Nguyễn Đỗ Quốc Duy - 1810867

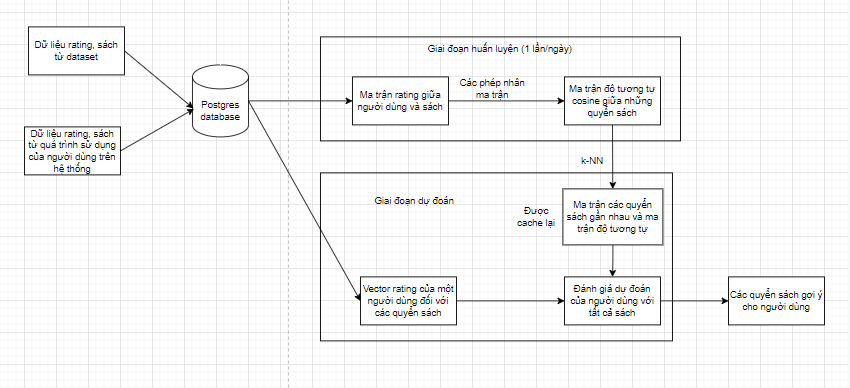
Câu 1:

1. Vấn đề dự án cần giải quyết là tạo một hệ thống có khả năng gợi ý các cuốn sách phù hợp cho người dùng, hệ thống sẽ dựa trên những đánh giá đã có của người dùng đối với các quyển sách để dự đoán các đánh giá chưa có, từ đó chọn ra những quyển sách mà người dùng có thể thích nhất.
2. Dữ liệu được lấy từ nguồn: [zygmuntz/goodbooks-10k: Ten thousand books, six million ratings (github.com)](https://github.com/zygmuntz/goodbooks-10k)

Trong đó nhóm đã sử dụng các file “ratings.csv” để lấy thông tin về đánh giá của người dùng đối với các quyển sách và file “books.csv” để lấy thông tin của các quyển sách. Dữ liệu có được cho qua xử lý một số bước:

* Lọc bỏ các cột không sử dụng, đối với file “ratings.csv” chỉ sử dụng các cột ID và đánh giá. Đối với file “books.csv” chỉ sử dụng một số thông tin của sách.
* Chuẩn hóa ma trận đánh giá (bằng cách trừ đi đánh giá trung bình của người dùng)
* Thực hiện tính toán để thu được ma trận độ giống nhau giữa các quyển sách.
* Từ ma trận độ giống nhau thực hiện dự đoán đánh giá của một người dùng cụ thể.

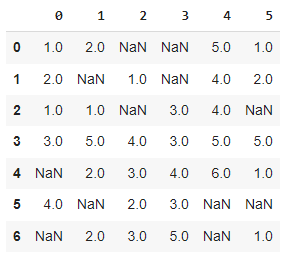
1. Pipeline xử lý dữ liệu của dự án:



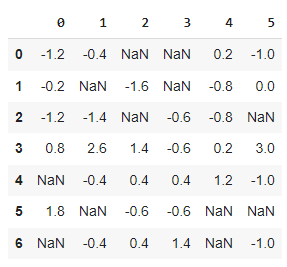
* Các dữ liệu nhập ban đầu và dữ liệu phát sinh được lưu trữ trong Postgres database.
* Trong giai đoạn huấn luyện, được hệ thống gọi theo chu kỳ, ma trận đánh giá giữa người dùng và sách được tải về. Từ đó các tính toán được thực hiện để tính ra ma trận độ tương tự giữa các quyển sách.
* Kỹ thuật k-NN được sử dụng để tính ra ma trận các quyển sách gần nhau. Kết quả được lưu lại cho quá trình dự đoán.
* Trong quá trình dự đoán, vector rating của người dùng A đối với các quyển sách được lấy từ database, dựa vào ma trận các quyển sách gần nhau, các đánh giá chưa có của từng quyển sách sẽ được tính bằng trung bình có trọng số từ các đánh giá đã có của đúng người dùng A (các trọng số cũng chính là độ tương tự), từ đó cho ra được đánh giá của tất cả các quyển sách.
* Những quyển sách có đánh giá cao nhất mà chưa được người dùng đánh giá sẽ được gợi ý cho người dùng.

1. Kỹ thuật học máy được sử dụng trong dự án là kỹ thuật Item-based Collaborative Filtering với k-NN. Các bước cơ bản của kỹ thuật như sau:

**Dữ liệu vào** là ma trận đánh giá giữa người dùng và sách, trong đó các người dùng được đánh số từ 0-5 và các quyển sách được đánh số từ 0-6.

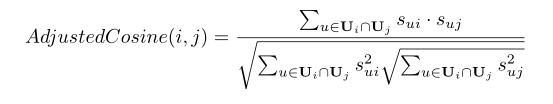


* **Ma trận đánh giá được chuẩn hóa** bằng cách ứng với mỗi người dùng, trừ các đánh giá của họ đi trung bình của chúng.



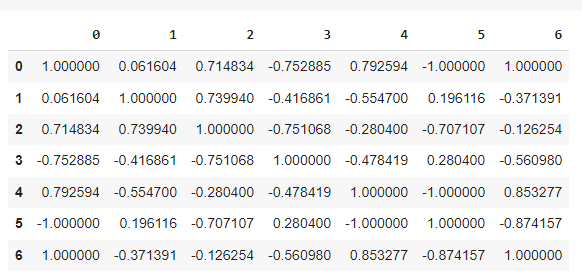
* **Tính ma trận độ tương tự giữa các quyển sách**:

Công thức tính độ tương tự được biến thể từ công thức độ tương tự cosine:



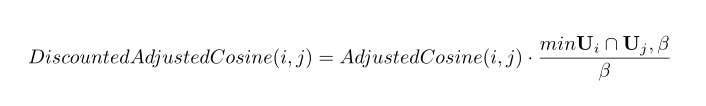
Do giữa hai quyển sách chỉ có một số người dùng là đã đánh giá cho cả hai, công thức chỉ xét các đánh giá ấy.

Sau khi tính độ tương tự ta được ma trận:



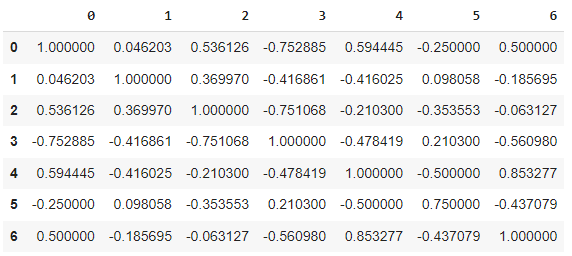
* **Hiệu chỉnh ma trận độ tương tự**:

Đối với các quyển sách có rất ít người dùng chung, độ tương tự giữa chúng không đáng tin cậy, do đó ta sử dụng công thức sau để hiệu chỉnh lại:



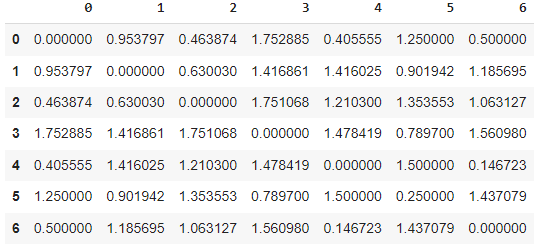
Trong đó Ui ^ Uj là số lượng các đánh giá chung của hai quyển sách. Nếu số lượng này nhỏ hơn một ngưỡng beta nhất định, ta bắt đầu giảm độ tương tự giữa chúng.

Ta được ma trận độ tương tự sau khi hiệu chỉnh:



* **Từ ma trận độ tương tự, ta có ma trận khoảng cách** bằng công thức:

distance = 1 – similarity



* **Sử dụng k-NN để tính ra danh sách các quyển sách gần nhất:** (cột đầu tiên là các quyển sách từ 0-6, các cột sau là các quyển sách gần với chúng).

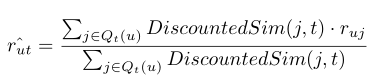


* **Dự đoán đánh giá của người dùng**: giả sử ta có vector đánh giá của người dùng với các quyển sách như sau:



Ta cần dự đoán đánh giá của người dùng đối với hai quyển sách số 1 và quyển sách số 5.

Ta sử dụng công thức:



Đối với quyển sách 1 ta có 3 quyển sách gần nhất là 2, 5, 0. Độ tương tự lần lượt là 0.3699, 0.098, 0.0462.

Đối với quyển sách 5 ta có 3 quyển sách gần nhất là 3, 1, 0. Độ tương tự lần lượt là

0.21, 0.098, -0.25.

Do quyển sách 5 chưa được đánh giá nên ta chỉ dùng 2 quyển sách 2 và 0 để dự đoán cho quyển sách 1.

Đánh giá dự đoán của người dùng với quyển sách 1:

(0.3699 \* 1 + 0.0462 \* 2) / (0.3699 + 0.0462) = 1.111 (sao)

Do quyển sách 1 chưa được đánh giá và độ tương tự của 5 với 0 là âm nên ta chỉ dùng quyển sách số 3 để dự đoán

Đánh giá dự đoán của người dùng với quyển sách 5:

0.21 \* 5 / 0.21 = 5 (sao)

Câu 2:

1. x là vector 7 chiều có cận dưới và cận trên của từng chiều là [-1,-2,-3,-4,-5,-6,-7] và [ 9 , 8, 7, 6, 5, 4, 3]. Do đó các khoảng giá trị của từng chiều trong x có độ dài lần lượt là [10, 10, 10, 10, 10, 10, 10].

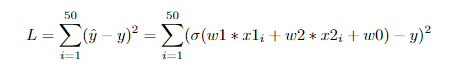
Vì đáp số cần chính xác đến 3 chữ số sau dấu thập phân nên ứng với các khoảng giá trị có độ dài là 10, ta cần 10 \* 1000 + 1 = 10001 giá trị khác nhau cần phải biểu diễn cho mỗi giá trị của x. Chẳng hạn, ứng với chiều thứ nhất ta cần biểu diễn -1, -0.999, -0.998… 8.999, 9.

Ứng với chiều thứ hai ta cần biểu diễn -2, -1.999, -1.998… 7.999, 8.

Do 2^13 = 8192 < 10001 < 2^14 = 16384 nên số bit cần để biểu diễn 10001 giá trị ứng với từng chiều là 14bit. Do vậy để biểu diễn tất cả 7 chiều của x ta cần nhiễm sắc thể có số bit là 14 \* 7 = 98 bit.

Ta chọn một khoảng và độ chính xác của các giá trị w1, w2, w0 rồi biểu diễn các giá trị này thành các dãy bit.

Sử dụng 50 điểm trên không gian 2 chiều và các nhãn của chúng, ta có hàm mất mát:



Bài toán trở thành sử dụng GA để cực tiểu hàm mất mát L.

Để giải bài toán, ta thực hiện các bước khởi tạo quần thể, trao đổi chéo nhiễm sắc thể và đột biến, loại bỏ các cá thể yếu và giữ lại các cá thể mạnh cho đến khi hàm mất mát L không còn giảm.