Xây dựng hệ khuyến nghị cho video game

Hồ Đình Long

18521022

KHDL-2018 UIT, Tp.HCM, Việt Nam

18521022@gm.uit.edu.com

Đỗ Hùng Dũng

18520629

KHDL-2018 UIT, Tp.HCM, Việt Nam

18520629@gm.uit.edu.vn

Trần Cao Phát

18521233

KHDL-2018 UIT, Tp.HCM, Việt Nam

18521233@gm.uit.edu.vn

Tóm tắt—Trong bài báo cáo này, chúng tôi sử dụng một bộ dữ liệu đánh giá về video game trên amazon. Thực hiện các phương pháp để đề xuất những video cho người dùng dựa trên lọc cộng tác và nội dung. Kết quả cho ra khá cao.

Từ khoá: Hệ khuyến Nghị, lọc cộng tác, lọc dựa trên nội dung, trò chơi điện tử

I. GIỚI THIỆU

Hệ khuyến nghị là một trong những ứng dụng phổ biến trong khoa học dữ liệu. Nhiều công việc, đề tài đã được thực hiện về chủ đề này. Mối quan tâm và nhu cầu ngày càng cao vì tốc độ phát triển nhanh chóng của internet và vấn đề quá tải thông tin. Giúp người dùng đối phó với tình trạng này và cung cấp các đề xuất, nội dung và các dịch vụ cá nhân trở nên cần thiết đối với các doanh nghiệp và các nền tảng trực tuyến.

Trò chơi điện tử hay còn được gọi là video game, là một dạng giải trí đối với con người sau những giờ học, giờ làm căng thẳng, mệt mỏi. Nó được sáng tạo bởi những người tài giỏi, thông minh, có trí óc tưởng tượng phong phú. Với đa dạng thể loại và số lượng khổng lồ, thế giới trò chơi điện tử đang dần trở nên rộng lớn hơn bao giờ hết, khiến cho người chơi khó có thể tìm được một trò chơi ưng ý theo sở thích của mình mà không có sự trợ giúp từ hệ thống.

Hiểu được vấn đề này, nhóm chúng tôi đã quyết định tiến hành xây dựng một mô hình hệ khuyến nghị để nhận biết từ sở thích của người dùng, những dữ liệu từ người chơi tương tự, thông tin từ các trò chơi mà đưa ra cho họ những gợi ý lựa chọn ưng ý nhất.

- Input: thông tin của các video game, lượt rating của người dùng.
- Output: các video game được khuyến nghị cho người dùng.

II. BỘ DỮ LIỆU

A. Giới thiệu bộ dữ liệu

Trong bài này, chúng tôi sử dụng bộ dữ liệu có tên: "Video games", được lấy từ trang web http://jmcauley.ucsd.edu.

Bộ dữ liệu là những thông tin và đánh giá của người dùng đối với những trò chơi điện tử tương ứng.

B. Thông tin bộ dữ liệu

- Tên bộ dữ liệu: Video games
- Nguồn: Amazon review data (ucsd.edu)
- Tác giả: Julian McAuley, UCSD
- Link download bộ dữ liệu :

https://drive.google.com/drive/folders/1WjV64HYv20CYD5PsOyPkLx6xNSiy5KSm?usp=sharing

- Tập dữ liệu: Gồm 2 tập dữ liệu là Ratings và Details:

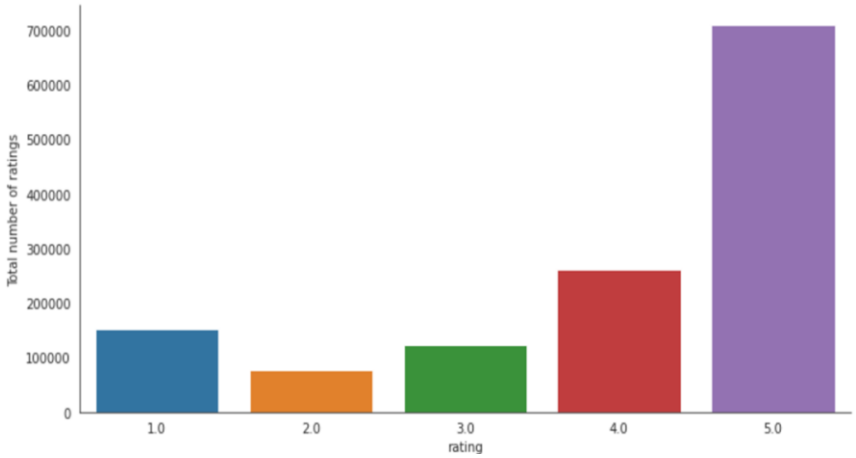
  - Ratings:
    + Số thuộc tính: 4
      - ➢ UserId: Id của người dùng
      - ➢ Ratings: Điểm đánh giá trò chơi của người dùng
      - ➢ ProductId: Id của trò chơi
      - ➢ Timestamp
    + Tổng số lượt đánh giá : 1324753
    + Tổng số người dùng: 826767
    + Tổng số video game: 50210
  - Details:
    + Số thuộc tính: 4
      - ➢ ProductId: Id của trò chơi
      - ➢ Title: Tên của trò chơi
      - ➢ Brand: Nhà phát hành

➢ Category: Các phân loại của trò chơi

+ Tổng số video game: 68137

A .Phân tích bộ dữ liệu:

- Lượt đánh giá:



Hình 1: Phân bố lượt đánh giá

+ Bộ dữ liệu phân bổ đánh giá từ 1 tới 5 sao, thang điểm tỉ lệ thuận với mức độ ưa thích của người dùng đối với trò chơi mà họ đánh giá.
+ Có thể thấy, phân bố dữ liệu của lượt đánh giá không đồng đều. Đa số đánh giá được nghiêng về hướng tích cực với lượng điểm đánh giá khá cao. Lượng đánh giá tiêu cực chiếm phần nhỏ, trong khi những đánh giá trung tính còn có số lượng thấp hơn. Cho thấy rằng, phần lớn người dùng đều ưa thích trò chơi mà họ đánh giá.

- Kích thước dữ liệu:

+ Với tổng số lượt đánh giá lên tới 1324753, khiến việc huấn luyện mô hình gặp khó khăn. Vì vậy, nhóm đã quyết định tiến hành chỉ lấy dữ liệu của những user đã đánh giá từ 20 video game trở lên và những video game có 15 lượt đánh giá trở lên để giúp quá trình huấn luyện trở nên suôn sẻ.

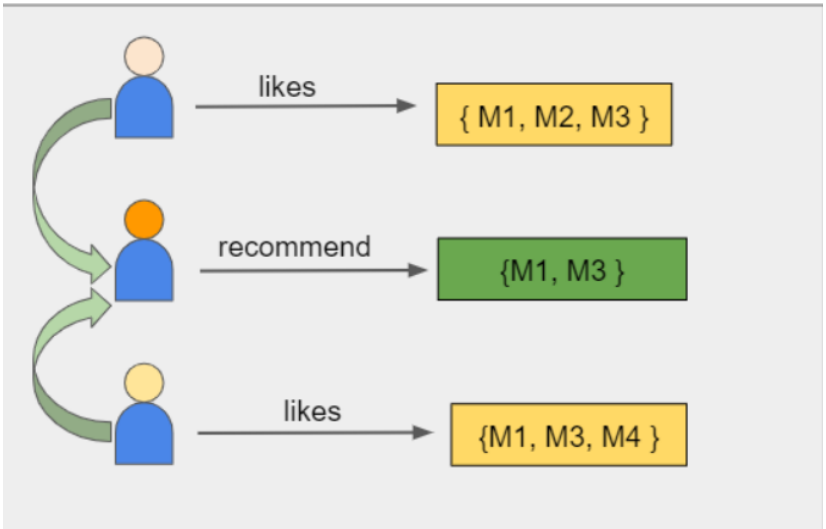
III. PHƯƠNG PHÁP

Trong bài này, chúng tôi sử dụng hai phương pháp chính là lọc cộng tác và lọc dựa trên nội dung để đề xuất cho người dùng.

A .Lọc cộng tác

Với lọc cộng tác, hệ thống sẽ dự đoán xếp hạng hoặc sở thích của người dùng dựa trên xếp hạng trước đây hoặc sở thích của những người dùng khác. Lọc cộng tác không yêu cầu dữ liệu lớn.

Ví dụ: nếu người dùng A xem M1, M2 và M3 và người dùng B xem M1, M3, M4, chúng tôi đề xuất M1 và M3 cho người dùng tương tự C.



Hình 2: Sơ đồ đơn giản của lọc cộng tác

Có 2 loại lọc cộng tác : Dựa trên người dùng (user-based) và dựa trên sản phẩm (item-based).

User-based: tính toán độ tương đồng giữa giữa nhóm người dùng. Từ đó đưa ra đề xuất cho một người dùng dựa trên những người dùng khác trong nhóm.

Hướng tiếp cận này được thực hiện như sau:

- Biểu diễn mỗi user bằng một vector thuộc tính được xây dựng từ những feedback trong quá khứ của user với các item. Từ đó, tính toán độ tương đồng giữa các user.
- Để tính toán độ yêu thích của user U với một item I, ta sẽ lựa chọn ra k users đã từng đánh giá I và có độ tương đồng với user U là cao nhất. Sau đó, dựa vào những feedback của k user đó với item I để tính toán ra kết quả.

- Cuối cùng, lựa chọn những items được dự đoán user U yêu thích nhất để gợi ý cho U.

Item-based: Tương tự như User-based , phương pháp này sẽ tìm ra những nhóm item tương tự nhau. Sau đó, dự đoán mức độ yêu thích của user với item dựa trên độ yêu thích của user đó với các item khác cùng loại.

Hướng tiếp cận này được thực hiện như sau:

- Biểu diễn mỗi item bằng một vector thuộc tính. Từ đó, tính toán độ tương đồng giữa các item.
- Để tính toán độ yêu thích của user U với một item I, ta sẽ lựa chọn ra k items đã từng được U đánh giá và có độ tương đồng với I là cao nhất. Sau đó, dựa vào những feedback của U với k item đó để tính toán ra kết quả.
- Cuối cùng, lựa chọn những items được dự đoán user U yêu thích nhất để gợi ý cho U. Vì số lượng items thường nhỏ hơn nhiều so với số lượng users.

Ngoài ra, còn một phương pháp khác của lọc cộng tác mà chúng tôi sử dụng là phương pháp phân rã ma trận (Matrix factorization). Các hệ thống đề xuất lọc cộng tác dựa trên người dùng và dựa trên mục thường bị thưa thớt dữ liệu. Phương pháp phân rã ma trận giúp giải quyết những nhược điểm này của lọc cộng tác dựa trên bộ nhớ bằng cách giảm số chiều của ma trận xếp hạng . Khi nói đến giảm số chiều, SVD là một phương pháp phổ biến trong đại số tuyến tính để phân tích nhân tử ma trận trong học máy. Thu nhỏ kích thước không gian thứ N nguyên thành thứ K nguyên (trong đó K <N) và giảm số lượng đối tượng. SVD xây dựng một ma trận với hàng là người dùng và cột là các mục và các phần tử được đưa ra bởi xếp hạng của người dùng. SVD phân tách một ma trận thành ba ma trận khác và trích xuất các yếu tố từ việc phân tích nhân tử của ma trận cấp cao (người dùng-mục-xếp hạng).

$$A = USV^T$$

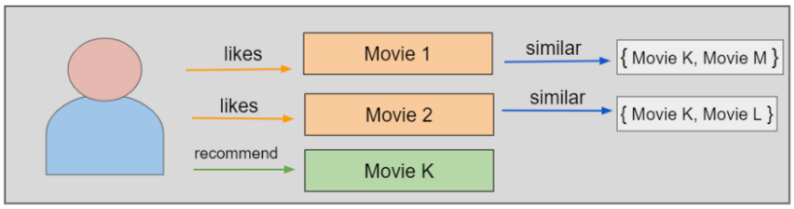
Ma trận U: ma trận số ít của (người dùng * các yếu tố tiềm ẩn)

Ma trận S: ma trận đường chéo (thể hiện sức mạnh của từng yếu tố tiềm ẩn)

Ma trận V: ma trận số ít của (mục * yếu tố tiềm ẩn)

B. Lọc dựa trên nội dung

Không giống như lọc cộng tác chỉ dựa vào tương tác giữa người dùng với sản phẩm, lọc dựa trên nội dung sử dụng hồ sơ của người dùng và sản phẩm hoặc dựa vào nội dung, thuộc tính của những item tương tự như item mà người dùng đã chọn trong quá khứ. Ý tưởng của phương pháp dựa trên nội dung là cố gắng xây dựng một mô hình, dựa trên các thuộc tính giải thích các tương tác giữa người dùng và sản phẩm được quan sát.



Hình 3: Sơ đồ đơn giản của lọc nội dung

Ví dụ: nếu người dùng xem một bộ phim hài có sự tham gia của diễn viên A, hệ thống sẽ giới thiệu cho họ những bộ phim cùng thể loại hoặc có sự tham gia của cùng một diễn viên hoặc cả hai. Đầu vào để xây dựng hệ thống đề xuất dựa trên nội dung là các thuộc tính phim.

D. Cài đặt thực nghiệm

Đầu tiên, chúng tôi tiến hành chia bộ dữ liệu theo tỉ lệ 60:40, trong đó 60% là tập huấn luyện, 40% là tập kiểm tra. Kết quả được trình bày trong bảng 2.

	userId	productId	rating
Train	2059	9509	5
Test	2055	8154	5
Total	2060	10913	5

Bảng 1: Thống kê các nhãn trên mỗi tập dữ liệu

Có khá nhiều thư viện và bộ công cụ trong Python cung cấp triển khai các thuật toán khác nhau mà bạn có thể sử dụng để xây dựng một đề xuất. Nhưng cái mà bạn nên thử khi hiểu các hệ thống khuyến nghị là Surprise. Surprise là một Python SciKit đi kèm với các thuật toán đề xuất khác nhau và các chỉ số tương tự để giúp dễ dàng xây dựng và phân tích các đề xuất.

Với lọc cộng tác, chúng tôi tiến hành tạo ID cho người dùng và sản phẩm, ratings của người dùng đối với sản phẩm.

User-based collaborative : sử dụng một thuật toán lọc cộng tác cơ bản KNNwithMeans

$$\hat{r}_{ui} = \mu_u + \frac{\sum\limits_{v \in N_i^k(u)} \text{sim}(u, v) \cdot (r_{vi} - \mu_v)}{\sum\limits_{v \in N_i^k(u)} \text{sim}(u, v)}$$

.

K - Số lượng láng giềng (tối đa) cần tính đến để tổng hợp (ở đây nhóm sử dụng K = 10)

sim_options - Từ điển các tùy chọn cho phép đo độ tương tự.

Để tìm điểm tương đồng, chỉ cần định cấu hình hàm bằng cách chuyển một từ điển làm đối số cho hàm đề xuất. Từ điển phải có các khóa bắt buộc, chẳng hạn như sau:

Name chỉ số tương tự để sử dụng. Các tùy chọn là cosine, msd, pearson hoặc pearson_baseline. Mặc định là msd.

User_based là một boolean cho biết cách tiếp cận sẽ dựa trên User-based hay Item-based. Giá trị mặc định là True, có nghĩa là phương pháp tiếp cận dựa trên người dùng sẽ được sử dụng.

Cấu hình hàm

```
from surprise import KNNWithMeans

# To use item-based cosine similarity
sim_options = {
    "name": "cosine",
    "user_based": True,
}
algo = KNNWithMeans(k=10,sim_options=sim_options)
```

Item-based collaborative : Sử dụng phương pháp tương tự **User-based collaborative ,** sử dụng tính tương tự cosine để tìm các Item tương đồng bằng cách sử dụng phương pháp Item-based. Với cấu hình hàm một từ điển làm đối số cho hàm đề xuất.

```
from surprise import KNNWithMeans

# To use item-based cosine similarity
sim_options = {
    "name": "cosine",
    "user_based": False,  # Compute  similarities between items
}
algo = KNNWithMeans(k=10,sim_options=sim_options)
```

Content-based : Ý tưởng của thuật toán này , là từ thông tin mô tả của item , biểu diễn item dưới dạng vector thuộc tính .Sau đó dùng các vector này để học mô hình của mỗi user , là ma trận trọng số của mỗi user với mỗi item.

Như vậy, thuật toán content-based gồm 2 bước:

- Bước 1: Biểu diễn items dưới dạng vector thuộc tính - item profile
- Bước 2: Học mô hình của mỗi user

Trong các hệ thống content-based chúng ta cần xây dựng một hệ thống (Profile) cho mỗi item . Profile được biểu diễn dưới dạng toán học là một " features vector " N chiều. Trong những trường hợp đơn giản (ví dụ item là dạng dữ liệu văn bản) features vector được trích xuất trực tiếp từ các item.Từ đó chúng ta có thể xác định các item có nội dung tương tự bằng cách tính độ tương đồng giữa các feature vector của chúng. Nội dụng sử dụng là Category , phân loại mục của các video game.

Phương pháp sử dụng để xây dựng features vector là :

- Sử dụng TF-IDF để tạo không gian vector
- Sau đó tính toán độ tương đồng cosine similarity

Đưa ra khuyến nghị giữa các video tương đồng

IV. kết quả và đánh giá

Kết quả chúng tôi thu được 3 mô hình như trong bảng. Vì bộ dữ liệu có sự chênh lệch, chúng tôi sử dụng 4 độ đo là Precision@k và Recall@k, RMSE , MAE. Trong đó, RMSE là một quy tắc tính điểm bậc hai cũng đo độ lớn trung bình của lỗi. Đó là căn bậc hai của trung bình của sự khác biệt bình phương giữa dự đoán và quan sát thực tế.

MAE đo độ lớn trung bình của các lỗi trong một tập hợp các dự đoán mà không cần xem xét hướng của chúng. Đó là giá trị trung bình trên mẫu thử nghiệm về sự khác biệt tuyệt đối giữa dự đoán và quan sát thực tế, trong đó tất cả các khác biệt riêng lẻ có trọng số bằng nhau.

Model	Pre@10	Rec@10	RMSE	MAE
User (KNNwithMeans)	0.0033	0.0037	0.7869	0.5532
Item (KNNwithMeans)	0.0014	0.0011	0.8062	0.5796
User (SVD)	0.0171	0.0116	0.8131	0.6246

Bảng 2: kết quả độ đo trên các mô hình

Từ bảng 2 cho thấy kết quả mô hình User (KNNwithMeans) cho kết quả tốt nhất dựa trên độ đo RMSE

V. **Demo**

- Khuyến nghị 10 video hàng đầu cho userId = A1Y5LUJZ8879PP

```
recommendation_video('A1Y5LUJZ8879PP',prediction,10)

 khuyến nghị 10 video cho user (user_id = A1Y5LUJZ8879PP):

9722     B000G6SPHI
4322     B001NIP3EG
14949    B000066TS5
24451    B00009VE68
1787     B0009A4EVM
26339    B009DYE3JI
15395    B0002RQ3ES
21409    B0000B0MNH
6192     B0007SL1ZI
7549     B000067DPL
Name: productId, dtype: object
```

- Khuyến nghị 10 video hàng đầu cho userId = A2QY0WD2JLWUKS

```
[84] recommendation_video('A2QY0WD2JLWUKS',prediction,10)

 khuyến nghị 10 video cho user (user_id = A2QY0WD2JLWUKS):

12377    B0013RC1W4
433      B0029LJIFG
164      B00H2VOELQ
5946     B00ATF5YY8
5399     B002I7KJ5O
5939     B00CIBDOF2
12119    B001AH8YSW
10499    B00AWSPCPI
7567     B003OAE79S
4        B0050SWS8O
Name: productId, dtype: object
```

- Khuyến nghị 10 video hàng đầu tương tự title_video = Turok PC

```
[ ] category_recommendations('Turok PC').head(10)

1                  Turok PC
8        Ship Simulator 2008
22             Uyku stasyonu
101          The X-Files Game
116         Final Fantasy VII
117         Final Fantasy VII
118         Final Fantasy VII
203    Grand Theft Auto - PC
267           Half-Life - PC
268           Half-Life - PC
Name: title, dtype: object
```

- Khuyến nghị 10 video hàng đầu tương tự title_video = Mega Man Xtreme 2

```
[ ] category_recommendations('Mega Man Xtreme 2').head(10)

1664        Rugrats Time Travelers
1692                   Yoda Stories
1753              Deja Vu I amp II
1754              Deja Vu I amp II
1770                  Test Drive 6
1782    Pro Darts - Game Boy Color
1800       Hello Kitty Cube Frenzy
2385           Castlevania Legends
2386           Castlevania Legends
2387           Castlevania Legends
Name: title, dtype: object
```

VI. Kết Luận

Trong bài báo này, chúng tôi đã xây dựng được hệ khuyến nghị cho người dùng dựa trên 2 phương pháp phổ biến là lọc cộng tác và lọc dựa trên nội dung. Với lọc cộng tác, chúng tôi đã xây dựng mô hình dựa trên người dùng, dựa trên mục(sản phẩm) và sử dụng phương pháp ma trận phân rã. Trong đó mô hình mang lại kết quả tốt nhất với RMSE đạt 0.78. Bên cạnh đó chúng tôi còn xây dựng hệ khuyến nghị giữa trên nội dung sử dụng TF-IDF kết hợp với độ đo cosine.

Trong tương lai, chúng tôi muốn phát triển và cải thiện thêm hệ thống khuyến nghị hơn nữa bằng cách sử dụng phương pháp deep learning, ALS để huấn luyện với dữ liệu lớn. Tiếp theo, chúng tôi muốn xây dựng một API có sẵn để mọi người có thể dễ dàng sử dụng.

Tài liệu tham khảo

1. Singular Value Decomposition (SVD) In Recommender System (analyticsindiamag.com)
2. Recommendation System using kNN - Auriga IT
3. Build a Recommendation Engine With Collaborative Filtering – Real Python
4. (PDF) Content-based Recommender Systems: State of the Art and Trends (researchgate.net)