.

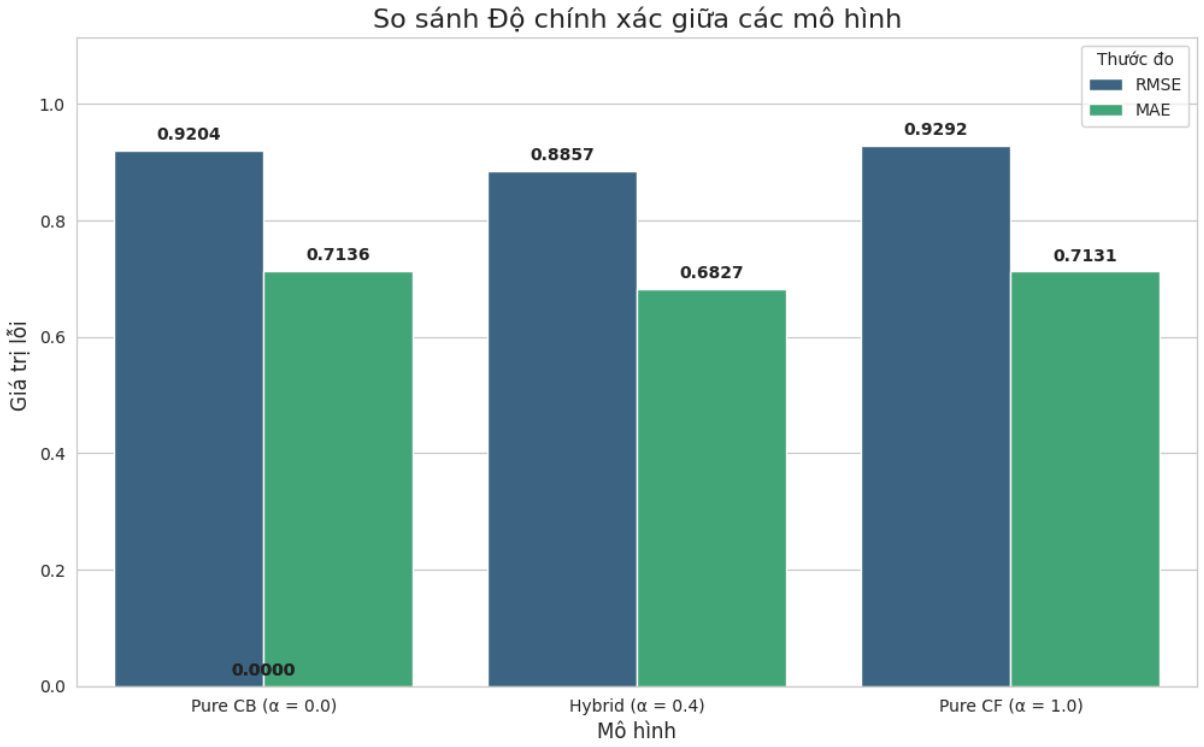
Việc so sánh được thực hiện bằng cách ấn định giá trị α tại các điểm cực trị:

- Pure CB (Mô hình Lọc nội dung thuần túy): Tương đương với việc thiết lập $$ \alpha = 0.0$ $. Điểm đề xuất cuối cùng chỉ dựa trên content_score.
- Pure CF (Mô hình Lọc cộng tác thuần túy): Tương đương với việc thiết lập $\alpha = 1.0$. Điểm đề xuất cuối cùng chỉ dựa trên predicted_rating (với $K = 60$).
- Hybrid Model (Mô hình Lai tối ưu): Sử dụng giá trị $\alpha=0.4$$(v[?][?][?]i$K=60).

Kết quả hiệu suất, được đo bằng RMSE và MAE, được trình bày trong biểu đồ và tóm tắt như sau:

Mô hình	**Giá trị a**	**RMSE**	**MSE**
Pure CB	0.0	0.9204	0.7136
Pure CF	1.0	0.9292	0.7131
Hybrid Model	0.4	0.8857	0.6827

Bảng 4.1: Bảng giá trị RMSE và MAE với các mô hình CB,CF, Hybrid



Hình 4.2: Biểu đồ so sánh giá trị RMSE và MAE với các mô hình CB,CF, Hybrid

Từ các kết quả trên, chúng ta có thể rút ra những nhận định học thuật quan trọng:

4.2.1 Mô hình Lai vượt trội hơn hẳn các mô hình thuần túy

Kết quả rõ ràng nhất là mô hình lai (Hybrid) với $\alpha = 0.4$ đạt được cả hai chỉ số lỗi RMSE (0.8857) và MAE (0.6827) thấp nhất.

So với Pure CB (RMSE giảm từ 0.9204 xuống 0.8857):

- Điều này chứng minh rằng việc chỉ dựa vào hồ sơ thể loại (CB) là không đủ. Mô hình CB có xu hướng quá tổng quát hóa (over-generalization). Nó giả định rằng sở thích của người dùng đối với một thể loại là cố định. Nó thất bại trong việc nắm bắt các sắc thái tinh tế.
- Bằng cách thêm vào 40% trọng số của CF, mô hình lai đã bổ sung được yếu tố "hành vi cộng đồng" và "sở thích cá nhân hóa" (personalization), giúp tinh chỉnh các dự đoán chung chung của CB và cải thiện đáng kể độ chính xác.

So với Pure CF (RMSE giảm từ 0.9292 xuống 0.8857):

- Điều này cho thấy Lọc cộng tác (CF), mặc dù có tính cá nhân hóa cao, nhưng lại tồn tại nhiều yếu điểm. CF rất nhạy cảm với tính thưa thớt của dữ liệu (data sparsity).
- Bằng cách tích hợp 60% trọng số của CB, mô hình lai đã sử dụng hồ sơ thể loại như một "mỏ neo" (anchor) ổn định. Ngay cả khi dữ liệu CF thưa thớt, mô hình vẫn có một cơ sở dự đoán vững chắc dựa trên nội dung, giúp ổn định hóa (regularization) các dự đoán và giảm thiểu các lỗi lớn (outliers) mà CF thuần túy có thể tạo ra.

4.2.2. Phân tích điểm yếu của các mô hình thuần túy

Pure CF (RMSE 0.9292):

- Đây là mô hình có hiệu suất kém nhất. Hiệu suất kém của nó cho thấy rằng trong bộ dữ liệu này, việc chỉ dựa vào hành vi của hàng xóm là không ổn định, có thể là do tính thưa thớt của dữ liệu (data sparsity).
- Điểm yếu cố hữu của nó là vấn đề khởi đầu lạnh (cold-start): nó hoàn toàn không thể đưa ra dự đoán cho các bộ phim mới (new items) vì chưa có ai đánh giá chúng.

Pure CB (RMSE 0.9204):

- Hiệu suất của Pure CB tốt hơn Pure CF một chút, cho thấy hồ sơ thể loại cung cấp một đường cơ sở dự đoán ổn định hơn.
- Tuy nhiên, nó vẫn kém hơn đáng kể so với mô hình lai. Như đã phân tích, điểm yếu cố hữu của nó là không có khả năng khám phá (serendipity). Nó sẽ chỉ đề xuất những gì tương tự như những gì người dùng đã xem, tạo ra một "bong bóng lọc" (filter bubble).

4.2.3 Kết luận

Biểu đồ so sánh này cung cấp bằng chứng thực nghiệm mạnh mẽ cho thấy các phương pháp Lọc cộng tác và Lọc nội dung có tính chất bổ trợ (complementary) cho nhau.

- Lọc nội dung (CB) giải quyết vấn đề khởi đầu lạnh cho phim mới và cung cấp một đường cơ sở (baseline) dự đoán ổn định.

- Lọc cộng tác (CF) cung cấp khả năng cá nhân hóa sâu và khám phá sở thích mới (serendipity) mà CB không thể.

Mô hình lai của chúng tôi, bằng cách kết hợp cả hai theo một tỷ lệ trọng số được tinh chỉnh (40% CF và 60% CB), đã khai thác thành công điểm mạnh của cả hai phương pháp đồng thời bù đắp cho điểm yếu của nhau. Kết quả là một hệ thống đề xuất tổng thể mạnh mẽ (robust), ổn định và có độ chính xác dự đoán cao nhất.

KẾT LUẬN

Bài báo cáo đã trình bày một cách toàn diện quy trình thiết kế, xây dựng và đánh giá một hệ thống gợi ý phim lai (Hybrid Recommendation System) hoàn chỉnh. Nhiệm vụ cốt lõi là giải quyết các thách thức cố hữu của các hệ thống đề xuất, đặc biệt là vấn đề khởi đầu lạnh (cold-start) và tính thưa thớt của dữ liệu (data sparsity).

Để thực hiện mục tiêu này, chúng tôi đã đề xuất một kiến trúc lai tiên tiến, kết hợp đồng thời hai phương pháp luận chính: Lọc cộng tác (CF) và Lọc nội dung (CB). Mô hình được xây dựng theo kiến trúc kết hợp theo tầng (Cascade) và trọng số (Weighted). Quá trình Lọc cộng tác (User-User CF) đã được tối ưu hóa về mặt hiệu suất bằng cách sử dụng phân cụm K-Means để giảm không gian tìm kiếm lân cận.

Sau khi xây dựng các thành phần, một quy trình tinh chỉnh siêu tham số (Hypertuning) nghiêm ngặt đã được thực hiện. Kết quả đã xác định được cấu hình tối ưu cho mô hình là $K = 60$ (số lân cận) và $\alpha = 0.4$ (trọng số lai), cùng với $M = 200$ (bộ lọc cascade).

Kết quả đánh giá trên tập kiểm thử (test set) đã chứng minh tính hiệu quả và độ chính xác vượt trội của phương pháp lai:

1. Mô hình lai tối ưu ($\alpha = 0.4$) đạt được chỉ số lỗi thấp nhất với RMSE = 0.8857 và MAE = 0.6827.
2. Hiệu suất này vượt trội hơn đáng kể so với cả hai mô hình cơ sở: Lọc nội dung thuần túy (Pure CB, RMSE 0.9204) và Lọc cộng tác thuần túy (Pure CF, RMSE 0.9292).

Điều này khẳng định rằng việc kết hợp cả hai phương pháp đã bù đắp hiệu quả cho các điểm yếu của nhau: Lọc nội dung (CB) cung cấp sự ổn định, giải quyết vấn đề khởi đầu lạnh cho phim mới, trong khi Lọc cộng tác (CF) mang lại khả năng cá nhân hóa sâu và khám phá sở thích bất ngờ (serendipity).

Hạn chế:

Mặc dù mô hình lai được đề xuất đã đạt được kết quả khả quan, vẫn tồn tại nhiều hạn chế cần được nhìn nhận một cách nghiêm túc để hướng đến việc hoàn thiện hệ thống gợi ý trong tương lai:

1. Đặc trưng nội dung còn đơn giản:

Thành phần Lọc nội dung (CB) hiện tại chỉ dựa trên yếu tố "thể loại" (genre) để mô tả sở thích người dùng. Cách tiếp cận này chưa phản ánh đầy đủ gu xem phim thực sự của người dùng, vốn chịu ảnh hưởng bởi nhiều yếu tố khác như phong cách đạo diễn, dàn diễn viên, cốt truyện (abstract), hoặc tông cảm xúc của phim. Do đó, các gợi ý đôi khi vẫn mang tính "đại khái", chưa đủ sắc sảo để chạm đúng thị hiếu cá nhân.

2. Mức độ cá nhân hóa còn hạn chế:

Mặc dù CF đã giúp hệ thống cá nhân hóa gợi ý, nhưng việc kết hợp CB dựa trên genre khiến kết quả cuối cùng đôi khi bị "làm phẳng" – mất đi sự khác biệt tinh tế giữa các người dùng có gu tương tự nhưng khác biệt nhỏ về cảm xúc hoặc thể loại phụ. Mô hình hiện tại chưa đủ nhạy để phản ánh những sắc thái tinh tế ấy.

3. Vấn đề khởi đầu lạnh cho người dùng mới:

Hệ thống vẫn yêu cầu người dùng mới đánh giá tối thiểu 6 bộ phim để khởi tạo hồ sơ, điều này gây bất tiện và ảnh hưởng đến trải nghiệm ban đầu. Trong thực tế, nhiều người dùng có xu hướng bỏ qua giai đoạn này, khiến mô hình không thể đưa ra gợi ý chính xác ngay từ đầu.

4. Thiếu khả năng thích ứng thời gian thực:

Mô hình hiện tại hoạt động theo cơ chế huấn luyện offline, chỉ dựa trên dữ liệu tĩnh. Điều này dẫn đến việc các thay đổi trong sở thích người dùng (ví dụ: chuyển từ thích phim hành động sang phim tâm lý) không được phản ánh kịp thời. Do đó, khả năng phản ứng của hệ thống với dữ liệu mới còn hạn chế.

5. Sự cân bằng giữa hai thành phần CB và CF còn thủ công:

Giá trị trọng số α được xác định qua quá trình hypertuning thủ công và không có khả năng tự điều chỉnh linh hoạt theo từng người dùng. Trong khi một số người dùng phù hợp với CB hơn, số khác lại thiên về CF – mô hình hiện tại chưa đủ thông minh để tự tối ưu trọng số theo từng trường hợp.α được xác định qua quá trình hypertuning thủ công và không có khả năng tự điều chỉnh linh hoạt theo từng người dùng. Trong khi một số người dùng phù hợp với CB hơn, số khác lại thiên về CF – mô hình hiện tại chưa đủ thông minh để tự tối ưu trọng số theo từng trường hợp.

6. Chưa tận dụng các đặc trưng ngữ nghĩa và ngữ cảnh:

Hệ thống chưa tích hợp các đặc trưng liên quan đến ngữ nghĩa phim (semantic features) hoặc ngữ cảnh người dùng (context-aware) như thời gian xem, nền tảng thiết bị, hoặc tâm trạng. Điều này khiến hệ thống còn "cứng nhắc", chưa đạt mức độ thông minh ngữ cảnh (contextual intelligence) của các hệ thống gợi ý tiên tiến.

Hướng phát triển tương lai:

Dựa trên các hạn chế đã phân tích, chúng tôi đề xuất một số hướng phát triển tiềm năng để cải thiện hệ thống trong tương lai:

1. **Làm giàu đặc trưng nội dung** (Feature Enrichment): Áp dụng các kỹ thuật Xử lý Ngôn ngữ Tự nhiên (NLP) (như TF-IDF hoặc mô hình nhúng từ như Word2Vec) trên trường dữ liệu "abstract" (tóm tắt phim) để xây dựng một vector đặc trưng nội dung phong phú, chính xác hơn.
2. **Nâng cấp thuật toán Lọc cộng tác**: Thay thế User-User CF bằng các phương pháp Model-based CF tiên tiến hơn như Phân rã ma trận (Matrix Factorization) (ví dụ: SVD, ALS) hoặc các mô hình Học sâu (Deep Learning) (như Neural Collaborative Filtering). Các phương pháp này thường xử lý tính thưa thớt của dữ liệu tốt hơn và có khả năng mở rộng (scalability) cao hơn.
3. **Xây dựng hệ thống học trực tuyến** (Online Learning): Phát triển cơ chế cho phép mô hình cập nhật trọng số và hồ sơ người dùng theo thời gian thực (real-time) ngay khi có một đánh giá mới, giúp hệ thống thích ứng nhanh chóng với sự thay đổi trong sở thích của người dùng.

TÀI LIỆU THAM KHẢO

[1] V. H. Tiệp, *Machine Learning cơ bản*. 2016. [Online]. Available: https://machinelearningcoban.com/. Accessed: Oct. 31, 2025.

[2] E. Çano and M. Morisio, "Hybrid recommender systems: A systematic literature review," *Intelligent Data Analysis*, vol. 21, no. 6, pp. 1487–1524, 2017. [Online]. Available: 1901.03888v1.pdf

[3] R. Burke, "Hybrid recommender systems: Survey and experiments," *User Modeling and User-Adapted Interaction*, vol. 12, no. 4, pp. 331–370, 2002. [Online]. Available:https://www.researchgate.net/publication/220445047_Hybrid_Recommender_Systems_Survey_and_Experiments

[4] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA: Springer, 2011, pp. 73–105. doi: 10.1007/978-0-387-85820-3_3.

[5] A. Jadon and A. Patil, "A comprehensive survey of evaluation techniques for recommendation systems," 2023. [Online]. Available: [2312.16015] A Comprehensive Survey of Evaluation Techniques for Recommendation Systems

[6] F. Furtado and A. Singh, "Movie recommendation system using machine learning," *International Journal of Research in Industrial Engineering*, vol. 9, no. 1, pp. 84–98, 2020. [Online]. Available: https://www.riejournal.com/article_106395.html

[7] Y. Dou, H. Yang, X. Deng, and P. S. Yu, "A survey of collaborative filtering algorithms for social recommender systems," in *Proc. 12th Int. Conf. Semantics, Knowledge and Grids (SKG)*, 2016, pp. 40–46. doi: 10.1109/SKG.2016.14. [Online]. Available:https://www.researchgate.net/publication/312485550_A_Survey_of_Collaborative_Filtering_Algorithms_for_Social_Recommender_Systems