Đại học quốc gia TP.HCM
Trường đại học công nghệ thông tin

Môn học: Khai phá dữ liệu trong doanh nghiệp

Lớp: DS317.P11

Báo cáo phân tích bộ dữ liệu

GVHD: ThS. Nguyễn Thị Anh Thư

Nhóm sinh viên thực hiện:

Nguyễn Hữu Nam	MSSV: 22520917
Nguyễn Khánh	MSSV: 22520641
Võ Đình Khánh	MSSV: 22520659
Nguyễn Minh Sơn	MSSV: 22521254
Bùi Hồng Sơn	MSSV: 22521246

Mục lục

1. Báo cáo phân tích bộ dữ liệu

1.1. Tìm hiểu dữ liệu

1.1.1. Giới thiệu bộ dữ liệu sử dụng

MOOCCubeX là một trong những bộ dữ liệu lớn nhất và chi tiết nhất về MOOCs (Massive Open Online Courses), hỗ trợ các nghiên cứu về hành vi học tập trực tuyến và cá nhân hóa học tập. Bộ dữ liệu được xây dựng bởi Nhóm Kỹ thuật Tri thức (Knowledge Engineering Group) tại Đại học Thanh Hoa (Tsinghua University), Trung Quốc, với sự hợp tác của XuetangX, một nền tảng MOOC lớn tại Trung Quốc. Đây là bộ dữ liệu đa dạng, phục vụ cho nghiên cứu trong các lĩnh vực như học máy, hệ thống học tập thích ứng, phân tích giáo dục, và trí tuệ nhân tạo.

MOOCCubeX bao gồm nhiều loại dữ liệu khác nhau, tập trung vào các khóa học và hành vi học tập của học viên. Các thành phần chính của bộ dữ liệu bao gồm

Courses
-Số lượng khóa học 4,216
-Nội dung: Mỗi khóa học bao gồm các video giảng dạy, bài tập, và bài kiểm tra. Thông tin về mỗi khóa học bao gồm tiêu đề, mô tả, người hướng dẫn, ngày bắt đầu và ngày kết thúc, ngôn ngữ giảng dạy và lĩnh vực học tập
Video
-Số lượng: 230,263
-Thông tin: Các video giảng dạy được thu thập từ các khóa học trên nền tảng MOOC. Mỗi vidfeo có các thuộc tính như tiêu đề, thời lượng, nội dung được giảng dạy, và số lần xem của học viên
Exercise
-Số lượng: 258,265
-Thông tin: bao gồm các bài tập tự luyện và kiểm tra đánh giá. Các

bài tập này được thiết kế để giúp học viên ôn luyện kiến thức và kiểm tra khả năng tiếp thu sau mỗi phần học

Problem

-Số lượng: 2,454,397 vấn đề

-Thông tin: Thường là các vấn đề hoặc câu hỏi phức tạp yêu cầu học viên giải quyết bằng cách áp dụng kiến thức học được từ khóa học

Student Profile

-Số lượng: 3,330,294 hồ sơ

-Thông tin: Hồ sơ học viên lưu trữ các thông tin về hành vi học tập, tiến trình học tập và các hoạt động của họ trên nền tảng

Video watching behavior

-Số lượng: 154,332,174 dữ liệu

-Thông tin: Dữ liệu hành vi xem video cung cấp thông tin chi tiết về cách học viên tương tác với video giảng dạy. Dữ liệu này giúp nghiên cứu thói quen học tập của học viên

Comment and Reply

-Số lượng: 8,422,134 bản ghi phản hồi bình luận

-Thông tin: Bình luận và phản hồi là phần quan trọng trong việc đánh giá mức độ tương tác của học viên với khóa học. Là cơ sở để phân tích cảm xúc của học viên, đánh giá mức độ hài lòng và tìm kiếm những khó khăn mà học viên gặp phải trong quá tình học

Bộ dữ liệu MOOCCubeX được cung cấp dưới dạng các tệp tin JSON và CSV, cho phép người dùng dễ dàng tải xuống và sử dụng. Đây là một bộ dữ liệu quý giá cho nghiên cứu về giáo dục trực tuyến và học tập thích ứng. Với khối lượng dữ liệu lớn và đa dạng, bộ dữ liệu này mở ra nhiều cơ hội cho các nhà nghiên cứu trong việc hiểu sâu hơn về hành vi học tập và xây dựng các hệ thống học tập tiên tiến, giúp cải thiện hiệu quả giáo dục trên các nền tảng trực tuyến.

1.1.2. Mô tả về tập dữ liệu

I. Courses

Giới thiệu: Phần này mô tả về khóa học (course) và các tài nguyên liên quan, bao gồm các file: course.json, video.json, problem.json, school.json, teacher.json, course-field.json, course-school.txt, course-teacher.txt, exercise-problem.txt, video_id-ccid.txt.

Đây là bảng sơ lược về các file:

Tên	Loại	Mô tả	Kích thước
course.json	entities	Tổ chức video và bài tập của khóa học.	43MB
video.json	entities	Tên video và phụ đề.	580MB
exercise-problem.txt	relations	Một nhóm các bài tập của khóa học.	129MB
problem.json	entities	Các bài tập thực hành trong một nhóm bài tập.	1.2GB
school.json	entities	Thông tin về trường học.	613KB
teacher.json	entities	Thông tin về giáo viên.	8.7MB
course-field.json	relations	Lĩnh vực mà khóa học thuộc về, được chú thích bởi con người.	62KB

Bảng course.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
about	Giới thiệu khóa học	string	
id	ID của khóa học	string	Bắt đầu bằng "C_"
field	Danh sách các lĩnh vực của khóa học	list<string>	
name	Tên trường	string	
prerequisites	Nội dung về kiến thức tiên quyết	string	
resource	Danh sách các tài nguyên	list<Resource>*	

***Bảng Resource**

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
resource_id	ID của tài nguyên.	string	Bắt đầu bằng "V_"nếu là video, "Ex_"nếu là bài tập.
chapter	Số chương.	list<string>	
titles	Danh sách các tiêu đề, bao gồm tiêu đề chương, tiêu đề video, v.v. Có tối đa 3 cấp tiêu đề.	list<string>	

Bảng video.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
ccid	ID duy nhất của video.	string	
name	Tên của video.	string	
start	Thời gian bắt đầu của từng câu trong phụ đề video.	list<float>	
end	Thời gian kết thúc của từng câu trong phụ đề video.	list<float>	
text	Phụ đề của từng câu trong video.	list<string>	

Bảng exercise-problem.json

Mô tả	Định dạng	Kích thước
Câu hỏi của bài tập.	exercise ID\tquestion ID	129MB

Bảng problem.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
id	ID của bài toán.	string	Bắt đầu với "Pm_"
exercise_id	ID của bài tập.	string	Bắt đầu với “Ex_”
language	Ngôn ngữ mô tả của bài toán, tiếng Trung/tiếng Anh.	string	Chinese hoặc English
title	Tiêu đề của bài tập.	string	

content	Mô tả bài toán.	string
option	Lựa chọn của bài toán.	json
answer	Đáp án của câu hỏi.	list<string>
score	Điểm số của câu hỏi.	string
type	Lựa chọn câu hỏi.	int
typetext	Lựa chọn câu hỏi.	string
location	Vị trí chương của bài toán.	string
context_id	leaf_id liên quan đến bài toán.	list<int>

Bảng school.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
id	ID của trường.	string	Bắt đầu với "S_"
name	Tên tiếng Trung của trường.	string	
name_en	Tên tiếng Anh của trường.	string	
sign	Chữ cái đầu của tên tiếng Anh của trường.	string	
about	Giới thiệu về trường.	string	
motto	Khẩu hiệu của trường.	string	

Bảng teacher.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
id	ID của giáo viên	string	Bắt đầu với "T_"
name	Tên tiếng Trung của giáo viên.	string	
name_en	Tên tiếng Anh của giáo viên.	string	
about	Hồ sơ giáo viên.	string	
job_title	Chức danh công việc.	string	
org_name	Cơ quan/đơn vị công tác.	string	

Bảng course-field.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
course_id	ID của khóa học.	int	
course_name	Tên của khóa học.	string	
field	Danh sách lĩnh vực được gán nhãn thủ công.	list<string>	

Các mối quan hệ khác

Tên	Mô tả	Định dạng	Kích thước
course-school.txt	Trường dạy khóa học.	course ID\tschool ID	60KB
course-teacher.txt	Giáo viên dạy khóa học.	course ID\tteacher ID	1.6MB
video_id-ccid.txt	Phụ đề của video	Video ID\tccid	115MB

II. User

Giới thiệu: Phần này mô tả hành vi người học (user), bao gồm các file: user.json, comment.json, reply.json, course-comment.txt, user-comment.txt, user-reply.txt, comment-reply.txt, user-problem,json, user-video.json, user-xiaomu.json.

Đây là bảng sơ lược về các file:

Tên	Loại	Mô tả	Kích thước
user.json	entities	Thông tin của học sinh (user)	770MB
comment.json	entities	Thông tin bình luận của user lên từng tài nguyên của course	2.1GB
reply.json	entities	Thông tin của phần trả lời bình luận (reply) của user trên từng tài nguyên của courses	50MB
user-problem.json	relations	Thông tin về bài tập mà user làm	50MB
user-video.json	relations	Quá trình của user xem video: số lần tua, giây bắt đầu, giây kết thúc,.	3.0GB
user-xiaomu.json	relations	Tương tác của người dùng với Xiaomu (bot QA của XuetangX).	50MB

Bảng user.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
id	Id người dùng	string	bắt đầu bằng "U_"
name	Tên người dùng	string	
gender	Giới tính	int	0, 1, hoặc 2
school	Tên trường	string	
year_of_birth	Năm sinh	list<int>	
course_order	Các mã khóa học đã chọn	Thông tin về bài tập mà user làm	
enroll_time	Thời gian đăng kí tương ứng với từng khoá học	list<DateTime>.	Định dạng DateTime là "YYYY-MM-DD HH:MM:SS"

Bảng comment.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
id	Comment ID	string	bắt đầu bằng "Cm_"
user_id	ID của người dùng đã bình luận	Int	bắt đầu bằng "U_"
text	Nội dung bình luận	String	
create_time	Thời gian bình luận	DateTime	định dạng "YYYY-MM-DD HH:MM:SS"
resource_id	ID của tài nguyên mà user bình luận	String	Có thể nhận giá trị null

Bảng reply.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
id	Reply ID	string	bắt đầu bằng "Rp_"
user_id	ID của người dùng đã bình luận	string	bắt đầu bằng "U_"
text	Nội dung phản hồi	string	
create_time	Thời gian phản hồi	DateTime	định dạng "YYYY-MM-DD HH:MM:SS"

Bảng user-problem.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
log_id	ID của bản ghi câu hỏi của người dùng	string	kết hợp với khóa duy nhất của user_id và problem_id
user_id	ID người dùng	string	bắt đầu bằng "U_"
problem_id	ID vấn đề	string	bắt đầu bằng "Pm_"
is_correct	Câu hỏi có đúng không	bool	0 hoặc 1

attempts	Số lượng câu hỏi đã thử	int	
score	Điểm của người dùng	float	
submit_time	Thời gian làm câu hỏi	DateTime	định dạng "YYYY-MM-DD HH:MM:SS"

Bảng user-video.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
user_id	ID của user	string	bắt đầu bằng "U_"
seq	Mảng chứa quá trình người dùng xem video, bao gồm thời gian xem video, thời gian bắt đầu và kết thúc của video, và tốc độ xem video, v.v.	list<object>.	Mỗi object sẽ gồm 2 trường video_id (string) và segment (list<object>). Mỗi phần tử trong segment bao gồm các trường start_point (float), end_point (float), speed (float), local_start_time (int)

Bảng user-xiaomu.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
user_id	ID của user	string	bắt đầu bằng "U_"
question_type	ID của user	string	
question	Câu hỏi hỏi bởi user	string	

Các mối quan hệ khác

Tên	Mô tả	Định dạng	Kích thước
course-comment.txt	Phản hồi bình luận của người dùng lên course	course ID\treview ID	60KB
user-comment.txt	bình luận của người dùng.	user ID\tcomment ID	1.6MB
user-reply.txt	Phản hồi bình luận của người dùng.	user ID\treply ID	1.6MB
comment-reply.txt	Phản hồi bình luận liên quan đến khái niệm (phần concept).	concept ID\treply ID	115MB

III. Concept

Giới thiệu: Phần này mô tả về khái niệm khóa học (course concept) và các file liên quan, bao gồm: concept.json, other.json, paper.json, concept-other.txt, concept-paper.txt, concept-problem.txt, concept-video.txt, concept-comment.txt.

Đây là bảng sơ lược về các file:

Tên	Loại	Mô tả	Kích thước
concept.json	entities	Thông tin về khái niệm khóa học	43MB
other.json	entities	Các tài liệu liên quan được thu thập bên ngoài những khoá học	580MB
paper.json	entities	Những bài báo khoa học liên quan	129MB
concept-other.txt	relations	Khái niệm liên quan tới các nguồn ngoài khóa học	1.2MB
concept-paper.txt	relations	Khái niệm liên quan đến luận án	613KB
concept-problem.txt	relations	Khái niệm liên quan đến vấn đề	8.7MB
concept-video.txt	relations	Khái niệm liên quan đến video	8.7MB
concept-comment.txt	relations	Khái niệm liên quan đến phần bình luận	62KB

Bảng concept.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
id	ID của khái niệm	string	Định dạng là K_concept name_field
name	Tên của khái niệm, và tên này sẽ giống với tên xuất hiện trong id	string	
context	Ngữ cảnh mà khái niệm đó xuất hiện	string	

Bảng other.json

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
id	Mã dữ liệu, không có ý nghĩa cụ thể (đơn thuần là một định danh duy nhất cho từng mục dữ liệu).	string	
concept	Khái niệm mà thông tin này liên quan đến hoặc được thu thập dựa trên	string	
type	Nguồn dữ liệu	string	Miền giá trị là [“zhihu”, ”baike”, “wiki”]
content	Nội dung của dữ liệu, có thể là văn bản hoặc thông tin được thu thập từ các nguồn đã nêu	string	

Các mối quan hệ khác

Tên	Mô tả	Định dạng	Kích thước
concept-other.txt	Lưu trữ mối quan hệ giữa các khái niệm và các tài liệu, tài nguyên ngoại khóa được thu thập từ các nguồn bên ngoài khóa học	concept ID\tresource ID	60KB
concept-paper.txt	Lưu trữ mối quan hệ giữa các khái niệm và các bài báo khoa học có liên quan	concept ID\tpaper ID	1.6MB
concept-problem.txt	Lưu trữ mối quan hệ giữa các khái niệm và các câu hỏi hoặc bài tập liên quan	concept ID\tquestion ID	1.6MB
concept-video.txt	Lưu trữ mối quan hệ giữa các khái niệm và các video liên quan	concept ID\tccid	1.6MB
concept-comment.txt	Lưu trữ mối quan hệ giữa các khái niệm và các bình luận của người dùng có liên quan	concept ID\treview ID	115MB

IV. Prerequisites

Bảng prerequisites/cs.json

- Nội dung: Chú thích và dự đoán về các điều kiện tiên quyết của môn Khoa học máy tính
- Số lượng mẫu: 492,102 mẫu

Thuộc tính	Nội dung	Kiểu dữ liệu	Miền giá trị
c1	Khái niệm điều kiện tiên quyết	string	
c2	Khái niệm điều kiện sau sửa chữa	string	
ground_truth	Chỉ ra có mối quan hệ sữa chữa tuần tự hay không	int	Miền giá trị là 0 hoặc 1
text_predict	Cung cấp kết quả dự đoán sử dụng đặc điểm văn bản	list<float>	
graph_predict	Mức độ tin cậy của dự đoán được đạt được bằng các đặc điểm đồ thị	list<float>	

Bảng prerequisites/math.json

- Nội dung: Chú thích và dự đoán các khái niệm trong lĩnh vực toán học, theo định dạng giống cs,json
- Số lượng mẫu: 331202

Bảng prerequisites/psy.json

- Nội dung: Chú thích và dự đoán các khái niệm trong lĩnh vực tâm lý học, theo định dạng giống cs.json
- Số lượng mẫu: 757771

1.1.3. Nhận xét

Sau khi khảo sát bộ dữ liệu MOOCCubeX, chúng em đã rút ra một số nhận xét như sau:

- Tính đa dạng và phong phú: Bộ dữ liệu MOOCCubeX chứa đựng nhiều loại thông tin khác nhau liên quan đến giáo dục trực tuyến, bao gồm các khóa học, bài giảng video, bài tập, hồ sơ học sinh, cũng như hành vi tương tác của học sinh với các tài nguyên học tập. Đây là một bộ dữ liệu có mức độ đa dạng cao, giúp cung cấp cái nhìn toàn diện về nhiều khía cạnh trong quá trình học tập trực tuyến.

- Quy mô lớn: Bộ dữ liệu có kích thước lớn và chứa đựng hàng triệu điểm dữ liệu, từ đó tạo cơ sở vững chắc cho các bài toán khai thác dữ liệu, học máy, học sâu. Nhờ quy mô này, người nghiên cứu có thể khám phá và áp dụng các phương pháp tiên tiến trong lĩnh vực phân tích dữ liệu giáo dục.

- Tính chi tiết và tổ chức linh hoạt: Mặc dù không đồng nhất về loại dữ liệu, bộ dữ liệu MOOCCubeX được tổ chức bài bản với cấu trúc rõ ràng và chi tiết. Điều này giúp người dùng dễ dàng tìm kiếm và trích xuất các thông tin quan trọng, đồng thời cung cấp sự linh hoạt trong việc áp dụng bộ dữ liệu vào nhiều mục tiêu khác nhau. Các yếu tố như hành vi học tập, bình luận của học sinh, và các tài liệu khóa học đều được ghi nhận chi tiết, tạo nền tảng tốt cho việc xây dựng các hệ thống hỗ trợ học tập thông minh.

1.1.4. Mục tiêu sử dụng bộ dữ liệu:

Với các đặc điểm nêu trên, chúng em định khai thác bộ dữ liệu MOOCCubeX để giải quyết các bài toán thuộc lĩnh vực Cố vấn học tập thông minh. Cụ thể, nhóm đã đưa ra bài toán sau:

Bài toán: Hệ khuyến nghị khóa học.

- Mục tiêu: Xây dựng hệ thống khuyến nghị giúp sinh viên chọn lựa môn học hoặc khóa học phù hợp với định hướng chuyên ngành dựa trên hành vi học tập của họ. Điều này có thể bao gồm các yếu tố như các khóa học mà sinh viên đã hoàn thành, kết quả học tập, thời gian dành cho mỗi môn học, và sự tương tác của họ với tài nguyên học tập (Như video, bài tập, và bài kiểm tra).

- Ứng dụng: Hệ thống sẽ hỗ trợ sinh viên đưa ra các quyết định học tập thông minh hơn, giúp họ lựa chọn các môn học phù hợp với năng lực và định hướng cá nhân. Điều này không chỉ giúp tối ưu hóa quá trình học tập mà còn tăng khả năng hoàn thành các chương trình học, đặc biệt trong các môi trường giáo dục trực tuyến hoặc bán trực tuyến.

- Khả năng áp dụng: Bài toán này hoàn toàn có thể được áp dụng trong bối cảnh giáo dục đại học, cụ thể là ở Việt Nam. Đặc biệt là tại các trường có chương trình học trực tuyến hoặc có nhu cầu xây dựng hệ thống cố vấn học tập dựa trên dữ liệu. Hệ thống có thể giúp sinh viên định hướng chuyên ngành, chọn lựa các môn học phù hợp, và điều chỉnh lộ trình học tập dựa trên kết quả học tập và hành vi của họ.

Nhìn chung, việc ứng dụng bộ dữ liệu MOOCCubeX vào các bài toán như vậy có tiềm năng lớn trong việc hỗ trợ sinh viên và nâng cao trải nghiệm học tập trong môi trường giáo dục điểm số.

1.2. Chuẩn bị dữ liệu

1.2.1. Dịch bảng

Trong quá trình chuyển ngữ từ Trung sang Việt, chúng em đã tận dụng thư viện "googletrans một công cụ Python không mất phí và không giới hạn số lần dịch. Thư viện này vận hành thông qua API Google Translate Ajax để thực hiện các tác vụ như nhận diện ngôn ngữ và dịch thuật.

Do khối lượng dữ liệu lớn, quá trình dịch gặp phải một số thách thức về thời gian và kết nối. Để khắc phục, chúng em đã triển khai các giải pháp sau:

- Lưu lại tiến trình dịch để tránh mất dữ liệu
- Thiết lập cơ chế tự động gửi lại yêu cầu khi mất kết nối
- Ứng dụng thư viện "asyncio"cho phép gửi đồng thời nhiều API, giúp tối ưu tốc độ xử lý

Đây là một phần code mẫu đã sử dụng phương pháp đã nêu trên:

```
async def translate(df, batch_start, batch_end):
    tasks = []
    for i in range(batch_start, batch_end):
        tasks.append(async_translate(df.loc[i, COL], i))

    df.loc[batch_start: batch_end - 1, COL] = await asyncio.gather(*tasks)


df = pd.read_json('teacher.json', lines=True)
batch_size = 1000
for i in range(0, len(df), batch_size):
    batch_end = min(len(df), i + batch_size)
    asyncio.run(translate(df, i, batch_end))

df[COL].to_csv(f"translated_{COL}.csv", index=False)
```

Ngoài ra, chúng em nhận thấy không cần thiết phải dịch toàn bộ các trường dữ liệu lớn để huấn luyện mô hình vì một số trường dữ liệu không hỗ trợ cho việc huấn luyện mô hình. Thay vào đó, chúng em chỉ tập trung dịch 1 số trường sau đây:

-**course.json:** dịch cột "name", "field", "prerequisites" và "about"

```
async def translate(df, batch_start, batch_end):
    tasks = []
    for i in range(batch_start, batch_end):
        tasks.append(async_translate(df.loc[i, COL], i))

    df.loc[batch_start: batch_end - 1, COL] = await asyncio.gather(*tasks)


df = pd.read_json('teacher.json', lines=True)
batch_size = 1000
for i in range(0, len(df), batch_size):
    batch_end = min(len(df), i + batch_size)
    asyncio.run(translate(df, i, batch_end))

df[COL].to_csv(f"translated_{COL}.csv", index=False)
```

-**user.json:** dịch cột "school"

```
user_df = pd.DataFrame(data_list)
user_df.head()
```

✓ 44.0s Python

	id	name	gender	school	year_of_birth	course_order	enroll_time
0	U_22	我	0.0	None	2015.0	[682129, 2294668]	[2019-10-12 10:28:02, 2020-11-21 14:03:28]
1	U_24	王帅国	1.0	Tsinghua University	6558.0	[597214, 605512, 597211, 597314, 597208, 62950...	[2019-05-20 16:06:48, 2019-05-24 19:34:43, 201...
2	U_25	王帅国	0.0	Tsinghua University	NaN	[1903985]	[2020-08-07 18:59:13]
3	U_53	于歆杰	1.0	Tsinghua University	1973.0	[696679, 1704639, 943255, 1729417, 682164, 177...	[2020-03-01 21:24:30, 2020-03-12 16:17:02, 202...
4	U_54	马显春	2.0	Tsinghua University	NaN	[682442, 682164, 1748240, 1778890, 1829031, 17...	[2019-10-09 02:17:49, 2019-11-08 00:49:03, 202...

-**teacher.json:** Tiến hành dịch tất cả (trừ "id" và "name")

```
teacher_df.head()
```

	id	name	name_en	about	job_title	org_name
0	T_1	刘燕妮	Yanni Liu	Graduated from the Philosophy Department of Pe...	lecturer	Tsinghua University
1	T_2	陈怡	Yi Chen	Born in Chongqing in 1945, he graduated from H...	professor	Tsinghua University
2	T_3	程钢	Gang Cheng	Cheng Gang is the course leader of "Introducti...	Associate Professor	Tsinghua University
3	T_4	谢维和	xie wei he	Xie Weihe, PhD, professor, doctoral supervisor...	professor	Tsinghua University
4	T_5	史静寰	Jing-huan Shi	Shi Jinghuan, female, professor and doctoral s...	professor	Tsinghua University

-**concept.json:** Dịch tất cả các cột của bảng này vì toàn bộ đều ở dạng chuỗi

```
df = pd.read_json("../translated/concept_translated.json", lines=True)
df.head()
```
✓ 3.1s

	id	name	context
0	K_Nervous system_Histology and Embryology	Nervous system	[]
1	K_TSH cells_Histology and Embryology	TSH cells	['The pituitary gland consists of two parts: t...
2	K_Chromophilic cells_Histology and Embryology	Chromophilic cells	[]
3	K_Growth hormone cells_Histology and Embryology	Growth hormone cells	['Answer: B\n13. Adenohypophysis eosinophils c...
4	K_Limonite_Materials Science and Engineering	Limonite	['\nLimonite is a common iron ore, often forme...

-**course-field.json:** Tiến hành dịch cột course_name và field mang các thông tin dưới dạng chuỗi của bảng.

```
df = pd.read_json("../original_translated/course-field-translated.json", lines=True)
df.head()
```
✓ 0.0s Python

	course_id	course_name	field
0	584313	Introduction to "Zi Zhi Tong Jian"	[Chinese language and literature, History]
1	681932	"Learning by doing" Java programming	[Computer Science and Technology]
2	674962	The spatial art of "Dream of Red Mansions"	[Chinese language and literature]
3	682709	Introduction to the Critique of Pure Reason	[philosophy]
4	682635	Introduction to "Tongwancheng"	[History]

1.2.2. Khám phá dữ liệu

a) Bảng course.json

Ta xem qua bảng course.json:

```
course_df = pd.DataFrame(data_list)
course_df.head()
```

Python

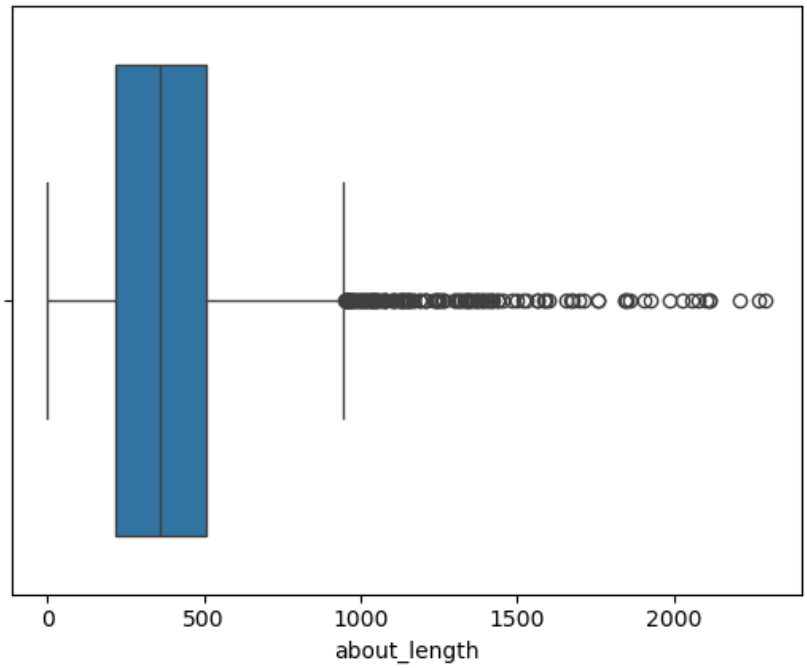
	id	name_trans	field	prerequisites_trans	about_trans	resource
0	C_584313	introduction to "zi zhi tong jian"	[history, chinese language and literature]		through the teacher's guidance, students can g...	[{'titles': ['第一课 导论与三家分晋', '导论', '导论'], 'reso...
1	C_584329	calculus - limit theory and functions of one v...	[applied economics, math, physics, theoretical...		this course is a basic mathematics course in s...	[{'titles': ['序言', '序言', '序言'], 'resource_id':...
2	C_584381	photojournalism	[art, journalism and communication]		master basic photography skills, understand ho...	[{'titles': ['第一章 绪论', '第一讲 引言1', '引言1'], 'res...
3	C_597208	data mining: theory and algorithms	[computer science and technology]		the most interesting theory + the most useful ...	[{'titles': ['走进数据科学：博大精深，美不胜收', '整装待发', 'Vide...
4	C_597225	university computer	[]		university computer courses will be guided by ...	[{'titles': ['第1周： 基于计算机的问题求解', '课程介绍', '开篇'],...

Ta xét độ dài của 3 cột "about", "name_trans" và "resource":

	about_length	name_length	resource_length
count	3781.000000	3781.000000	3781.000000
mean	393.445120	36.942343	71.685533
std	267.904934	21.575065	74.802345
min	0.000000	2.000000	1.000000
25%	217.000000	22.000000	38.000000
50%	361.000000	32.000000	59.000000
75%	509.000000	46.000000	88.000000
max	2293.000000	193.000000	2728.000000

Ta có thể thấy được 1 số thông tin từ dữ liệu trên:

- Có những dòng dữ liệu không tồn tại cột "about", tồn tại giá trị ngoại tệ ở cột "about" vì mean là 393 mà max lên đến 2293. Ta thể hiện trên boxplot độ dài của cột "about":



- Có thể thấy thật sự nhiều giá trị ngoại tệ cần được xử lí.
- Có những dòng dữ liệu không có resource_length, mean cũng rất ngắn (71) chứng tỏ ít thông tin về khoá học.

Ta phân tích sâu cột "resource":

```
course_df['resource'][0][0]
```

```
{'titles': ['第一课 导论与三家分晋', '导论', '导论'],
 'resource_id': 'V_849',
 'chapter': '1.1.1'}
```

Mỗi resource trong bảng 2 là 1 tập hợp các video hay một tập các exercise. Mỗi resource sẽ có thêm 1 resource_id là id của resource, chapter là chương chứa resource trong khóa học, titles gồm các tiêu đề như tiêu đề chương, video chương.

Thông tin của resource có thể tìm thấy trong file course.json. Một resource có 2 loại: Video và Exercise. Nếu loại tài nguyên là video, nó được xác định bằng ID video bắt đầu bằng ký tự V_. Nhiều video_id khác nhau tương ứng với một ccid, và ccid xác định duy nhất một video. Các video_id này tương ứng với việc hiển thị cùng một video ccid tại các thời gian bắt đầu khác nhau. Mối liên hệ giữa video_id và ccid được lưu trong relations/video_id-ccid.txt. Phụ đề video có thể được tìm thấy trong tệp entities/video.json thông qua ccid.

Ta sẽ kiểm tra xem có bao nhiêu ID video không hợp lệ để phục vụ cho quá trình xử lý dữ liệu sau này:

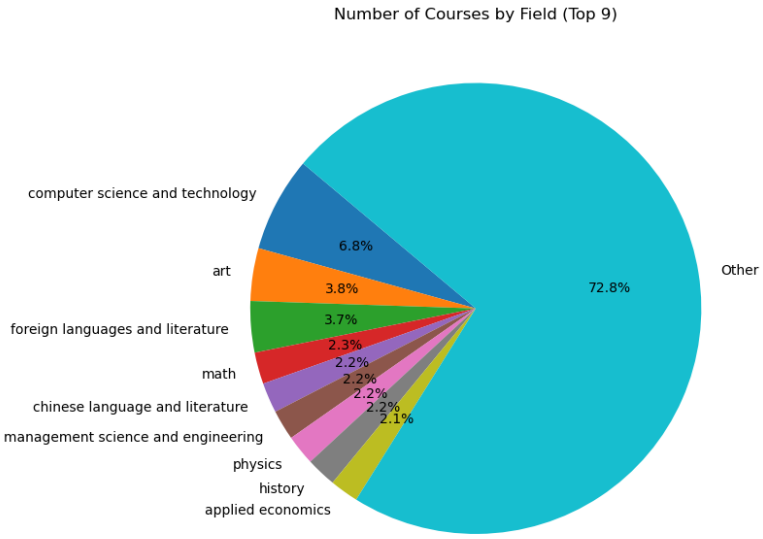
```
videoID = ccid_df['video_id'].unique()

valid_videoID = set(videoID)

non_existent_ids = unique_video_ids - valid_videoID

# Hiển thị kết quả
print(f"Tổng số lượng các video ID không tồn tại: {len(non_existent_ids)}")
print(f"Các video ID không tồn tại: {non_existent_ids}")
```

```
Tổng số lượng các video ID không tồn tại: 2397
Các video ID không tồn tại: {'V_543429', 'V_543378', 'V_543519', 'V_1056006', 'V_374
```

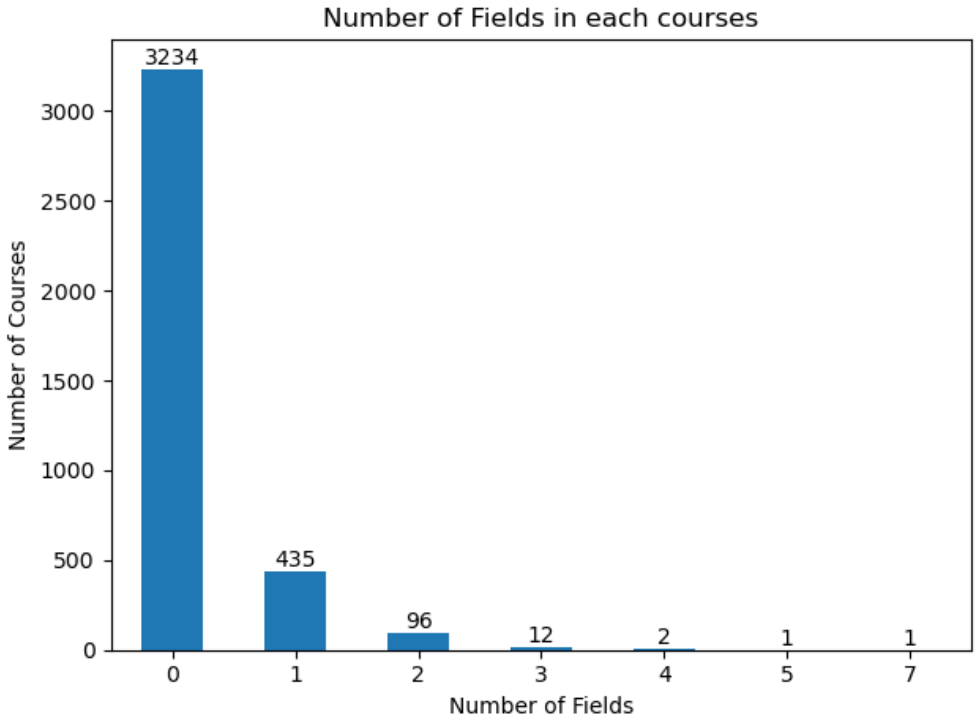
Có 2397 video ID không tồn tại, ta sẽ lọc đi hỗ trợ cho hiển thi thông tin trong tương lai.

Ta bắt đầu tiến hành đếm số khoá học trong cột "name_trans", chia bởi lĩnh vực (cột "field"):

```
2) Number of courses by field:
field
computer science and technology       63
art                                   35
foreign languages and literature      34
math                                  21
history                               20
                                      ..
marine science                         1
ship and marine engineering            1
army command science                   1
metallurgical engineering              1
basic chinese medicine                 1
Name: count, Length: 81, dtype: int64
```



Ta thấy có tổng 3781 khoá học và 81 lĩnh vực, với "computer science and technology" đứng đầu với 63 khoá học, chiếm 6.8% trên tổng khoá học. Ta cũng kiểm tra với mỗi khoá học được xếp bao nhiêu lĩnh vực (cột "field"):



Ta có thể thấy có rất nhiều khoá học không thuộc lĩnh vực nào, có rất nhiều khóa học không có field nào, có thể cột "field" sẽ không đóng góp nhiều trong xây dựng thuật toán hoặc cần xử lí.

b) Bảng user.json

Đầu tiên, ta đọc dữ liệu và quan sát dữ liệu thông qua dạng bảng (DataFrame):

	id	name	gender	school	year_of_birth	course_order	enroll_time
0	U_22	我	0.0	None	2015.0	[682129, 2294668]	[2019-10-12 10:28:02, 2020-11-21 14:03:28]
1	U_24	王帅国	1.0	Tsinghua University	6558.0	[597214, 605512, 597211, 597314, 597208, 62950...	[2019-05-20 16:06:48, 2019-05-24 19:34:43, 201...
2	U_25	王帅国	0.0	Tsinghua University	NaN	[1903985]	[2020-08-07 18:59:13]
3	U_53	于歆杰	1.0	Tsinghua University	1973.0	[696679, 1704639, 943255, 1729417, 682164, 177...	[2020-03-01 21:24:30, 2020-03-12 16:17:02, 202...
4	U_54	马昱春	2.0	Tsinghua University	NaN	[682442, 682164, 1748240, 1778890, 1829031, 17...	[2019-10-09 02:17:49, 2019-11-08 00:49:03, 202...

Ta tiến hành thống kê đặc điểm từng cột có trong bảng:

```
len(user_df)
✓ 0.0s                                                    Python
1128390
```

Hình 1: Số lượng users

```
user_df['gender'].describe()
✓ 1.7s                                                    Python
count    3.330240e+06
mean     9.455748e-01
std      8.321099e-01
min      0.000000e+00
25%      0.000000e+00
50%      1.000000e+00
75%      2.000000e+00
max      2.320000e+02
Name: gender, dtype: float64
```

Hình 2: Cột "gender"

```
user_df['gender'].value_counts()
✓ 0.0s                                                    Python
gender
0.0      1221931
1.0      1067858
2.0      1040449
232.0          1
3.0            1
Name: count, dtype: int64
```

Hình 3: Phân bố các các giá trị trong cột "gender":

```
user_df['school'].describe()
✓ 0.3s                                                    Python

count                  1128399
unique                   25848
top        Tsinghua University
freq                     18318
Name: school, dtype: object
```

Hình 4: Thông tin cột "school"

```
len(user_df["school"].unique())
✓ 0.0s                                                    Python

25849
```

Hình 5: Số lượng trường học trong bảng

```
user_df.info()
✓ 0.0s                                                    Python

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3330294 entries, 0 to 3330293
Data columns (total 7 columns):
 #   Column         Dtype
---  ------         -----
 0   id             object
 1   name           object
 2   gender         float64
 3   school         object
 4   year_of_birth  float64
 5   course_order   object
 6   enroll_time    object
dtypes: float64(2), object(5)
memory usage: 177.9+ MB
```

Hình 6: Kiểm tra thông tin tổng quan sau cùng

```
user_df.info()
```
✓ 0.0s Python

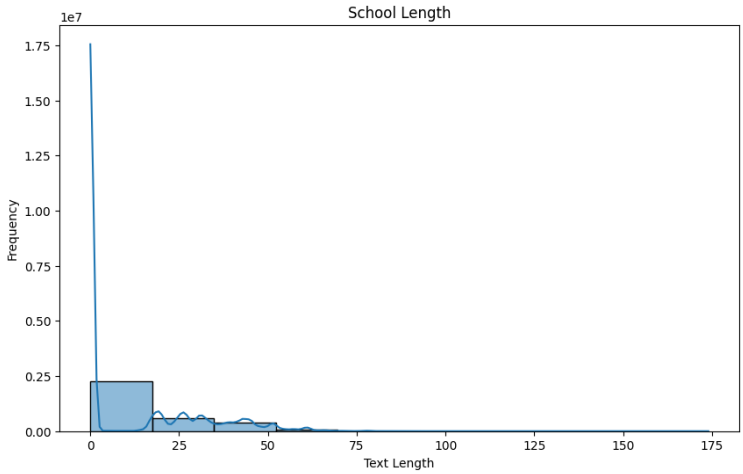
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3330294 entries, 0 to 3330293
Data columns (total 7 columns):
 #   Column         Dtype
---  ------         -----
 0   id             object
 1   name           object
 2   gender         float64
 3   school         object
 4   year_of_birth  float64
 5   course_order   object
 6   enroll_time    object
dtypes: float64(2), object(5)
memory usage: 177.9+ MB
```

Hình 7: Số lượng sample (users) có trong bảng và số lượng users thuộc về mỗi trường học

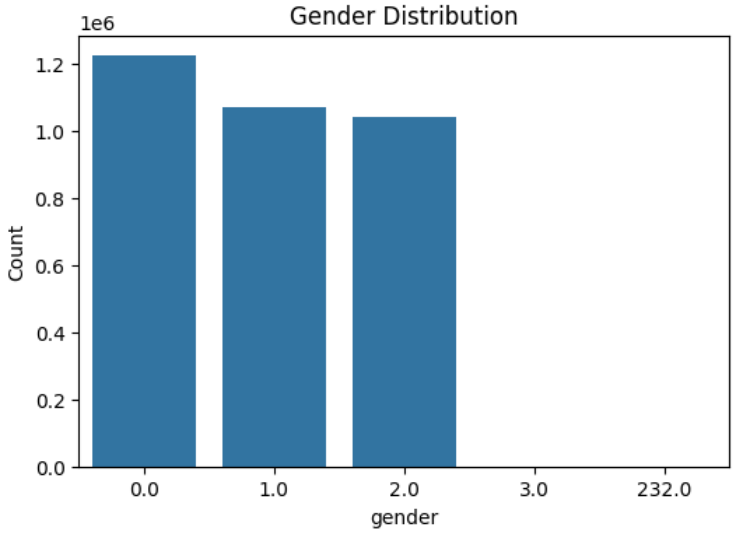
```
user_df['school_length'] = user_df['school'].apply(lambda x: len(x) if x is not None else 0)

user_df['school_length'].describe()
```
✓ 1.3s Python

```
count    3.330294e+06
mean     1.137576e+01
std      1.756154e+01
min      0.000000e+00
25%      0.000000e+00
50%      0.000000e+00
75%      2.400000e+01
max      1.740000e+02
Name: school_length, dtype: float64
```

Hình 8: Tạo một cột "school_length" để phân tích độ dài mỗi sample của cột



Hình 9: Trực quan hóa độ dài của sample cột "school"



Hình 10: Trực quan hóa phân bố các giá trị của cột "gender"

c) Bảng teacher.json

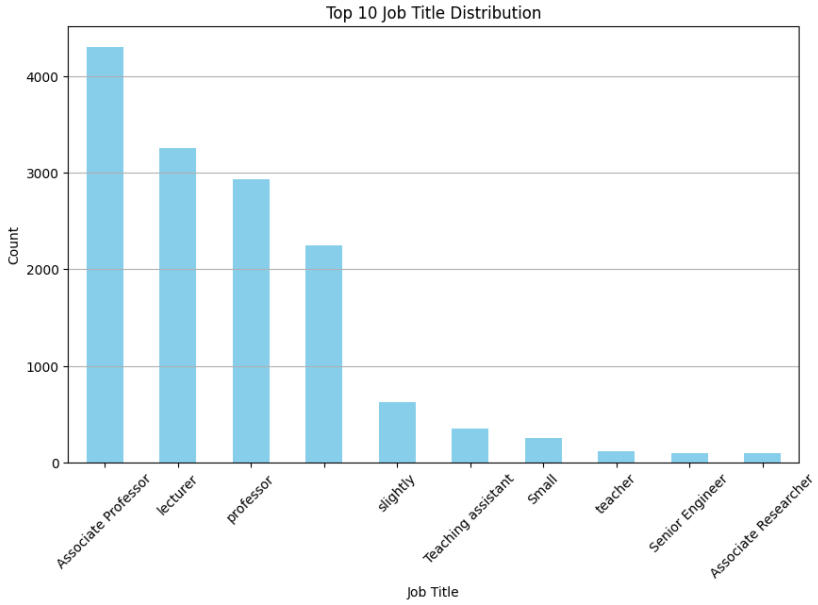
Sau đây là các thống số cơ bản của bảng

	id	name	name_en	about	job_title	org_name
count	17018	17018	17018	13893	14768	17018
unique	17018	13967	11061	12536	1323	998
top	T_1	顾礼平		slightly	Associate Professor	Tsinghua University
freq	1	20	4142	626	4305	1140

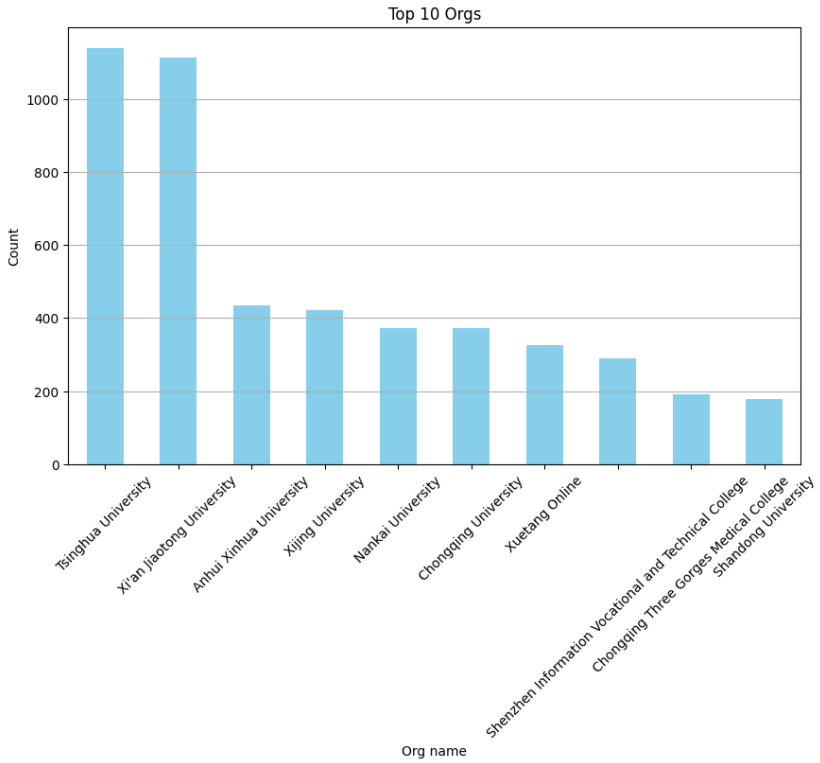
	0
id	object
name	object
name_en	object
about	object
job_title	object
org_name	object

dtype: object

Tham khảo phân phối của top 10 tên việc xuất hiện nhiều nhất trong bảng



Tham khảo phân phối của top 10 tổ chức xuất hiện nhiều nhất trong bảng

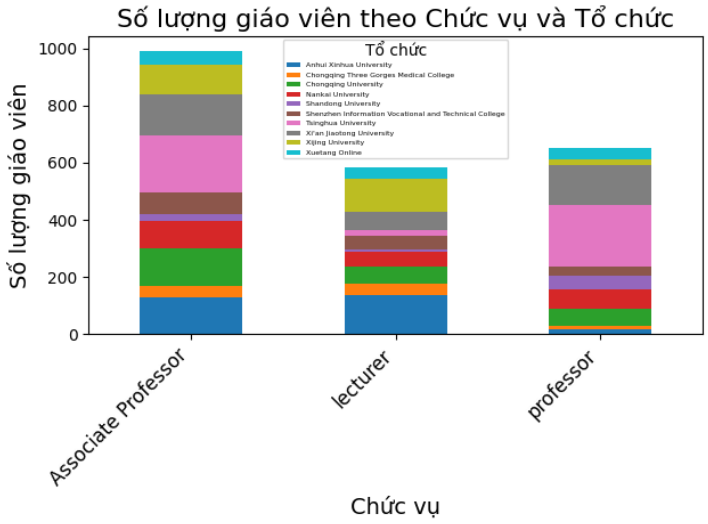


Ta thực hiện phân tích mối quan hệ giữa ba chức vụ (job titles) có số lượng giáo viên nhiều nhất và mười tổ chức (organizations) có số lượng giáo viên cao nhất

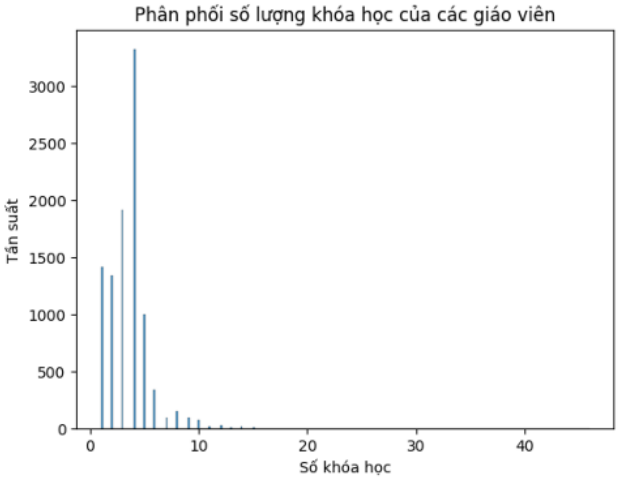
```
print("Bảng tần suất giữa job_title và org_name:")
contingency_table
```

Bảng tần suất giữa job_title và org_name:

org_name	Anhui Xinhua University	Chongqing Three Gorges Medical College	Chongqing University	Nankai University	Shandong University	Shenzhen Information Vocational and Technical College	Tsinghua University	Xi'an Jiaotong University	Xijing University	Xuetang Online
job_title										
Associate Professor	130	39	130	97	23	78	199	144	105	47
lecturer	136	42	58	52	10	48	19	62	117	41
professor	16	14	57	68	49	33	217	138	19	40

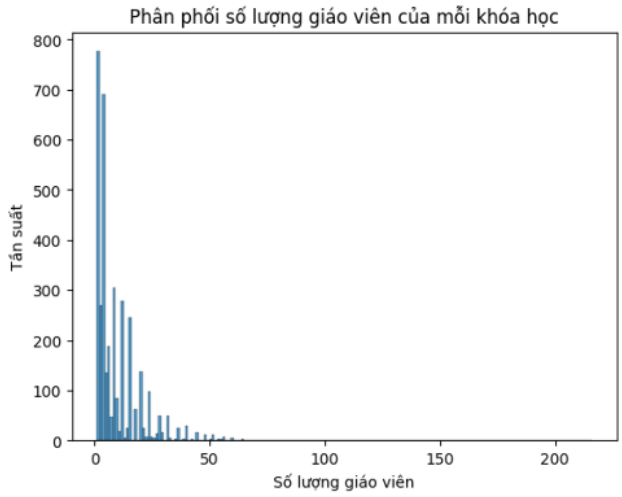


Sau khi lọc bỏ các liên kết có khóa học hoặc teacher không tồn tại dựa vào file course-teacher.txt, số hàng còn lại là 35593. Các thông tin được trực quan hóa như sau



	count
count	9876.000000
mean	3.603989
std	2.322770
min	1.000000
25%	2.000000
50%	4.000000
75%	4.000000
max	46.000000

Hình 11: Histogram thể hiện số lượng khóa học của mỗi teacher và bảng thống kê mô tả tương ứng



	count
count	3673.000000
mean	9.690444
std	11.383169
min	1.000000
25%	3.000000
50%	5.000000
75%	12.000000
max	216.000000

Hình 12: Histogram thể hiện số lượng teacher của mỗi khóa học và bảng thống kê mô tả tương ứng

d) Bảng school.json

Ta đếm dữ liệu ở từng cột, đếm các giá trị đặc biệt, giá trị xuất hiện nhiều nhất với tần số của nó:

```
df.describe(include='all')
```

Python

	id	name	name_en	sign	about	motto
count	428	428	428	428	428	428
unique	428	421	423	420	420	138
top	S_1	长安大学	Dalian University of Technology	hzic	Hebei Normal University Of Science & Technolog...	
freq	1	2	2	2	2	282

Kiểm tra kiểu dữ liệu của từng cột:

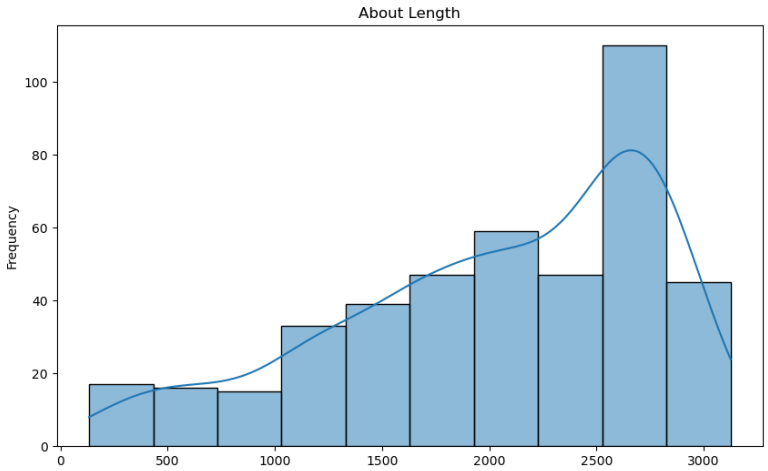
```
[99]  pd.DataFrame(df.info())

...   <class 'pandas.core.frame.DataFrame'>
      RangeIndex: 428 entries, 0 to 427
      Data columns (total 6 columns):
       #   Column   Non-Null Count  Dtype
      ---  ------   --------------  -----
       0   id       428 non-null    object
       1   name     428 non-null    object
       2   name_en  428 non-null    object
       3   sign     428 non-null    object
       4   about    428 non-null    object
       5   motto    428 non-null    object
      dtypes: object(6)
      memory usage: 20.2+ KB
```

Ta tạo 2 cột mới là "about_length" và "motto_length" để lần lượt thể hiện độ dài của giá trị dữ liệu ở 2 cột "about" và "motto":

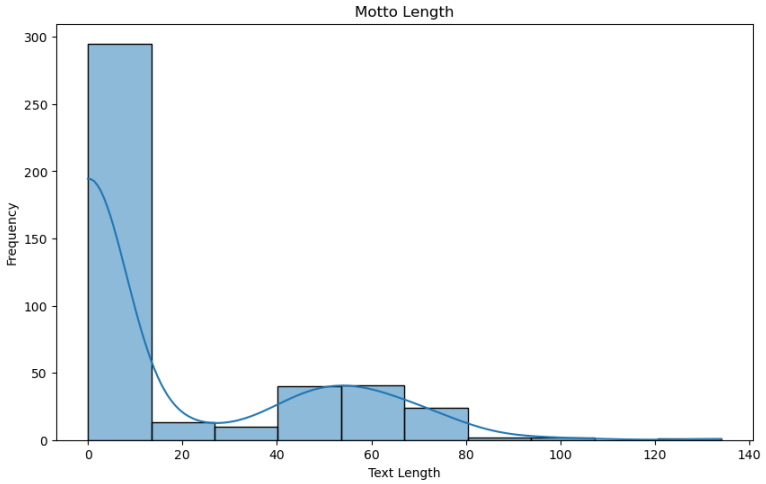
```
00]  lengths_df = pd.DataFrame({
         'about_length': df['about'].apply(len),
         'motto_length': df['motto'].apply(len)
     })
     # Display summary
     lengths_df[['about_length', 'motto_length']].describe()
```

	about_length	motto_length
count	428.000000	428.000000
mean	2023.530374	16.955607
std	737.761007	26.937787
min	134.000000	0.000000
25%	1552.750000	0.000000
50%	2157.000000	0.000000
75%	2647.750000	42.000000
max	3126.000000	134.000000

Có 2 cột ta cần là “about_length” và “motto_length” để ta tìm phân bố độ dài của giá trị lên đồ thị:



Dựa vào biểu đồ ta có thể nhận xét rằng mô tả của các trường đều rất chi tiết, số lượng trường với số lượng từ phần mô tả > 2000 chiếm phần lớn. Tuy nhiên thông tin này có vẻ không hữu ích với hệ thống khuyến nghị.



Hầu hết các trường đại học đều có một khẩu hiệu ngắn gọn dưới 20 từ vì chủ yếu khẩu hiệu sẽ đơn giản nhất có thể để truyền đạt tầm nhìn và mục tiêu của trường một cách trực tiếp ngắn gọn, đọng lại trong trí nhớ người xem. Một phần nhỏ hơn các trường có khẩu hiệu tương đối dài với 40 đến 88 chữ.

e) Bảng course-field.json

```python
# 1. Số lượng khóa học
num_courses = df['course_id'].nunique()
print(f"Số lượng khóa học: {num_courses}")

# 2. Số lượng các lĩnh vực khác nhau
unique_fields = set(field for fields_list in df['field'] for field in fields_list)
num_unique_fields = len(unique_fields)
print(f"Số lượng các lĩnh vực khác nhau: {num_unique_fields}")
```

✓ 0.0s Python

```
Số lượng khóa học: 632
Số lượng các lĩnh vực khác nhau: 82
```

Hình 13: Tổng số lượng khóa học và tổng số lượng các lĩnh vực khác nhau

```python
# 3. Phân bố số lượng khóa học theo từng lĩnh vực
field_distribution = df.explode('field')['field'].value_counts()
print("\nPhân bố số lượng khóa học theo từng lĩnh vực:")
print(field_distribution)
```

✓ 0.0s Python

```
Phân bố số lượng khóa học theo từng lĩnh vực:
field
Computer Science and Technology                   75
foreign languages and literature                  43
Art                                               38
Chinese language and literature                   26
Management Science and Engineering                25
                                                  ..
Battle Science                                     1
Military Logistics and Military Equipment Science  1
Weapons Science and Technology                     1
Army Command Science                               1
Mining Engineering                                 1
Name: count, Length: 82, dtype: int64
```

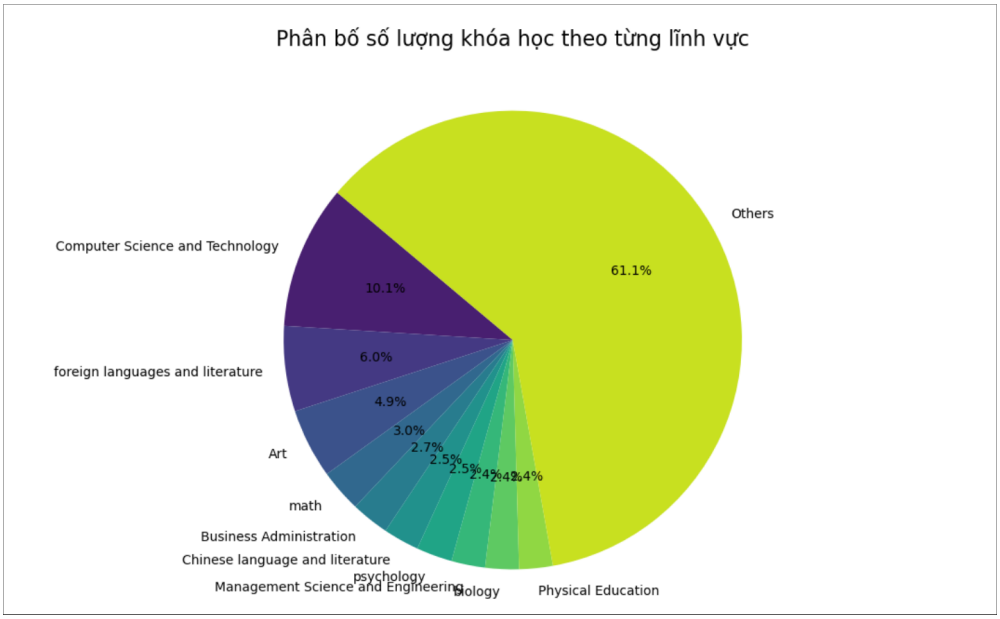
Hình 14: Phân bố số lượng khóa học theo từng lĩnh vực

```python
# 4. Phân bố độ dài tên khóa học (số ký tự)
course_name_length = df['course_name'].apply(len)
print("\nThống kê độ dài tên khóa học:")
print(course_name_length.describe())
```

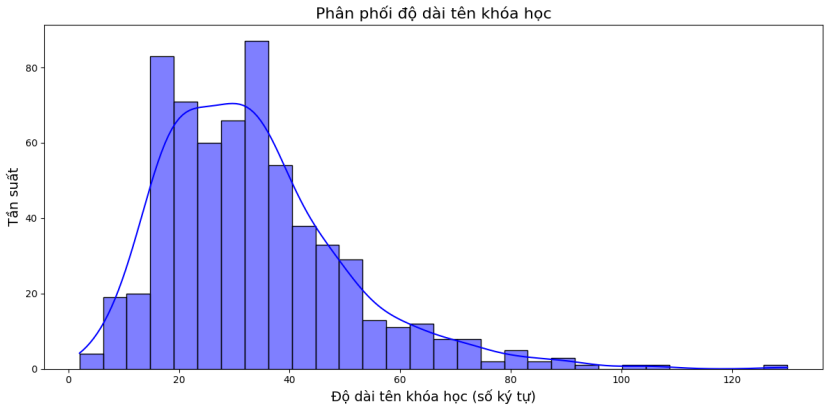
0.0s Python

```
Thống kê độ dài tên khóa học:
count    632.000000
mean      33.507911
std       16.882082
min        2.000000
25%       21.000000
50%       31.000000
75%       41.000000
max      130.000000
Name: course_name, dtype: float64
```

Hình 15: Phân bố độ dài tên khóa học



Hình 16: Biểu đồ thanh thể hiện sự phân bố số lượng khóa học theo từng lĩnh vực



Hình 17: Biểu đồ phân phối cho độ dài tên khóa học

1.2.3. Làm sạch dữ liệu

a) Bảng course.json

Ta kiểm tra dữ liệu thiếu, dữ liệu không nhất quán, dữ liệu trùng lặp và dữ liệu trống:

Đầu tiên ta thấy được có 647 giá trị ở cột "name_trans" bị trùng lặp cho dù id không bị trùng, chứng tỏ có sự lỗi nhất định trong bộ dữ liệu, cũng như nãy đã thống kê ta thấy được có rất nhiều giá trị trống ở cột "field_trans".

Ta kiểm tra kĩ hơn về các dòng có giá trị trong cột "name" bị trùng lặp:

	NaN values	NA values	Duplicated rows	Empty values
id	0	0	0	0
name_trans	0	0	224	0
field_trans	0	0	618	603
prerequisites_trans	0	0	459	413
about_trans	0	0	52	13
resource	0	0	0	0
course_name	603	603	607	0
field	603	603	618	0

	id	name_trans	field_trans	prerequisites_trans	about_trans	resource	course_name	field
1490	769273	introduction to the basic principles of marxism	(political science,)		"basic principles of marxism" educates college...	{'titles': ['导论', '1.1 马克思，何许人也？ ', 'Video'], ...	马克思主义基本原理概论（2019春）	[政治学]
1598	837985	introduction to the basic principles of marxism	()	"ideological and moral cultivation and legal b...	why is marx right? what are "universal values"...	{'titles': ['专题一：为什么是马克思？ ', '1.1 为什么是马克思主义？ ',...	NaN	NaN
2826	1891061	introduction to the basic principles of marxism	()		the basic principles of marxism are the basic ...	{'titles': ['绪论', '1.青年马克思', '青年马克思'], 'resou...	NaN	NaN
3757	2342515	introduction to the basic principles of marxism	()	ideological and moral cultivation and legal fo...	the course "introduction to the basic principl...	{'titles': ['绪论', None, '序言'], 'resource_id':...	NaN	NaN

Ta thấy được đa số dữ liệu trong này cột "field" đa số bị trống và trùng lặp, cũng như các cột khác không có ý nghĩa hoặc trùng với các cột khác, thực hiện chi square test, ta có được kết quả với P-value rất thấp, chứng tỏ các giá trị phụ thuộc với nhau chứ không hề có giá trị mới. Chứng tỏ ta có thể xoá được các dòng dữ liệu này, cũng như các khoá học không tồn tại trong "course-field.json".

b) Bảng user.json

Ta thấy cột "year_of_birth" bị thiếu dữ liệu hơn 97% trong khi các cột còn lại tỉ lệ % thiếu là rất thấp. Ta tiến hành loại bỏ cột này, sau đó ta sẽ tiến hành xử lý dữ liệu nhiễu trên cột gender với 2 giá trị nhiễu là 232 và 3

c) Bảng concept.json

Xử lý dữ liệu thiếu giúp cải thiện độ chính xác của mô hình, đảm bảo tính toàn vẹn của phân tích, tránh lỗi tính toán và giảm độ thiên lệch. Một số cách xử lý phổ biến gồm:

- Loại bỏ hàng/cột: Áp dụng khi dữ liệu thiếu quá nhiều.
- Điền giá trị thay thế: Điền trung bình, trung vị, hoặc giá trị dự đoán vào chỗ thiếu.
- Dùng mô hình dự đoán: Áp dụng các thuật toán để dự đoán giá trị thiếu.

Việc xử lý phù hợp giúp dữ liệu chính xác và đáng tin cậy hơn trong phân tích và dự đoán.

```
print("Số lượng giá trị thiếu trong từng cột:")
print(df.isnull().sum())
df = df.dropna()
print("Số lượng thiếu sau khi xử lý:", df.isna().sum().sum())
```

✓ 0.3s

```
Số lượng giá trị thiếu trong từng cột:
id         207
name         0
context      0
dtype: int64
Số lượng thiếu sau khi xử lý: 0
```

Xử lý dữ liệu trùng lặp là bước quan trọng trong tiền xử lý dữ liệu nhằm loại bỏ các bản ghi trùng lặp để đảm bảo tính chính xác và hiệu quả của mô hình. Dữ liệu trùng lặp có thể gây sai lệch và làm chậm quá trình xử lý.
Các phương pháp xử lý dữ liệu trùng lặp phổ biến bao gồm:

- Xóa các bản ghi trùng lặp: Loại bỏ các hàng hoàn toàn trùng lặp trong DataFrame bằng hàm drop_duplicates() trong Pandas.
- Giữ lại bản ghi đầu tiên hoặc cuối cùng: Nếu cần giữ lại một bản ghi đại diện, có thể chỉ xóa các bản trùng lặp sau hoặc trước.
- Xác định tiêu chí trùng lặp: Tìm và xóa bản ghi trùng lặp dựa trên một số cột cụ thể thay vì toàn bộ hàng.

Loại bỏ dữ liệu trùng lặp giúp dữ liệu trở nên nhất quán, giảm dung lượng và cải thiện độ chính xác của phân tích và mô hình.
Loại bỏ dữ liệu với phương thức drop_duplicates():

3.2 Xử lý dữ liệu trùng lặp

```
print("Số lượng bản ghi trùng lặp:", df.duplicated().sum())
duplicates = df[df.duplicated(keep=False)]
display(duplicates)
df = df.drop_duplicates()
print("Số lượng trùng lặp sau khi xử lý:", df.duplicated().sum())
```

✓ 4.7s

```
Số lượng bản ghi trùng lặp: 11543
```

	id	name	context
16	K_Dorsal digital vein_Human anatomy	Dorsal digital vein	[]
240	K_Hilar lymph node tuberculosis_Tuberculosis	Hilar lymph node tuberculosis	[]
246	K_Dynamic stability_Ship engineering	Dynamic stability	[]
287	K_Vesicouterine fistula_Urology	Vesicouterine fistula	[]
289	K_Vesicouterine fistula_Urology	Vesicouterine fistula	[]
...	...	...	...
637335	K_Genetically modified corn_Food Science and E...	Genetically modified corn	[]
637374	K_Meringue_Food Science and Engineering	Meringue	[]
637376	K_Hip fracture_Food Science and Engineering	Hip fracture	[]
637527	K_Esters_Food Science and Engineering	Esters	[]
637570	K_Genetically modified corn_Food Science and E...	Genetically modified corn	[]

21106 rows × 3 columns

```
Số lượng trùng lặp sau khi xử lý: 0
```

d) Bảng course-field.json

Sử dụng isnull().sum() để tính số lượng giá trị thiếu trong từng cột. Sau đó loại bỏ hàng chứa giá trị thiếu bằng cách sử dụng dropna()

```
Số lượng giá trị thiếu trong từng cột:
course_id      0
course_name    0
field          0
dtype: int64

Tỷ lệ dữ liệu thiếu trong từng cột (%):
course_id      0.0
course_name    0.0
field          0.0
dtype: float64

Số lượng giá trị thiếu sau khi xử lý:
course_id      0
course_name    0
field          0
dtype: int64
```

Dữ liệu văn bản thường chứa nhiều thông tin nhiễu chẳng hạn như các ký tự không mong muốn: Các ký tự đặc biệt, dấu câu, hoặc ký tự không phải chữ cái có thể làm giảm chất lượng phân tích. Ở đây chúng ta sẽ tiến hành loại bỏ các ký tự không cần thiết, các khoảng trắng dư thừa và thường hóa các ký tự viết hoa

Để kiểm tra dữ liệu trùng lặp, chúng ta sử dụng phương thức duplicated() trong pandas. Đầu tiên xác định các bản ghi trùng lặp, sau đó đếm số lượng và hiển thị các bảng ghi trùng lặp đó. Sau đó tiến hành xóa bản ghi trùng lặp bằng cách sử dụng drop_duplicates()

```python
# Chuyển đổi cột 'field' thành chuỗi
df['field'] = df['field'].apply(lambda x: ', '.join(x))

# Kiểm tra dữ liệu trùng lặp
duplicate_rows = df.duplicated()

# Đếm số lượng bản ghi trùng lặp
num_duplicates = duplicate_rows.sum()
print(f"Số lượng bản ghi trùng lặp: {num_duplicates}")

# Hiển thị các bản ghi trùng lặp
if num_duplicates > 0:
    print("Các bản ghi trùng lặp:")
    print(df[duplicate_rows])

# Xóa bản ghi trùng lặp (nếu cần)
df_cleaned = df.drop_duplicates()

# Kiểm tra lại số lượng bản ghi sau khi xóa trùng lặp
print(f"Số lượng bản ghi sau khi xóa trùng lặp: {len(df_cleaned)}")
```

✓ 0.0s Python

```
Số lượng bản ghi trùng lặp: 0
Số lượng bản ghi sau khi xóa trùng lặp: 632
```

e) Bảng school.json

Ta xoá cột "name" đi vì trùng với ý nghĩa với cột "name_en" (tên nhưng trong Tiếng Anh)

Ta thống nhất cột "sign" (kí hiệu đại diện cho trường) đều là tất cả in hoa:

```
df['sign'] = df['sign'].str.upper() ## convert to uppercase for consistent
df['sign'].value_counts()
```

[113]

```
sign
XJTU        2
HZIC        2
TJU         2
QDU         2
ECUST       2
           ..
PASTEURX    1
DELFTX      1
RICE        1
BURGUNDYX   1
HLJUCM      1
Name: count, Length: 418, dtype: int64
```

Vì ở đây tên trường ("name_en") cũng như kí hiệu ("sign") là chìa khoá chính, hay nói cách khác là giá trị duy nhất nên không thể có dòng trùng với nhau, ta tiến hành xoá các dòng trùng giá trị:

Xử lí dữ liệu trùng lặp

```
df.drop_duplicates(subset=['name_en'], keep='first', inplace=True)
df.drop_duplicates(subset=['sign'], keep='first', inplace=True)
```

4]

```
name_en_counts = df['name_en'].value_counts()
name_en_counts[ name_en_counts > 1]
```

0]

```
Series([], Name: count, dtype: int64)
```

```
sign_counts = df['sign'].value_counts()
sign_counts[ sign_counts > 1]
```

6]

```
Series([], Name: count, dtype: int64)
```

g) Bảng teacher.json

Ở đây có cột name_en bị thiếu nên điền vào cột đó bằng cách lấy phiên âm của cột name là được. Để làm việc này có thể sử dụng thư viện pypinyin để lấy phát âm dùng cho tên tiếng anh.

```python
from pypinyin import pinyin, Style
from functools import reduce

def get_reading(name):
    return ' '.join(r[0] for r in pinyin(name, style=Style.NORMAL))

print(f'Example name translation: "陈怡" to "{get_reading('陈怡')}"')

print("Before filling missing data:")
missing_data = df.isna().any(axis=1)
display(df[missing_data].head())

print("After filling missing data:")
df['name_en'] = df['name_en'].fillna(df['name'].apply(get_reading))
df[missing_data].head()
```

✓ 1.3s

```
Example name translation: "陈怡" to "chen yi"
Before filling missing data:
```

	id	name	name_en	about	job_title	org_name
3	T_4	谢维和	None	谢维和，博士、教授、博士生导师、教育研究院院长。研究方向：教育学原理、教育社会学、高等教育和...	教授	清华大学
5	T_6	王孙禺	None	王孙禺，汉族，教授、博士生导师，出生于1947年10月，曾任清华大学人文社会科学学院党委书记...	教授	清华大学
6	T_7	袁本涛	None	袁本涛，博士、教授、博士生导师，现任教育研究院副院长，主要研究领域为高等教育政策，高等教育管...	教授	清华大学
7	T_8	林健	None	林健，福建福州人，英国Lancaster大学管理学博士、博士后，清华大学公共管理学教授、博士...	教授	清华大学
8	T_9	程建钢	None	程建钢，博士、教授、博士生导师，教育技术学学科负责人暨学术带头人， 中国教育技术协会学术委员...	教授	清华大学

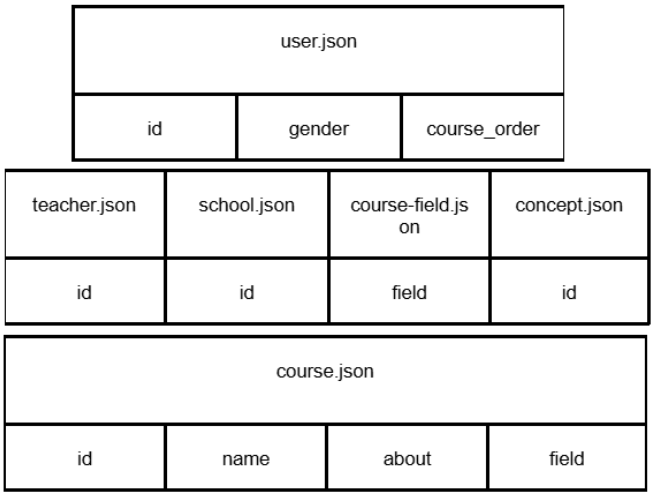
```
After filling missing data:
```

	id	name	name_en	about	job_title	org_name
3	T_4	谢维和	xie wei he	谢维和，博士、教授、博士生导师、教育研究院院长。研究方向：教育学原理、教育社会学、高等教育和...	教授	清华大学
5	T_6	王孙禺	wang sun yu	王孙禺，汉族，教授、博士生导师，出生于1947年10月，曾任清华大学人文社会科学学院党委书记...	教授	清华大学
6	T_7	袁本涛	yuan ben tao	袁本涛，博士、教授、博士生导师，现任教育研究院副院长，主要研究领域为高等教育政策，高等教育管...	教授	清华大学
7	T_8	林健	lin jian	林健，福建福州人，英国Lancaster大学管理学博士、博士后，清华大学公共管理学教授、博士...	教授	清华大学
8	T_9	程建钢	cheng jian gang	程建钢，博士、教授、博士生导师，教育技术学学科负责人暨学术带头人， 中国教育技术协会学术委员...	教授	清华大学

1.2.4. Chuyển đổi dữ liệu

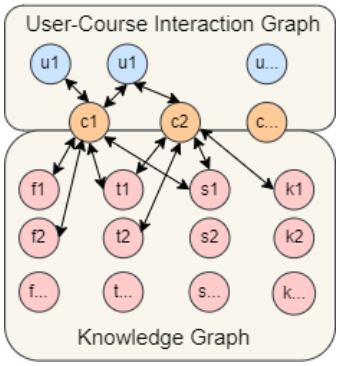
Feature Engineering: Nhóm sẽ chọn các bảng và thuộc tính có thể sử dụng để tạo ra feature các mô hình khuyến nghị dựa trên bộ dữ liệu đã xử lý và làm sạch trước đó:

Các bảng được chọn và thuộc tính sử dụng:



- Với ‘user.json’: ‘course_order’ gồm các khóa học mà user đã đăng ký với khóa học sau cùng là khóa học gần đây nhất, dùng để tạo liên kết giữa ‘user.json’ và ‘course.json’.
- Với ‘course.json’: Đây là table quan trọng chứa thông tin về các khóa học như ‘name’, ‘about’ và ‘field’.
- Với ‘teacher.json’, ‘school.json’: dùng để tạo relation với ‘course.json’ chứa thông tin về trường tổ chức khóa học và giáo viên giảng dạy.
- Với ‘course-field.json’: chứa các field của mỗi khóa học, dùng để kiểm tra với trường ‘field’ trong ‘course.json’.
- Với ‘concept.json’: id theo quy ước ‘K_concept namefield’, tạo thêm feature concept-name_field với mỗi khoá học.

Tạo knowledge graph:



Tạo interaction giữa người dùng với khóa học: sử dụng 5-core filtering, lọc người dùng với ít hơn 5 khóa học và những khóa học có số lượng đăng ký dưới 5.

Kết quả: Vì data đã được xử lý trước đó nên ta thấy không có thay đổi đáng kể

Trước khi filter	Sau khi filter
1.183.774 interactions	1.182.745 interactions

Tạo relation giữa các entities: course-relation-attribute. Sau đó ta tiến hành lọc theo tiêu chí, số lần course xuất hiện tối thiểu là 5 và số lần xuất hiện tối thiểu của một relation là 25.

Kết quả:

Trước khi filter	Sau khi filter
376.093 interactions	71.787 interactions

1.3. Phân tích vấn đề

Hệ thống học tập trực tuyến MOOC cung cấp số lượng lớn các khóa học đa dạng, nhưng khó khăn lớn đối với người học là tìm kiếm khóa học phù hợp với sở thích và nhu cầu cá nhân. Để giải quyết vấn đề này, hệ thống khuyến nghị khóa học được phát triển nhằm cá nhân hóa trải nghiệm học tập cho từng người dùng dựa trên dữ liệu về hành vi học tập và các đặc điểm cá nhân.

Bài toán đặt ra trong dự án này là: **Làm thế nào để xây dựng một hệ thống khuyến nghị khóa học cá nhân hóa cho từng người học trên nền tảng MOOC?**

1.3.1. Câu hỏi nghiên cứu

- Làm thế nào để dự đoán chính xác các khóa học mà một người dùng có khả năng sẽ đăng ký tiếp theo?
- Làm sao tận dụng các đặc điểm của người dùng như giới tính, độ tuổi, trường học, và lịch sử khóa học để tăng độ chính xác của mô hình khuyến nghị?
- Làm sao đánh giá chất lượng các gợi ý khóa học và xác định mức độ hiệu quả của hệ thống (metric đánh giá là gì) ?

1.3.2. Kết quả đề tài

Dự án hướng tới xây dựng một hệ thống khuyến nghị khóa học hiệu quả, dựa trên dữ liệu của người học từ bộ MOOCCubeX. Kết quả mong muốn bao gồm:

- **Xác định yếu tố ảnh hưởng đến việc đăng ký khóa học:** Khám phá các đặc điểm người dùng (giới tính, trường học, năm sinh, các khóa học đã đăng ký...) ảnh hưởng đến hành vi chọn khóa học. Điều này giúp hệ thống có cái nhìn rõ ràng hơn về các yếu tố quan trọng khi gợi ý khóa học.
- **Khả năng khuyến nghị khóa học cá nhân hóa:** Kỳ vọng hệ thống sẽ đưa ra những gợi ý chính xác cho từng người học, dựa trên hành vi đăng ký khóa học trước đây và các yếu tố liên quan. Mục tiêu là hệ thống có thể dự đoán tốt các khóa học mà người dùng có khả năng quan tâm trong tương lai.
- **Định hướng cải thiện trải nghiệm học tập:** Hệ thống khuyến nghị dự kiến sẽ giúp người học tiết kiệm thời gian tìm kiếm, đồng thời cung cấp cho họ trải nghiệm học tập tốt hơn thông qua việc gợi ý các khóa học phù hợp với mục tiêu và sở thích cá nhân.
- **Đánh giá các phương pháp tiếp cận:** Tìm hiểu các mô hình Recommendation System và thử nghiệm với bộ dữ liệu để so sánh độ hiệu quả của các mô hình.

1.4. Khả năng ứng dụng

Hệ thống khuyến nghị này có tiềm năng ứng dụng rộng rãi trong các nền tảng học tập trực tuyến. Cụ thể:

- **Cá nhân hóa trải nghiệm học tập:** Giúp người dùng nhanh chóng tìm được các khóa học phù hợp với mục tiêu học tập và sở thích cá nhân. Tăng cường trải nghiệm người dùng.
- **Thu hút người dùng:** Các gợi ý chính xác và kịp thời có thể dẫn đến tỷ lệ đăng ký khóa học cần thiết cao hơn và cải thiện sự gắn bó của người dùng

với nền tảng. Các khóa học phù hợp và hấp dẫn có thể giúp giảm tỷ lệ người học từ bỏ giữa chừng, cải thiện tỷ lệ hoàn thành khóa học.

- **Nâng cao hiệu suất học tập của người dùng:** Từ những hành vi học tập của người dùng trong quá khứ, hệ thống sẽ căn cứ vào và tự động đề xuất các khóa học tương thích nhất với khả năng và kỹ năng của người học để tối ưu hóa nhất hiệu suất học tập của người dùng.