TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

**KHOA ĐIỆN TỬ**

**Bộ môn: Công nghệ Thông tin**.

**BÀI TẬP KẾT THÚC MÔN HỌC**

MÔN HỌC

**KHOA HỌC DỮ LIỆU**

  Sinh viên: **Trần Chiến Thắng**

  Lớp: **K57KMT01**

Giáo viên giảng dạy: **Nguyễn Văn Huy**

Link GitHub: https://github.com/thangtran1904

**Thái Nguyên – 2025**

|  |  |
| --- | --- |
| **TRƯỜNG ĐHKTCN** | **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM** |
| **KHOA ĐIỆN TỬ** | ***Độc lập - Tự do - Hạnh phúc*** |

**BÀI TẬP KẾT THÚC MÔN HỌC**

**MÔN HỌC: KHOA HỌC DỮ LIỆU**

BỘ MÔN : CÔNG NGHỆ THÔNG TIN

*Sinh viên:* **Trần Chiến Thắng**

*Lớp*: **K57KMT**

*Ngành:* **Kỹ thuật máy tính**

*Giáo viên hướng dẫn:* **Nguyễn Văn Huy**

*Ngày giao đề:* 20/5/2025  *Ngày hoàn thành* 30/5/2025

*Tên đề tài :***Hệ thống Khuyến nghị phim cá nhân hoá**

*Yêu cầu :* Danh sách phim được khuyến nghị cho người dùng

|  |
| --- |
| **GIÁO VIÊN HƯỚNG DẪN** |
| *(Ký và ghi rõ họ tên)* |

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Thái Nguyên, ngày….tháng…..năm 20....*

**GIÁO VIÊN HƯỚNG DẪN**

*(Ký ghi rõ họ tên)*

**Mục Lục**

[**CHƯƠNG 1: GIỚI THIỆU ĐỀ BÀI** 5](#_Toc199482490)

[**1.1** **Khảo sát hiện trạng.** 5](#_Toc199482491)

[**1.2 Phân tích yêu cầu bài toán** 6](#_Toc199482492)

[a. Hệ thống tìm kiếm và đề xuất phim 6](#_Toc199482493)

[**b. Dữ liệu** 7](#_Toc199482494)

[**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT** 7](#_Toc199482495)

[**2.1 Giới thiệu** 7](#_Toc199482496)

[**2.2 Numpy** 8](#_Toc199482497)

[**2.3 Matplotlib** 9](#_Toc199482498)

[**2.4 Pandas** 10](#_Toc199482499)

[**2.5 Scikit Learn - Thư viện Machine Learning** 11](#_Toc199482500)

[**2.6 Flask** 12](#_Toc199482501)

[**CHƯƠNG 3 THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH** 13](#_Toc199482502)

[**3.1 Sơ đồ khối hệ thống** 13](#_Toc199482503)

[**3.2 Sơ đồ thuật toán** 15](#_Toc199482504)

[**3.3 Xây dựng chương trình** 16](#_Toc199482505)

[**Chương 4: Kiểm thử & Đánh giá kết quả** 17](#_Toc199482506)

[**4.1Môi trường kiểm thử** 17](#_Toc199482507)

[**4.2 Chạy kiểm thử** 18](#_Toc199482508)

[**4.3 Đánh giá** 18](#_Toc199482509)

[**Chương 5:Kết Luận** 19](#_Toc199482510)

[**Tài liệu tham khảo** 20](#_Toc199482511)

# **CHƯƠNG 1: GIỚI THIỆU ĐỀ BÀI**

## **Khảo sát hiện trạng.**

Hiện nay, nhu cầu giải trí qua phim ảnh ngày càng tăng, kéo theo sự phát triển mạnh mẽ của các nền tảng xem phim trực tuyến như Netflix, Amazon Prime hay Disney+. Tuy nhiên, người dùng thường gặp khó khăn trong việc tìm kiếm phim phù hợp với sở thích cá nhân giữa hàng triệu lựa chọn, dẫn đến tình trạng quá tải thông tin. Các hệ thống đề xuất hiện tại tuy đã tiên tiến, nhưng vẫn tồn tại những hạn chế như thiếu cá nhân hóa (chỉ dựa vào xu hướng chung), vấn đề Cold Start (khó đề xuất cho người dùng mới hoặc phim mới), và đôi khi thiếu sự đa dạng trong gợi ý.

Tuy nhiên hiện nay việc xem phim trên mạng cũng còn gặp phải nhiều vấn đề cần phải giải quyết như:

- Người tiêu dùng lo ngại về chất lượng phim.

- Lo ngại về phim không đến được tay người xem.

- Lo ngại về tính bảo mật thông tin của khách hàng trên website.

Mặt khác việc xem phim qua mạng có nhiều lợi ích hơn so với xem phim truyền thống như:

- Quảng bá thông tin và tiếp thị trong thị trường toàn cầu với chi phí thấp.

- Cung cấp dịch vụ tốt hơn cho khách hàng.

- Tăng doanh thu và giảm chi phí.

- Tạo lợi thế cạnh tranh.

Do đó, để tạo nên một hệ thống xem phim chuyên nghiệp và tạo được lòng tin cho khách hàng là một việc không dễ dàng.Từ những lý do trên em đề tài “Xây dựng hệ thống đề xuất và gợi ý phim” là một đề tài phù hợp với xu hướng hiên tại và tương lai. Các công ty đang tận dụng AI và học máy để sử dụng dữ liệu này theo những cách sáng tạo. Hệ thống đề xuất được hỗ trợ bởi có thể sử dụng dữ liệu khách hàng một cách hiệu quả để cá nhân hóa trải nghiệm người dùng, tăng mức độ tương tác và giữ chân, đồng thời cuối cùng thúc đẩy doanh số bán hàng cao hơn.

Ví dụ, vào năm 2021, [Netflix](https://www.spiceworks.com/marketing/customer-experience/articles/recommendation-engines-how-amazon-and-netflix-are-winning-the-personalization-battle/)báo cáo rằng hệ thống khuyến nghị của nó đã giúp tăng doanh thu thêm 1 tỷ đô la mỗi năm. Amazon là một công ty khác được hưởng lợi từ việc cung cấp các đề xuất được cá nhân hóa cho khách hàng của mình. Năm 2021, một số web báo cáo rằng hệ thống khuyến nghị của nó đã giúp tăng doanh số bán hàng lên 35%.

## **1.2 Phân tích yêu cầu bài toán**

### a. Hệ thống tìm kiếm và đề xuất phim

Mục tiêu: Xây dựng một hệ thống cho phép người dùng tìm kiếm phim dựa trên các tiêu chí (tên phim, thể loại, diễn viên, v.v.) và nhận đề xuất phim dựa trên sở thích, lịch sử tìm kiếm, và đánh giá của người dùng.

Đối tượng người dùng:

* Người dùng thông thường: đánh giá phim

Tính năng chính:

* Tìm kiếm phim theo từ khóa (tên phim, thể loại, diễn viên, v.v.).
* Đề xuất phim dựa trên đánh giá của người dùng khác (hệ thống cộng tác).
* Hiển thị thông tin chi tiết của phim (tóm tắt, diễn viên, đạo diễn, thể loại, v.v.).
* Biểu đồ đánh giá phim

Sử dụng:

* Tích hợp các thuật toán: Content-Based Filtering và Collaborative Filtering
* Ngôn ngữ: Python

### 

### **b. Dữ liệu**

TMDB 5000 Movie Dataset:

* Chứa thông tin chi tiết về phim: tên phim, thể loại, tóm tắt (overview), diễn viên, đạo diễn, năm sản xuất, điểm đánh giá, v.v.
* Phù hợp để xây dựng hệ thống đề xuất dựa trên nội dung (Content-Based Filtering), vì dataset này cung cấp các đặc trưng nội dung của phim (thể loại, tóm tắt, v.v.).

MovieLens 25M Dataset từ Kaggle:

* Chứa thông tin về đánh giá của người dùng: userID, movieID, rating, timestamp.
* Phù hợp để xây dựng hệ thống đề xuất dựa trên cộng tác (Collaborative Filtering), vì dataset này có dữ liệu về hành vi người dùng (đánh giá phim).

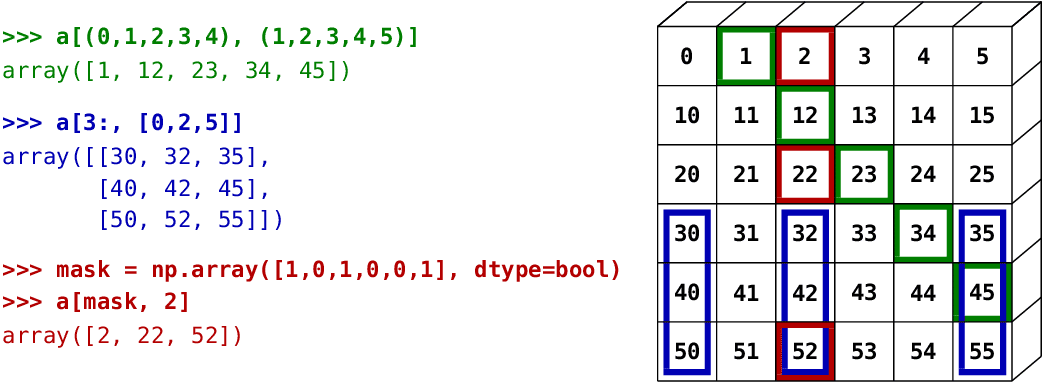
# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## **2.1 Giới thiệu**

Python là một ngôn ngữ lập trình được sử dụng rộng rãi trong các ứng dụng web, phát triển phần mềm, khoa học dữ liệu và máy học (ML). Các nhà phát triển sử dụng Python vì nó hiệu quả, dễ học và có thể chạy trên nhiều nền tảng khác nhau. Phần mềm Python được tải xuống miễn phí, tích hợp tốt với tất cả các loại hệ thống và tăng tốc độ phát triển.Trong đó các tập thư viện được sử dụng phổ biến như:

## **2.2 Numpy**

Là một gói xử l (Processing Package) phổ biến của Python. NumPy làm phong phú ngôn ngữ lập trình Python với các cấu trúc dữ liệu mạnh mẽ để tính toán hiệu quả các mảng và ma trận đa chiều. Numpy không chỉ là một gói mô-đun để xử lý mảng mà nó còn cung cấp khả năng quản lý mảng cực kỳ vượt trội. Nhanh chóng, vượt trội, hiệu quả là những gì tôi được trải nghiệm với Numpy.

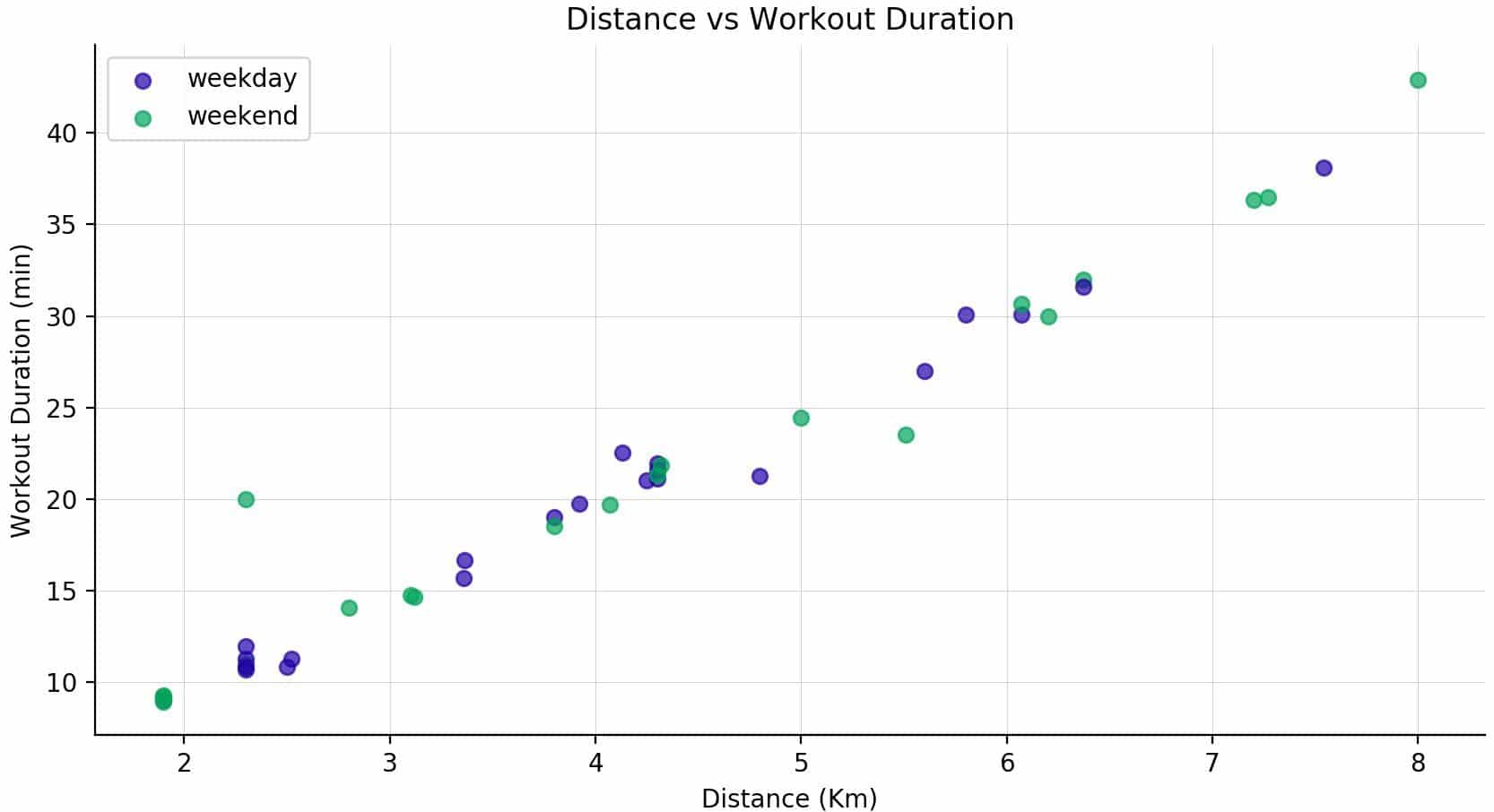


Những khả năng của Numpy:

* Numpy là một mô-đun mở rộng mã nguồn mở cho Python, cung cấp các chức năng biên dịch nhanh cho các thao tác toán học và số, thậm chí là với những ma trận và mảng có lượng dữ liệu khổng lồ. Bên cạnh đó các mô-đun cung cấp một thư viện lớn các chức năng toán học cấp cao để hoạt động trên các ma trận và mảng một cách dễ dàng và thuận tiện.
* Numpy cung cấp những masked arrays đồng thời với mảng gốc. Nó cũng đi kèm với các chức năng như thao tác với hình dạng logic, biến đổi Fourier rời rạc, đại số tuyến tính tổng quát, và nhiều hơn nữa.
* Mỗi khi bạn thay đổi đặc điểm của bất kỳ mảng N chiều nào, Numpy sẽ tạo các mảng mới cho mảng đó và xóa các mảng cũ.
* Gói mô-đun này cung cấp các công cụ hữu ích để tích hợp với các ngôn ngữ lập trình khác. Chẳng hạn như C, C++, và ngôn ngữ lập trình Fortran.
* Numpy cung cấp các chức năng tương đương với MATLAB. Cả hai đều cho phép người dùng thao tác nhanh hơn.

## **2.3 Matplotlib**

Matplotlib là một thư viện Python sử dụng Python Script để giúp chúng ta tạo ra các đồ thị 2D thường được ứng dụng trong toán học và khoa học dữ liệu. Thư viện này có hỗ trợ tạo nhiều giao điểm giữa hai trục số trong cùng một lúc. Bên cạnh đó, chúng ta cũng có thể dùng Matplotlib để thao tác trực tiếp đến các đặc điểm khác nhau của đồ thị.

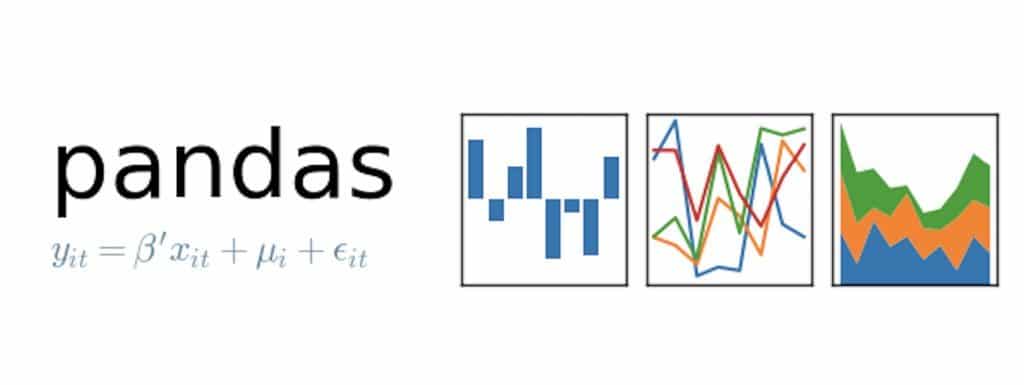


Những khả năng của Matplotlib:

* Matplotlib có thể tạo ra những đồ thị chất lượng và xuất ra một cách dễ dàng và thuận tiện, hoàn toàn đáp ứng nhu cầu của mọi ngành học. Các đồ thị được tạo ra bằng Matplotlib có sẵn bản sao cứng trên các nền tảng tương tác khác nhau.
* Bạn có thể dùng Matplotlib với nhiều bộ công cụ như Python Scripts, IPython Shells, Jupyter Notebook, và nhiều công cụ khác.
* Một số thư viện của bên thứ ba có thể được tích hợp với các ứng dụng Matplotlib. Chẳng hạn như seaborn, ggplot, và các bộ công cụ chiếu xạ, mapping khác như basemap.
* Một cộng đồng các nhà phát triển tích cực luôn sẵn sàng trợ giúp bạn với bất kỳ thắc mắc nào của bạn với Matplotlib. Sự đóng góp của họ cho Matplotlib là rất đáng khen ngợi.
* Ngoài ra, bạn còn có thể theo dõi bất kỳ lỗi nào phát sinh trong quá trình coding, các bản vá mới, đồng thời còn có thể đóng góp các tính năng mới tại GitHub. Đó là một trang chính thức để nêu ra các vấn đề liên quan đến Matplotlib và cùng giải quyết chúng.

## **2.4 Pandas**

Pandas là một gói phần mềm của Python. Nếu muốn trở thành một data scientist, bạn bắt buộc phải học Pandas, nó được viết chuyên dụng cho Python. Pandas mang lại hiệu suất cao cho các dự án, bởi tính trực quan, tốc độ và mô hình trực quan hóa cấu trúc dữ liệu của nó. Bạn có thể dễ dàng thao tác với bất kỳ loại dữ liệu nào như - dữ liệu định lượng (structured data) hoặc chuỗi thời gian (time-series data) với gói tuyệt vời này.

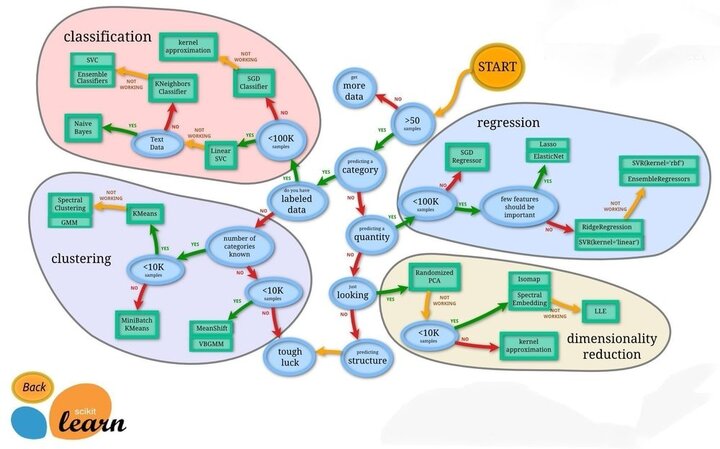


Những khả năng của Pandas:

* Pandas cung cấp cho bạn nhiều Series và DataFrames. Cho phép bạn có thể tổ chức, đi sâu, trình bày, và thao tác với dữ liệu.
* Căn chỉnh và lập chỉ mục thông minh có trong Pandas, nhờ đó mà bạn sẽ có một hệ thống tổ chức và dán nhãn dữ liệu gần như hoàn hảo.
* Pandas có một số tính năng đặc biệt cho phép bạn xử lý dữ liệu hoặc giá trị bị thiếu bằng một biện pháp thích hợp.
* Cú pháp của nó đơn giản đến mức ngay cả những người thiếu hoặc không có kiến thức cơ bản về lập trình cũng có thể dễ dàng làm việc với nó.
* Nó cung cấp một bộ sưu tập các công cụ tích hợp cho phép bạn cả đọc và ghi dữ liệu trong các dịch vụ web, cấu trúc dữ liệu và cơ sở dữ liệu khác nhau.
* Pandas có thể hỗ trợ JSON, Excel, CSV, HDF5 và nhiều định dạng khác. Trên thực tế, bạn có thể hợp nhất các cơ sở dữ liệu khác nhau cùng một lúc với Pandas.

## **2.5 Scikit Learn - Thư viện Machine Learning**

Scikit Learn là một thư viện cực kỳ đơn giản và hữu dụng cho machine learning. Nó được viết bằng Python, Cython, C và C++. Dẫu vậy, phần lớn mã nguồn được viết bằng Python. Bên cạnh đó, Scikit Learn là một thư viện hoàn toàn miễn phí nhằm vào machine learning, nó có thể hoạt động rất linh hoạt, không xung đột với các thư viện khác, chẳng hạn như NumPy hay SciPy, nó hoạt động song song và vô cùng hài hòa.



Những khả năng của Scikit Learn:

* Scikit Learn đi kèm với một API sạch sẽ và gọn gàng. Nó cũng cung cấp tài liệu rất hữu ích cho người mới bắt đầu.
* Nhiều thuật toán được tích hợp sẵn, chẳng hạn như- thuật toán phân lớp, thuật toán phân cụm dữ liệu và đệ quy. Nó cũng hỗ trợ random forest, k-means, gradient boosting, DBSCAN và hơn thế nữa.
* Scikit Learn là một lựa chọn tuyệt vời để làm quen với machine learning. Khi bạn đã hiểu rõ về các chức năng cơ bản của Scikit Learn, việc chuyển sang các nền tảng khác sẽ không là vấn đề.
* Scikit Learn cung cấp các phương pháp dễ dàng để biểu diễn dữ liệu. Cho dù bạn muốn trình bày dữ liệu dưới dạng bảng hay ma trận, tất cả đều có thể thực hiện được với Scikit Learn.
* Nó còn có thể phân tích và đi sâu vào nhận diện ký tự được viết bằng tay, bị biến dạng, ... Không chỉ nhận diện, bạn còn có thể trực quan hóa dữ liệu chữ số sau khi được phân tích.

Ngoài ra trong hệ thống này em sẽ lựa chọn sử dụng Scikit-Surprise là một thư viện học máy (machine learning) chuyên biệt – cụ thể là cho hệ thống gợi ý (Recommender Systems), một lĩnh vực con trong học máy. Surprise là một bộ công cụ Python dùng để xây dựng và phân tích các hệ thống đề xuất xử lý dữ liệu xếp hạng rõ ràng.

Surprise được thiết kế với mục đích sau :

Cung cấp cho người dùng quyền kiểm soát hoàn hảo đối với các thử nghiệm của họ. Để đạt được mục đích này, chúng tôi tập trung mạnh vào tài liệu , chúng tôi đã cố gắng làm cho tài liệu rõ ràng và chính xác nhất có thể bằng cách chỉ ra mọi chi tiết của thuật toán.

Giảm bớt nỗi đau khi xử lý Bộ dữ liệu . Người dùng có thể sử dụng cả bộ dữ liệu tích hợp ( Movielens , Jester ) và bộ dữ liệu tùy chỉnh của riêng họ .

Cung cấp nhiều thuật toán dự đoán sẵn sàng sử dụng như thuật toán cơ sở , phương pháp lân cận , dựa trên phân tích ma trận ( SVD , PMF , SVD++ , NMF ) và nhiều thuật toán khác . Ngoài ra, nhiều biện pháp tương tự (cosine, MSD, pearson…) cũng được tích hợp sẵn.

Dễ dàng triển khai các ý tưởng thuật toán mới .

Cung cấp các công cụ để đánh giá , phân tích và so sánh hiệu suất của các thuật toán. Các thủ tục xác thực chéo có thể được chạy rất dễ dàng bằng cách sử dụng các trình lặp CV mạnh mẽ (lấy cảm hứng từ các công cụ tuyệt vời của scikit-learn ), cũng như tìm kiếm toàn diện trên một tập hợp các tham số .

## **2.6 Flask**

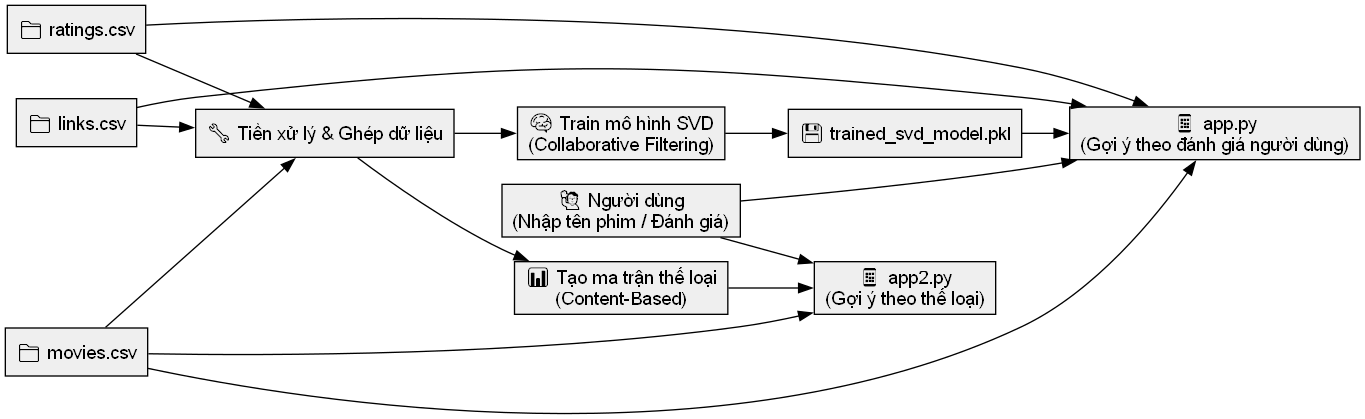
Flask là một framework trang web phát triển nhanh, được tạo ra cho quy trình thiết kế API hiệu quả hơn. Đây chỉ là một trong những cách sử dụng của Flask. Nói chung, nó là một khung framework để phát triển ứng dụng web.

Flash rất nhẹ, cung cấp hỗ trợ để kiểm tra đơn vị và cookie an toàn cho các phiên phía khách hàng. Các nhà phát triển khen rằng khung framework này là tài liệu tốt, có nghĩa là bạn sẽ tìm thấy nhiều trường hợp sử dụng để tìm hiểu.



# **CHƯƠNG 3 THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH**

## **3.1 Sơ đồ khối hệ thống**



Khối movies.csv mô tả danh sách bộ phim gồm movieId,title,genres.

Khối rating.csv mổ tả dữ liệu đánh giá người dung gồm UserId,movieId,rating

Khối links.csv mổ tả bản đồ nối movieId với ID từ các dịch vụ khác như TMDb (dung để lấy ảnh poster)

Khối tiền xử lý và ghép dữ liệu chức năng là gộp dữ liệu từ các file trên thành một khối thông tin hoàn chỉnh.Chuẩn bị để huấn luyện mô hình hoặc xây ma trận thể loại.

Khối tạo ma trận thể loại(Content-Based Filtering) dùng genres từ movies.csv để xây dựng ma trận vector thể loại. Tính toán độ tương đồng giữa các phim bằng **cosine similarity**. Và dữ liệu ngày dùng cho **app2.py**

**Huấn luyện mô hình SVD (Collaborative Filtering)** :

Khối Train mô hình SVD dùng ratings.csv để huấn luyện mô hình SVD (Singular Value Decomposition), học cách dự đoán điểm người dùng chưa đánh giá.

Khối trained\_svd\_model.pkl là File chứa mô hình đã huấn luyện, được lưu để sử dụng sau.

Và mô hình này được nạp trong **app.py**

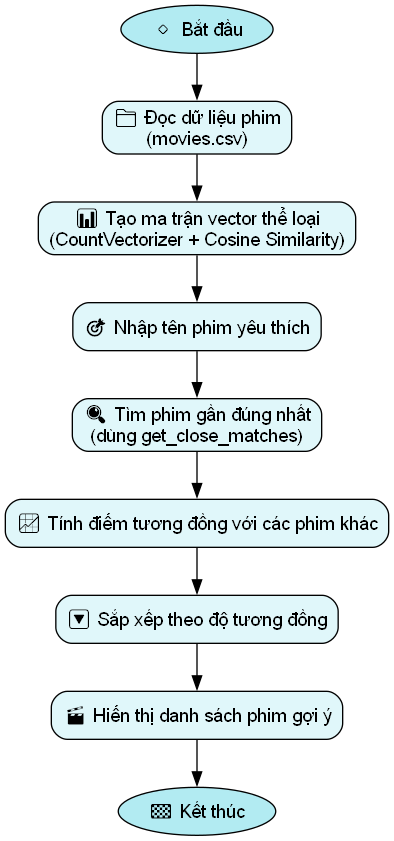
Khối người dùng (Nhập tên phim / Đánh giá) Có hai cách tương tác:   
1. Nhập tên phim yêu thích → dùng lọc theo thể loại.   
2. Đánh giá vài phim → dùng SVD để dự đoán.

Ứng dụng giao diện:

Khối **app.py** Gợi ý dựa trên đánh giá của người dùng mới. Dùng mô hình SVD đã huấn luyện. (Collaborative filtering)

Khối **app2.py** Gợi ý phim tương tự dựa trên thể loại của phim bạn thích. (Content-based filtering)

## **3.2 Sơ đồ thuật toán**



## **3.3 Xây dựng chương trình**

Hệ thống gợi ý phim được xây dựng gồm hai thành phần chính: gợi ý theo **nội dung (Content-Based Filtering)** và gợi ý theo **đánh giá người dùng (Collaborative Filtering)**. Chương trình được phát triển bằng ngôn ngữ **Python** và sử dụng thư viện **Streamlit** để xây dựng giao diện người dùng đơn giản, dễ tương tác.

3.3.1 Chuẩn bị dữ liệu

Hệ thống sử dụng ba tập dữ liệu từ tập MovieLens:

* **movies.csv**: Chứa thông tin tên phim và thể loại (movieId, title, genres).
* **ratings.csv**: Chứa đánh giá của người dùng cho từng phim (userId, movieId, rating).
* **links.csv:** Dùng để liên kết phim với TMDb, phục vụ hiển thị ảnh poster.

Dữ liệu được tải vào và xử lý bằng thư viện pandas.

3.3.2 Gợi ý dựa trên nội dung (app2.py)

Sử dụng thể loại phim (genres) để tạo ma trận đặc trưng sau đó dùng CountVectorizer và cosine\_similarity để tính toán mức độ tương đồng giữa các phim và khi người dùng nhập tên phim yêu thích, chương trình sẽ:

* Tìm phim gần đúng (sử dụng difflib.get\_close\_matches).
* Tính điểm tương đồng giữa phim được chọn và các phim khác.
* Hiển thị danh sách gợi ý cùng với poster.

**Thư viện sử dụng**: pandas, sklearn, difflib, streamlit, requests.

3.3.3 Gợi ý dựa trên đánh giá người dùng (app.py)

Sử dụng mô hình **SVD** từ thư viện surprise để huấn luyện dữ liệu đánh giá. Người dùng được yêu cầu đánh giá ít nhất 5 phim.Sau đó mô hình dự đoán điểm cho các phim còn lại, sau đó sắp xếp và hiển thị top phim có điểm dự đoán cao nhất.

**Thư viện sử dụng**: surprise, pandas, pickle, streamlit, requests.

3.3.4 Lưu và sử dụng mô hình

Mô hình SVD sau khi huấn luyện được lưu vào file trained\_svd\_model.pkl bằng thư viện pickle. Trong app.py, mô hình này được nạp lại để đưa ra gợi ý dựa trên đánh giá người dùng mới.

3.3.5 Giao diện người dùng

Giao diện được xây dựng bằng **Streamlit**, hỗ trợ:

* Nhập tên phim để tìm kiếm.
* Đánh giá phim qua thanh trượt.
* Hiển thị kết quả gợi ý với ảnh poster và điểm tương đồng/dự đoán.

Hệ thống chia thành 2 tab:

* **Gợi ý theo thể loại** (dành cho người dùng chưa từng đánh giá).
* **Gợi ý cá nhân hóa** (dành cho người dùng đã đánh giá vài phim).

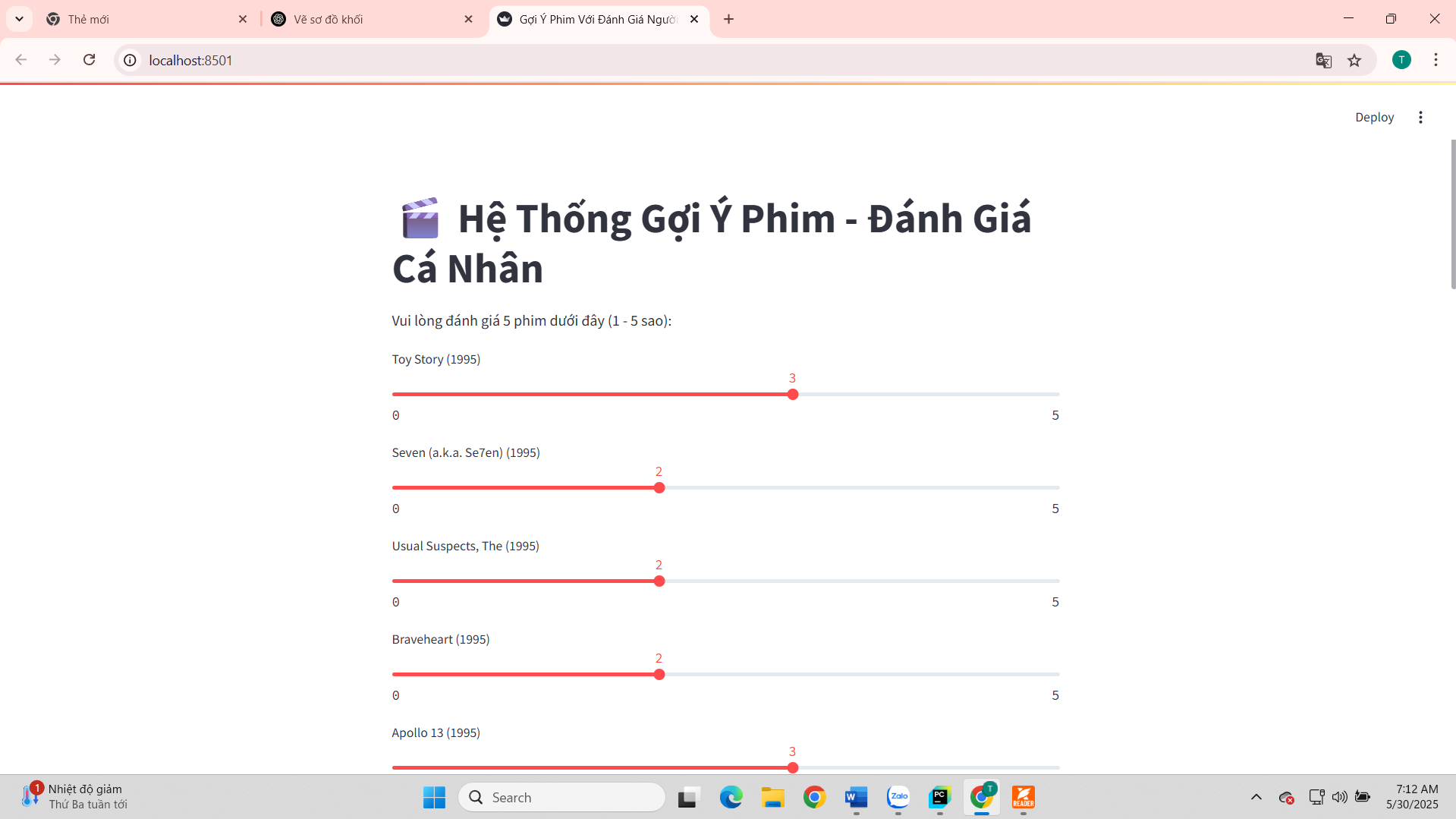
# **Chương 4: Kiểm thử & Đánh giá kết quả**

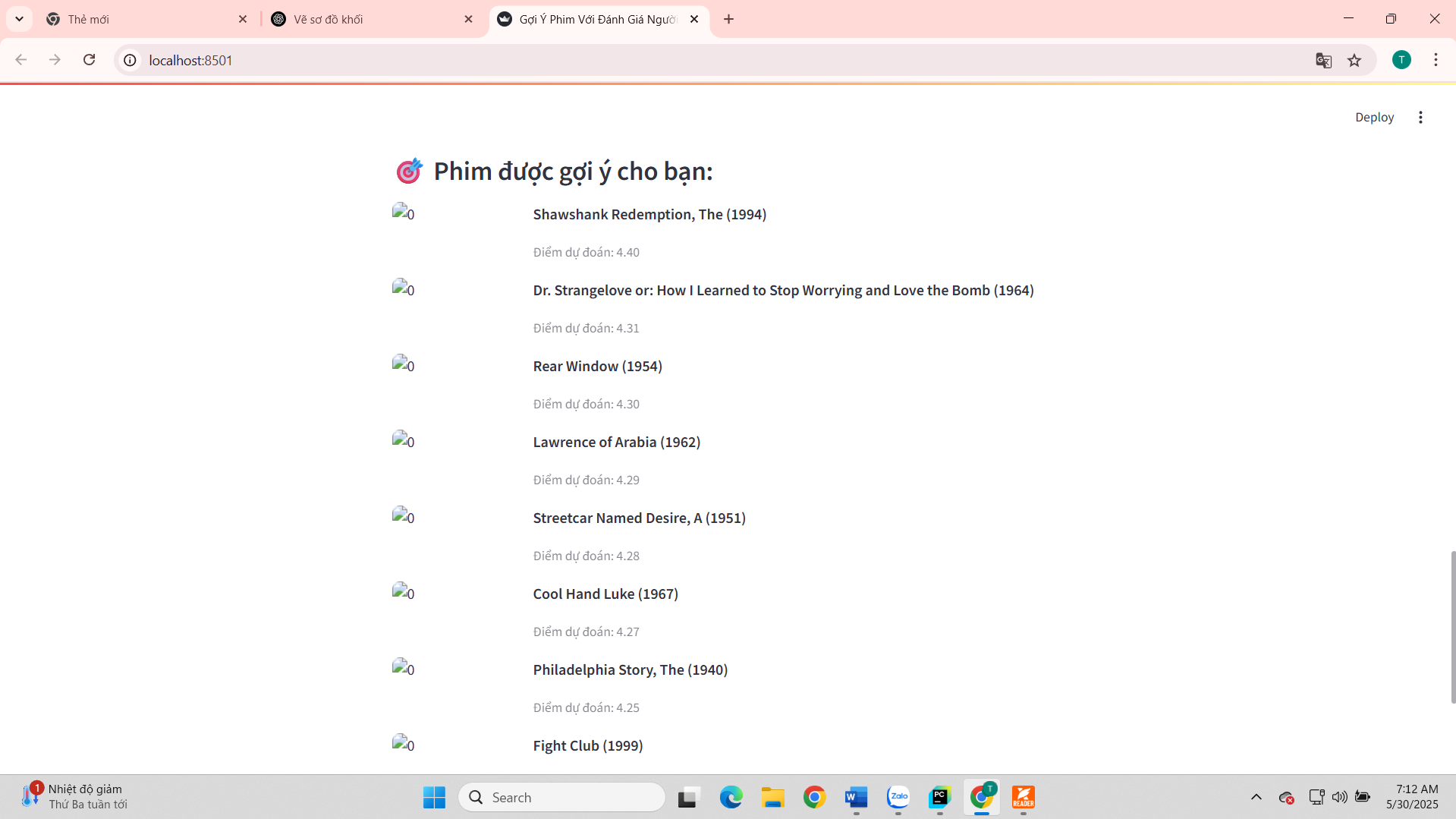
Sau khi xây dựng và triển khai hệ thống gợi ý phim, nhóm tiến hành kiểm thử tính năng và đánh giá chất lượng gợi ý nhằm xác minh độ chính xác, tính khả dụng và hiệu quả của hệ thống trong thực tế.

## **4.1Môi trường kiểm thử**

* **Ngôn ngữ lập trình: Python 3.9**
* **Thư viện chính: streamlit, pandas, scikit-learn, surprise, graphviz, requests**
* **Trình duyệt kiểm thử: Google Chrome 124+**
* **Hệ điều hành: Windows 10**
* **Dữ liệu: Tập MovieLens nhỏ (movies.csv, ratings.csv, links.csv)**

## **4.2 Chạy kiểm thử**

****

****

## **4.3 Đánh giá**

**A. Content-Based (app2.py):**

Hệ thống hoạt động tốt với các phim có nhiều thể loại phổ biến.

Khi người dùng nhập tên phim, các gợi ý thường cùng thể loại (hành động, phiêu lưu, lãng mạn...).

**Hạn chế: Không phân biệt được độ nổi bật hoặc đánh giá cộng đồng vì không dùng dữ liệu tương tác.**

B. **Collaborative Filtering (app.py)**:

Dự đoán khá chính xác cho người dùng đã đánh giá một số phim.

Kết quả được cá nhân hóa rõ ràng hơn: hai người dùng đánh giá khác nhau sẽ nhận gợi ý khác nhau.

**Hạn chế:** Cold-start – người dùng mới chưa đánh giá gì thì không dùng được mô hình SVD.

# **Chương 5:Kết Luận**

Trong đề tài này, chúng tôi đã nghiên cứu, thiết kế và xây dựng thành công một **hệ thống gợi ý phim** dựa trên hai phương pháp phổ biến:

* **Lọc theo nội dung (Content-Based Filtering)**: Gợi ý các bộ phim có đặc điểm thể loại tương đồng với phim người dùng yêu thích.
* **Lọc cộng tác (Collaborative Filtering)**: Gợi ý phim dựa trên hành vi đánh giá của người dùng thông qua mô hình **SVD** (Singular Value Decomposition).

Thông qua việc sử dụng các thư viện mạnh mẽ như **Streamlit**, **pandas**, **scikit-learn**, và **surprise**, hệ thống không chỉ cho kết quả chính xác, mà còn có giao diện người dùng đơn giản, trực quan và dễ sử dụng.

Hệ thống đã được kiểm thử thành công với tập dữ liệu MovieLens, cho kết quả gợi ý hợp lý và phù hợp với sở thích đầu vào của người dùng. Đồng thời, chương trình còn dễ mở rộng và có khả năng triển khai thực tế trong các ứng dụng giải trí.

Hướng phát triển trong tương lai, hệ thống có thể được mở rộng theo các hướng như :Tích hợp gợi ý theo nhiều thuộc tính khác như: đạo diễn, diễn viên, từ khóa.Hỗ trợ người dùng đăng nhập và lưu lịch sử đánh giá. Kết hợp hai phương pháp lọc (hybrid recommender system) để tăng độ chính xác. Đưa lên nền tảng web thực tế như **Heroku**, **Streamlit Cloud** hoặc **Docker**.

# **Tài liệu tham khảo**

**1.Ricci, F., Rokach, L., & Shapira, B. (2015). *Recommender Systems Handbook* (2nd ed.). Springer.  
Tổng quan lý thuyết về hệ thống gợi ý, các thuật toán lọc cộng tác, lọc nội dung.**

**2.MovieLens Dataset – GroupLens Research.  
Truy cập: https://grouplens.org/datasets/movielens/  
Nguồn dữ liệu phim được sử dụng trong huấn luyện và kiểm thử hệ thống.**

**3.Surprise: A Python scikit for building and analyzing recommender systems  
Truy cập:** [**https://surprise.readthedocs.io/**](https://surprise.readthedocs.io/) **Thư viện dùng để xây dựng mô hình SVD.**

**4.Streamlit Documentation – Simple apps for machine learning and data science.  
Truy cập: https://docs.streamlit.io/  
Hướng dẫn tạo giao diện web tương tác với Python.**

**5.Pedregosa et al. (2011). *Scikit-learn: Machine Learning in Python*, Journal of Machine Learning Research, 12, pp. 2825–2830.  
Cơ sở cho CountVectorizer, cosine\_similarity.**

**6.Python Official Documentation  
Truy cập:** [**https://docs.python.org/3/**](https://docs.python.org/3/) **Ngôn ngữ lập trình chính được sử dụng cho toàn bộ hệ thống.**

**7.TMDb API Documentation  
Truy cập: https://developer.themoviedb.org/  
API được dùng để lấy ảnh poster phim phục vụ giao diện người dùng.**