

**Hệ thống đề xuất sách sử dụng thuật toán
Content-based và Collaborative Filtering, kết hợp
chức năng Tìm kiếm ngữ nghĩa.**

MỤC LỤC

1. Giới thiệu.....	3
1.1. Thành viên.....	3
1.2. Giới thiệu đề tài.....	3
1.2.1. Ý tưởng.....	3
1.2.1.1. Demographic Filtering.....	3
1.2.1.2. Collaborative Filtering.....	4
1.2.1.3. Content-Based Filtering.....	4
1.2.1.4. Semantic search.....	4
1.2.2. Mục tiêu.....	4
1.3. Lý do thực hiện đề tài.....	5
1.4. Lời cảm ơn.....	6
2. Phân tích và xác định vấn đề.....	7
2.1. Thiếu dữ liệu người dùng mới.....	7
2.2. Chất lượng và tính chính xác của dữ liệu.....	7
2.3. Tính cá nhân hóa và sự đa dạng trong sở thích.....	7
2.4. Khó khăn trong việc đánh giá hiệu quả của hệ thống đề xuất.....	7
3. Mô hình giải pháp đề xuất.....	8
3.1. Thu thập và xử lý dữ liệu.....	8
3.1.1. Thu thập dữ liệu.....	8
3.1.2. Xử lí dữ liệu.....	9
3.2. Thiết kế và triển khai mô hình.....	9
3.2.1. Thiết kế mô hình.....	9
3.2.1.1. Demographic filtering.....	9
3.2.1.2. Content-based filtering và collaborative filtering.....	10
3.2.2. Triển khai mô hình.....	11
3.2.2.1. Demographic filtering.....	11
3.2.2.2. Content-based filtering.....	11
3.2.2.3. Collaborative filtering.....	11
3.2.2.4. Semantic search.....	12
3.3. Huấn luyện mô hình.....	12
3.3.1. Demographic filtering.....	12
3.3.2. Content-based filtering.....	12
3.3.3. Collaborative filtering.....	13
3.3.4. Semantic search.....	14
3.4. Đánh giá hiệu suất mô hình.....	16
3.5. So sánh và tinh chỉnh mô hình.....	16

3.5.1. Content-based filtering.....	16
4. Ứng dụng và demo thực tế.....	26
4.1. Mức độ tích hợp mô hình vào ứng dụng web.....	26
4.2. Giao diện và trải nghiệm của người dùng.....	28
4.2.1. Đăng ký.....	28
4.2.4. Trang chủ.....	30
4.2.5. Hiển thị danh sách tìm kiếm sách.....	31
4.2.6. Danh sách các quyền sách yêu thích của bạn.....	31
4.2.7. Danh sách các tác giả mà bạn đang theo dõi.....	32
4.2.8. Trang hiển thị thông tin chi tiết của sách.....	32
4.2.8.1. Thông tin chung của sách.....	32
4.2.8.2. Các quyền sách có liên quan.....	33
4.2.9. Khám phá các tác giả.....	33
4.2.10. Khám phá các thể loại.....	34
5. Tài liệu tham khảo.....	35

1. Giới thiệu

1.1. Thành viên

STT	MSSV	Họ và tên	Ghi chú
1	22127029	Lê Nguyễn Gia Bảo	Nhóm trưởng
2	22127086	Nguyễn Lâm Anh Duy	Thành viên
3	22127149	Huỳnh Anh Huy	Thành viên
4	22127277	Võ Thị Hồng Minh	Thành viên
5	22127254	Trương Nguyễn Hiền Lương	Thành viên

1.2. Giới thiệu đề tài

1.2.1. Ý tưởng

Trong bối cảnh thị trường sách ngày càng phát triển với số lượng đầu sách đa dạng, việc tìm kiếm và đề xuất những cuốn sách phù hợp với sở thích cá nhân trở thành một thách thức lớn. Đề tài "Hệ thống đề xuất sách sử dụng thuật toán Content-based và Collaborative Filtering, kết hợp Semantic Search với LLM và Vector Database" hướng tới việc xây dựng một giải pháp thông minh, hiệu quả để giải quyết vấn đề này.

Hệ thống đề xuất sách được xây dựng dựa trên việc ứng dụng các thuật toán học máy để phân tích sở thích, thói quen đọc sách của người dùng cũng như các đặc trưng của sách (tên sách, mô tả, thể loại...). Hệ thống được thiết kế dựa trên 04 yếu tố cốt lõi:

1.2.1.1. Demographic Filtering

- Sử dụng số liệu tổng hợp từ người dùng, bao gồm:
 - Số lượt đánh giá:** Thể hiện mức độ phổ biến của sách.
 - Số điểm đánh giá trung bình:** Đánh giá mức độ chất lượng và sự yêu thích của độc giả đối với sách.
- Gợi ý các sách được đánh giá cao hoặc có số lượt đánh giá lớn, phù hợp với độ tuổi, thể loại yêu thích, và xu hướng chung của người dùng.

- Thích hợp để giới thiệu sách nổi bật, bán chạy hoặc đang được quan tâm trong cộng đồng.

1.2.1.2. Collaborative Filtering

Phân tích hành vi của người dùng khác có sở thích tương tự để đưa ra các gợi ý dựa trên sách mà họ đã đọc và yêu thích. Phương pháp này giúp khai thác tối đa dữ liệu từ cộng đồng người dùng, mang lại khả năng gợi ý những cuốn sách mà một cá nhân có thể chưa từng khám phá.

1.2.1.3. Content-Based Filtering

Dựa vào đặc điểm nội dung của từng cuốn sách, hệ thống sẽ đề xuất các đầu sách có chủ đề hoặc phong cách tương đồng với sở thích cá nhân của người dùng. Điều này giúp tăng tính chính xác khi gợi ý các sách có nội dung liên quan trực tiếp đến sở thích hoặc nhu cầu cụ thể.

1.2.1.4. Semantic search

Sử dụng các mô hình transformer và vector hóa dữ liệu để tìm kiếm ngữ nghĩa. Semantic Search không chỉ tập trung vào các từ khóa mà còn hiểu được ý nghĩa ngữ cảnh trong nội dung của sách và truy vấn của người dùng. Việc kết hợp với Vector Database giúp tăng tốc độ xử lý, cải thiện độ chính xác, và cung cấp trải nghiệm tìm kiếm sách tự nhiên và hiệu quả hơn.

1.2.2. Mục tiêu

Hệ thống đề xuất sách được xây dựng với các mục tiêu cụ thể như sau:

- Cá nhân hóa trải nghiệm người dùng:

Phân tích dữ liệu chi tiết để hiểu sâu hơn về sở thích và hành vi của từng người dùng, từ đó đưa ra các gợi ý phù hợp nhất.

- Hỗ trợ người dùng mới:

Thiết kế các gợi ý ban đầu dựa trên xu hướng hiện tại, sách bán chạy và các danh mục được yêu thích, cung cấp lựa chọn phù hợp cho người dùng trong trường hợp chưa xác định được sở thích cụ thể.

- Tối ưu hóa tốc độ tìm kiếm:

Sử dụng công nghệ Semantic Search kết hợp với Vector Database để đảm bảo kết quả tìm kiếm được trả về nhanh chóng và chính xác, ngay cả khi số lượng sách trong cơ sở dữ liệu rất lớn.

- **Đa dạng hóa lựa chọn và tiết kiệm thời gian:**

Đảm bảo rằng hệ thống không chỉ gợi ý những sách phổ biến mà còn giới thiệu các sách mới hoặc ít được biết đến nhưng phù hợp với sở thích người dùng, qua đó mở rộng tầm nhìn và trải nghiệm đọc. Giúp người dùng tiết kiệm thời gian tìm kiếm trên cả database.

- **Thân thiện với ngôn ngữ tự nhiên:**

Cho phép người dùng tìm kiếm sách thông qua các câu truy vấn ngôn ngữ tự nhiên, nâng cao tính tiện lợi và trực quan khi sử dụng hệ thống.

1.3. Lý do thực hiện đề tài

- **Đáp ứng nhu cầu cá nhân hóa:**

Hệ thống đề xuất sách giúp người dùng dễ dàng tiếp cận những cuốn sách mà họ thực sự quan tâm. Việc áp dụng các phương pháp học máy như collaborative filtering và content-based filtering cho phép học hỏi từ hành vi người dùng

⇒ **Có thể liên tục cải thiện gợi ý, giúp tăng sự hài lòng và tương tác của người dùng với nền tảng.**

- **Giảm thiểu sự quá tải thông tin:**

Với kho sách khổng lồ, việc đưa ra lựa chọn giữa hàng nghìn cuốn sách có thể khiến người dùng cảm thấy choáng ngợp. Hệ thống đề xuất giải quyết vấn đề này bằng cách đưa ra các gợi ý phù hợp và có giá trị, giúp người dùng không phải mất nhiều thời gian để tìm kiếm.

- **Thúc đẩy văn hóa đọc:**

Hệ thống không chỉ gợi ý sách theo thói quen hiện tại mà còn khuyến khích người dùng mở rộng phạm vi sở thích, khám phá các thể loại hoặc tác giả mới. Điều này đóng góp vào việc phát triển văn hóa đọc trong cộng đồng, đặc biệt là trong thời đại số.

- **Cải thiện trải nghiệm người dùng:**

Việc có một hệ thống gợi ý sách chính xác và dễ dàng sử dụng giúp người dùng tìm thấy những cuốn sách phù hợp nhanh chóng, từ đó nâng cao trải nghiệm của họ. Đồng thời, với việc tích hợp tính năng đánh giá và chia sẻ, người dùng cũng có thể kết nối và giao lưu với cộng đồng đọc sách rộng lớn.

1.4. Lời cảm ơn

Nhóm em xin chân thành cảm ơn thầy Bùi Tiến Lê, thầy Lê Nhựt Nam, thầy Võ Nhật Tân, thầy Nguyễn Thanh Tình là giảng viên môn “Nhập môn học máy” lớp 22KHMT1 đã nhiệt tình giảng dạy, đưa ra hướng dẫn chi tiết và hỗ trợ cho nhóm em trong quá trình thực hiện đề tài này. Những kiến thức mà thầy cô truyền đạt cùng với những kinh nghiệm mà nhóm em đạt được trong quá trình thực hiện đồ án sẽ là hành trang quý báu giúp nhóm em tiếp tục phát triển trong tương lai.

2. Phân tích và xác định vấn đề

2.1. Thiếu dữ liệu người dùng mới

- **Vấn đề:** Hệ thống đề xuất thường gặp phải vấn đề không có đủ dữ liệu từ hành vi của người dùng để học lướt đánh giá sách. Điều này gây khó khăn trong việc đưa ra các gợi ý chính xác cho người dùng mới (*Cold Start*).
- **Giải pháp:** Sử dụng phương pháp *collaborative filtering* để tìm những người dùng có độ tương đồng cao (tương đồng về lĩnh vực quan tâm, mục đích đọc sách). Ngoài ra, có thể sử dụng dữ liệu từ các nguồn bên ngoài (như xu hướng sách bán chạy, sách nổi bật) để gợi ý cho người dùng mới.

2.2. Chất lượng và tính chính xác của dữ liệu

- **Vấn đề:** Dữ liệu thu thập từ các nguồn khác nhau có thể không đồng nhất hoặc có sự sai lệch, dẫn đến việc đề xuất sách không chính xác. Ví dụ, một cuốn sách có thể có thông tin bị thiếu, hoặc người dùng có thể không cung cấp đủ thông tin cá nhân để hệ thống hoạt động hiệu quả.
- **Giải pháp:** Cải thiện quá trình thu thập và làm sạch dữ liệu, loại bỏ dữ liệu không hợp lệ và bổ sung thông tin thiếu.

2.3. Tính cá nhân hóa và sự đa dạng trong sở thích

- **Vấn đề:** Mỗi người dùng có sở thích đọc sách rất riêng biệt và khó khăn để có thể cá nhân hóa trong việc đề xuất sách cho người dùng. Điều này khiến việc xây dựng một hệ thống đề xuất chính xác và cá nhân hóa gặp khó khăn.
- **Giải pháp:** Xây dựng các mô hình học máy cá nhân hóa, điều chỉnh gợi ý dựa trên các đặc điểm và hành vi riêng của từng người dùng.

2.4. Khó khăn trong việc đánh giá hiệu quả của hệ thống đề xuất

- **Vấn đề:** Đánh giá chính xác hiệu quả của hệ thống đề xuất sách là một thách thức, vì không có một phương pháp duy nhất có thể đo lường mọi khía cạnh của trải nghiệm người dùng. Các chỉ số như độ chính xác, độ bao phủ, và độ đa dạng của các gợi ý cần phải được tính toán và phân tích.
- **Giải pháp:** Áp dụng chỉ số đánh giá là độ lỗi AE để đo lường hiệu quả của hệ thống.

3. Mô hình giải pháp đề xuất

3.1. Thu thập và xử lý dữ liệu

3.1.1. Thu thập dữ liệu

- Sách:

Thu thập từ Google APIs:

Dữ liệu về sách sẽ được thu thập thông qua Google Books API. API này cung cấp một loạt các thông tin chi tiết về sách từ nhiều nguồn, bao gồm:

- **Thông tin cơ bản:** Tên sách, tác giả, nhà xuất bản, ngày phát hành, mô tả, và ISBN.
 - **Đánh giá:** Điểm đánh giá trung bình và số lượt đánh giá của mỗi cuốn sách từ cộng đồng người dùng Google.
 - **Thể loại và từ khóa:** Dữ liệu về thể loại sách, các thẻ từ khóa giúp phân loại sách theo các chủ đề hoặc chủ đề cụ thể (ví dụ: "Khoa học", "Lịch sử", "Tiểu thuyết").
 - **Liên kết và hình ảnh:** Liên kết đến các trang bán sách trực tuyến như Google Play Books và hình ảnh bìa sách.
- Người dùng:

Thu thập từ Google Form: Dữ liệu người dùng sẽ được thu thập trực tiếp thông qua các biểu mẫu Google Form. Các thông tin cơ bản bao gồm:

- **Thông tin cá nhân:** Tên, độ tuổi, giới tính, trường học, chuyên ngành.
- **Mục đích đọc sách của người dùng, được chia thành:**
 - Người dùng đọc sách để trau dồi kiến thức chuyên môn
 - Người dùng đọc sách để tìm hiểu về thế giới xung quanh
 - Người dùng đọc sách để phát triển bản thân.
- **Lịch sử hành vi người dùng:** Sau khi người dùng bắt đầu tương tác với hệ thống (đọc sách, đánh giá sách, tìm kiếm sách), các dữ liệu hành vi sẽ được thu thập để phục vụ việc cải thiện các gợi ý trong tương lai. Cụ thể trong đồ án này, nhóm em sẽ sử dụng lịch sử yêu thích sách để đề xuất những cuốn sách có nội dung tương tự như những cuốn sách người dùng đã thích.

Câu hỏi Câu trả lời Cài đặt

[AI Avengers] - KHẢO SÁT THÓI QUEN ĐỌC VÀ TRA CỨU SÁCH

B I U G K

Sau đây sẽ là mẫu khảo sát thói quen đọc và tra cứu sách của nhóm AI Avengers. Rất mong mọi người sẽ dành một chút ít thời gian để hoàn thành khảo sát để giúp nhóm mình có thêm thông tin cho dự án sắp tới nhé vì những gì bạn sắp điền dưới đây sẽ là những thông tin cực kỳ quý giá đối với dự án này của tụi mình.

Thông tin nhận được từ khảo sát chỉ phục vụ cho mục đích giáo dục!

Biểu mẫu này đang tự động thu thập email của tất cả người trả lời. [Thay đổi cài đặt](#)

Giới tính của bạn là? *

Nam
 Nữ
 Khác...

(+)
()
()
()
()

3.1.2. Xử lý dữ liệu

Làm sạch dữ liệu: bao gồm việc xử lý các giá trị thiếu (missing values), loại bỏ dữ liệu trùng lặp, phát hiện và sửa các lỗi trong dữ liệu như định dạng sai hoặc giá trị bất hợp lý.

Gán nhãn, phân loại, chuyển đổi dữ liệu: gắn các nhãn cụ thể vào dữ liệu thô để tạo ra các bộ dữ liệu có thể sử dụng cho các bài toán học máy, phân tích dữ liệu, hoặc huấn luyện mô hình.

Ví dụ: Ở trường “school” của người dùng, dữ liệu sẽ được xử lý bằng cách gán nhãn như sau:

“Khoa học tự nhiên” -> "HCMUS"

“Bách Khoa” -> “HCMUT”

=> Việc gán nhãn như vậy sẽ giúp quá trình so sánh chuỗi, giúp quá trình huấn luyện mô hình sau này trở nên dễ dàng hơn.

3.2. Thiết kế và triển khai mô hình

3.2.1. Thiết kế mô hình

3.2.1.1. Demographic filtering

Sử dụng Weighted Rating

Tiền xử lý dữ liệu đầu vào bằng cách chọn các mẫu có **Weighted Rating** đủ điều kiện, tức là có số lượt đánh giá lớn hơn ít nhất là QuantileRate (tỷ lệ) của các cuốn sách trong tập dữ liệu.

+Tính toán Weighted Rating của tất cả dữ liệu đã được tiền xử lí:

$$WR(sample) = \frac{sample.RC}{sample.RC+CT} * sample.AR + \frac{CT}{sample.RC+CT} * AM$$

Trong đó:

CT: CountThreshold = quantile_{QuantileRate}(Dataset.RC): Nguồn số lượng

AM: AvgMean = mean(Dataset.AR): Trung bình cộng điểm đánh giá

RC: RatingCount : Số lượng đánh giá

AR: AvgRating : Điểm đánh giá trung bình

3.2.1.2. Content-based filtering và collaborative filtering

Mô hình Exponent sử dụng thuật toán Gradient Descent:

- **Đầu vào:**

- W: Vectơ trọng số
- lr: Tỷ lệ học (Learning Rate)
- max_epoch: Số vòng lặp tối đa (Max Epoch)
- X (kích thước: (D-1) x (d+1)): Ma trận khoảng cách giữa một mục tiêu và (D-1) mẫu khác, với d thuộc tính và thuộc tính bias cho mỗi mục tiêu trong tập dữ liệu gồm D mẫu.

- **Đầu ra:**

Top N mẫu có Điểm tổng thể (Overall Score) cao nhất, liên quan nhất đến mục tiêu.

- **Thuật toán:**

- **Bước 1:** Khởi tạo Vectơ trọng số W.
- **Bước 2:** Với mỗi vòng lặp, lấy ma trận đầu vào X đã tiền xử lý với kích thước (D-1) x (d+1) cho mỗi mục tiêu trong tập dữ liệu.
- **Bước 3:** Tính điểm tổng thể (Overall Score - OS) bằng cách lấy tổng trọng số của lũy thừa X với Vectơ Lambda:

$$OS(X) = W \cdot \exp_{lambda}(X) = W \cdot e^{-lambda * X} = \sum_{i=0}^d w_i e^{-lambda_i x_i}$$

- **Bước 4:** Chọn top N mẫu có Overall Scores cao nhất làm mẫu dự đoán (N predicts).
- **Bước 5:** Tính Ma trận Delta bằng cách tính khoảng cách Cosine (trừ đi 1 để đưa về khoảng [-1, 1]) giữa nhãn dự đoán (predict labels) và nhãn mục tiêu (target labels):

$$\text{Delta}_i = \text{CosineDistance}(\text{predict}_i, \text{label}, \text{target.label}) - 1$$

- **Bước 6:** Tính Vector Gradient bằng cách lấy tích của các vector đầu vào của dự đoán với Delta trung bình:

$$\text{Gradient} = X[\text{predicts}]^T * \text{Delta} / N$$

- **Bước 7:** Cập nhật Vector trọng số W bằng Vector Gradient và Tỷ lệ học lr:

$$W' = W - lr * \text{Gradient}$$

3.2.2. Triển khai mô hình

3.2.2.1. Demographic filtering

- Đầu vào:
 - Tập dữ liệu sách.
- Đầu ra:
 - Top N những cuốn sách có Weighted Rating cao nhất.

3.2.2.2. Content-based filtering

- Đầu vào:
 - Tập dữ liệu sách
- Đầu ra:
 - Top N những cuốn sách liên quan nhất với cuốn sách mục tiêu (theo thể loại sách)

3.2.2.3. Collaborative filtering

- Đầu vào:
 - Tập dữ liệu người dùng.

- Đầu ra:

Top N người dùng liên quan nhất với người dùng mục tiêu (dựa trên mục đích đọc sách).

3.2.2.4. Semantic search

- Đầu vào:

Câu hỏi truy vấn của người dùng.
- Đầu ra:

Top N sách liên quan nhất đến câu hỏi của người dùng.

3.3. Huấn luyện mô hình

3.3.1. Demographic filtering

- **Bước 1:** Tải các thuộc tính đầu vào của tập dữ liệu nội sách:
 - Content ID (từ ‘id’)
 - Content Avg Rating (từ ‘avg_rating’)
 - Content Rating Count (từ ‘rating_count’)
- **Bước 2:** Tiền xử lý dữ liệu đầu vào bằng cách chọn các mẫu có Số lượng đánh giá (Content Rating Count) đủ điều kiện, tức là có số lượt đánh giá lớn hơn ít nhất là QuantileRate của các nội dung trong tập dữ liệu (mặc định: QuantileRate = 90%).
- **Bước 3:** Tính Weighted Rating - WR của tất cả các mẫu đã được tiền xử lý
- **Bước 4:** Trả về top N cuốn sách có điểm đánh giá có trọng số cao nhất.

3.3.2. Content-based filtering

- **Bước 1:** Tải các thuộc tính đầu vào từ Tập dữ liệu nội dung:
 - Content ID: từ trường ‘id’
 - Content ISBN13: từ trường ‘isbn13’
 - Content Title: từ trường ‘title’
 - Content Authors: từ trường ‘authors’
 - Content Published Year: từ trường ‘published_year’
 - Content Page Count: từ trường ‘page_count’

- Content Language: từ trường ‘language’
- Content Description: từ trường ‘description’
- Content Category: từ trường ‘category’

- **Bước 2:** Mã hóa các thuộc tính dạng văn bản bằng Sentence Transformer Model ('all-mpnet-base-v2'):

- Content Title
- Content Authors
- Content Description
- Content Category

- **Bước 3:** Tiền xử lý các thuộc tính bằng cách tính ma trận khoảng cách chuẩn hóa tuyệt đối (Standardized Abs Distance Matrix) (theo Chuẩn hóa Z-score):

- Khoảng cách Cosine: Title, Authors, Description
- Khoảng cách Dữ liệu số (Digits Distance): ISBN13
- Khoảng cách Số (Number Distance): Published Year, Page Count
- Khoảng cách Nhị phân (One Distance): Language

- **Bước 4:** Chia Tập dữ liệu đầu vào thành Tập huấn luyện (Trainset) và Tập kiểm tra (Testset) với tỷ lệ 3:1. Tạo Từ điển Nhãn (Label Dictionary) để ánh xạ Content ID với nhãn tương ứng.
- **Bước 5:** Xác định Vectơ Lambda và tiền xử lý Tập huấn luyện.
- **Bước 6:** Xác định Vectơ trọng số (Weight Vector), Tỷ lệ học (Learning Rate), và Số vòng lặp tối đa (Max Epoch), sau đó sử dụng Mô hình Lũy thừa (Exponent Model) với Gradient Descent để huấn luyện Tập huấn luyện.
- **Bước 7:** Đánh giá mô hình đã được huấn luyện bằng Tập kiểm tra (Testset) thông qua việc tính Lỗi Trung bình (Average Error - AE).

3.3.3. Collaborative filtering

- **Bước 1:** Tải các thuộc tính đầu vào từ Tập dữ liệu người dùng:

- User ID: từ trường ‘id’
- User Gender: từ trường ‘gender’

- User School: từ trường ‘school’
- User Faculty: từ trường ‘faculty’
- User Age: từ trường ‘age’
- User Language: từ trường ‘language’
- User Factor: từ trường ‘factor’
- và nhãn: User Goal: từ trường ‘goal’

- **Bước 2:** Mã hóa thuộc tính User School, User Goal bằng Sentence Transformer Model (‘all-mpnet-base-v2’).
- **Bước 3:** Tiền xử lý các thuộc tính bằng cách tính ma trận khoảng cách chuẩn hóa tuyệt đối (Standardized Abs Distance Matrix) (theo Chuẩn hóa Z-score):
 - Khoảng cách Cosine: School
 - Khoảng cách Dữ liệu số (Digits Distance): Language, Factor
 - Khoảng cách Số (Number Distance): Age
 - Khoảng cách Nhị phân (One Distance): Gender, Faculty
- **Bước 4:** Chia Tập dữ liệu đầu vào thành Tập huấn luyện (Trainset) và Tập kiểm tra (Testset) với tỷ lệ 3:1. Tạo Từ điển Nhãn (Label Dictionary) để ánh xạ User ID với nhãn tương ứng.
- **Bước 5:** Xác định Vector Lambda và tiền xử lý Tập huấn luyện.
- **Bước 6:** Xác định Vector trọng số (Weight Vector), Tỷ lệ học (Learning Rate), và Số vòng lặp tối đa (Max Epoch), sau đó sử dụng Mô hình Lũy thừa (Exponent Model) với Gradient Descent để huấn luyện Tập huấn luyện.
- **Bước 7:** Đánh giá mô hình đã được huấn luyện bằng Tập kiểm tra (Testset) thông qua việc tính Lỗi Trung bình (Average Error - AE).

3.3.4. Semantic search

- Bước 1: Lấy và chuẩn bị dữ liệu đầu vào
 - Dữ liệu sách: Hệ thống tải danh sách sách từ cơ sở dữ liệu, bao gồm các thuộc tính chính: *id, title, authors, description, category, avg_rating, rating_count*.

- Tiền xử lý văn bản: Các thuộc tính như *title*, *authors*, *description*, *category* được làm sạch để loại bỏ ký tự đặc biệt, khoảng trắng dư thừa, và định dạng không cần thiết.
- Bước 2: Chuẩn bị mô hình embedding
 - Khởi tạo mô hình Sentence Transformer: Sử dụng mô hình ***all-MiniLM-L6-v2*** để ánh xạ các thuộc tính văn bản của sách thành các vector nhúng (embeddings).
 - Tích hợp mô hình ngữ nghĩa lớn (Large Language Model - LLM): Mô hình hỗ trợ tìm kiếm ngữ nghĩa dựa trên ngữ cảnh của truy vấn và dữ liệu sách.
- Bước 3: Vector hóa dữ liệu sách
 - Kết hợp thuộc tính: Gộp các thuộc tính quan trọng của từng cuốn sách (ví dụ: tiêu đề, tác giả, mô tả) thành một chuỗi văn bản duy nhất.
 - Tạo vector nhúng: Tạo các vector nhúng chuẩn hóa cho mỗi cuốn sách thông qua ***Sentence Transformer***.
- Bước 4: Xây dựng FAISS Index
 - Khởi tạo FAISS Index: Sử dụng ***FAISS*** (Facebook AI Similarity Search) để xây dựng chỉ mục tìm kiếm hiệu quả.
 - Thêm vector vào chỉ mục: Các vector nhúng của sách được thêm vào FAISS Index để phục vụ cho việc tìm kiếm tương đồng.
- Bước 5: Tìm kiếm ngữ nghĩa (Semantic Search)
 - Nhận truy vấn từ người dùng: Truy vấn được xử lý qua mô hình Sentence Transformer để tạo vector nhúng cho câu hỏi của người dùng.
 - Tìm kiếm trên FAISS Index: FAISS tính toán độ tương đồng giữa truy vấn và các vector sách đã lưu. Kết quả trả về là danh sách các cuốn sách tương đồng với điểm số tương ứng.
- Bước 6: Lọc và sắp xếp kết quả
 - Áp dụng ngưỡng tương đồng: Chỉ giữ lại các kết quả có điểm số tương đồng lớn hơn ngưỡng đã định (ví dụ: 0.1).
 - Sắp xếp kết quả: Danh sách kết quả được sắp xếp giảm dần theo điểm số để đảm bảo các sách phù hợp nhất xuất hiện đầu tiên.
- Bước 7: Xuất kết quả
 - Trả về thông tin chi tiết của các sách được gợi ý.

3.4. Đánh giá hiệu suất mô hình

Độ lỗi Average Error (AE): được dùng để đo lường mức độ sai lệch trung bình giữa các giá trị dự đoán của mô hình và các giá trị thực tế trong một tập dữ liệu. Cụ thể, nó tính toán trung bình của các "Average Absolute Deltas" (AAD) trên toàn bộ tập dữ liệu.

$$AE = \text{mean}(\text{AvgAbsDeltas}(\text{Dataset})) = \sum_{\text{target}} \text{AvgAbsDelta}(\text{target}) / D$$

$$= \sum_{\text{target}} (\sum_{\text{sample}} |\text{Delta}_{\text{target}}(\text{sample})| / N) / D$$

3.5. So sánh và tinh chỉnh mô hình

3.5.1. Content-based filtering

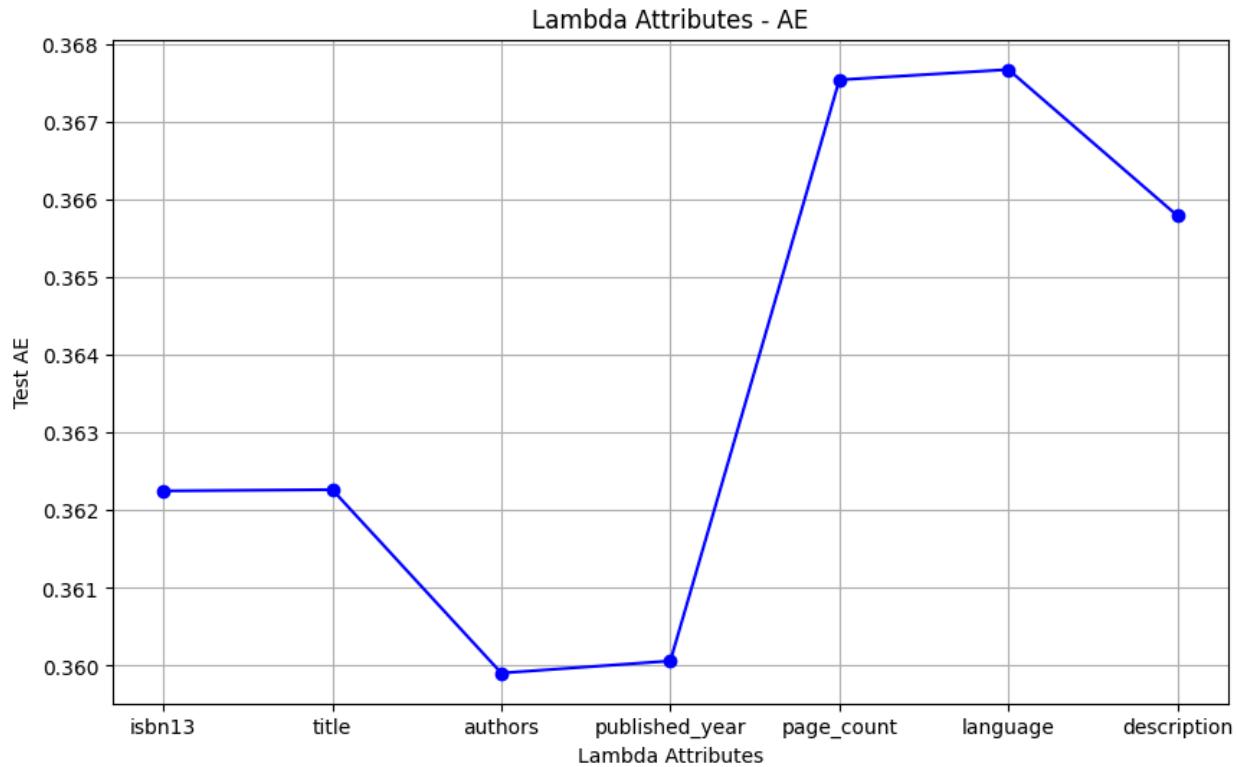
Với $N=10$ và trọng số initial_W = None (được khởi tạo một cách ngẫu nhiên).

a) Lambda

Đối với mỗi thuộc tính, gán giá trị Lambda của thuộc tính đó bằng 2 và các thuộc tính khác bằng 1.

Cố định LearningRate = 0.03 và MaxEpoch = 100.

Phân tích Lỗi trung bình (AE) dựa trên từng thuộc tính Lambda.



Lambda tốt nhất: $(1, 1, 2, 2, 1, 1, 1)$ với:

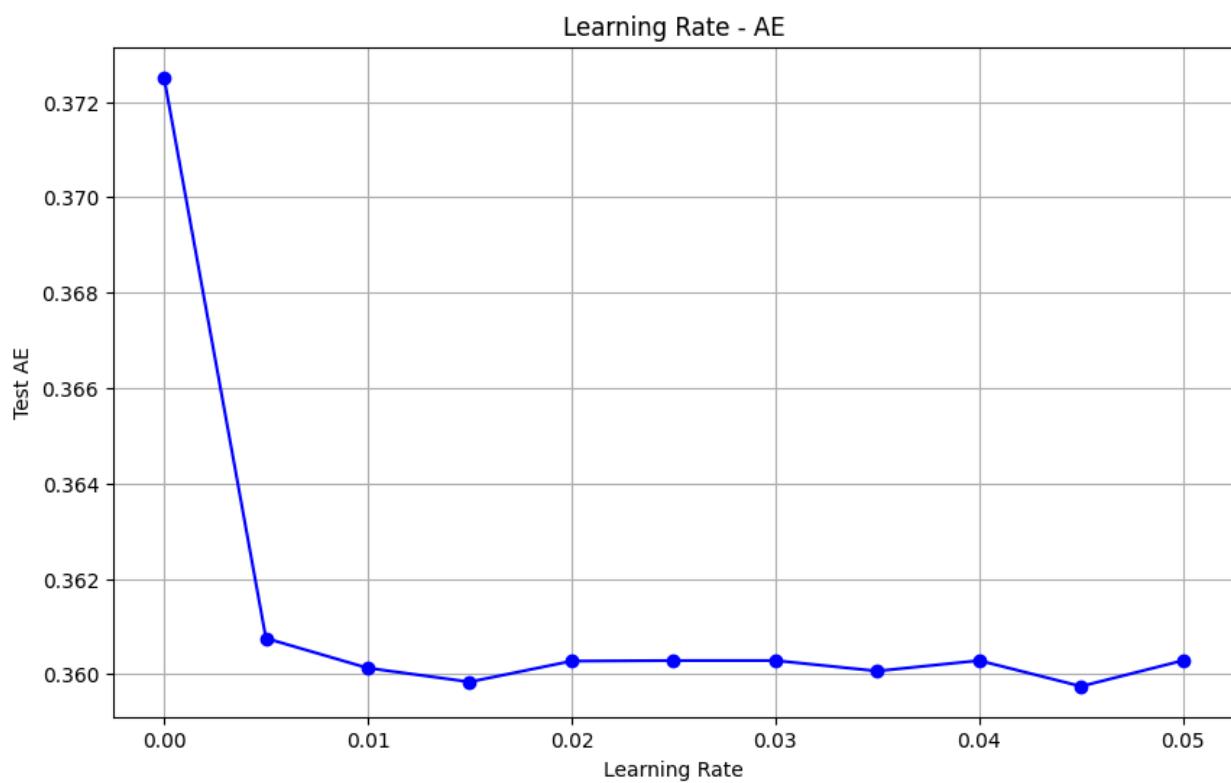
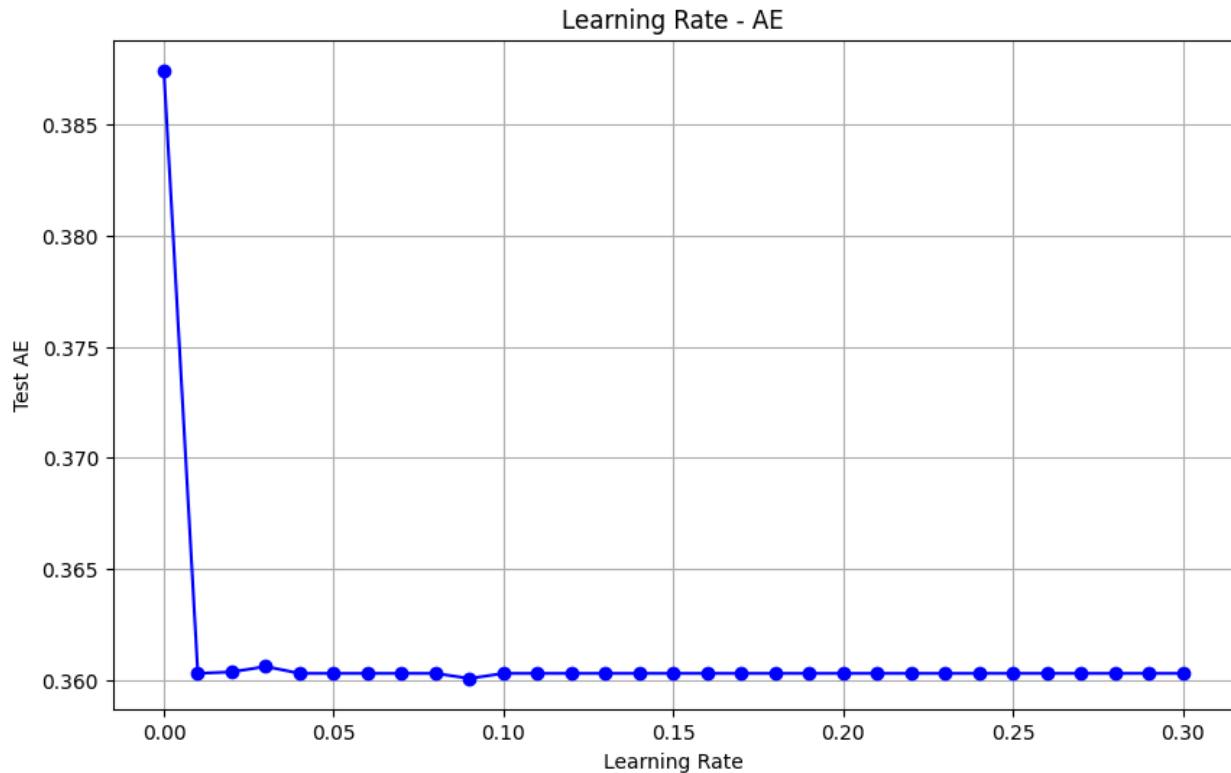
$$\lambda_{Authors} = \lambda_{PublishedYear} = 2$$

Mô hình tốt nhất:

$$f(X) = w_0 + w_1 e^{-x_1} + w_2 e^{-x_2} + w_3 e^{-2x_3} + w_4 e^{-2x_4} + w_5 e^{-x_5} + w_6 e^{-x_6} + w_7 e^{-x_7}$$

b) Tốc độ học (Learning Rate)

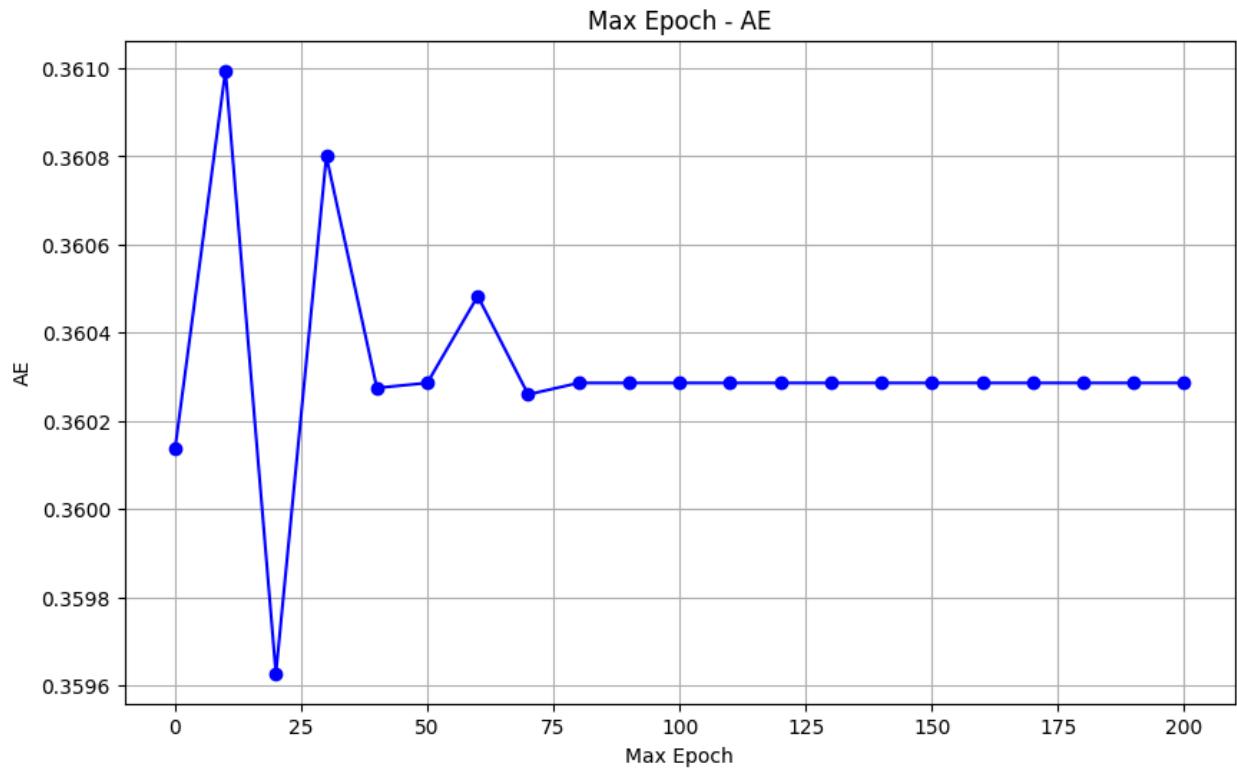
- Cố định λ = $(1, 1, 2, 2, 1, 1, 1)$ và $MaxEpoch = 100$
- Phân tích độ lỗi AE dựa trên tốc độ học (Learning Rate).

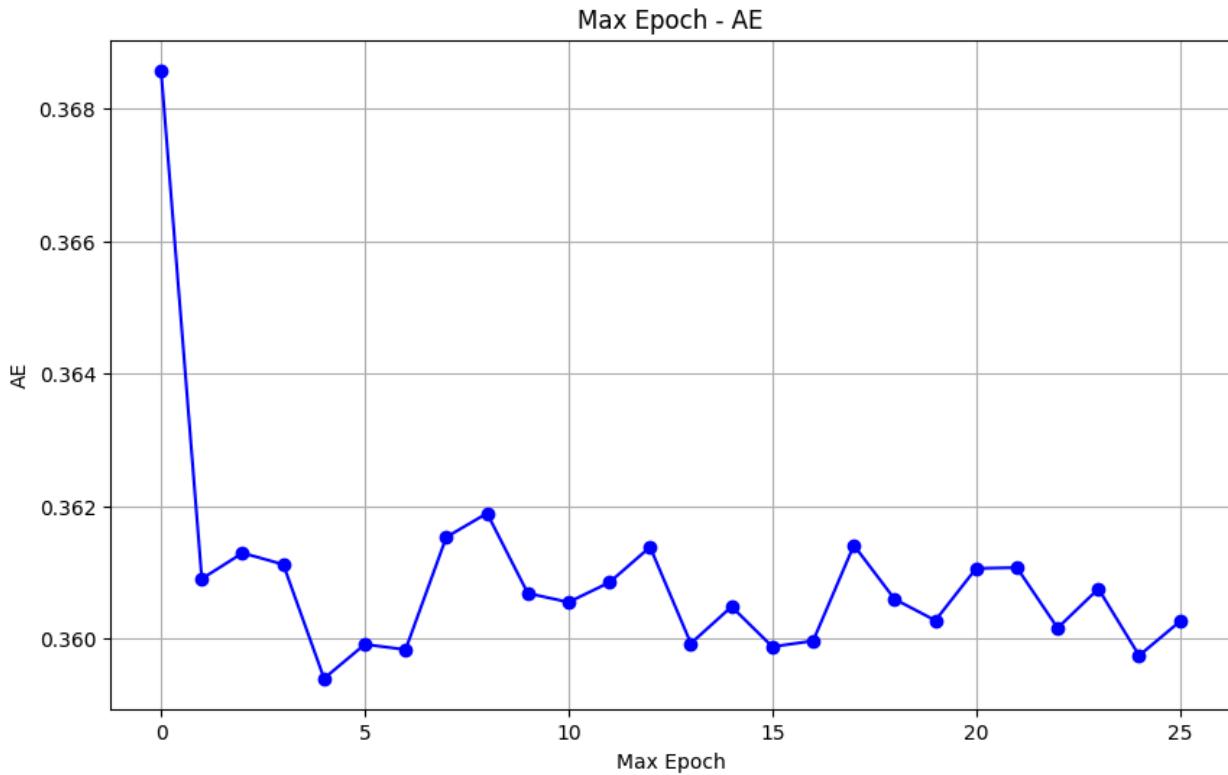


*Tốc độ học tốt nhất: 0.045

c) Số lần lặp (Max_epoch)

- Cố định $\Lambda = (1, 1, 2, 2, 1, 1, 1)$ và $LearningRate = 0.045$.
- Phân tích độ lỗi AE dựa trên số lần lặp (Max_Epoch).





Số lần lặp (Max Epoch) tốt nhất: 4

d) Kết quả

Các tham số tốt nhất:

- i) $N = 10$
- ii) $initialW = None$
- iii) $Lambda = (1, 1, 2, 2, 1, 1, 1)$
- iv) $LearningRate = 0.045$
- v) $MaxEpoch = 4$

⇒ Kết quả khi áp dụng các tham số trên $TestAE = 0.36991610927581775$

Công thức mô hình:

$$f(X) = w_0 + w_1 e^{-x_1} + w_2 e^{-x_2} + w_3 e^{-2x_3} + w_4 e^{-2x_4} + w_5 e^{-x_5} + w_6 e^{-x_6} + w_7 e^{-x_7}$$

With $w_0 = 28.49500599, w_1 = 6.27121165, w_2 = 24.11881559,$

$w_3 = 20.08518844, w_4 = 27.37318616, w_5 = 25.78043362,$

$$w_6 = 28.15006633, w_7 = 23.79187828$$

Với một cuốn sách đích, sử dụng mô hình trên để trả về kết quả là mười cuốn sách liên quan nhất với cuốn sách đích và có Overall Scores cao nhất.

3.5.2. Collaborative filtering

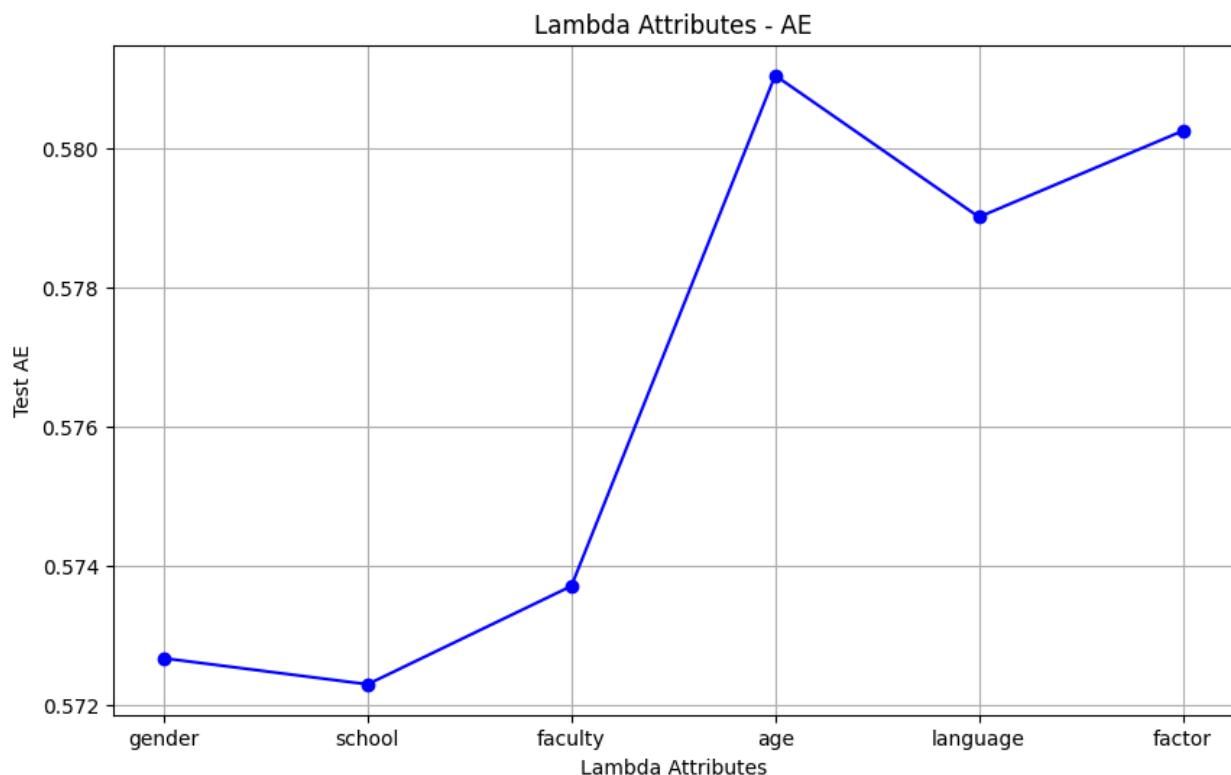
Với $N = 10$ và trọng số $initialW = None$ (được khởi tạo một cách ngẫu nhiên)

a) Lambda

Đối với mỗi thuộc tính, gán giá trị Lambda của thuộc tính đó bằng 2 và các thuộc tính khác bằng 1.

Cố định LearningRate = 0.03 và MaxEpoch = 100.

Phân tích Lỗi trung bình (AE) dựa trên từng thuộc tính Lambda.



*Lambda tốt nhất: (2, 2, 1, 1, 1, 1):

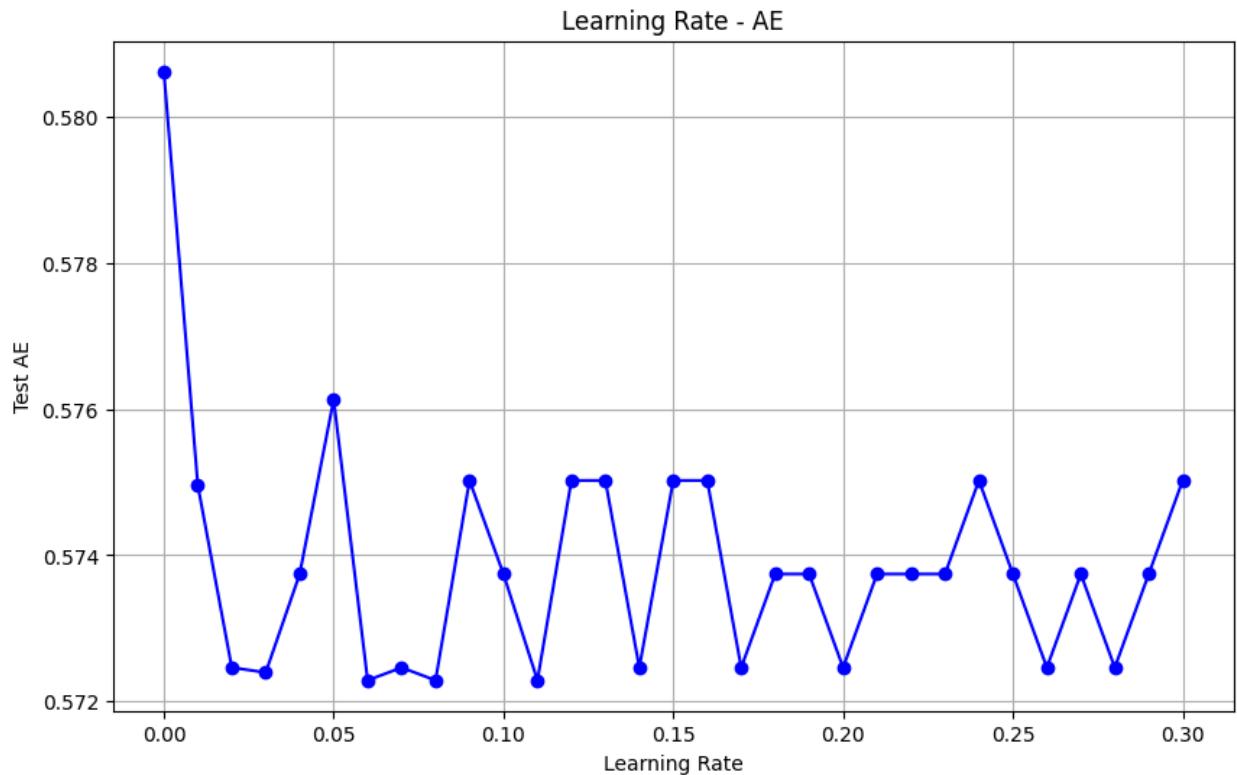
$$\lambda_{Gender} = \lambda_{School} = 2$$

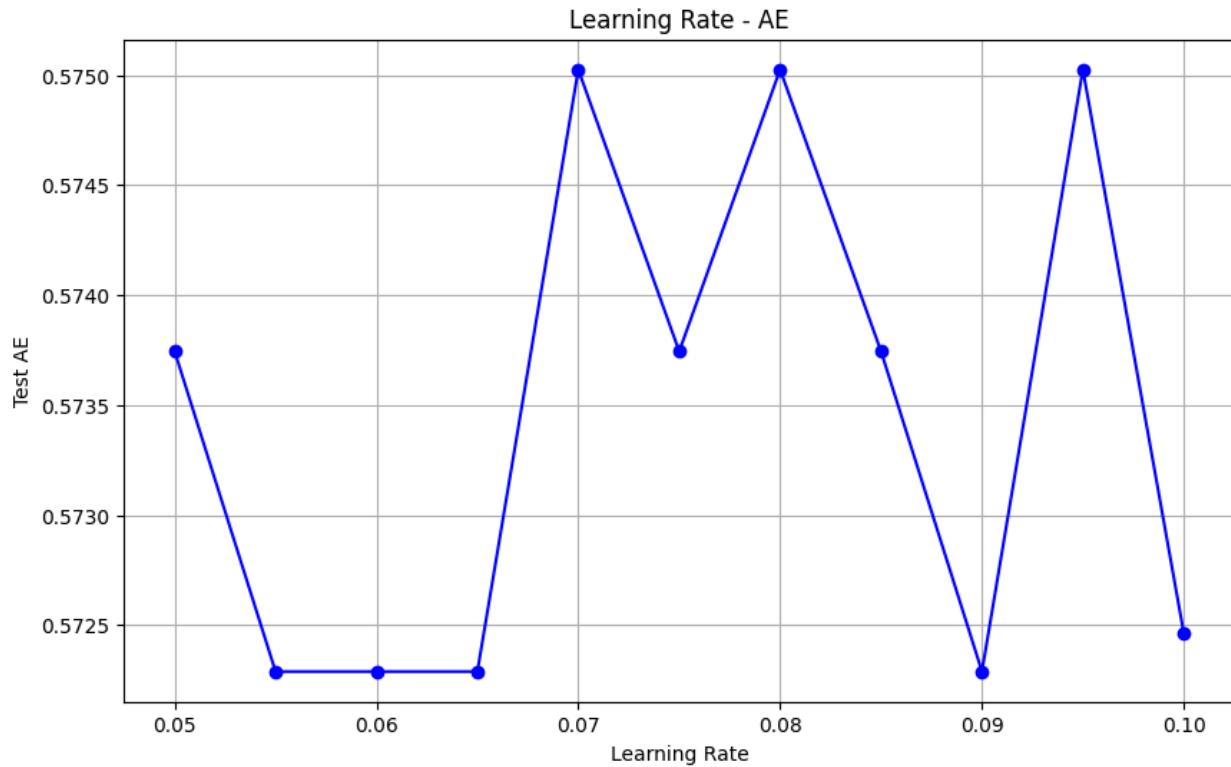
*Mô hình tốt nhất:

$$f(X) = w_0 + w_1 e^{-2x_1} + w_2 e^{-2x_2} + w_3 e^{-x_3} + w_4 e^{-x_4} + w_5 e^{-x_5} + w_6 e^{-x_6}$$

b) Tốc độ học (learning rate)

- Cố định $\Lambda = (2, 2, 1, 1, 1, 1)$ và $MaxEpoch = 100$
- Phân tích độ lỗi AE dựa trên learning rate

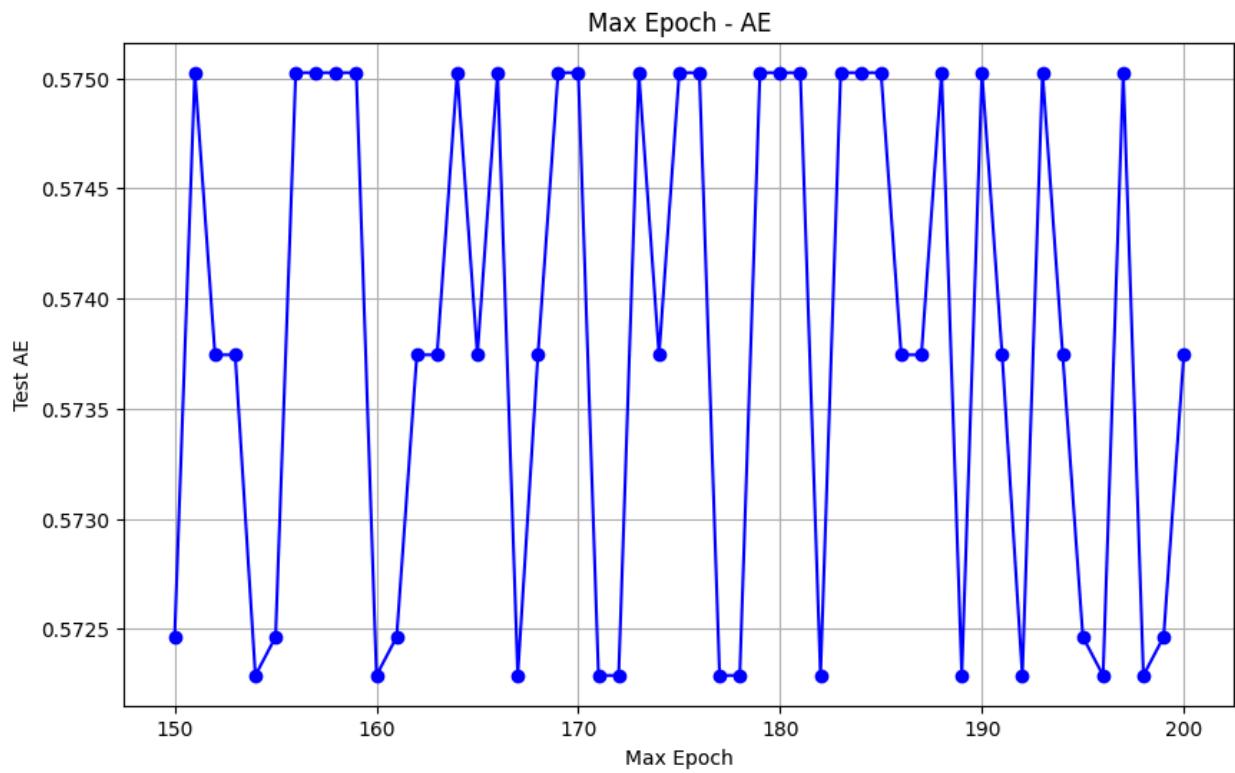
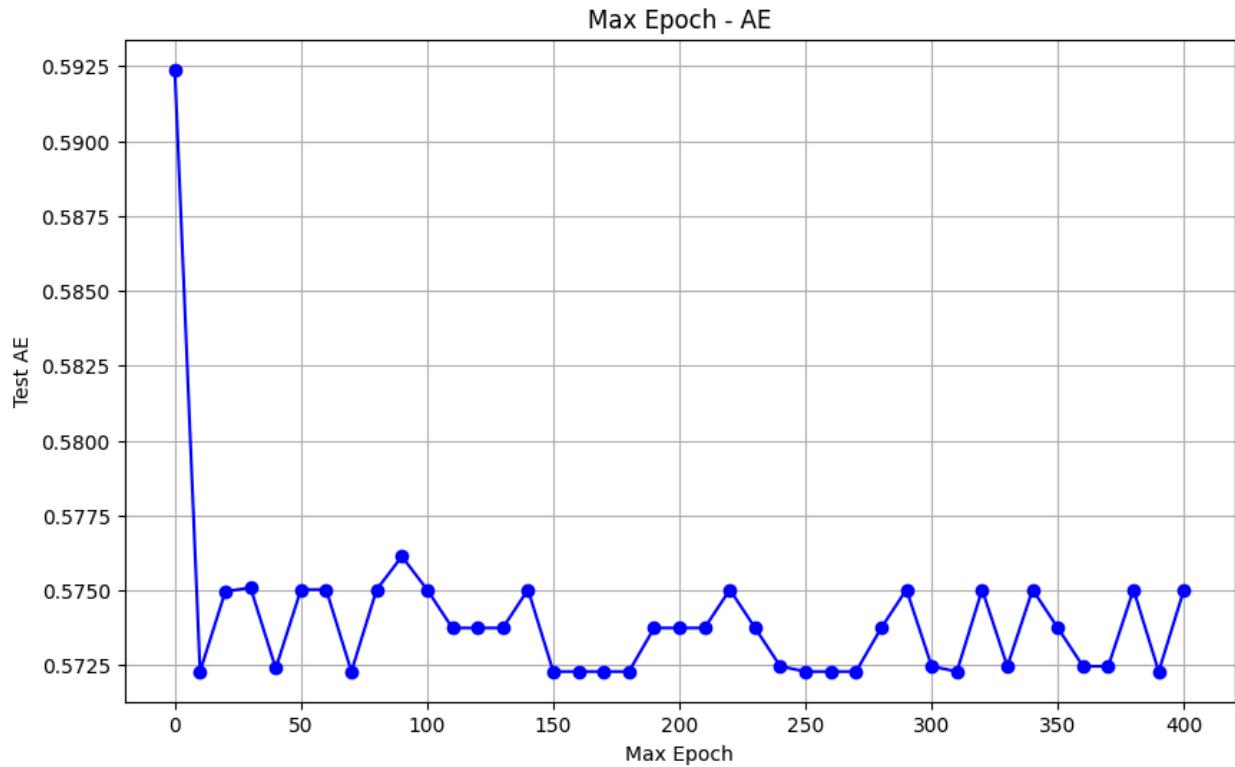




*Tốc độ học (learning rate) tốt nhất: 0.06

c) Số lần lặp (Max Epoch)

- Cố định $\Lambda = (2, 2, 1, 1, 1, 1)$ và $LearningRate = 0.06$
- Phân tích độ lỗi AE dựa trên mỗi số lần lặp (Max Epoch).



* Số lần lặp (Max Epoch) tốt nhất: 160

d) Kết quả

Những tham số tốt nhất:

- $N = 10$
- $initialW = None$
- $Lambda = (2, 2, 1, 1, 1, 1)$
- $LearningRate = 0.06$
- $MaxEpoch = 160$

=> Kết quả khi áp dụng các tham số trên vào $TestAE = 0.6170126835505165$

Công thức mô hình:

$$f(X) = w_0 + w_1 e^{-2x_1} + w_2 e^{-2x_2} + w_3 e^{-x_3} + w_4 e^{-x_4} + w_5 e^{-x_5} + w_6 e^{-x_6}$$

Với $w_0 = 1247.49043159, w_1 = 1247.31508567,$

$w_2 = 824.14274302, w_3 = 764.67227592,$

$w_4 = 1181.92896903, w_5 = 1242.0339137,$

$w_6 = 1006.54043616$

Với một người dùng dịch, sử dụng mô hình trên để trả về N người dùng liên quan nhất với người dùng dịch (những người có Overall Scores cao nhất).

Hệ thống sẽ đề xuất những cuốn sách mà những người dùng này yêu thích cho người dùng dịch.

4. ... và hướng phát triển

5. Ứng dụng và demo thực tế

5.1. Mức độ tích hợp mô hình vào ứng dụng web

a) Demographic filtering

Gửi query tới API: /popular_books

Responses

Curl

```
curl -X 'GET' \
'http://127.0.0.1:8000/popular_books?n=45' \
-H 'accept: application/json'
```

Request URL

```
http://127.0.0.1:8000/popular_books?n=45
```

Server response

Code	Details
200	Response body <pre>{ "status": "success", "message": "Popular books retrieved successfully", "data": ["480", "223", "81", "327", "34", "416", "154", "108", "12", "287", "22", "497", "431", "145", "83", "110", "324", "7", "111", "343", "253", "159", "87", "146"] }</pre> <p>Download</p>

Response headers

```
content-length: 338
content-type: application/json
date: Fri, 03 Jan 2025 08:21:56 GMT
server: uvicorn
```

b) Content-based filtering

Gửi query tới API: /recommend

Responses

Curl

```
curl -X 'POST' \
'http://127.0.0.1:8000/recommend' \
-H 'accept: application/json' \
-H 'Content-type: application/json' \
-d '{
    "book_id": 5,
    "n": 10
}'
```

Request URL

<http://127.0.0.1:8000/recommend>

Server response

Code	Details
200	Response body

```
{
  "status": "success",
  "message": "Recommendations retrieved successfully",
  "data": [
    {
      "book_id": "105",
      "label": "Cooking",
      "best_value": 0.9
    },
    {
      "book_id": "461",
      "label": "Science",
      "best_value": 0.9
    },
    {
      "book_id": "454",
      "label": "Biography & Autobiography",
      "best_value": 0.9
    },
    {
      "book_id": "349",
      "label": "Business & Economics",
      "best_value": 0.9
    },
    {
      "book_id": "248",
      "label": "Business & Economics",
      "best_value": 0.9
    }
  ]
}
```

Download

Response headers

```
content-length: 715
content-type: application/json
date: Fri, 03 Jan 2025 09:23:28 GMT
server: unicorn
```

c) Semantic Search

Nội dung truy vấn: “History books”

Responses

Curl

```
curl -X 'POST' \
'http://127.0.0.1:8000/search' \
-H 'accept: application/json' \
-H 'Content-type: application/json' \
-d '{
    "query": "History books",
    "top_k": 5
}'
```

Request URL

<http://127.0.0.1:8000/search>

Server response

Code	Details
200	Response body

```
[
  {
    "id": 45,
    "title": "America's Hidden History",
    "authors": "Kenneth C. Davis",
    "category": "History",
    "rating": 4.5,
    "similarity_score": "0.51",
    "preview_url": "https://books.google.com.vn/books?id=PIIVpN0sUHC&dq=isbn:9780061118180&hl=&cd=1&source=gbs_api",
    "img_url": "https://images.ismbdb.com/covers/81/80/9780061118180.jpg"
  },
  {
    "id": 259,
    "title": "The LEGO Book",
    "authors": "Daniel Lipkowitz",
    "category": "Juvenile Nonfiction",
    "rating": 1.4,
    "similarity_score": "0.40",
    "preview_url": "https://books.google.com.vn/books?id=4bj_ngECAAQ&dq=isbn:9780756669348&hl=&cd=1&source=gbs_api",
    "img_url": "https://images.ismbdb.com/covers/09/34/978075666934.jpg"
  },
  {
    "id": 151,
    "title": "The Marsh of Other Suns",
    "authors": "Isabel Wilkerson",
    "category": "History",
    "rating": 2.6,
    "similarity_score": "0.40",
  }
]
```

Download

Response headers

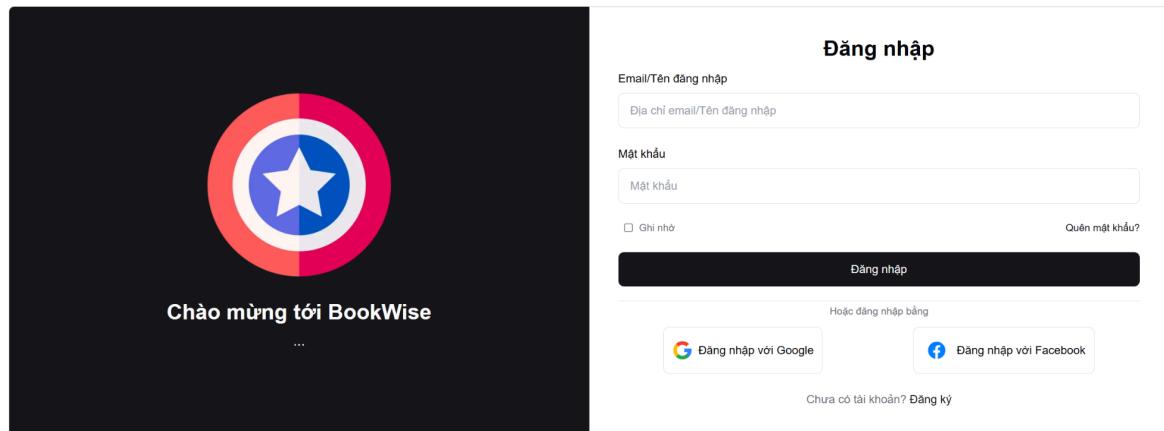
5.2. Giao diện và trải nghiệm của người dùng

5.2.1. Đăng ký

The image displays a user flow for account creation on the BookWise platform, consisting of four sequential screens:

- Welcome Screen:** Shows a Captain America shield logo and the text "Chào mừng tới BookWise".
- Registration Step:** Titled "Đăng ký", it contains fields for "Username" (Tên đăng nhập), "Email" (Địa chỉ email), "Mật khẩu" (Mật khẩu), and "Xác nhận mật khẩu" (Xác nhận mật khẩu). It includes a "Đăng ký" button and a "Đã có tài khoản? Đăng nhập" link.
- Profile Step:** Titled "Hoàn thiện hồ sơ", it features five circular icons for profile picture, gender, birth date, and interests. It includes dropdowns for "Giới tính" (Gender) and "Ngày sinh" (Birth Date), and buttons for "Quay lại" (Back) and "Tiếp tục" (Next).
- Education Step:** Titled "Hoàn thiện hồ sơ", it includes dropdowns for "Trường" (School) and "Khoa" (Faculty). It has "Quay lại" and "Tiếp tục" buttons.
- Interests Step:** Titled "Hoàn thiện hồ sơ", it has a text input field for "Mục tiêu lớn nhất khi đọc sách" (Primary goal when reading books) with placeholder text "Chia sẻ mục tiêu của bạn". It includes "Quay lại" and "Tiếp tục" buttons.
- Final Step:** Titled "Hoàn thiện hồ sơ", it shows "Lĩnh vực quan tâm" (Areas of interest) with options like Công nghệ, Khoa học, Kinh doanh, Tâm lý học, Văn hóa, and Ngôn ngữ thường đọc (Languages read) with options like Tiếng Việt and Tiếng Anh. It includes "Quay lại" and "Hoàn thành" (Finish) buttons.

5.2.2. Đăng nhập



5.2.3. Trang thông tin người dùng

A screenshot of the BookWise user profile page. The top navigation bar includes the BookWise logo, a search bar with fields for "Tim theo" and "Tên sách...", and a "Đăng nhập" button. The left sidebar contains various icons for account management. The main content area is titled "Trang cá nhân" and shows the following profile information: Username (gnoul), Email (tnhl.us@gmail.com), Trường (UIT), Khoa (MT), and Ngày sinh (Sun Jan 01 1911 00:00:00 GMT+0706 (Indochina Time)). There is a blue "Update Profile" button at the bottom of this section. At the very bottom of the page, there is a footer bar with sections for "Book Wise - HCMUS" (including a brief description and social media links), "Thông tin liên lạc" (with contact details like email support@book-wlse.com and phone +91 945 658 3256), "Products" (with links to Home, About, Pricing, and Features), and a "Hãy tham gia cùng chúng tôi" section for newsletter subscription.

5.2.4. Trang chủ

Top Book

Best Seller

Có thể bạn sẽ thích

Mọi người cũng đang xem

Book Wise

Tìm theo... Tên sách... Đăng nhập Đăng ký

Minecraft Wonder Vogue and The TouchPoints Dirt on My Shirt

Mọi người cũng đang xem

Book Wise - HCMUS Thông tin liên lạc support@book-wise.com Products Home

Hãy cùng chúng tôi khám phá thế giới sách với hàng ngàn cuốn sách phong phú và đa dạng.

5.2.5. Hiển thị danh sách tìm kiếm sách

The screenshot shows the Book Wise search interface. On the left, there is a sidebar with various filters and a main search area with a search bar and a 'Tim theo' button. The main content area displays a list of books based on the search term 't'. Each book entry includes the book cover, title, author, genre, rating, and a 'Xem' (View) button.

Tìm theo: t

Bộ lọc tìm kiếm

Tác giả: Nhập tên tác giả

Năm xuất bản: 1900 - 2025

Thể loại: Chọn thể loại

Danh giá: 0 sao - 5 sao

Ngôn ngữ: Chọn ngôn ngữ

Áp dụng bộ lọc

Từ khóa tìm kiếm: t

Sắp xếp: ★ Danh giá cao

The Fast Metabolism Diet
Viết bởi: Haylie Pomroy
Health & Fitness
★★★★★ 5 (734)

The Future, Declassified
Viết bởi: Mathew Burrows
History
★★★★★ 5 (440)

5.2.6. Danh sách các quyển sách yêu thích của bạn

The screenshot shows the Book Wise favorite books page. It features a sidebar with various filters and a main content area titled 'Danh sách yêu thích của bạn'. The page displays a grid of book covers with their titles, authors, genres, ratings, and a brief description.

Danh sách yêu thích của bạn

JEFFERY DEAVER
The Sleeping Doll
Fiction
The Sleeping Doll
Jeffery Deaver
3.0 - 149 đánh giá

JAMES PATTERSON
Michael Bennett: Tick Tock
Fiction
Tick Tock
James Patterson, Michael ...
5.3 - 968 đánh giá

KATHRYN STOCKETT
The Help
Fiction
The Help
Kathryn Stockett
0.2 - 720 đánh giá

CHRISTOPHER KNIGHT AND ALAN BUTLER
Before the Pyramids
Health & Fitness
Before the Pyramids
Christopher Knight and Alan Butler
0.2 - 734 đánh giá

MATTHEW BURROWS
The Future, Declassified
History
The Future, Declassified
Matthew Burrows
0.7 - 440 đánh giá

THE 4-HOUR WORKWEEK
The Short and Tragic Life of Robert Peace

5.2.7. Danh sách các tác giả mà bạn đang theo dõi

The screenshot shows a list of authors on the Book Wise platform. On the left is a vertical sidebar with icons for navigation. The main area has a title "Tác giả bạn đang theo dõi". Below it are six author profiles in cards:

- Michael Pollan** (108 lượt xem) - A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative...
- Michael Lewis** (11 lượt xem) - A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative...
- Haylie Pomroy** (297 lượt xem) - A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative...
- Mark Murphy** (151 lượt xem) - A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative...
- Jeff Guinn** (46 lượt xem) - A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative...
- Teresa Rhyne** (604 lượt xem) - A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative...
- Michael Moss** (335 lượt xem) - A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative...

5.2.8. Trang hiển thị thông tin chi tiết của sách

5.2.8.1. Thông tin chung của sách

The screenshot shows the details page for the book "The Fast Metabolism Diet" by Haylie Pomroy. On the left is a vertical sidebar with icons for navigation. The main area features the book cover for "BEFORE THE PYRAMIDS" by Christopher Knight, followed by the book details for "The Fast Metabolism Diet".

The Fast Metabolism Diet
ISBN: 9780307988276 • 273 trang • Ngôn ngữ: English
viết bởi Haylie Pomroy (Hủy theo dõi)
Thể loại: Health & Fitness • Sản xuất: 2013-04-09

#1 NEW YORK TIMES BESTSELLER • Kick your metabolism into gear with a diet program that uses the fat-burning power of food to help you lose up to 20 pounds in 28 days! "This is not a fad diet. It's a medically proven method of food as medicine to fight obesity, cure chronic illness, and heal a broken metabolism."—Jacqueline Fields, M.D. Hailed as "the metabolism whisperer," Haylie Pomroy reminds us that food is not the enemy but medicine needed to rev up your sluggish, broken-down metabolism to turn your body into a fat-burning furnace. On this plan you're going to eat a lot—three full meals and at least two snacks a day—and you're still going to lose weight. What... Đọc thêm

★★★★★ (734 đánh giá)

Đọc Thêm vào yêu thích

5.2.8.2. Các quyền sách có liên quan

The screenshot shows the Book Wise website interface. At the top, there is a search bar with the placeholder "Tim theo" and "Tên sách...". Below the search bar, there is a navigation menu with icons for search, history, and account. The main content area displays a book cover for "CHRISTOPHER KNIGHT Co-author of the Bestseller The Imitation Game". Below the book cover, there is a section titled "Bạn cũng có thể thích" (You might also like) which lists five other books:

- Patricia Cornwell - The Front**: Fiction, 5.4 rating, 445 reviews.
- Bilquis Sheikh - I Dared to Call Him Father**: Biography and Autobiography, 1.9 rating, 317 reviews.
- Seraphina**: Juvenile Fiction, 5.3 rating, 820 reviews.
- Fern Michaels - Hot Shot**: Fiction, 2.1 rating, 798 reviews.
- Dave Ramsey - The Total Money Makeover**: Business and Economics, 4.7 rating, 970 reviews.

5.2.9. Khám phá các tác giả

The screenshot shows the Book Wise website interface, similar to the previous one, but with a different section title. The main content area displays a section titled "Khám phá các tác giả" (Discover authors) which lists ten authors in a grid format:

- Abraham Verghese**: 167 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Adam Bryant**: 479 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Adam Goodheart**: 256 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Adam Hochschild**: 581 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Adam Mansbach**: 725 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Adele Faber**: 182 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Akira Toriyama**: 432 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Alan Moore**: 330 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Alastair Bonnett**: 120 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Alex Kendrick**: 703 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Alex Tresniowski**: 48 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...
- Alexander Rose**: 138 lượt xem. A creator of written works, including books, articles, scripts, poetry, and more. Authors may specialize in various genres such as fiction, nonfiction, academic writing, or creative writing. They often engage ...

5.2.10. Khám phá các thể loại

Book Wise

Tìm theo

Tên sách...

»

Khám phá các thể loại sách

Các thể loại sách

Humor
Works intended to amuse and entertain through comedy and wit
4 views 16 books

Cooking
Books about food preparation, recipes, and culinary arts
857 views 16 books

Family and Relationships
Content about interpersonal connections and family dynamics
671 views 15 books

Comics and Graphic Novels
Sequential art storytelling in comic or graphic novel format
475 views 14 books

Self-help
Books focused on personal improvement and development
507 views 9 books

Social Science
Study of human society and social relationships
333 views 9 books

Religion
Study of religious beliefs, behaviors, and institutions
281 views 8 books

Sports and Recreation
Topics related to athletic activities and leisure pursuits
532 views 6 books

Previous 1 2 3 4 5 6 7 8 9 Next

6. Tài liệu tham khảo

- Nghiên cứu và hướng dẫn:
 - [1] [Getting Started with a Movie Recommendation System](#) - by Ibtesama, Kaggle.
 - [2] [Building RAG using Gemma & Faiss Vector DB](#) - by Youssef Hosni, Kaggle.
- Công nghệ sử dụng:
 - [3] Sentence-Transformers, *all-MiniLM-L6-v2*, <https://www.sbert.net>.
 - [4] Facebook AI Research, *FAISS (Facebook AI Similarity Search)*,
<https://github.com/facebookresearch/faiss>.
-