TRƯỜNG ĐẠI HỌC KINH TẾ ĐÀ NẴNG

KHOA THƯƠNG MẠI ĐIỆN TỬ

BÁO CÁO ĐỀ ÁN THỰC HÀNH 1

ĐỀ TÀI

Thành viên: Hoàng Nguyên Khang

Nguyễn Đăng Quốc Khánh

Lớp: 48K29.2

GVHD: Trần Danh Nhân

Đà Nẵng, ngày __ tháng 4 năm 2025

LỜI CẢM ƠN

Trước hết, chúng em xin được gửi lời cảm ơn tới thầy Trần Danh Nhân – giảng viên Khoa Thương mại điện tử, Trường Đại học Kinh tế - Đại học Đà Nẵng. Thầy không chỉ truyền đạt cho chúng em những kiến thức vô cùng quý báu mà còn luôn tận tình hướng dẫn, đồng hành và tạo động lực để chúng em có thể hoàn thành bài tập lớn này một cách tốt nhất.

Nhờ có sự giảng dạy tâm huyết và sự hỗ trợ nhiệt tình của thầy, chúng em đã được tiếp cận và hiểu sâu hơn về các kiến thức chuyên môn như thu thập dữ liệu, xử lý và trực quan hóa dữ liệu, từ đó rút ra những insights có giá trị cho doanh nghiệp. Chính những kiến thức này đã giúp chúng em không chỉ hoàn thành bài tập một cách hiệu quả mà còn mở ra cho chúng em một nền tảng vững chắc cho hành trình phát triển bản thân cũng như chuyên môn trong tương lai.

Tuy nhiên, do hạn chế về kiến thức lý luận cũng như kinh nghiệm thực tế, bài báo cáo của chúng em chắc chắn sẽ không tránh khỏi những thiếu sót. Trong quá trình thực hiện bài tập, có thể còn những điểm chưa hoàn thiện hoặc chưa sâu sắc, mong thầy bỏ qua. Chúng em rất mong sẽ nhận được những góp ý chân thành, những nhận xét mang tính xây dựng từ thầy để có thể rút ra bài học và kinh nghiệm quý báu, từ đó cải thiện và hoàn thiện hơn các kỹ năng của bản thân nhằm có thể cạnh tranh trong thời điểm mà AI lớn manh như hiện nay.

Một lần nữa, chúng em xin chân thành cảm ơn thầy vì tất cả sự giúp đỡ và đồng hành trong suốt thời gian qua.

1. Giới thiệu

a. Bối cảnh

Trong thời đại chuyển đổi số hiện nay, giáo dục trực tuyến đã trở thành một xu hướng tất yếu và phát triển mạnh mẽ trên toàn cầu. Trong số các nền tảng học trực tuyến, Udemy nổi bật như một "chợ giáo dục mở", nơi giảng viên từ khắp nơi có thể chia sẻ kiến thức đến hàng triệu người học trên toàn thế giới. Tuy nhiên, để tối ưu hóa hiệu quả giảng dạy và nâng cao trải nghiệm học tập, việc phân tích dữ liệu người học, khóa học và mô hình kinh doanh trên nền tảng này là vô cùng cần thiết.

b. mục tiêu dự án

Dự án này là một nghiên cứu thị trường nhằm phân tích tình hình kinh doanh hiện tại của Udemy – một nền tảng học trực tuyến hàng đầu thế giới. Với vai trò là chuyên viên phân tích dữ liệu, nhóm tác giả có nhiệm vụ tổng hợp, xử lý và trực quan hóa dữ liệu từ Udemy nhằm đưa ra các góc nhìn thực tế về hoạt động kinh doanh, xu hướng khóa học, hành vi người học, và tiềm năng phát triển trong từng chuyên mục. Đồng thời áp dụng công nghệ học máy nhằm giới thiệu đến người dùng những khóa học đáng để trải nghiệm trên nền tảng này.

c. Giới thiệu về Udemy

Udemy là một nền tảng học trực tuyến phổ biến, mang đến hàng loạt khóa học ở nhiều lĩnh vực khác nhau, chẳng hạn như: lập trình, marketing, thiết kế,... cho người học trên toàn thế giới. Ra đời vào năm 2010, nền tảng này hoạt động như một chợ giáo dục nơi giảng viên có thể tạo và chia sẻ các bài giảng, còn người học thì dễ dàng tiếp cận để trau dồi kỹ năng hoặc khám phá kiến thức mới. Nội dung giảng dạy rất phong phú, từ công nghệ, kinh doanh cho đến phát triển bản thân hay nghệ thuật sáng tạo, biến Udemy thành công cụ hữu ích phục vụ cả mục tiêu nghề nghiệp lẫn cá nhân. Mô hình học tập ở đây đề cao sự linh hoạt, cho phép người dùng tự quyết định tốc độ học qua video, bài kiểm tra và hoạt động tương tác. Nhờ khả năng tiếp cận rộng rãi và thân thiện với người dùng, Udemy đã góp phần quan trọng vào việc mở rộng cơ hội giáo dục, giúp mọi người ở nhiều hoàn cảnh khác nhau có thể học tập, phát triển đa lĩnh vực.

2. Mục tiêu nghiên cứu

- Phân tích thực trạng kinh doanh của nền tảng Udemy thông qua dữ liệu khóa học, học viên và giảng viên.
- Ứng dụng công cụ trực quan hóa Tableau để trình bày dữ liệu một cách sinh động, hỗ trợ ra quyết định hiệu quả.
- Ứng dụng mô hình học máy vào nghiên cứu

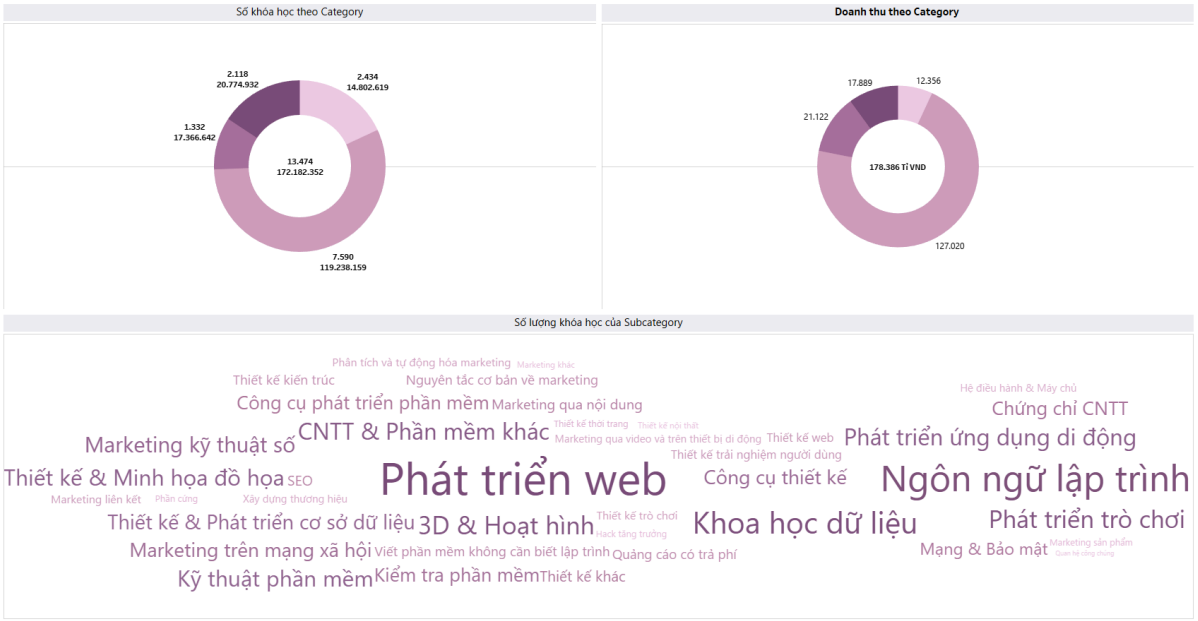
3. Phương pháp thực hiện

- Triển khai thu thập dữ liệu sử dụng Selenium và BeautifulSoup để lấy được dữ liệu từ website của Udemy
    - Thu thập liên kết các khóa học từ các danh mục trên Udemy và lưu vào file CSV.
        - Sử dụng thư viện Selenium để tự động hóa việc truy cập và lấy dữ liệu từ website của Udemy.
        - Khởi tạo trình duyệt chrome với tính năng tắt mở tự động, đồng thời kết hợp với user- agent để tránh bị chặn khi thực hiện quy trình thu thập.
        - Trích xuất liên kết khóa học bằng cách sử dụng các selector CSS tương ứng với các phần tử chứa URL khóa học, kết hợp xử lý nhiều khả năng thay đổi qua các trang Udemy.
        - Lưu trữ liên kết khóa học vào file CSV riêng biệt cho từng danh mục (category) sau mỗi lần quét. Đồng thời tích hợp cơ chế tránh cào trùng lặp và phục hồi sau lỗi để đảm bảo tính ổn định khi thu thập khối lượng lớn URL.
    - Truy cập vào các liên kết để lấy được dữ liệu từ Udemy.
        - Sử dụng thư viện Selenium và BeautifulSoup để phân tích cấu trúc HTML và trích xuất dữ liệu từ các liên kết khóa học Udemy.
        - Viết script điều khiển trình duyệt tự động để gửi yêu cầu đến từng URL khóa học, lấy toàn bộ mã nguồn HTML, và sử dụng BeautifulSoup để trích xuất các thông tin khóa học cần thiết.
        - Tích hợp cơ chế xử lý lỗi và tránh bị chặn từ phía Udemy bằng các kỹ thuật nâng cao như sử dụng user-agent ngẫu nhiên, điều khiển trình duyệt không bị phát hiện (undetected_chromedriver), và delay ngẫu nhiên giữa các lần truy cập.
        - Cài đặt cơ chế làm sạch trình duyệt sau mỗi lần truy cập bằng cách xóa cookie và localStorage nhằm tránh bị theo dõi hoặc khóa IP.

        - Sau khi hoàn tất việc cào dữ liệu từ danh sách URL, toàn bộ dữ liệu sẽ được hợp nhất trong một file duy nhất để phục vụ cho các bức tiền xử lý và trực quan hóa dữ liệu sau này.
- Tiền xử lý dữ liệu
    - Tổng hợp tất cả các file dữ liệu thành một và sử dụng python để đặt lại giá trị index cho toàn bộ khóa học, tránh việc trùng lặp.
    - Dùng các kỹ thuật bóc tách chuỗi (string) để tạo thêm các trường mới liên quan đến rating (đánh giá khóa học) và những thông tin về instructor (người cung cấp khóa học). Sau đó chuyển đổi kiểu dữ liệu thành số.
    - Sử lý cột giá bằng cách tách những ký tự không mong muốn như “đ”, “,” ra khỏi cột cũng như chuyển đổi kiểu dữ liệu.
    - Sử dụng các hàm của python để thực hiện tìm và loại bỏ các giá trị null ra khỏi dữ liệu, đảm bảo được độ chính xác của phân tích cũng như độ tin cậy từ kết quả của mô hình học máy.
- Trực quan hóa dữ liệu sử dụng tableau
    - Sử dụng Tableau để tạo biểu đồ, bảng tính và bảng điều khiển tương tác từ dữ liệu đã thu thập.
    - Phân tích các yếu tố quan trọng như doanh số bán hàng theo khóa học, phân khúc giá và danh mục sản phẩm.
    - Xây dựng bảng điều khiển giúp người dùng tùy chỉnh, lọc và khám phá dữ liệu theo nhu cầu phân tích.
    - Tận dụng các tính năng trực quan hóa nâng cao để trình bày dữ liệu một cách rõ ràng, sinh động và dễ hiểu.
- Machine learning
    - Sử dụng kết hợp các kỹ thuật tiền xử lý dữ liệu, trích xuất đặc trưng và thuật toán học máy để xây dựng hệ thống gợi ý khóa học thông minh.
    - Trích xuất đặc trưng từ dữ liệu văn bản bằng TF-IDF và kết hợp với các đặc trưng dạng số và phân loại thông qua kỹ thuật xử lý đặc trưng hỗn hợp.
    - Giảm chiều không gian dữ liệu đầu vào bằng phương pháp phân tích thành phần chính rút gọn (Truncated SVD), sau đó áp dụng thuật toán Nearest Neighbors để gợi ý các khóa học tương đồng.
    - Tạo prompt ngữ cảnh từ danh sách các khóa học gợi ý và gọi mô hình sinh ngôn ngữ Gemini (**Google Generative AI)** để phân tích nội dung và trình bày mô tả chi tiết lý do nên học.
    - Hiển thị kết quả qua giao diện người dùng sử dụng Streamlit, bao gồm danh sách khóa học và phần phân tích từ AI.

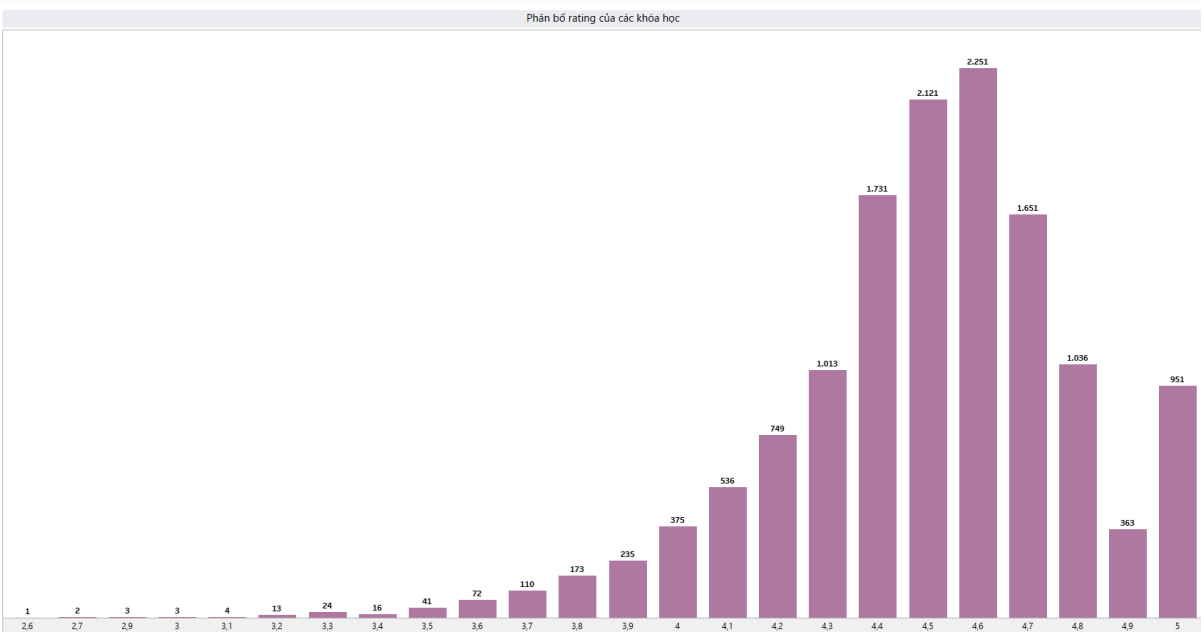
4. Phân tích tình hình kinh doanh của Udemy

a. Tổng quan



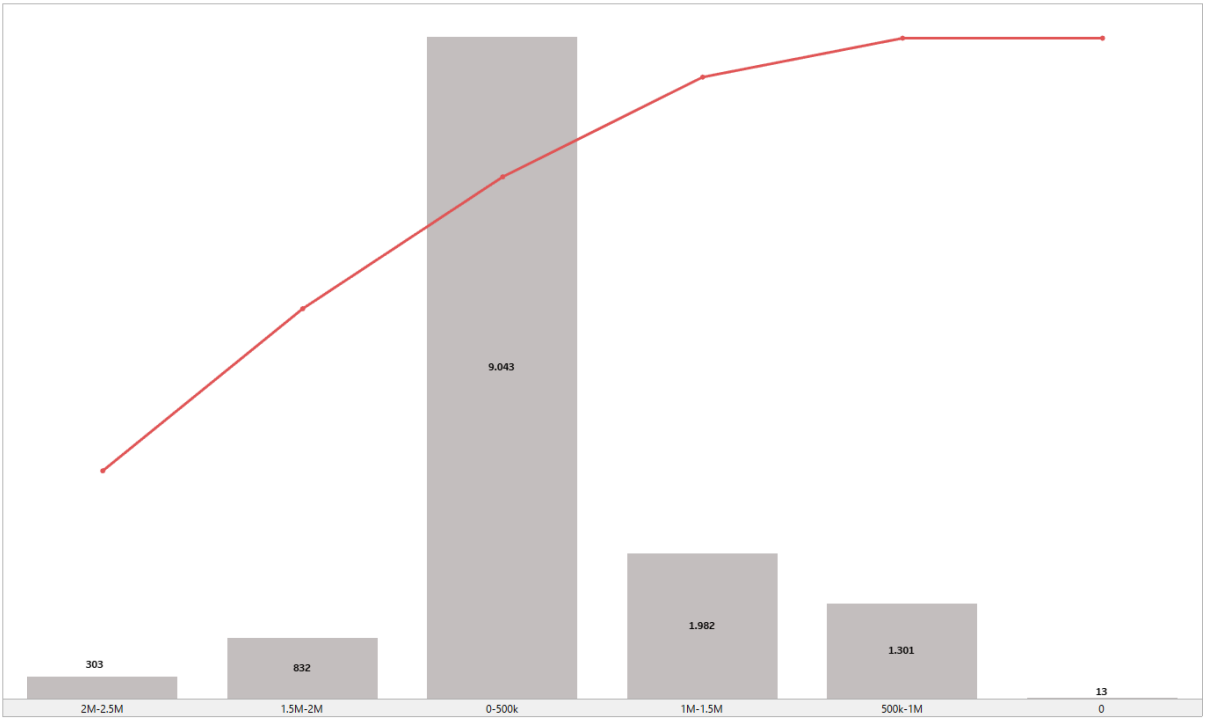
Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất
Tổng quan dữ liệu ta có được 13.520 số lượng khóa học được chia ra thành 4 danh mục khóa học chính bao gồm: Design, Development, IT & Sorfware và Marketing. Trong đó thì Development dẫn đầu về số lượng danh mục khóa học phụ (7.611 khóa học) cùng với 127.156 Tỉ VND doanh thu, chiếm 71,15% về doanh thu của các khóa học trên Udemy, kế sau đó là Design (2.436 khóa học, 12.357 Tỉ VND, chiếm 6,91% doanh thu) và Marketing (2.134 khóa học, tuy nhiên có lượng doanh thu cao hơn so với Design (18.060 Tỉ VND, chiếm 10,11% doanh thu),	Có thể do xu hướng học viên dần hướng tới các khóa học mang tính công nghệ và kĩ thuật máy tính nhiều hơn.	

cuối cùng là IT&Sorfware với 1.339 khóa học nhưng lại có doanh thu đạt mức 21.139 Tỉ VND, chiếm 11,83% tổng doanh thu gần như là gấp đôi so với Design. Trong các danh mục khóa học, các khóa học theo chủ đề về Phát triển Web, Khoa học dữ liệu và Ngôn ngữ lập trình đang dẫn đầu về số lượng khóa học cũng như là số người đăng ký với các con số rất ấn tượng		

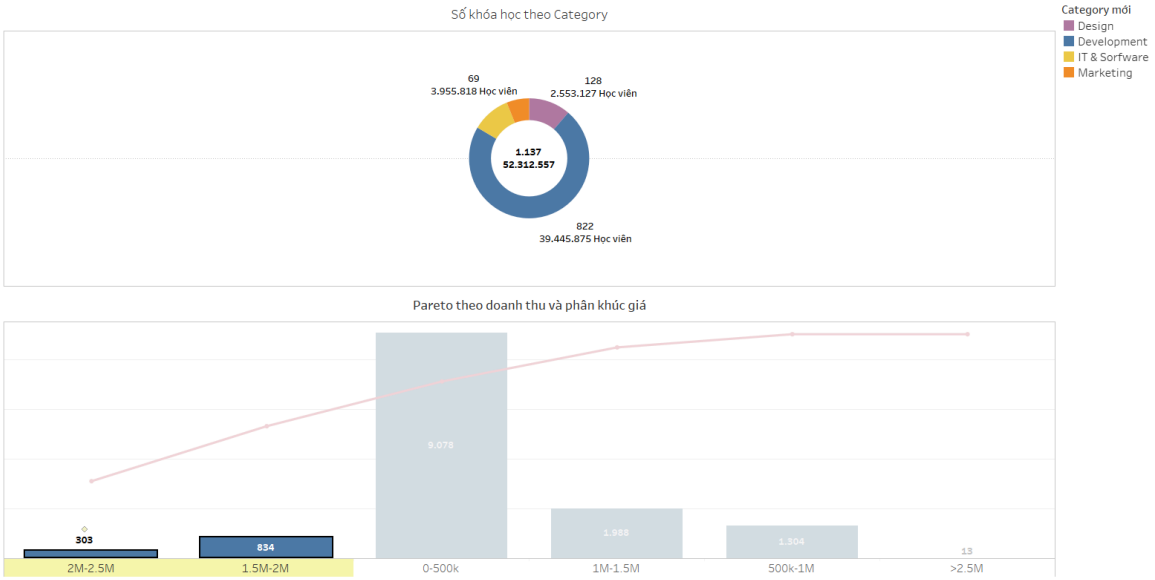


Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất
Nhìn chung các Rating của khóa học đa số đều nằm trên mức đánh giá tốt, đặc biệt số lượng đánh giá 4.5 và 4.6 sao cực kì	Giá trị các khóa học của Udemy cao, phù hợp với chi phí mà học viên bỏ ra. Một số các học viên ít có xu hướng đánh giá tiêu	

cao	cực khi thấy khóa học có nhiều đánh giá tích cực.	

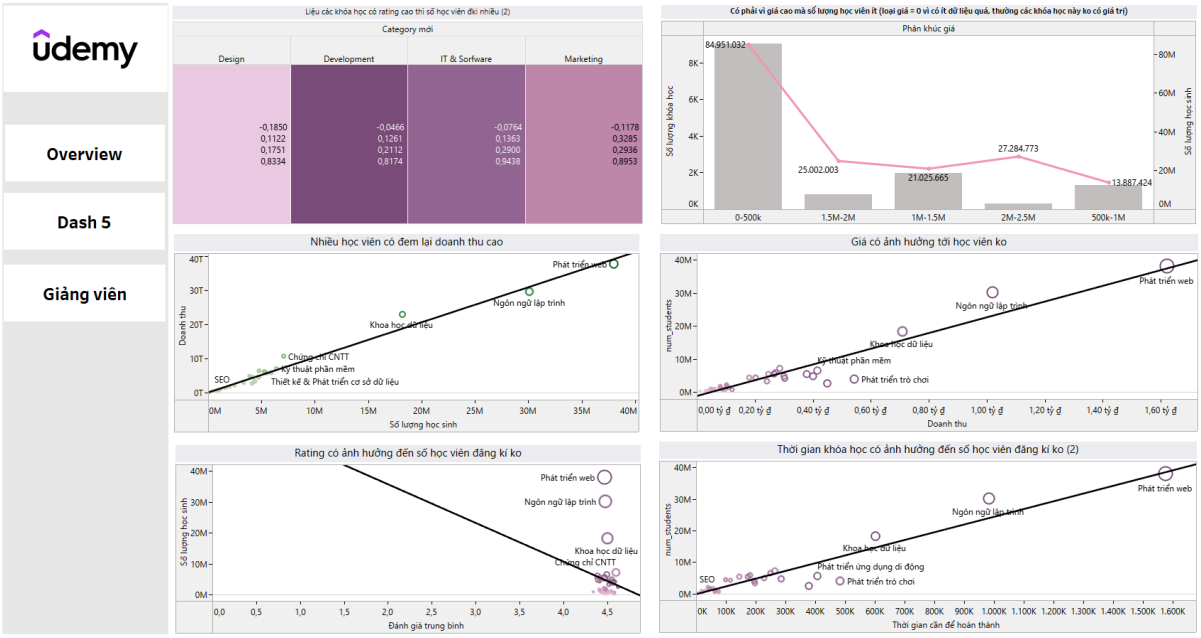


Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất
Phân khúc giá của Udemy tập trung nhiều trong phân khúc 0-500k, tuy nhiên 59% doanh thu đến từ các khóa học có số lượng khóa học ít nhưng giá trị cao, các phân khúc còn lại đóng góp khoảng 22% doanh thu.	Udemy sử dụng các chiến lược mức giá thấp để thu hút lượng lớn học viên. Phân khúc giá 0-500k dễ tiếp cận, chi phí thấp nên khách hàng ít cân nhắc khi bỏ ra để mua một khóa học online, điều mà các khóa học có phân khúc giá cao khó làm được	

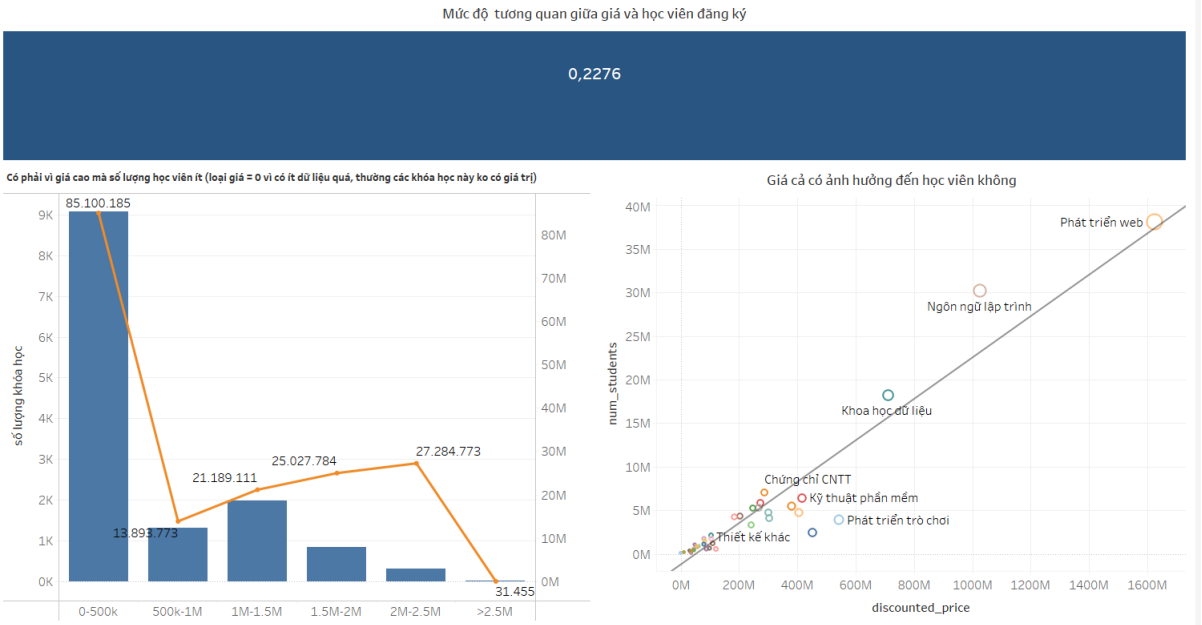


Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất
	2 Phân khúc giá 1.5M-2M và 2M-2.5M chiếm 59% doanh thu bởi vì nó có số lượng học viên đăng kí nhiều cùng với giá tiền cao nên nó đem lại doanh thu khổng lồ.	

b. Ảnh hưởng của các yếu tố lên số lượng học viên đăng ký



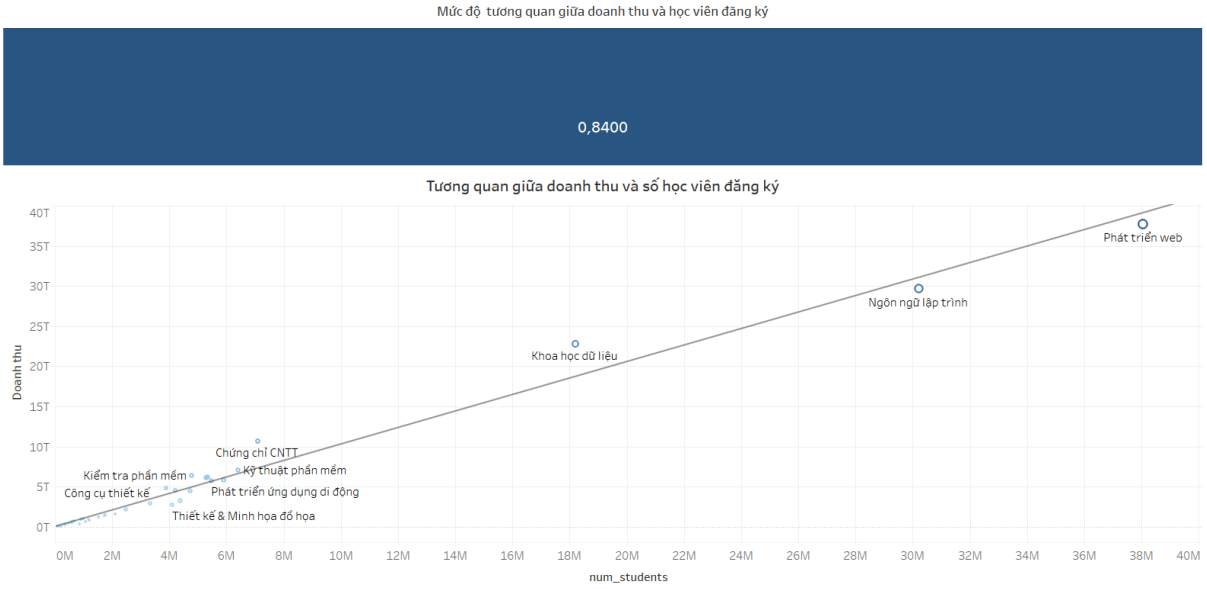
1. Ảnh hưởng của giá tác động lên học viên



Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất
Ta thấy hệ số tương quan giữa giá khóa học và số lượng học viên là 0.2276, tương quan thuận, giá	Có các biến ảnh hưởng nhiều hơn đến số lượng học viên đăng ký khóa học	

khóa học tăng thì học viên đăng ký khóa học sẽ tăng tuy nhiên nó chỉ tác động nhỏ lên số lượng học viên. Dựa vào phân phối của 2 đồ thị ở dưới thấy được phân khúc giá không cho thấy được sự tăng lên của số lượng học viên, chỉ thấy được sự tăng trưởng ở phân khúc 500k cho đến 2.5M, tuy nhiên con số này không quá lớn.		

2. Tác động tương quan giữa doanh thu và số học viên đăng ký

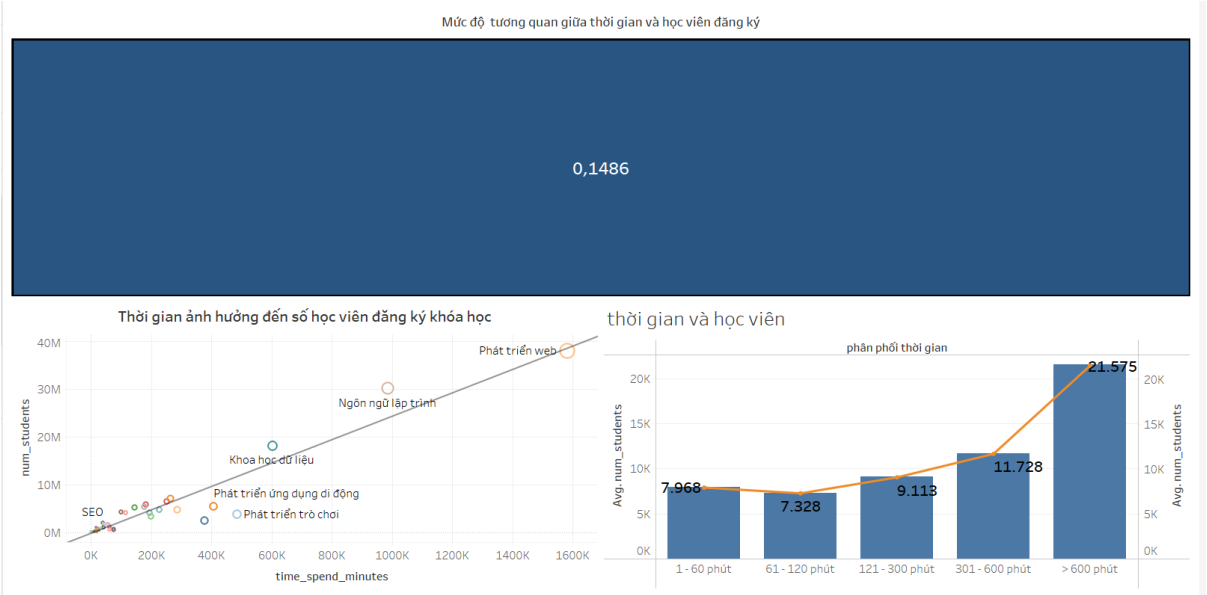


Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất
Ta thấy hệ số tương quan giữa doanh thu và số lượng học viên đăng ký là 0.84, tương quan thuận tuyến tính mạnh, hai biến này tác động qua lại lẫn nhau,70% doanh thu được		

giải thích bởi số lượng học viên đăng ký.		

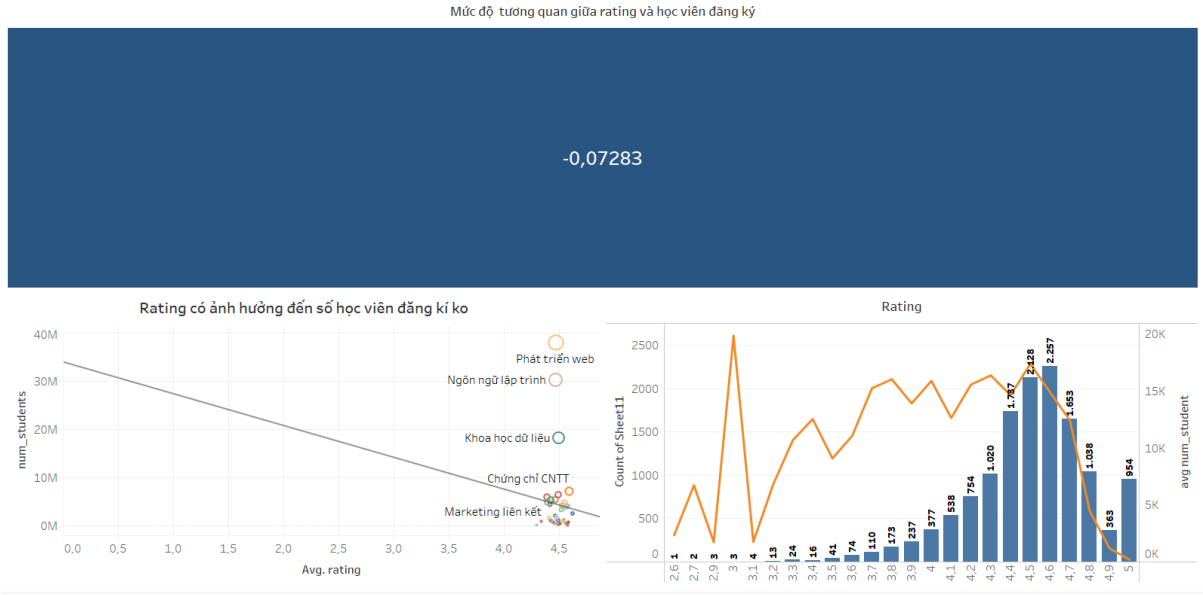
Ta đã phân tích về 2 yếu tố giá cả và doanh thu, bây giờ ta sẽ thực hiện phân tích đến 2 yếu tố còn lại khi đánh giá 1 khóa học là thời gian học và rating của khóa học

3. 2 yếu tố thời gian và rating ảnh hưởng gì đến khóa học



Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất
Biểu đồ trên phân tích mối quan hệ giữa thời lượng khóa học và số lượng học viên đăng ký trên Udemy. Hệ số tương quan giữa hai biến là 0.1486, cho thấy đây là mối tương quan thuận nhưng rất yếu – tức thời gian khóa học có ảnh hưởng đến số học viên đăng ký, nhưng không phải là yếu tố quyết định chính. Tuy nhiên, khi phân nhóm theo khoảng thời lượng, ta	Có thể do tâm lý của khách hàng, khách hàng thường so sánh chi phí mình bỏ ra so với giá trị mà mình nhận được, các khóa học cùng mức giá nhưng có thời gian học nhiều hơn khiến họ thấy nhận được nhiều hơn với cùng một mức giá. Các khóa học có thời gian ngắn thường không đem lại lượng kiến thức đầy đủ, chi mang tính tóm tắt hay nhập môn	Đề xuất cho học viên các khóa học có thời lượng từ 5 giờ trở lên.

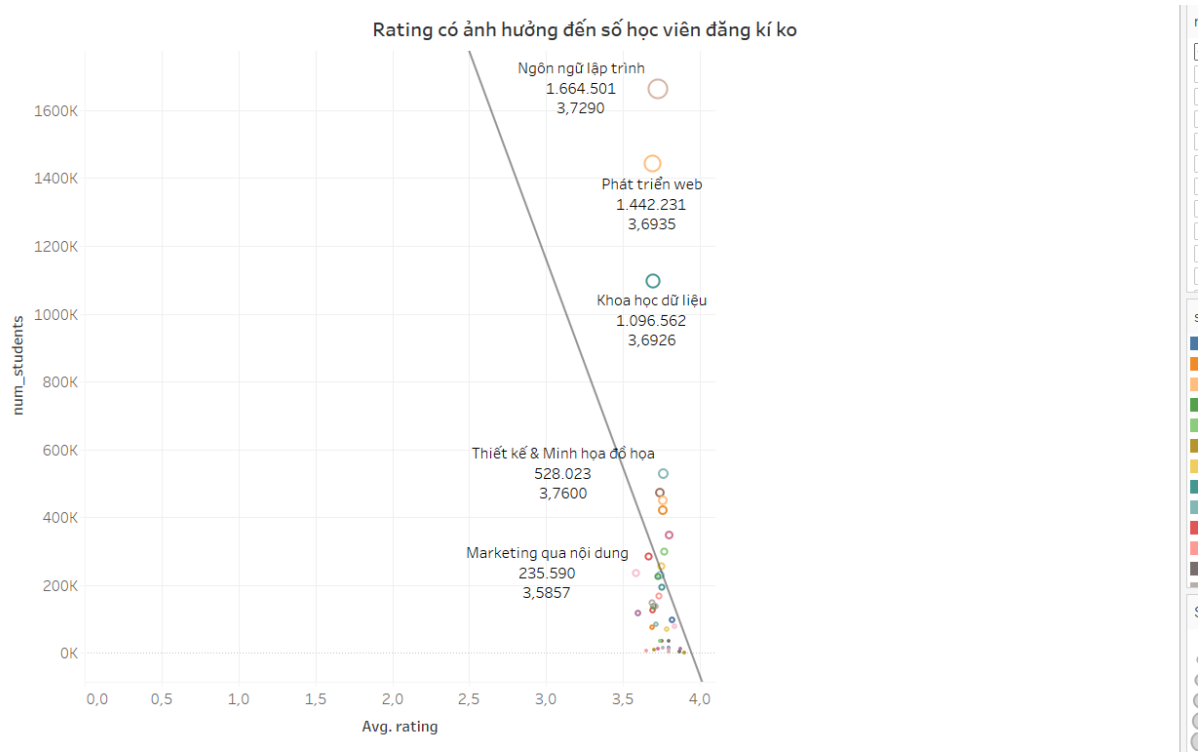
thấy xu hướng rõ rệt hơn. Các khóa học có thời gian trên 600 phút (tức hơn 10 giờ) có số lượng học viên trung bình lên tới 21.575, cao gấp 3 lần so với nhóm khóa học ngắn (1–60 phút, chỉ 7.968 học viên trung bình). Từ mức 300 phút trở lên, số học viên trung bình bắt đầu tăng rõ rệt, cho thấy người học có xu hướng đánh giá cao những khóa học có thời lượng đủ dài để đào sâu kiến thức.		



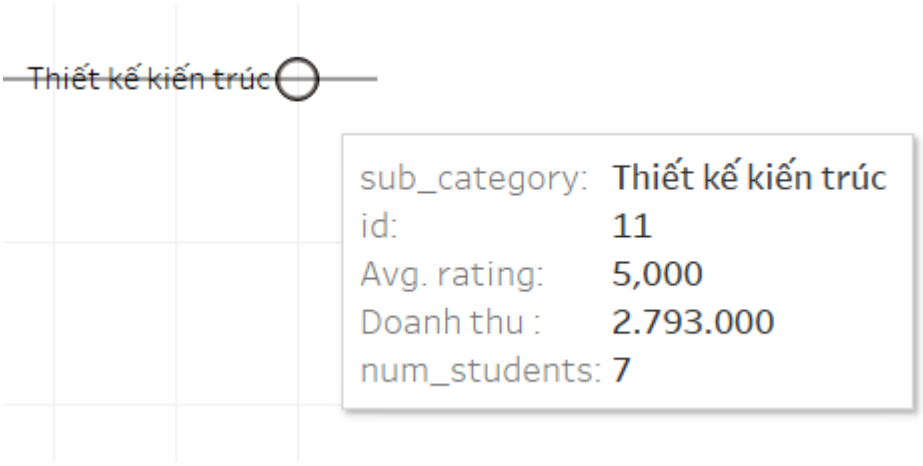
Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất
Biểu đồ thể hiện mối quan hệ giữa điểm đánh giá trung bình (rating) của khóa học và số lượng học viên đăng ký là -0.07 cho thấy mối tương quan yếu và nghịch chiều với số	Có thể do các khóa học có số lượng học viên lớn cùng với lượng rating cao nhưng chất lượng lại không phù hợp	

lượng học viên. Bằng chứng là ở biểu đồ phân phối ta có thể thấy khi rating tăng thì học viên đăng kí khóa học không tăng theo tỷ lệ thậm chi có xu hướng giảm mạnh từ mức rating 4.8-5		

Minh chứng:

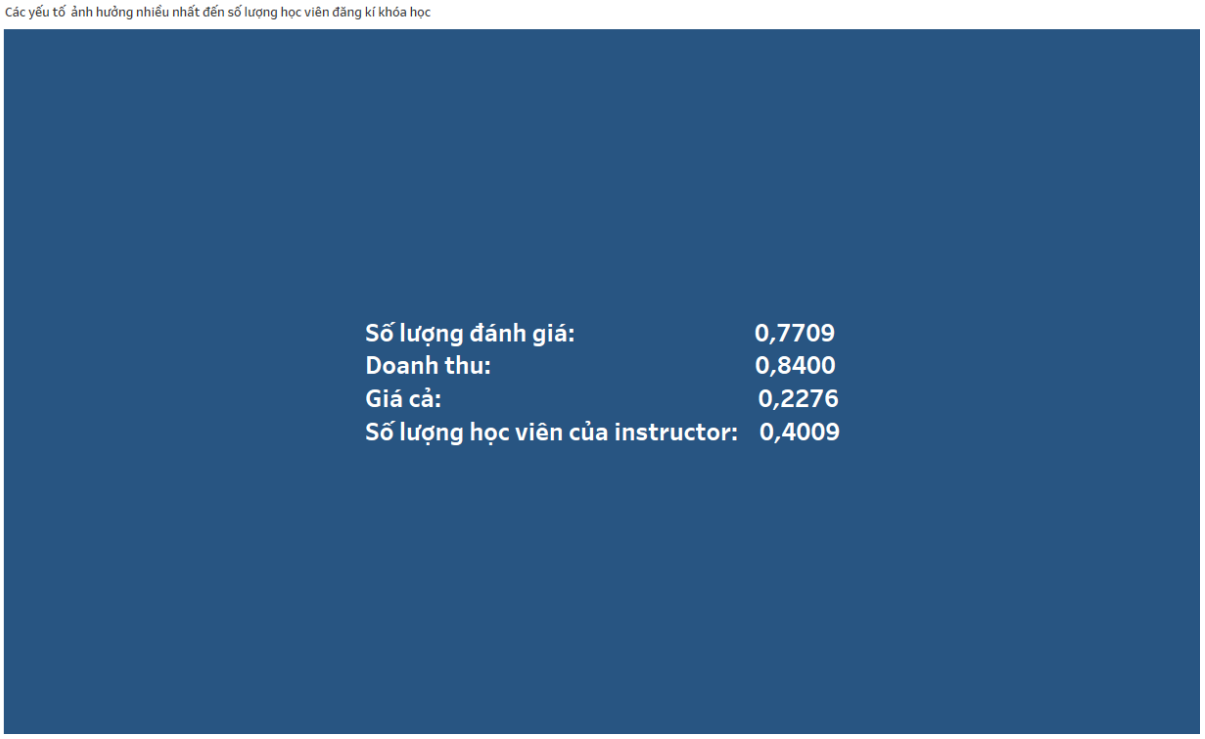


Các khóa học có số lượng học viên lớn nhưng không hài lòng với chất lượng của khóa học.



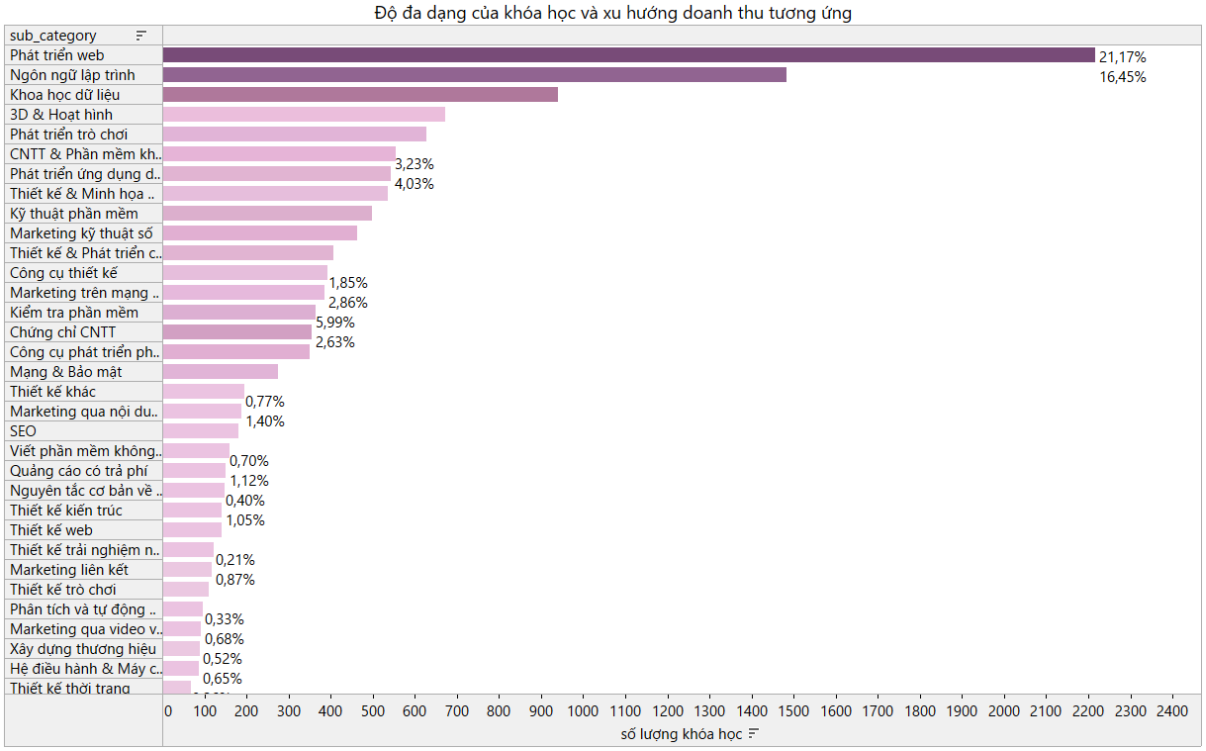
Một số nhóm học viên với các ngành học có số lượng học viên ít có kỳ vọng đồng nhất và dễ dàng

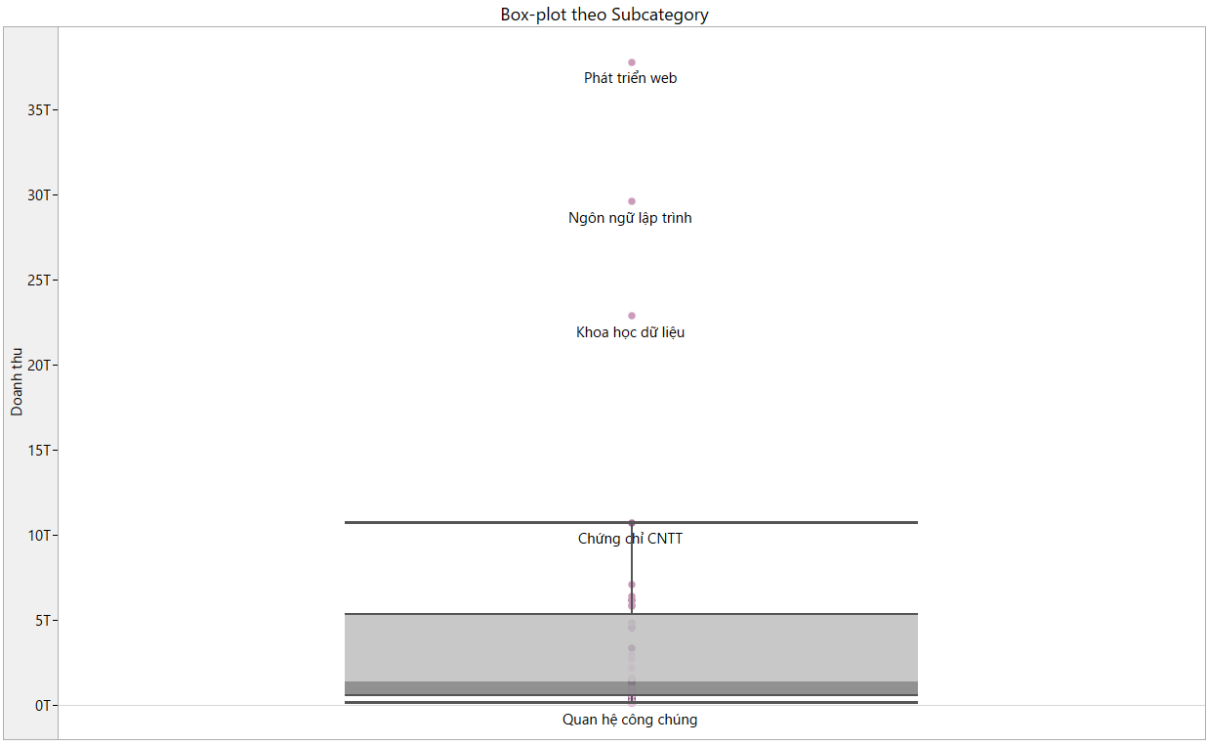
4. Vậy các yếu tố nào sẽ là yếu tố ảnh hưởng nhiều nhất đến số lượng học viên đăng ký khóa học



Dựa vào các chỉ số tương quan đã được tính thì các yếu tố như số lượng đánh giá (num_review), Doanh thu, Giá cả, Số lượng học viên của Instructor là các yếu tố có mức ảnh hưởng tương quan khá mạnh đến số lượng học viên đăng ký khóa học. Vì vậy ta sẽ dùng các yếu tố này để thực hiện đánh giá xem khóa học thuộc danh mục nào được cho là chất lượng và tốt với học viên nhất.

5. Độ đa dạng của khóa học và xu hướng doanh thu tương ứng



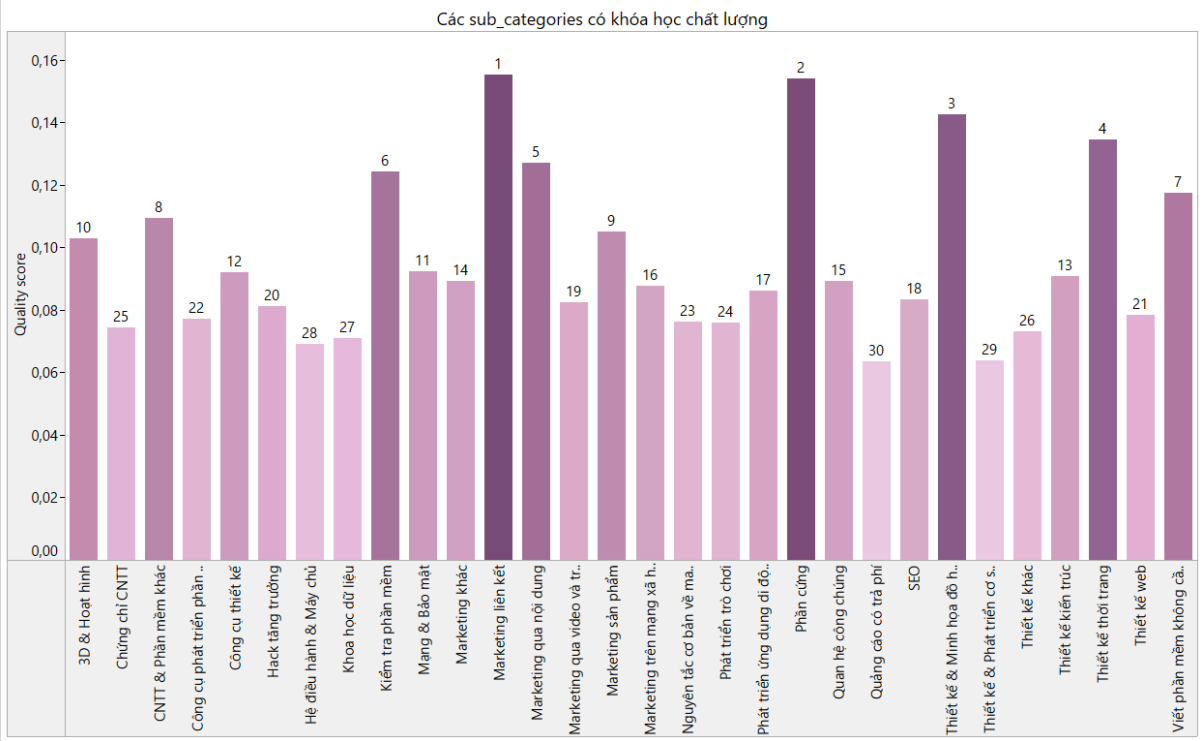


Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất

Các khóa học liên quan đến Development dẫn đầu với các chỉ số ấn tượng: “Phát triển web” dẫn đầu với 21,17% doanh thu của toàn bộ các khóa học cùng với 16,45% số lượng khóa học. "Ngôn ngữ lập trình" chiếm 16.45%, đứng thứ hai về số lượng khóa học, phản ánh sự quan tâm đáng kể đến kỹ năng lập trình. “Khoa học dữ liệu” là khóa học đứng thứ 3 với 12,81% doanh thu và 6,98% số lượng khóa học	Phần lớn do các công ty đang dần phụ thuộc vào công nghệ. nên 2 khóa học này đáp ứng được nhu cầu đó Các khóa học thuộc ngành học Development có thể là xu thế trong tương lai với doanh thu rất lớn kèm với số lượng khóa học khổng lồ.	
"3D & Hoạt hình" chiếm 4.03%, cho thấy mức độ quan tâm đáng kể nhưng thấp hơn so với hai lĩnh vực trên.		
Các lĩnh vực như "Marketing kỹ thuật số" (1.40%), "Thiết kế kiến trúc" (1.05%) và "SEO" (0.77%) chiếm tỷ lệ nhỏ, cho thấy chúng chưa được đầu tư mạnh và thu hút được sự chú ý của các học viên đăng ký.		

Các lĩnh vực như "Xây dựng bản phân cảnh" (0.52%), "Quản lý năng lượng" (0.52%) và "Thiết kế trải nghiệm người dùng" (0.33%) có số lượng khóa học rất ít, thể hiện sự phân bổ hạn chế.		

6. Thực hiện đánh giá khóa học nào là tốt nhất dựa trên Quality_Score



Diễn giải	Nguyên nhân	Đánh giá/ Đề xuất
Marketing liên kết là danh mục khóa học phụ đứng đầu trong các khóa học có chất lượng cao nhất được đánh giá, tiếp theo là	Các khóa học này có điểm chất lượng trung bình cao vượt trội, mặt khác nó được đánh giá bằng các yếu tố ảnh hưởng nhiều	Ưu tiên quảng bá các khóa học được xếp hạng cao, sử dụng các tag cho các khóa học này nhiều hơn giúp học viên dễ dàng nhận

“phần cứng”,”Thiết kế & Minh họa đồ họa” ,”Thiết kế web”,Marketing qua nội dung”	nhất đến số học viên đăng ký khóa học, các khóa học này đứng đầu chứng tỏ các khóa học đó đều được đánh giá cao, chất lượng	biết và chú ý.
Các khóa học thuộc nhóm Marketing chiếm ưu thế mặc dù cho doanh thu và số lượng khóa học trong Development lại vượt trội hơn		
Các lĩnh vực như "Xây dựng bản phân cảnh" (0.52%), "Quản lý năng lượng" (0.52%) và "Thiết kế trải nghiệm người dùng" (0.33%) có số lượng khóa học rất ít, thể hiện sự phân bổ hạn chế.		

3. Machine learning

a. Lý thuyết của mô hình

Trong đánh giá hệ thống gợi ý dựa trên nội dung (content-based recommender), metrics mà nhóm tác giả lựa chọn để đánh giá hiệu quả là: **Cosine Similarity.**

Là một trong những phép đo phổ biến nhất trong việc đo lường mức độ tương đồng giữa hai vector trong không gian vector cao chiều (high-dimensional space). Trong ngữ cảnh này:

- Mỗi khóa học được biểu diễn bằng một vector embedding.

- Cosine similarity giữa hai vector AAA và BBB được tính bằng công thức:

$$\text{Cosine Similarity}(A, B) = \frac{A \cdot B}{\|A\|\|B\|}$$

Giá trị nằm trong khoảng [0,1] trong đó 1 nghĩa là hai vector hoàn toàn giống nhau về hướng (tương đồng hoàn toàn). Trong pipeline, trung bình của cosine similarity giữa một khóa học và 5 khóa học được gợi ý được dùng để đánh giá hiệu năng của từng mô hình.

b. Các mô hình đề xuất

- Nearest Neighbors (NN) trên Vector Đặc Trưng
    - Mô hình Nearest Neighbors (k-NN) là một thuật toán không tham số thường được sử dụng trong hệ thống gợi ý. Với bài toán này, mỗi khóa học được biểu diễn bằng một vector đặc trưng (embedding vector), và thuật toán tìm ra kkk khóa học gần nhất (tương tự nhất) dựa trên khoảng cách cosine. Kết hợp sử dụng thuật toán brute-force, tức là tính khoảng cách cosine từ mỗi vector truy vấn đến toàn bộ các vector khác.
- Truncated SVD + Nearest Neighbors
    - Truncated SVD (Singular Value Decomposition) là một kỹ thuật giảm chiều được sử dụng phổ biến trong xử lý dữ liệu thưa (sparse data) và giảm nhiễu.
    - Sau khi giảm chiều, mô hình k-NN được áp dụng trong không gian mới (low-dimensional space), giảm thiểu chi phí tính toán và khử nhiễu.
- K-Means Clustering
    - KMeans là thuật toán phân cụm chia tập dữ liệu thành kkk cụm bằng cách tối thiểu hóa khoảng cách Euclidean từ mỗi điểm đến tâm cụm gần nhất. Trong bài toán gợi ý, ý tưởng là:
        - Gom các khóa học tương tự vào cùng một cụm.
        - Đối với mỗi truy vấn, tìm các khóa học cùng cụm để gợi ý.
        - Mặc dù cosine similarity được dùng ở bước embedding, KMeans vẫn có thể áp dụng trên không gian đã giảm chiều để tăng hiệu quả.

c. Quy trình thực hiện

- Import các thư viện cần thiết vào google colab:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.impute import SimpleImputer
from sklearn.neighbors import NearestNeighbors
from sklearn.decomposition import TruncatedSVD
from sklearn.cluster import KMeans
from sklearn.metrics.pairwise import cosine_similarity
from scipy.sparse import hstack
import numpy as np
```

- Sử dụng thư viện pandas để đọc bộ dữ liệu đã được tiền xử lý:

```
df = pd.read_excel("/content/drive/MyDrive/Capstone/Data_finalver5.xlsx")
df.head()
```

_5_star	publish_date	...	num_reviews	num_students	instructor_name	instructor_rating	instructor_num_reviews	instructor_num_students	instructor_num_courses	estimated_revenue	course_length_category	calculated_avg_rating
1.0	2023-01-01	...	7	457	Diemante SulciuteHospitality Expert, Digital D...	4.5	231.0	4758.0	13.0	182343000	Medium (1-5h)	5.0
1.0	2023-01-01	...	2	1015	Sydney HeightInstructor	5.0	3.0	1021.0	2.0	404985000	Medium (1-5h)	5.0
1.0	2023-12-01	...	2	15	Being CommerceBiggest Commerce Community	4.3	1187.0	3731.0	83.0	13935000	Medium (1-5h)	5.0
1.0	2022-12-01	...	2	1185	Dr. Noble Arya Full-Stack Data Scientist, AI R...	4.4	1070.0	21242.0	138.0	472815000	Medium (1-5h)	5.0
1.0	2023-01-01	...	2	29	Devasis SamantryCreativity in Every Frame	4.1	6.0	73.0	4.0	11571000	Medium (1-5h)	5.0

- Xây dựng đặc trưng

```
[25] text_data = df['text']
     numeric_features = ['discounted_price', 'rating', 'num_reviews', 'num_students', 'time_spend_minutes']
     categorical_features = ['category', 'sub_category', 'language']

[26] tfidf = TfidfVectorizer(max_features=1000)
     text_features = tfidf.fit_transform(text_data.fillna(""))

[27] numeric_transformer = Pipeline([
         ('imputer', SimpleImputer(strategy='median')),
         ('scaler', StandardScaler())
     ])
     categorical_transformer = Pipeline([
         ('imputer', SimpleImputer(strategy='most_frequent')),
         ('onehot', OneHotEncoder(handle_unknown='ignore'))
     ])
     preprocessor = ColumnTransformer([
         ('num', numeric_transformer, numeric_features),
         ('cat', categorical_transformer, categorical_features)
     ])
     structured_features = preprocessor.fit_transform(df)
```

Trong giai đoạn này, hệ thống tiến hành xử lý và chuyển đổi các đặc trưng đầu vào của khóa học để chuẩn bị cho mô hình gợi ý dựa trên nội dung (content-based). Dữ liệu được chia thành ba nhóm chính: đặc trưng văn bản, đặc trưng định lượng, và đặc trưng phân loại. Mỗi nhóm sẽ được xử lý thông qua một pipeline riêng biệt nhằm đảm bảo đầu ra phù hợp với yêu cầu của mô hình học máy.

Đối với đặc trưng văn bản, cụ thể là cột mô tả khóa học, phương pháp TfidfVectorizer được áp dụng để trích xuất các đặc trưng. Kỹ thuật TF-IDF giúp xác định những từ khóa nổi bật bằng cách gán trọng số cao cho các từ xuất hiện thường xuyên trong một tài liệu nhưng lại hiếm gặp ở những tài liệu khác. Để giảm nhiễu và rút gọn không gian đặc trưng, chỉ giữ lại 1.000 từ phổ biến nhất. Kết quả là một ma trận thưa đại diện cho nội dung khóa học, dùng để tính toán mức độ tương đồng giữa các học phần.

Với dữ liệu định lượng như discounted_price, rating, num_reviews, num_students và time_spend_minutes, pipeline xử lý số được sử dụng. Trước tiên, các giá trị thiếu được thay thế bằng trung vị thông qua SimpleImputer. Sau đó, toàn bộ dữ liệu được chuẩn hóa bằng StandardScaler để đưa về cùng một thang đo, giúp mô hình học hiệu quả hơn và tránh hiện tượng một số đặc trưng chi phối kết quả do giá trị lớn.

Đối với các đặc trưng phân loại như category, sub_category và language, pipeline phân loại sẽ đảm nhiệm. Các giá trị thiếu được xử lý bằng cách thay thế bằng giá trị phổ biến nhất (most_frequent). Sau đó, dữ liệu được mã hóa dưới dạng one-hot thông qua OneHotEncoder, chuyển các biến phân loại thành dạng nhị phân – định dạng thích hợp cho mô hình tuyến tính và các phép đo khoảng cách.

Cuối cùng, tất cả pipeline này được kết hợp thông qua ColumnTransformer để hợp nhất các đặc trưng định lượng và phân loại vào một ma trận đầu vào có cấu trúc. Kết quả là một tập hợp các vector biểu diễn cho từng khóa học, có thể dùng riêng biệt hoặc kết hợp với đặc trưng văn bản trong bước vector hóa tổng hợp và xây dựng mô hình gợi ý.

- Training 3 mô hình đã được đề xuất

```
[ ]  course_embeddings = hstack([text_features, structured_features])
     course_ids = df['id'].values
     course_titles = df['title'].values

[ ]  nn_full = NearestNeighbors(metric='cosine', algorithm='brute', n_neighbors=6)
     nn_full.fit(course_embeddings)

     svd = TruncatedSVD(n_components=100, random_state=42)
     course_embeddings_svd = svd.fit_transform(course_embeddings)
     nn_svd = NearestNeighbors(metric='cosine', algorithm='brute', n_neighbors=6)
     nn_svd.fit(course_embeddings_svd)

     kmeans = KMeans(n_clusters=50, random_state=42)
     kmeans_labels = kmeans.fit_predict(course_embeddings_svd)
```

Sau khi hoàn tất bước xử lý, các đặc trưng văn bản và có cấu trúc được nối ngang để tạo thành vector biểu diễn đầy đủ cho mỗi khóa học – gọi là course_embeddings. Các trường id và title cũng được lưu lại để phục vụ việc hiển thị gợi ý.

Tiếp đến, mô hình Nearest Neighbors được huấn luyện trên toàn bộ không gian embedding, sử dụng khoảng cách cosine và chọn 6 khóa học gần nhất. Thuật toán brute được áp dụng để đảm bảo tính chính xác tuyệt đối trong việc tính toán.

Để giảm nhiễu và rút gọn chiều, TruncatedSVD được dùng để giảm vector về 100 thành phần chính – phương pháp này phù hợp với dữ liệu thưa từ TF-IDF. Sau đó, mô hình NN được huấn luyện lại trên không gian giảm chiều (course_embeddings_svd) nhằm đánh giá hiệu quả giảm chiều.

Cuối cùng, thuật toán K- Means với 50 cụm được áp dụng trên không gian mới. Các nhãn cụm (kmeans_labels) giúp nhóm các khóa học theo nội dung hoặc ý nghĩa tương đồng.

- Lựa chọn mô hình tốt nhất

```python
# Step 5: Evaluate Models
np.random.seed(42)
sample_indices = np.random.choice(len(df), size=30, replace=False)

eval_results = []

for model_name, model, data in [
    ("Original NN", nn_full, course_embeddings),
    ("SVD NN", nn_svd, course_embeddings_svd),
    ("KMeans", None, kmeans_labels)
]:
    avg_sim = 0
    for idx in sample_indices:
        if model_name == "KMeans":
            cluster_id = data[idx]
            cluster_indices = np.where(data == cluster_id)[0]
            cluster_indices = [i for i in cluster_indices if i != idx][:5]
            vectors = course_embeddings_svd[cluster_indices]
            query_vector = course_embeddings_svd[idx].reshape(1, -1)
        else:
            query_vector = data[idx]
            if hasattr(query_vector, 'reshape'):
                query_vector = query_vector.reshape(1, -1)
            elif hasattr(query_vector, 'toarray'):
                query_vector = query_vector.toarray()
            distances, indices = model.kneighbors(query_vector, n_neighbors=6)
            neighbors = indices.flatten()[1:]
            vectors = data[neighbors]
        sim_matrix = cosine_similarity(query_vector, vectors)
        avg_sim += sim_matrix.mean()
    avg_sim /= len(sample_indices)
    eval_results.append({"Model": model_name, "Average Cosine Similarity": avg_sim})

metrics_df = pd.DataFrame(eval_results)
print(metrics_df)
```

```
          Model  Average Cosine Similarity
0  Original NN                   0.854996
1       SVD NN                   0.953778
2       KMeans                   0.711161
```

Kết quả đánh giá mở rộng với 30 mẫu cho thấy mô hình **SVD + Nearest Neighbors** đạt hiệu suất tốt nhất với độ tương đồng cosine trung bình là 0.9538, cao hơn rõ rệt so với mô hình **Original NN** (0.8550) và **KMeans** (0.7112). Điều này cho thấy việc giảm chiều dữ liệu bằng TruncatedSVD giúp tăng cường khả năng biểu diễn và loại bỏ nhiễu trong không gian vector hóa. Trong khi đó, KMeans cho kết quả thấp hơn do không tối ưu theo tiêu chí tương đồng cá thể, mà chỉ phân cụm tổng thể. Từ kết quả này, có thể kết luận rằng SVD + NN là lựa chọn phù hợp nhất để triển khai hệ thống gợi ý trong bài toán này.

d. Xây dựng trang web đề xuất, sử dụng LLM để đưa ra gợi ý cho người dùng.

- Import những thư viện cần thiết

```
import streamlit as st
st.set_page_config(page_title="Gợi ý khóa học thông minh", layout="wide")

import pandas as pd
import numpy as np
from sklearn.neighbors import NearestNeighbors
from sklearn.decomposition import TruncatedSVD
from sklearn.metrics.pairwise import cosine_similarity
from scipy.sparse import hstack
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.impute import SimpleImputer
from sklearn.compose import ColumnTransformer
import google.generativeai as genai
```

- Khởi tạo Gemini thông qua API và load dataset

```
# --- CONFIGURE GEMINI ---
genai.configure(api_key="AIzaSyDO9lUAeHNqPQZw1Oia43wtG-mtQmIt-BA")

# --- LOAD NEW DATASET ---
df = pd.read_excel("Data_finalver5.xlsx")
df.dropna(subset=['title', 'subtitle'], inplace=True)
df['text'] = df['title'] + " " + df['subtitle']
```

- Xây dựng pipeline để thực hiện xây dựng, biến đổi đặc trưng tương tự như lúc tạo mô hình. Sau đó xây dựng hàm đề xuất sử dụng mô hình **SVD + NN** là mô hình tốt nhất sau khi so sánh.

```python
# --- FEATURES ---
numeric_features = ['discounted_price', 'rating', 'num_reviews', 'num_students', 'time_spend_minutes']
categorical_features = ['category', 'sub_category', 'language']

text_features = TfidfVectorizer(max_features=1000).fit_transform(df['text'].fillna(""))

numeric_transformer = Pipeline([
    ('imputer', SimpleImputer(strategy='median')),
    ('scaler', StandardScaler())
])
categorical_transformer = Pipeline([
    ('imputer', SimpleImputer(strategy='most_frequent')),
    ('onehot', OneHotEncoder(handle_unknown='ignore'))
])
preprocessor = ColumnTransformer([
    ('num', numeric_transformer, numeric_features),
    ('cat', categorical_transformer, categorical_features)
])
structured_features = preprocessor.fit_transform(df)
course_embeddings = hstack([text_features, structured_features])

# --- MODEL ---
svd = TruncatedSVD(n_components=150, random_state=42)
course_embeddings_refined = svd.fit_transform(course_embeddings)
nn_refined = NearestNeighbors(metric='cosine', algorithm='brute', n_neighbors=6)
nn_refined.fit(course_embeddings_refined)

# --- FUNCTIONS ---
def recommend_similar_courses(course_index, top_n=5):
    query_vector = course_embeddings_refined[course_index].reshape(1, -1)
    distances, indices = nn_refined.kneighbors(query_vector, n_neighbors=top_n + 1)
    recommended_indices = indices.flatten()[1:]
    return df.iloc[recommended_indices][['title', 'subtitle', 'rating', 'num_students', 'time_spend_minutes', 'instructor_name', 'url']].reset_index(drop=True)
```

- Xây dựng context cho câu trả lời của Gemini

```python
def build_context(course_df):
    context = ""
    for _, row in course_df.iterrows():
        context += (
            f"- {row['title']}: {row['subtitle']}\n"
            f"  • Đánh giá: {row['rating']}/5\n"
            f"  • Giảng viên: {row['instructor_name']}\n"
            f"  • Học viên: {int(row['num_students']):,}\n"
            f"  • Thời lượng: {int(row['time_spend_minutes'])} phút\n"
            f"  • Link: {row['url']}\n\n"
        )
    return context.strip()
```

- Tạo hàm trả về kết quả gợi ý tại sao nên thử những khóa học này, sau khi người dùng đã đưa vào mô hình khóa học hay lĩnh vực mà họ hứng thú.

```python
def answer(context):
    prompt = f"""
    Dưới đây là danh sách 5 khóa học mà người dùng đang quan tâm:

    --- THÔNG TIN KHÓA HỌC ---
    {context}
    --- HẾT ---

    Với vai trò là cố vấn giáo dục, bạn hãy viết mô tả chi tiết cho từng khóa học (mỗi khóa học nhiều nhất 5–7 câu), trình bày:

    - Vì sao khóa học này đáng học (lý do cụ thể, nội dung hấp dẫn, ứng dụng thực tiễn)
    - Ai là người phù hợp (mức độ kiến thức, đối tượng học viên lý tưởng)
    - Những kỹ năng, lợi ích hoặc kết quả người học có thể đạt được sau khi học

    Hãy viết bằng tiếng Việt, giọng văn truyền cảm hứng, rõ ràng, dễ hiểu và tránh lặp lại nội dung.
    """
    model = genai.GenerativeModel("models/gemini-2.5-flash-preview-04-17")
    try:
        response = model.generate_content(
            prompt,
            generation_config={"max_output_tokens": 10000, "temperature": 0.7, "top_p": 1.0}
        )
        if response.candidates and response.candidates[0].content.parts:
            return response.text.strip()
        else:
            return "⚠️ AI không phản hồi được cho danh sách khóa học này."
    except Exception as e:
        return f"❌ Lỗi khi gọi AI: {str(e)}"
```

- Kế tiếp để giúp người dùng dễ dàng tương tác với hệ thống gợi ý khóa học, nhóm đã triển khai một giao diện người dùng đơn giản và trực quan bằng thư viện **Streamlit** – một công cụ mã nguồn mở hỗ trợ nhanh chóng tạo dashboard và ứng dụng web cho các bài toán học máy.
    - Giao diện được thiết kế để thực hiện các chức năng chính sau:
    - **Chọn một khóa học làm truy vấn đầu vào**: Người dùng có thể tìm kiếm và chọn một khóa học bất kỳ từ danh sách để hệ thống sử dụng

làm điểm bắt đầu gợi ý.

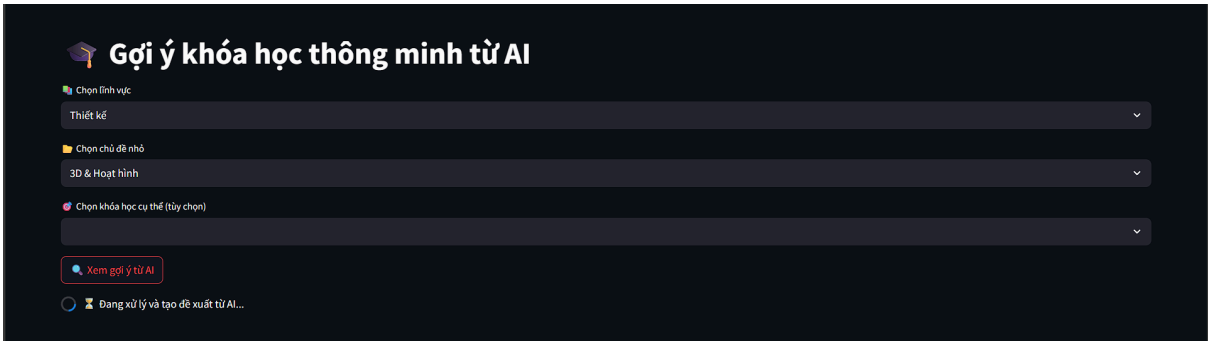
- **Hiển thị danh sách các khóa học được đề xuất**: Sau khi chọn truy vấn, hệ thống sẽ sử dụng mô hình tốt nhất (SVD + Nearest Neighbors) để tính toán độ tương đồng và hiển thị danh sách các khóa học có nội dung gần giống nhất.

- **Trình bày thông tin chi tiết của các khóa học**: Các gợi ý được hiển thị cùng với các đặc trưng quan trọng như tiêu đề, rating, số học viên, giá, giúp người dùng dễ dàng so sánh và đưa ra lựa chọn phù hợp.

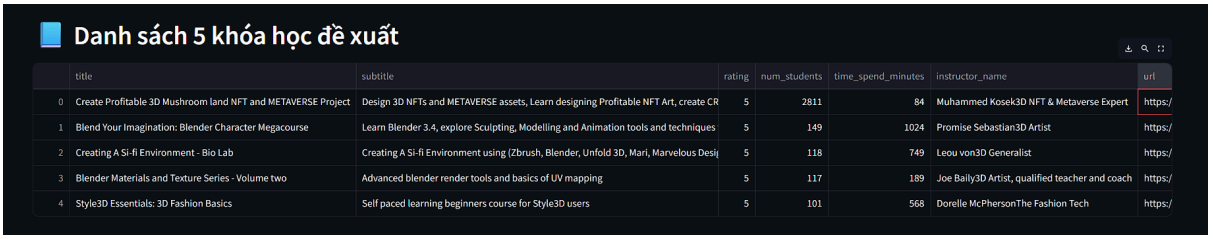
```
# --- UI ---
st.title("🔍 Gợi ý khóa học thông minh từ AI")

category = st.selectbox("📚 Chọn lĩnh vực", sorted(df['category'].unique()))
sub_df = df[df['category'] == category]
subcategory = st.selectbox("📂 Chọn chủ đề nhỏ", sorted(sub_df['sub_category'].unique()))
sub_sub_df = df[df['sub_category'] == subcategory]
course_list = sorted(sub_sub_df['title'].unique())
course = st.selectbox("🎯 Chọn khóa học cụ thể (tùy chọn)", [""] + course_list)

if st.button("🔎 Xem gợi ý từ AI"):
    with st.spinner("⏳ Đang xử lý và tạo đề xuất từ AI..."):
        if course:
            idx = df[df['title'] == course].index[0]
            result_df = recommend_similar_courses(idx)
        else:
            result_df = recommend_by_subcategory(subcategory)

        context = build_context(result_df)
        explanation = answer(context)

        st.markdown("## 📘 Danh sách 5 khóa học đề xuất")
        st.dataframe(result_df, use_container_width=True)

        st.markdown("## 💡 Phân tích tổng quan từ AI")
        st.markdown(explanation)
```

- UI (users interface) khi thực hiện chạy chương trình, có thể thấy hệ thống cho phép người dùng chọn lĩnh vực hay chuyên ngành mà mình mong muốn, thậm chí có thể chọn khóa học mà bản thân quan tâm tới. Lời khuyên của Gemini sẽ được in ra sau khi người dùng đã nhập input vào và nhấn xem gợi ý từ AI.

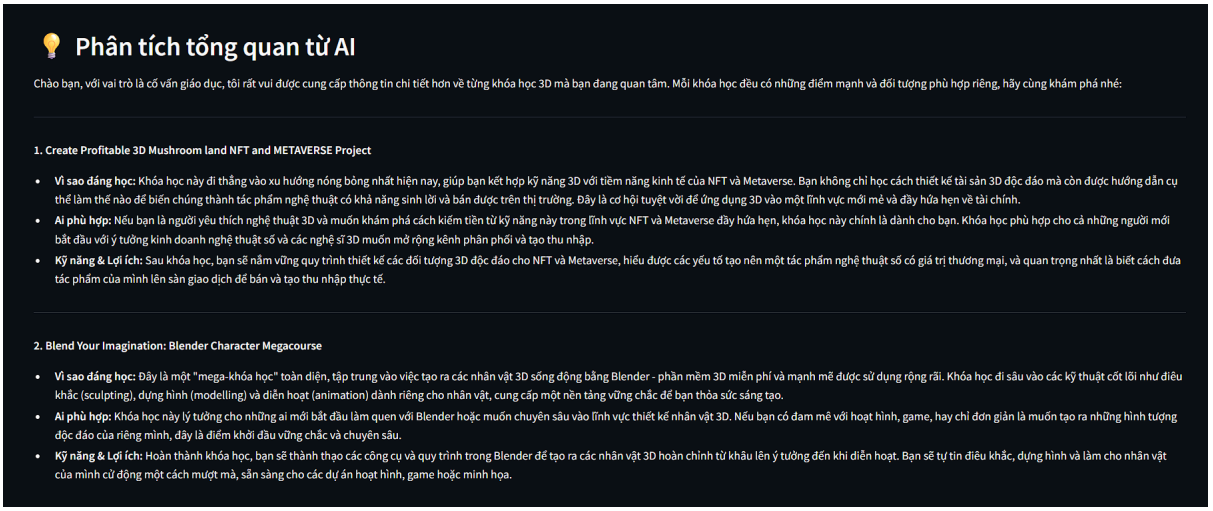
- Ví dụ về cách thức hoạt động của mô hình



Nhận input được chọn từ người dùng, trong trường hợp này lĩnh vực người dùng quan tâm là "Thiết kế (design)" và chuyên môn là "3D & Hoạt hình". Có thể bỏ qua khóa học cụ thể nếu người dùng chỉ mới tò mò tìm hiểu về lĩnh vực hay chuyên môn. Kết quả gợi ý từ AI sẽ được in ra sau khi người dùng nhấn vào nút "Xem gợi ý từ AI".



	title	subtitle	rating	num_students	time_spend_minutes	instructor_name	url
0	Create Profitable 3D Mushroom land NFT and METAVERSE Project	Design 3D NFTs and METAVERSE assets, Learn designing Profitable NFT Art, create CR	5	2811	84	Muhammed Kosek3D NFT & Metaverse Expert	https:/
1	Blend Your Imagination: Blender Character Megacourse	Learn Blender 3.4, explore Sculpting, Modelling and Animation tools and techniques	5	149	1024	Promise Sebastian3D Artist	https:/
2	Creating A Si-fi Environment - Bio Lab	Creating A Si-fi Environment using (Zbrush, Blender, Unfold 3D, Mari, Marvelous Desi	5	118	749	Leou von3D Generalist	https:/
3	Blender Materials and Texture Series - Volume two	Advanced blender render tools and basics of UV mapping	5	117	189	Joe Baily3D Artist, qualified teacher and coach	https:/
4	Style3D Essentials: 3D Fashion Basics	Self paced learning beginners course for Style3D users	5	101	568	Dorelle McPhersonThe Fashion Tech	https:/



💡 Phân tích tổng quan từ AI

Chào bạn, với vai trò là cố vấn giáo dục, tôi rất vui được cung cấp thông tin chi tiết hơn về từng khóa học 3D mà bạn đang quan tâm. Mỗi khóa học đều có những điểm mạnh và đối tượng phù hợp riêng, hãy cùng khám phá nhé:

1. Create Profitable 3D Mushroom land NFT and METAVERSE Project

- **Vì sao đáng học:** Khóa học này đi thẳng vào xu hướng nóng bỏng nhất hiện nay, giúp bạn kết hợp kỹ năng 3D với tiềm năng kinh tế của NFT và Metaverse. Bạn không chỉ học cách thiết kế tài sản 3D độc đáo mà còn được hướng dẫn cụ thể làm thế nào để biến chúng thành tác phẩm nghệ thuật có khả năng sinh lời và bán được trên thị trường. Đây là cơ hội tuyệt vời để ứng dụng 3D vào một lĩnh vực mới mẻ và đầy hứa hẹn về tài chính.
- **Ai phù hợp:** Nếu bạn là người yêu thích nghệ thuật 3D và muốn khám phá cách kiếm tiền từ kỹ năng này trong lĩnh vực NFT và Metaverse đầy hứa hẹn, khóa học này chính là dành cho bạn. Khóa học phù hợp cho cả những người mới bắt đầu với ý tưởng kinh doanh nghệ thuật số và các nghệ sĩ 3D muốn mở rộng kênh phân phối và tạo thu nhập.
- **Kỹ năng & Lợi ích:** Sau khóa học, bạn sẽ nắm vững quy trình thiết kế các đối tượng 3D độc đáo cho NFT và Metaverse, hiểu được các yếu tố tạo nên một tác phẩm nghệ thuật số có giá trị thương mại, và quan trọng nhất là biết cách đưa tác phẩm của mình lên sàn giao dịch để bán và tạo thu nhập thực tế.

2. Blend Your Imagination: Blender Character Megacourse

- **Vì sao đáng học:** Đây là một "mega-khóa học" toàn diện, tập trung vào việc tạo ra các nhân vật 3D sống động bằng Blender - phần mềm 3D miễn phí và mạnh mẽ được sử dụng rộng rãi. Khóa học đi sâu vào các kỹ thuật cốt lõi như điêu khắc (sculpting), dựng hình (modelling) và diễn hoạt (animation) dành riêng cho nhân vật, cung cấp một nền tảng vững chắc để bạn thỏa sức sáng tạo.
- **Ai phù hợp:** Khóa học này lý tưởng cho những ai mới bắt đầu làm quen với Blender hoặc muốn chuyên sâu vào lĩnh vực thiết kế nhân vật 3D. Nếu bạn có đam mê với hoạt hình, game, hay chỉ đơn giản là muốn tạo ra những hình tượng độc đáo của riêng mình, đây là điểm khởi đầu vững chắc và chuyên sâu.
- **Kỹ năng & Lợi ích:** Hoàn thành khóa học, bạn sẽ thành thạo các công cụ và quy trình trong Blender để tạo ra các nhân vật 3D hoàn chỉnh từ khâu lên ý tưởng đến khi diễn hoạt. Bạn sẽ tự tin điêu khắc, dựng hình và làm cho nhân vật của mình cử động một cách mượt mà, sẵn sàng cho các dự án hoạt hình, game hoặc minh họa.

Kết quả nhận được sẽ là những thông tin vì sao nên chọn khóa học này, nó phù hợp với những ai và cuối cùng là những kỹ năng có thể đạt được sau khi hoàn thành khóa học này.

3. Hạn chế của đề tài

- Hạn chế về thời gian thu thập dữ liệu và tính bao quát của dữ liệu:
    - Khi thực hiện cào dữ liệu từ Udemy, nhóm sử dụng 2 thư viện chính là Selenium và BeautifulSoup. Để tránh bị ngăn chặn bởi trang web khi thực hiện hoạt động cào, nhóm phải cài đặt một khoảng thời gian chờ tương đối, dẫn tới việc số lượng dữ liệu cào được còn khá ít.
    - Như đã nói ở trên, thời gian để cào 1 khóa học có thể lên tới 1-2 phút / 1 khóa học, chưa kể thời gian thu thập đường dẫn (url) truy cập vào khóa học. Vì vậy nhóm tác giả đã bỏ sót một vài trường dữ liệu có thể phân tích được, làm giảm đi độ đa dạng của bài phân tích

    → Đề xuất: Tối ưu hóa thời gian chờ, thực hiện cào trên nhiều máy hoặc cân nhắc thay đổi công cụ phổ biến hiện nay như Scrapy, Playwright, … là những công cụ đã được nâng cấp để phù hợp với tình hình hiện tại, khi mà các website lớn đã quan tâm nhiều hơn đến vấn đề bảo mật dữ liệu.
- :Hạn chế về mô hình đề xuất
    - Ban đầu, nhóm tác giả dự định triển khai mô hình RAG (Retrieval-Augmented Generation) nhằm phát triển chatbot tư vấn khóa học cho người dùng. Tuy nhiên, do thiếu hụt các trường dữ liệu văn bản mang tính diễn giải, việc xây dựng một database context đủ sâu cho bước truy xuất và sinh ngôn ngữ gặp nhiều trở ngại.

    - Trong điều kiện đó, nhóm quyết định chuyển hướng sang mô hình gợi ý dựa trên nội dung (content-based recommendation), tận dụng vector hóa đặc trưng và đo độ tương đồng giữa các khóa học để đưa ra gợi ý thay vì sinh câu trả lời tự động.
    - Đề xuất: Bổ sung và làm phong phú dữ liệu văn bản, kết hợp sử dụng thêm các loại mô hình retrival cũng như mô hình sinh ngôn ngữ khác. Hướng đến việc xây dựng được một hệ thống RAG hoàn chỉnh, là xu hướng trong thời điểm cạnh tranh về trí tuệ nhân tạo như hiện nay.
- Hạn chế trong quá trình thực hiện dự án
    - Do không thể vạch rõ một lộ trình đầy đủ, rõ ràng khi thực hiện dữ án. Nhóm gặp nhiều vấn đề ở các khâu tổng hợp dữ liệu cũng như tiền xử lý, khiến việc này lặp lại nhiều lần dẫn đến giảm mất thời gian thực hiện những phân tích chuyên sâu hơn
    - Đề xuất: Sử dụng nhiều công cụ quản trị dự án, nhằm có một lộ trình rõ ràng, hướng đến một mục tiêu cụ thể. Đồng thời đối với việc tiền xử lý, nên xây dựng một hệ thống pipeline cụ thể, tránh việc khi bắt đầu sử dụng dữ liệu mới phát hiện ra còn những vấn đề cần phải được xử lý.

4. Phụ lục

Link Tableau public	Capstone