

**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: Lý Thành An. ....

Lớp: K57KMT .....

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

Link GitHub: [https://github.com/lythanhan03/Baitaplon\\_Recommender](https://github.com/lythanhan03/Baitaplon_Recommender)

**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:* Lý Thành An

*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 :* Đề số 2: 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

<b>CHƯƠNG 1: GIỚI THIỆU ĐỀ BÀI</b> .....	1
<b>1.1 Khảo sát hiện trạng.</b> .....	1
<b>1.2 Phân tích yêu cầu bài toán</b> .....	2
a. Hệ thống tìm kiếm và đề xuất phim .....	2
b. Dữ liệu .....	3
<b>CHƯƠNG 2: CƠ SỞ LÝ THUYẾT</b> .....	3
<b>2.1 Giới thiệu</b> .....	3
<b>2.2 Numpy</b> .....	3
<b>2.3 Matplotlib</b> .....	4
<b>2.4 Pandas</b> .....	6
<b>2.5 Kỹ thuật xử lý ngôn ngữ tự nhiên TF-IDF</b> .....	7
<b>2.6 Scikit Learn - Thư viện Machine Learning</b> .....	8
<b>2.7 Flask</b> .....	10
<b>3.1 Sơ đồ khối hệ thống</b> .....	11
<b>3.2 Tiền xử lý dữ liệu</b> .....	12
<b>3.3 Thuật toán Content-Based Filtering</b> .....	13
<b>3.4 collaborative</b> .....	14
<b>3.5 Kiểm thử phần mềm</b> .....	17
<b>CHƯƠNG 4 KẾT LUẬN</b> .....	20
<b>4.1 Những kết quả đã đạt được</b> .....	20
<b>4.2 Những kiến thức đã học được</b> .....	20
<b>4.3 Hướng cải tiến trong tương lai</b> .....	20

# CHƯƠNG 1: GIỚI THIỆU ĐỀ BÀI

## 1.1 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 hiện 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 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 lý (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.

```
>>> a[(0,1,2,3,4), (1,2,3,4,5)]
array([1, 12, 23, 34, 45])

>>> a[3:, [0,2,5]]
array([[30, 32, 35],
       [40, 42, 45],
       [50, 52, 55]])

>>> mask = np.array([1,0,1,0,0,1], dtype=bool)
>>> a[mask, 2]
array([2, 22, 52])
```

0	1	2	3	4	5
10	11	12	13	14	15
20	21	22	23	24	25
30	31	32	33	34	35
40	41	42	43	44	45
50	51	52	53	54	55

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.



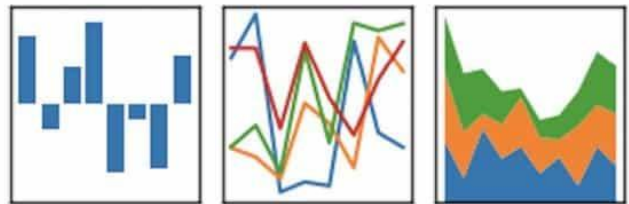
Những khả năng của Matplotlib:

- 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.

pandas  
 $y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$



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 Kỹ thuật xử lý ngôn ngữ tự nhiên TF-IDF

**TF-IDF** (Term Frequency – Inverse Document Frequency) là 1 kỹ thuật sử dụng trong khai phá dữ liệu văn bản. Trọng số này được sử dụng để đánh giá tầm quan trọng của một từ trong một văn bản. Giá trị cao thể hiện độ quan trọng cao và nó phụ thuộc vào số lần từ xuất hiện trong văn bản nhưng bù lại bởi tần suất của từ đó trong tập dữ liệu. Một vài biến thể của tf-idf thường được sử dụng trong các hệ thống tìm kiếm như một công cụ chính để đánh giá và sắp xếp văn bản dựa vào truy vấn của người dùng. Tf-idf cũng được sử dụng để lọc những từ stopwords trong các bài toán như tóm tắt văn bản và phân loại văn bản.

**TF:** Term Frequency(Tần suất xuất hiện của từ) là số lần từ xuất hiện trong văn bản. Vì các văn bản có thể có độ dài ngắn khác nhau nên một số từ có thể xuất hiện nhiều lần trong một văn bản dài hơn là một văn bản ngắn. Như vậy, term frequency thường được chia cho độ dài văn bản( tổng số từ trong một văn bản).

$$tf(t, d) = \frac{f(t, d)}{\max\{f(w, d) : w \in d\}}$$

Trong đó:

- $tf(t, d)$ : tần suất xuất hiện của từ  $t$  trong văn bản  $d$
- $f(t, d)$ : Số lần xuất hiện của từ  $t$  trong văn bản  $d$
- $\max(\{f(w, d) : w \in d\})$ : Số lần xuất hiện của từ có số lần xuất hiện nhiều nhất trong văn bản  $d$

**IDF:** Inverse Document Frequency(Nghịch đảo tần suất của văn bản), giúp đánh giá tầm quan trọng của một từ . Khi tính toán TF , tất cả các từ được coi như có độ quan trọng bằng nhau. Nhưng một số từ như “is”, “of” và “that” thường xuất hiện rất nhiều lần nhưng độ quan trọng là không cao. Như thế chúng ta cần giảm độ quan trọng của những từ này xuống.

$$\text{idf}(t, D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

Trong đó:

- $\text{idf}(t, D)$ : giá trị idf của từ  $t$  trong tập văn bản
- $|D|$ : Tổng số văn bản trong tập  $D$
- $|\{d \in D : t \in d\}|$ : thể hiện số văn bản trong tập  $D$  có chứa từ  $t$ .

Cơ số logarit trong công thức này không thay đổi giá trị idf của từ mà chỉ thu hẹp khoảng giá trị của từ đó. Vì thay đổi cơ số sẽ dẫn đến việc giá trị của các từ thay đổi bởi một số nhất định và tỷ lệ giữa các trọng lượng với nhau sẽ không thay đổi. (nói cách khác, thay đổi cơ số sẽ không ảnh hưởng đến tỷ lệ giữa các giá trị IDF). Việc sử dụng logarit nhằm giúp giá trị tf-idf của một từ nhỏ hơn, do chúng ta có công thức tính tf-idf của một từ trong 1 văn bản là tích của tf và idf của từ đó.

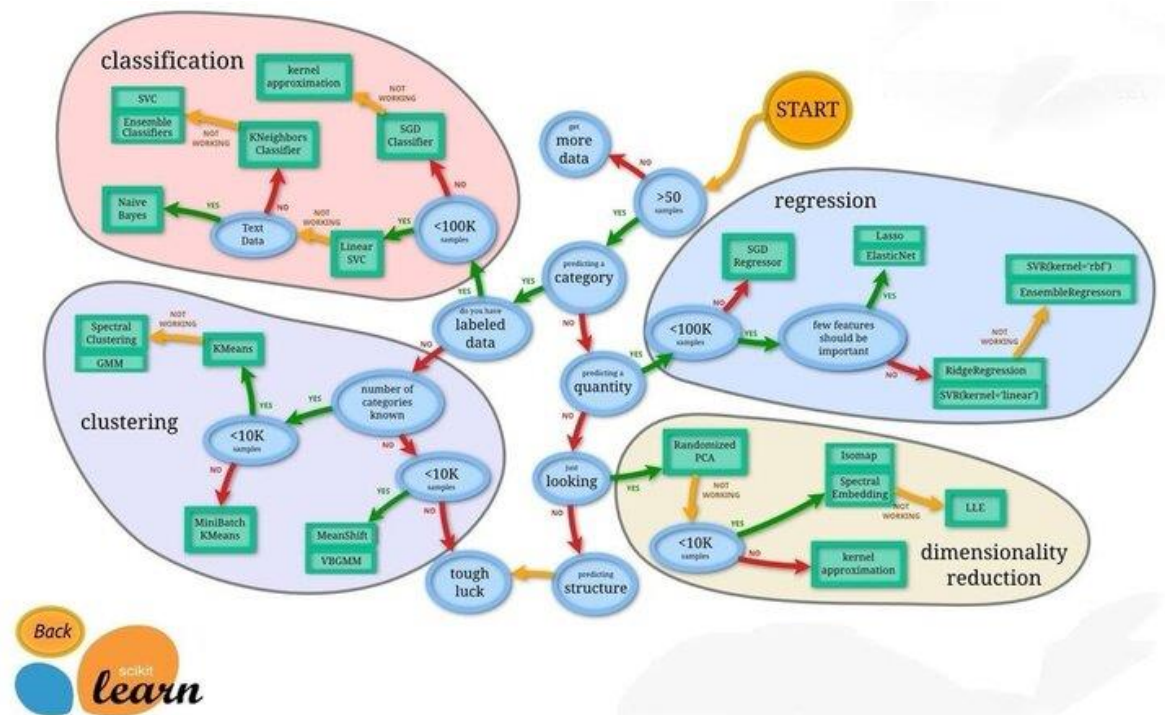
Cụ thể, chúng ta có **công thức tính tf-idf** hoàn chỉnh như sau:

$\text{TF-IDF}(t, d, D) = \text{TF}(t, d) \times \text{IDF}(t, D)$
--------------------------------------------------------------------

Những từ có giá trị TF-IDF cao là những từ xuất hiện nhiều trong văn bản này, và xuất hiện ít trong các văn bản khác. Việc này giúp lọc ra những từ phổ biến và giữ lại những từ có giá trị cao (từ khoá của văn bản đó).

## 2.6 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.

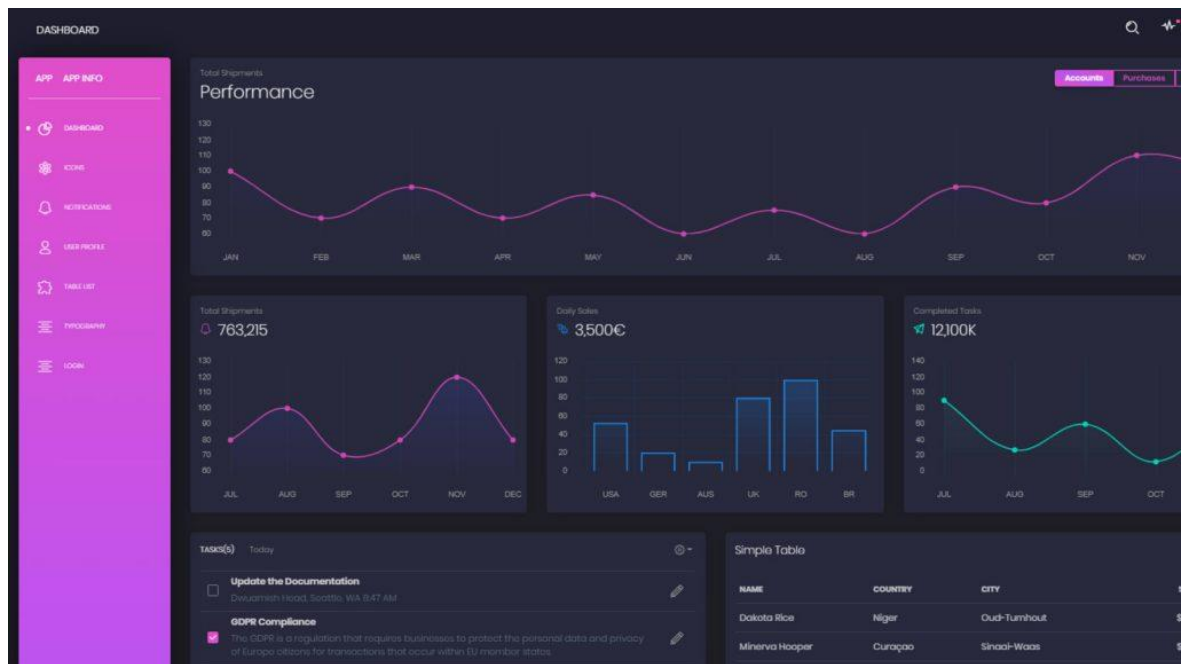
Đễ dàng triển khai các ý tưởng thuật toán mới .

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 lập 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 .

## 2.7 Flask

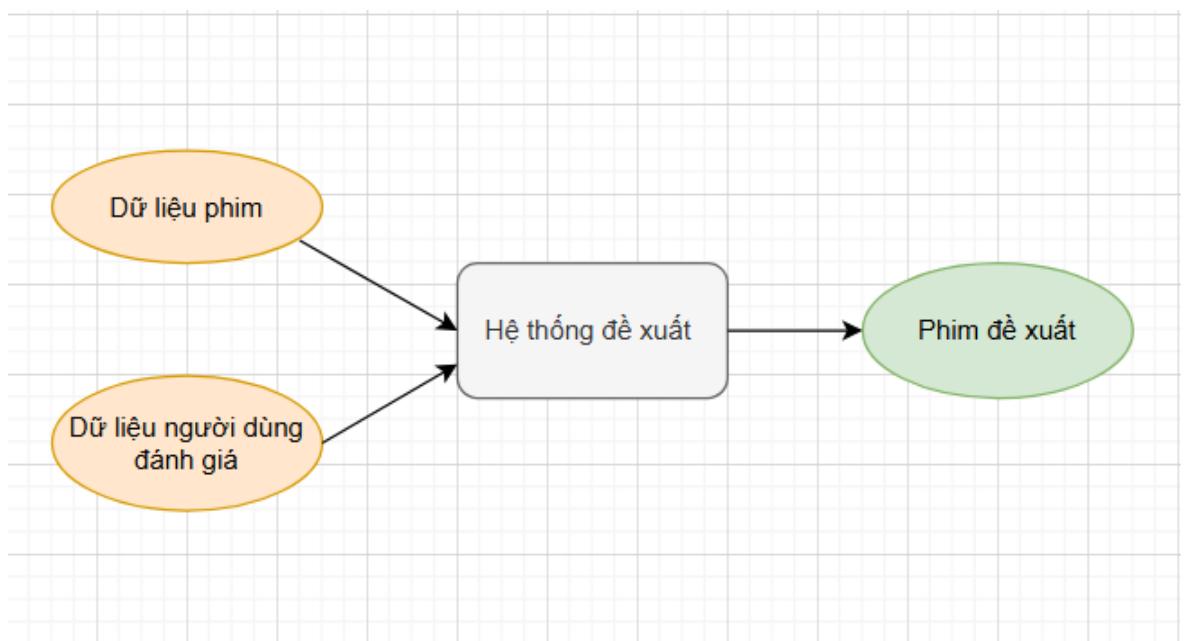
Flask là một framework trang web phát triển nhanh, được tạo ra cho quy trình thiết kế API hiệu quả hơn. Đây chỉ là một trong những cách sử dụng của Flask. Nói chung, nó là một khung framework để phát triển ứng dụng web.

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



### 3.2 Tiền xử lý dữ liệu

#### **TMDB 5000 Movie Dataset:**

- Chứa các thông tin liên quan đến phim
- Tmdb\_5000\_credits.csv: gồm 20 cột và 4803 dòng
- Tmdb\_5000\_credits.csv: gồm 4 cột và 4803 dòng
- Xử lý các trường chính: title, genres, keywords, overview, movie\_id, cast, crew

Các bước tiền xử lý:

- Chuyển đổi dữ liệu dạng Json trong các trường genres, keywords, overview, cast, crew
- Loại bỏ các ký tự đặc biệt
- Vector hoá: TF-IDF

#### **MovieLens 25M Dataset từ Kaggle:**

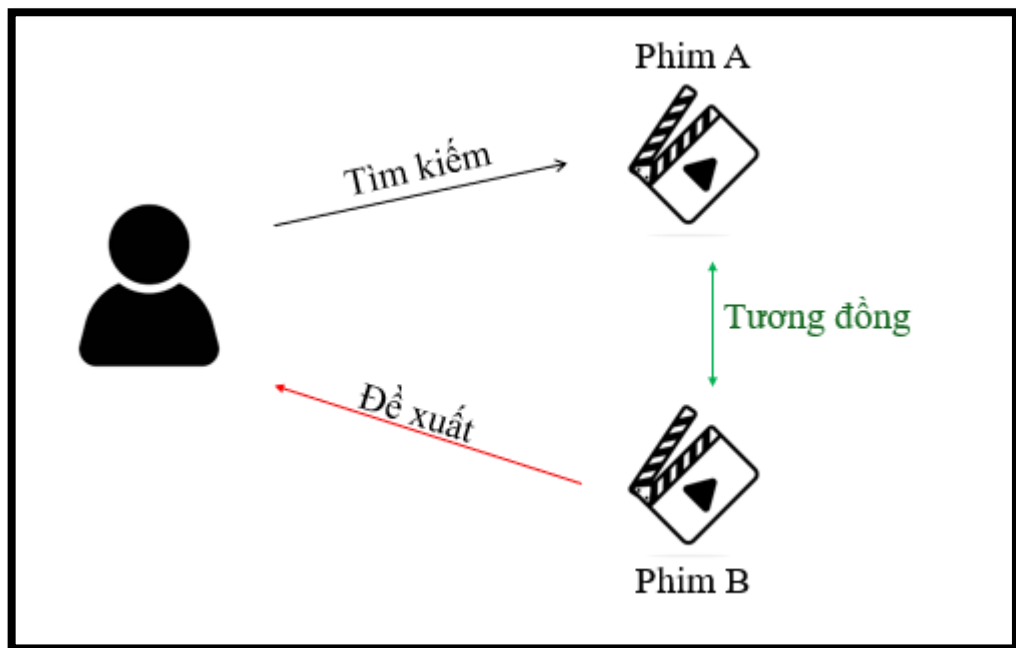
- Chứa 25,000,095 đánh giá (ratings) từ 162,541 người dùng (user) với 62,423 bộ phim (movie)

Các bước tiền xử lý:

- Lọc dữ liệu không đủ thông tin: loại bỏ các user hoặc movie có quá ít đánh giá để đảm bảo độ tin cậy cho mô hình học máy.
- Giảm nhiễu và tăng tính đại diện: giữ lại các user và phim có đủ dữ liệu để mô hình học được xu hướng đánh giá một cách chính xác.
- Chuẩn hóa dữ liệu cho thư viện Surprise: chỉ giữ lại các cột cần thiết (userId, movieId, rating) và chuyển đổi định dạng phù hợp



### 3.3 Thuật toán Content-Based Filtering

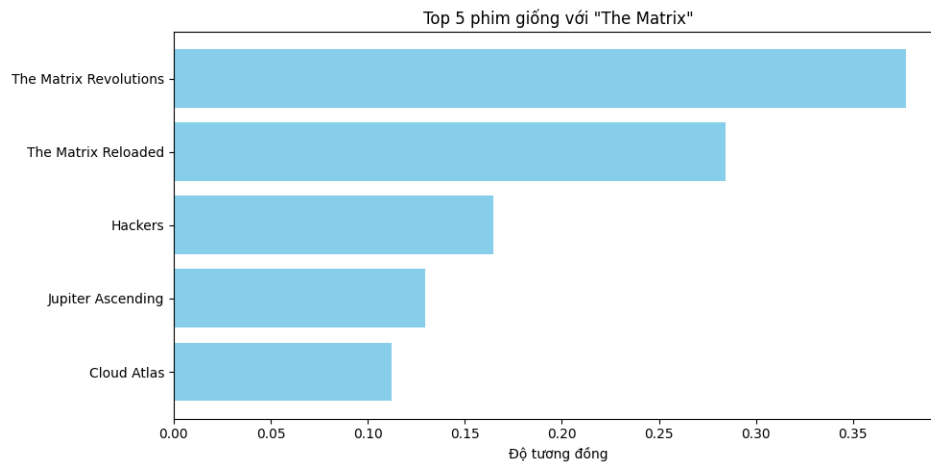


- Đầu vào : tên một bộ phim
  - Mô tả , từ khoá của tất cả các bộ phim
  - Tiền xử lý dữ liệu
  - Vector hoá bằng phương pháp TF-IDF
  - Sử dụng độ tương đồng Cosine
  - Sắp xếp và lựa chọn các bộ phim có độ tương đồng lớn nhất với bộ phim được tìm kiếm
- Đầu ra:
  - Top N bộ phim được đề xuất

```
print(recommend('The Dark Knight Rises'))
```

65	The Dark Knight
428	Batman Returns
119	Batman Begins
299	Batman Forever
1359	Batman
3854	Batman: The Dark Knight Returns, Part 2
210	Batman & Robin
9	Batman v Superman: Dawn of Justice
2507	Slow Burn
3819	Defendor

Name: title, dtype: object



### 3.4 collaborative

- Đầu vào : thông tin đánh giá của người dùng cho các bộ phim
  - Tiền xử lý dữ liệu
  - Chuẩn bị dữ liệu cho mô hình Collaborative Filtering
  - Huấn luyện mô hình Collaborative Filtering (SVD)
  - Đánh giá mô hình
  - Đề xuất phim cho người dùng cụ thể
  - Đề xuất cho người dùng mới
- Đầu ra:
  - Danh sách Top N phim được đề xuất cho người dùng

```
recommendations = recommend_movies(1, svd, movies, trainset)
print(recommendations)
```

	Title	Predicted Rating
1	Play Time (a.k.a. Playtime) (1967)	5.0
2	127 Hours (2010)	5.0
3	Shall We Dance (1937)	5.0
4	Captain Fantastic (2016)	5.0
5	City of Lost Children, The (Cité des enfants p...	5.0
6	Memento (2000)	5.0
7	Boyhood (2014)	5.0
8	Hustler, The (1961)	5.0
9	Hoop Dreams (1994)	5.0
10	Harold and Maude (1971)	5.0

```
new_user_ratings = [(1, 5.0), (2, 1.0)]
new_recommendations = recommend_for_new_user(new_user_ratings, svd, movies, trainset)
print(new_recommendations)
```

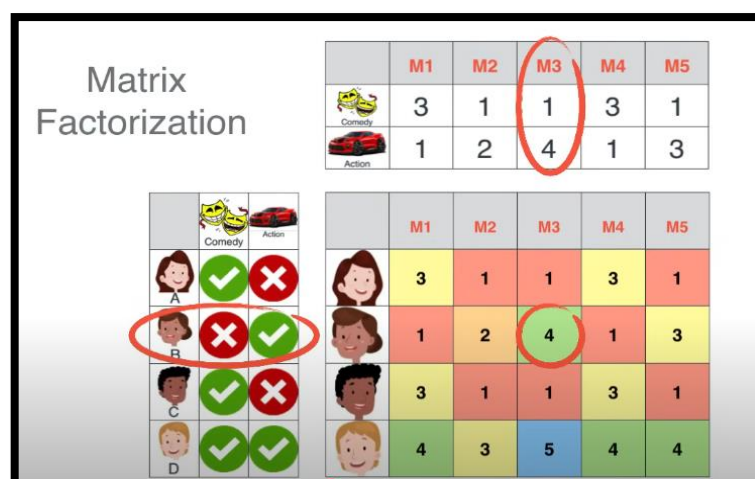
	Title	Predicted Rating
1	Shawshank Redemption, The (1994)	4.435438
2	Lawrence of Arabia (1962)	4.365897
3	Touch of Evil (1958)	4.334683
4	Wallace & Gromit: The Wrong Trousers (1993)	4.333141
5	Casablanca (1942)	4.311231
6	Forrest Gump (1994)	4.307941
7	Godfather: Part II, The (1974)	4.288378
8	American History X (1998)	4.284341
9	Hoop Dreams (1994)	4.256962
10	Amadeus (1984)	4.246918

## Thuật toán Matrix Factorization – Collaborative Filtering:

Biểu diễn ma trận đánh giá  $R$  (người dùng  $\times$  sản phẩm) dưới dạng tích của hai ma trận nhỏ hơn:

$$R \approx P \cdot Q^T$$

- $P \in \mathbb{R}^{m \times k}$ : ma trận đặc trưng của người dùng.
- $Q \in \mathbb{R}^{n \times k}$ : ma trận đặc trưng của sản phẩm.
- $k$ : số lượng yếu tố ẩn (latent factors).



## Phương pháp:

Sử dụng Surprise.SVD là một thuật toán lọc cộng tác (Collaborative Filtering) dựa trên phân rã ma trận (Matrix Factorization), được cung cấp bởi thư viện Surprise – một thư viện Python chuyên về hệ thống gợi ý:

Cách thức hoạt động Surprise.SVD:

Không trực tiếp trả về các ma trận  $P(\text{svd.pu})$  và  $Q(\text{svd.qi})$  (tức là ma trận đặc trưng của người dùng và sản phẩm), nhưng bên trong nó vẫn thực hiện việc phân rã ma trận đánh giá thành các vector đặc trưng thông qua thuật toán học máy – cụ thể là Stochastic Gradient Descent (SGD).

- Mục tiêu của SVD: Với mỗi cặp đánh giá  $(u, i)$  mô hình cố gắng ước lượng:

$$\hat{r}_{ui} = \mu + b_u + b_i + \mathbf{p}_u^T \mathbf{q}_i$$

Trong đó:

- $\mu$ : trung bình toàn bộ đánh giá
- $b_u$ : độ lệch (bias) của người dùng  $u$
- $b_i$ : độ lệch của sản phẩm  $i$
- $\mathbf{p}_u$ : vector đặc trưng (latent factors) của người dùng  $u$  (hàng của ma trận  $P$ )
- $\mathbf{q}_i$ : vector đặc trưng của sản phẩm  $i$  (hàng của ma trận  $Q$ )

Sử dụng thuật toán SGD:

Thuật toán cập nhật các vector  $\mathbf{p}_u$ ,  $\mathbf{q}_i$ ,  $b_u$  và  $b_i$  để giảm sai số bình phương giữa  $r_{ui}$  và  $\hat{r}_{ui}$ :

$$\min \sum_{(u,i)} (r_{ui} - \hat{r}_{ui})^2 + \lambda (\|\mathbf{p}_u\|^2 + \|\mathbf{q}_i\|^2 + b_u^2 + b_i^2)$$

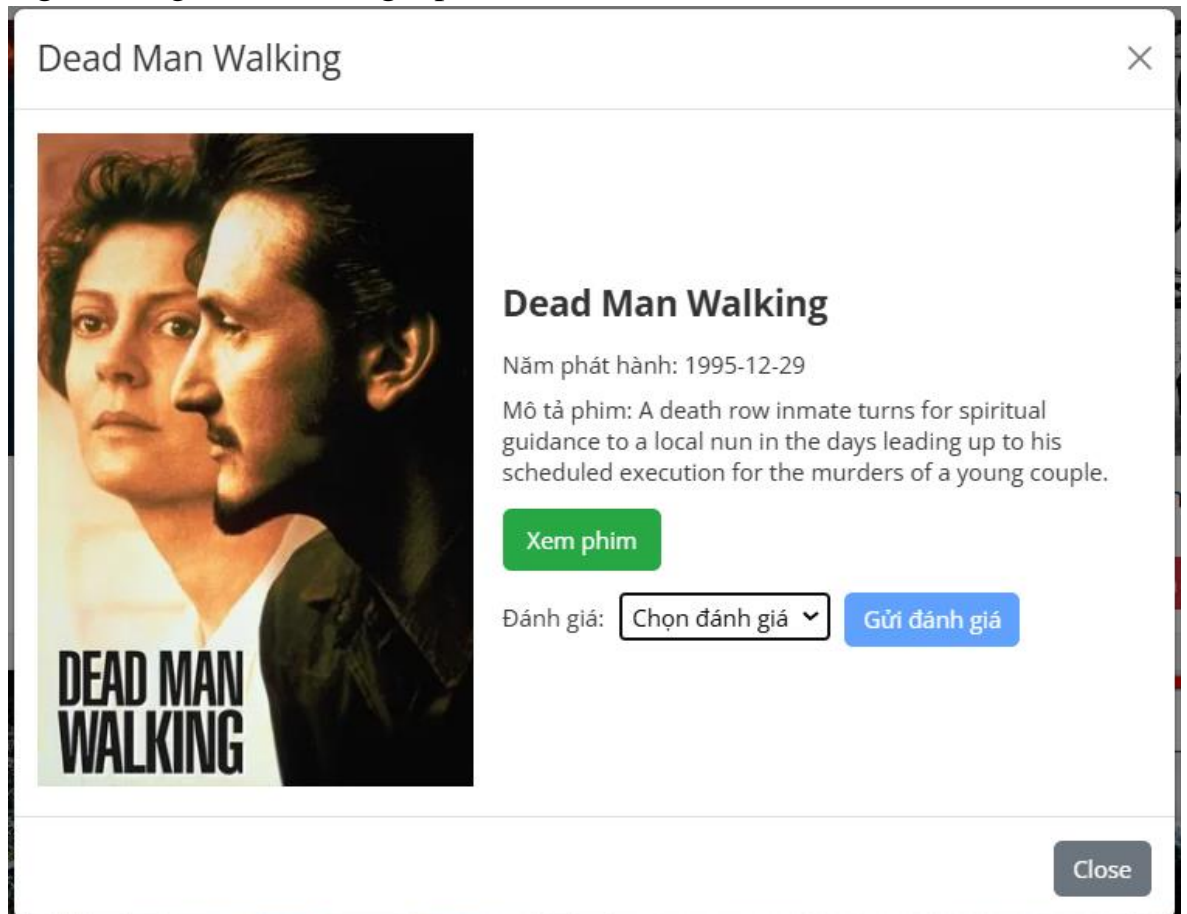
Lý do dùng Surprise.SVD cho Matrix Factorization:

- Phù hợp với dữ liệu thưa
- Hiệu quả tính toán
- Tự động xử lý bias
- Dễ sử dụng

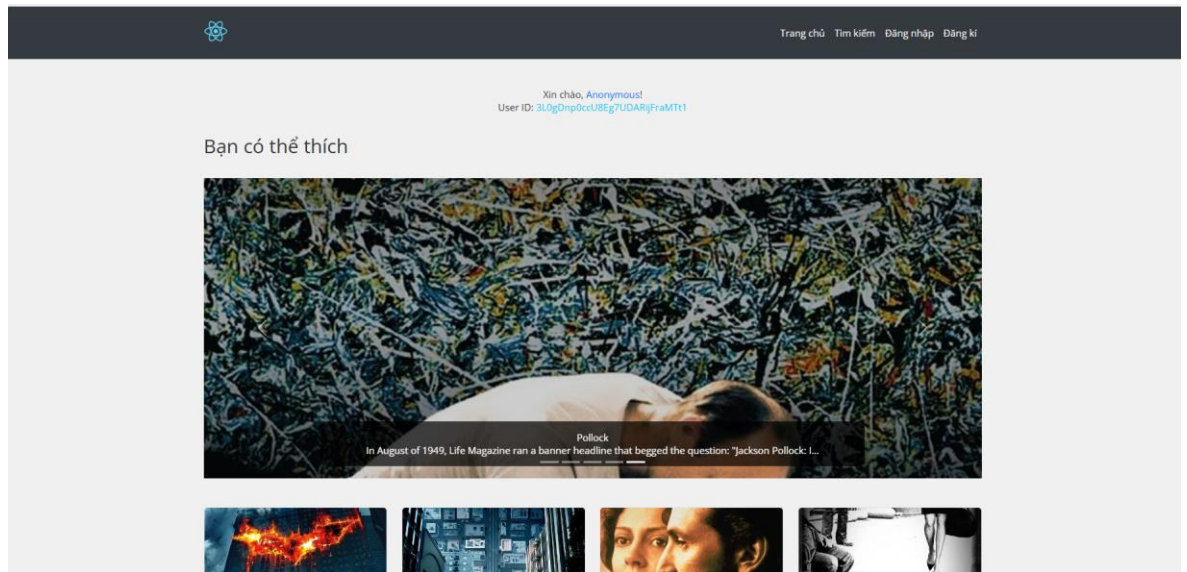
### 3.5 Kiểm thử phần mềm

Thuật toán collaborative :

- Người dùng có thể đánh giá phim

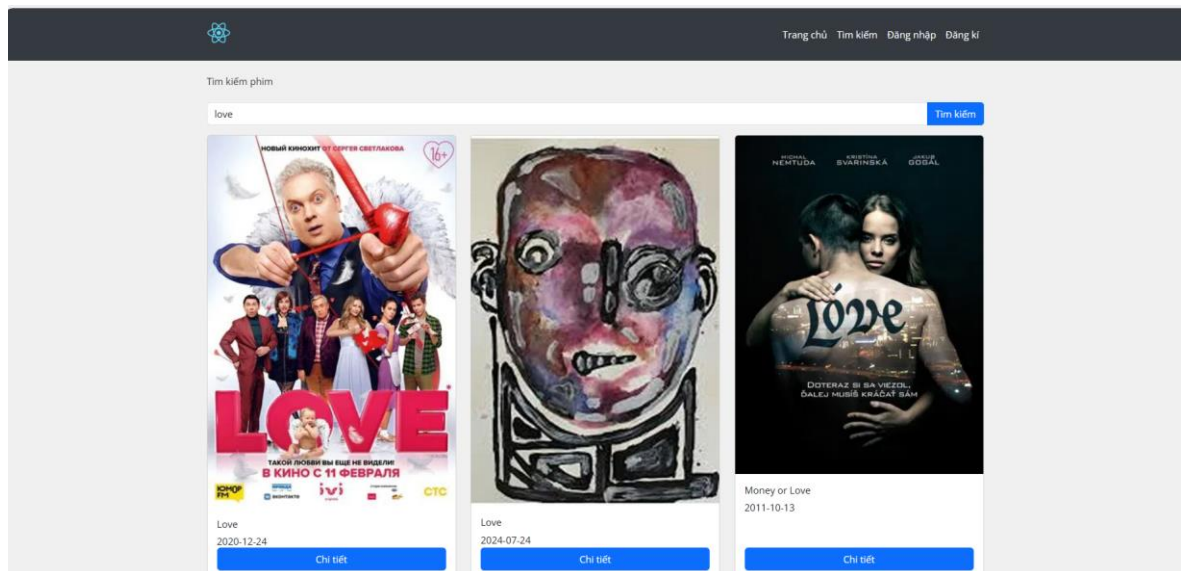


- Dưa ra các đề xuất dựa trên dữ liệu đánh giá

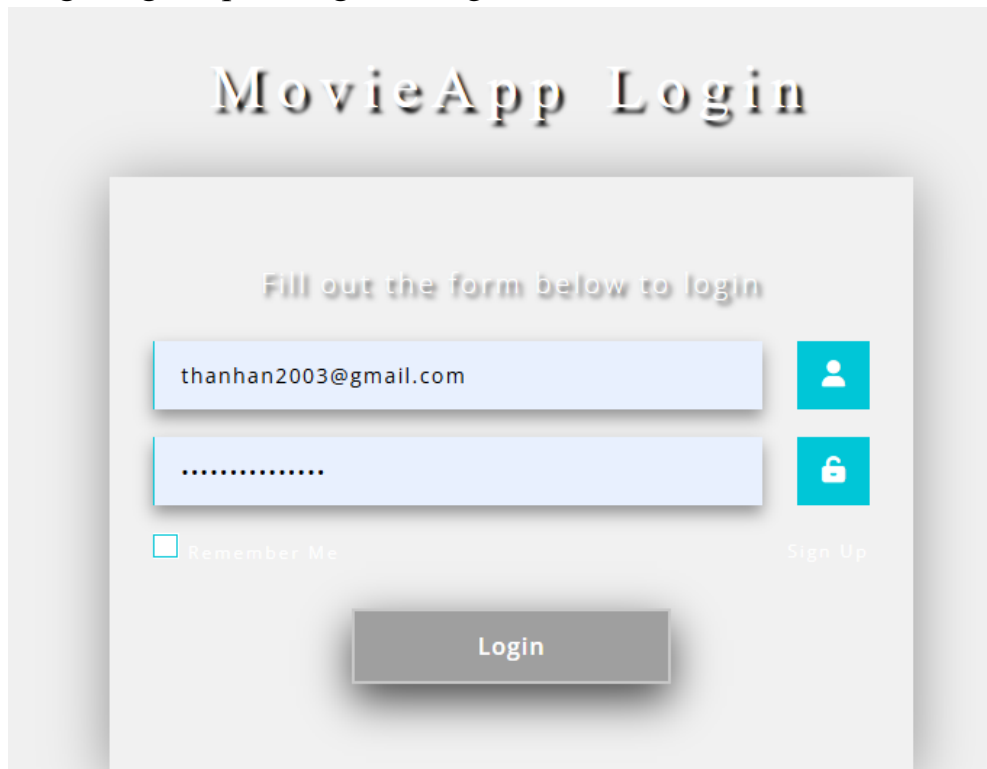


Thuật toán content-based :

- Người dùng có thể tìm kiếm phim và nhận được danh sách các bộ phim đề xuất



Chức năng đăng nhập cho người dùng:



The image shows a login form for 'MovieApp'. The title 'MovieApp Login' is at the top in a stylized font. Below it, the instruction 'Fill out the form below to login' is centered. The form consists of two light blue input fields: the first contains the email 'thanhan2003@gmail.com' and the second contains masked characters '.....'. To the right of each field is a teal icon: a person icon for the email field and a padlock icon for the password field. Below the password field is a checkbox labeled 'Remember Me' and a link 'Sign Up'. At the bottom center is a grey 'Login' button.

MovieApp Login

Fill out the form below to login

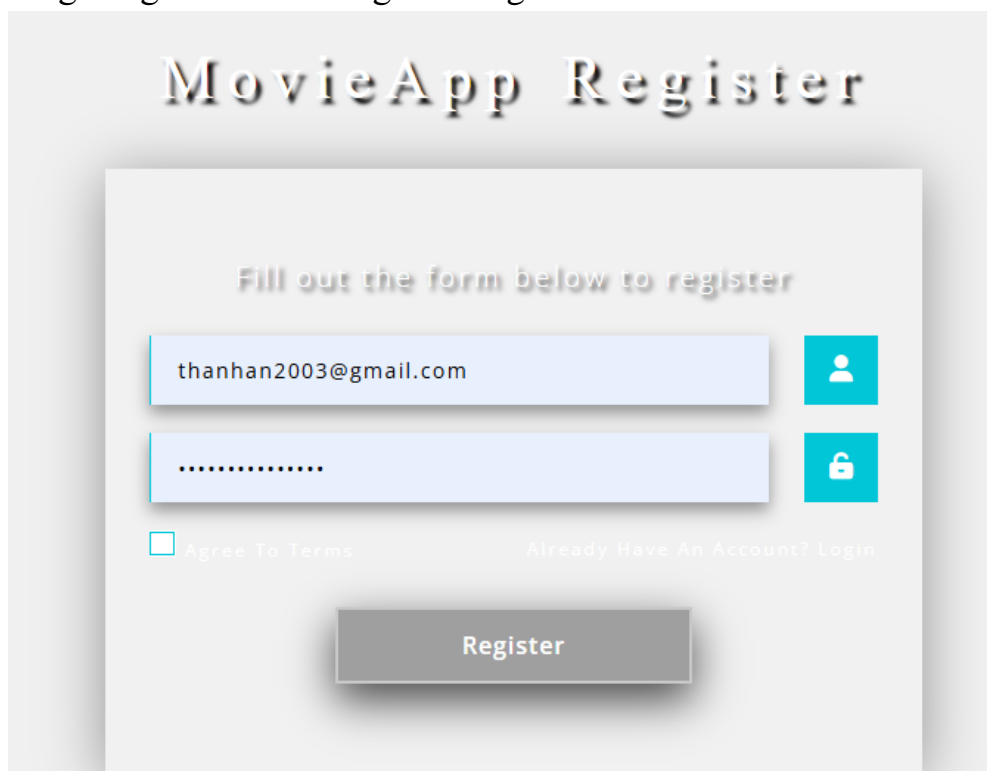
thanhan2003@gmail.com

.....

☐ Remember Me [Sign Up](#)

Login

Chức năng đăng kí tài khoản người dùng:



The image shows a registration form for 'MovieApp'. The title 'MovieApp Register' is at the top in a stylized font. Below it, the instruction 'Fill out the form below to register' is centered. The form consists of two light blue input fields: the first contains the email 'thanhan2003@gmail.com' and the second contains masked characters '.....'. To the right of each field is a teal icon: a person icon for the email field and a padlock icon for the password field. Below the password field is a checkbox labeled 'Agree To Terms' and a link 'Already Have An Account? Login'. At the bottom center is a grey 'Register' button.

MovieApp Register

Fill out the form below to register

thanhan2003@gmail.com

.....

☐ Agree To Terms [Already Have An Account? Login](#)

Register

## CHƯƠNG 4 KẾT LUẬN

### 4.1 Những kết quả đã đạt được

- Xây dựng thành công một hệ thống đề xuất Content-Based: Áp dụng các kỹ thuật tiền xử lý dữ liệu như trích xuất đặc trưng từ thể loại, từ khóa, diễn viên, đạo diễn và tóm tắt phim, kết hợp vector hóa bằng TF-IDF và tính toán độ tương đồng cosine để đề xuất phim phù hợp.
- Triển khai giao diện tìm kiếm và trực quan hóa: Phát triển một giao diện web sử dụng React và Flask, tích hợp biểu đồ Chart.js để hiển thị độ tương đồng giữa các phim, hỗ trợ người dùng khám phá các bộ phim tương tự một cách trực quan.
- Tích hợp dữ liệu từ TMDB: Kết nối với API TMDB để lấy thông tin bổ sung như hình ảnh poster và tóm tắt phim, cải thiện trải nghiệm người dùng với dữ liệu phong phú.

### 4.2 Những kiến thức đã học được

- Hiểu và áp dụng xử lý dữ liệu đa phương tiện: Nắm vững quy trình trích xuất đặc trưng từ văn bản, xử lý dữ liệu thô từ các nguồn như TMDB, và tối ưu hóa bằng cách sử dụng cache.
- So sánh và đánh giá thuật toán: Học cách so sánh hiệu quả giữa các phương pháp vector hóa (TF-IDF) và độ đo tương đồng (cosine similarity), cũng như tối ưu hóa thuật toán đề xuất.
- Kỹ năng triển khai hệ thống web: Thực hành tích hợp backend (Flask) và frontend (React), quản lý trạng thái ứng dụng, và xử lý lỗi thực tế như kết nối API hoặc dữ liệu không hợp lệ.
- Đánh giá và cải thiện hiệu suất: Hiểu cách sử dụng dữ liệu thực tế để kiểm tra và cải thiện hệ thống, bao gồm xử lý lỗi mạng và tối ưu hóa giao diện người dùng.

### 4.3 Hướng cải tiến trong tương lai

- Nâng cao độ chính xác của đề xuất: Sử dụng các mô hình học sâu như BERT hoặc Doc2Vec để cải thiện việc hiểu ngữ nghĩa trong tóm tắt phim, kết hợp với kỹ thuật xử lý mất cân bằng dữ liệu nếu có.
- Tích hợp thêm tính năng nâng cao: Phát triển chức năng đề xuất theo sở thích cá nhân dựa trên lịch sử xem phim của người dùng, hoặc thêm bộ lọc theo thể loại và năm phát hành.



- Cải thiện trải nghiệm người dùng: Nâng cấp giao diện với thông báo thời gian thực khi có phim mới tương tự, hoặc tích hợp chatbot để hỗ trợ tìm kiếm tự động.
- Mở rộng dữ liệu: Kết nối với nhiều nguồn dữ liệu hơn (như IMDb hoặc Netflix) để tăng độ đa dạng và chính xác của các đề xuất.

Hệ thống đề xuất này không chỉ là một sản phẩm kỹ thuật mà còn mở ra tiềm năng ứng dụng trong các nền tảng streaming hoặc trang web đánh giá phim, hứa hẹn sẽ được cải tiến liên tục để đáp ứng nhu cầu ngày càng cao của người dùng.

### TÀI LIỆU THAM KHẢO

Recommendation System using Machine Learning with Python:

<https://youtu.be/7rEagFH9tQg?si=zwOu3qtVHPeIEnh3>

Matrix Factorization:

<http://youtube.com/watch?v=gZgftF5hZOs&t=1075s>