TÓM TẮT QUÁ TRÌNH LÀM ĐỒ ÁN KHOA HỌC DỮ LIỆU

Nhóm 4:

18120280 - Trần Đức Anh

18120202 - Trần Quốc Long

1. THU THẬP DỮ LIỆU

- Bọn em quyết định lấy dữ liệu về chủ đề anime (phim hoạt hình của Nhật Bản) là đề tài cho đồ án này.
- Bọn em lấy dữ liệu bằng cách sử dụng web API từ trang web myanimelist.net .
- Quá trình thu thập dữ liệu khá là đơn giản và không gặp bất kì khó khăn nào.
- Do trong file json của API có dạng như thế này

```
json_text = json_url.json()
json_text['genres']
```

```
[{'mal_id': 1,
  'type': 'anime',
  'name': 'Action',
  'url': 'https://myanimelist.net/anime/genre/1/Action'},
 {'mal_id': 24,
  'type': 'anime',
  'name': 'Sci-Fi',
  'url': 'https://myanimelist.net/anime/genre/24/Sci-Fi'},
 {'mal_id': 2
```

- Nên em đã dùng hàm “collect_multichoice” để lấy dữ liệu.
- Sau khi hoàn thành, bọn em ghi ra 1 file mới là anime_rating_table.

2. KHÁM PHÁ DỮ LIỆU

- Bắt đầu đọc dữ liệu từ file.
- Cột genres là cột multichoice và cột source, rating là cột onechoice.
- Nhận ra ở cột duration thì dữ liệu nó sẽ hiện theo kiểu xx min per ep, cột scored, scored_by và episodes có những dòng dữ liệu không phải dạng số.
- Tìm hiểu xem với mỗi cột có dữ liệu dạng chữ và dạng số thì phân bố như thế nào và thống kê ra.
- Còn lại thì không có gì bất thường.

3. ĐƯA RA CÂU HỎI CẦN TRẢ LỜI

- Câu hỏi: Điểm số của một bộ anime được tính từ thông tin của bộ đó theo công thức nào ?
- Lợi ích: Đối với người xem, việc có nên dành thời gian để xem một bộ phim hay không đa số dựa vào điểm đánh giá của bộ phim đó. Nhưng với một bộ phim mới ra thì sao. Nhà sản xuất chỉ cung cấp tên phim, thể loại, độ dài, số tập phim… và với việc trả lời được câu hỏi này. Người xem có thể dự đoán được bộ phim đó có thể đạt bao nhiêu điểm và có đáng xem hay không.
- Nguồn cảm hứng: Em và teammate của em đều rất thích xem anime, bọn em cũng dựa trên gợi ý vào 1 buổi học của thầy khi thầy lấy ví dụ về doanh thu là 1 dữ kiện lỗi khi đánh giá phim.

4. TIỀN XỬ LÍ DỮ LIỆU

- Đầu tiên là giải quyết những cột đã nêu ở phần khám phá dữ liệu bằng hàm "convert_dtypes" để chuyển hoàn toàn về dạng số và chuyển cột "title" làm cột index.
- Tiếp theo là tách các tập train, tập validation và tập test.
- Sau đó xử lí drop những cột không cần thiết (class colDrop) như "producers", "studios" vì quá nhiều giá trị thiếu; "favorites", "scored_by" vì phim mới sẽ không có thông tin người thích và số người đánh giá về phim đó.
- Ở đây có 1 vấn đề là: cần xử lí cột multichoice(cột genres) và cột onechoice(cột rating, source). Vì thế nên bọn em đã viết thêm 2 class mới "customOneHotEncoder_1" để xử lí cột multichoice và "customOneHotEncoder_2" để xử lí cột onechoice. Mục đích là để biến thành 2 giá trị: top_ và Others_ để tránh bị overfitting do giá trị xuất hiện gần nhau khá nhiều.
- Tiếp theo là tạo một cái pipeline tên là full_pipeline bao gồm các bước: colDrop, customOneHotEncoder_1, customOneHotEncoder_2, điền các giá trị thiếu bằng SimpleImpute và mô hình học máy cơ bản Linear Regression.
- Sau khi đã tạo xong pipeline thì dùng nó để đi tìm mô hình tốt nhất bằng cách thử nghiệm với các giá trị khác nhau của siêu tham số và chọn ra các giá trị tốt nhất.

5. HUẤN LUYỆN DỮ LIỆU

- Huấn huyện tập dữ liệu bằng linear regression.
- Lưu lại mô hình tốt nhất ở file “best model” do máy học khá là tốn thời gian.
- Khi cần máy học thì dùng file “best model” là được.

6. MÔ HÌNH HÓA DỮ LIỆU

- Mô hình hóa dữ liệu bằng cách chọn mô hình tối ưu của 1 giá trị so với kết quả.
- Mô hình hóa độ lỗi của tập dữ liệu dựa trên genres.
- Mô hình hóa độ lỗi của tập dữ liệu dựa trên rating.
- Mô hình hóa độ lỗi của tập dữ liệu dựa trên source.
- Huấn luyện mô hình cuối cùng.
- Tìm w_0 và w_1-w_n để tính theo công thức của LNR: $h_w(x) = w_0+w_1x_1+ \dots + w_nx_n$.
- Tìm ra output được tính theo input từ công thức nào.
- Kiểm tra độ lỗi r^2 trên tập test.

7. NHÌN LẠI QUÁ TRÌNH LÀM ĐỒ ÁN

- Dự kiến lúc đầu: Tính toán doanh thu của những bộ Light Novel từ những yếu tố thể loại, tác giả,…Bọn em đã quyết định là lấy dữ liệu từ trang lndb.info .
- Nhưng có 1 vấn đề xảy ra là trang đó đã bị sập. ☹
- Sau đó em đã quyết định không sử dụng Light Novel làm dữ liệu chính mà chuyển sang làm anime bởi vì nó cũng gần như tương tự với Light Novel, đều có các yếu tố trên. Bọn em đã lấy dữ liệu từ trang myanimelist.net.
- Bởi vì trang web trên chưa hoàn thiện API nên bọn em dùng web API của myanimelist.net qua trang jikan.moe (1 trang unofficial API nhưng khá là đầy đủ).
- Quá trình thu thập dữ liệu khá là đơn giản. Có một chút khó khăn về việc thời gian sleep time giữa 2 lần lấy API của trang web là 4s (chủ trang web đề nghị như thế). Tập dữ liệu gồm hơn 1200 bộ phim nên thời gian lấy khá là lâu.
- Quá trình tiền xử lý: Trải qua 1 số khó khăn nhất định. Khá nhiều dữ liệu dạng category và có một cột là dạng multichoice nên việc xử lí gặp 1 chút rắc rối. Nên thay vì xử dụng theo One-hot Encoder truyền thống, em đã làm class khác dựa trên ý tưởng của One-hot Encoder để xử lý các cột category trong đồ án.
- Học dữ liệu bằng linear regression. Điều khiến bọn em bất ngờ là độ đo r^2 ban đầu của tập test bị quá tệ (-0.06). Sau đó bọn em đã thu thập lại 1 bộ dữ liệu và có 1 số thay đổi trong việc tạo ra class custom của One-hot Encoder để cho dữ liệu học có thể fit ổn hơn vào dữ liệu test.

- Sau khi chỉnh sửa bọn em đã tìm ra được 1 giải pháp và nó giúp độ đo r^2 tăng lên khoảng 0.16. Mặc dù vẫn còn khá nhỏ nhưng bọn em nghĩ rằng đó đã là giải pháp tối ưu nhất hiện tại. Em nghĩ việc có độ đo r^2 nhỏ như vậy có thể là do việc đánh giá phim nghiêng khá nhiều về ý kiến chủ quan nên dữ liệu bị nhiễu khá nhiều, một phần do bọn em chưa có nhiều kinh nghiệm về việc thu thập và xử lí dữ liệu nên vẫn chưa tối ưu được bài toán này.
- Việc mô hình hóa dữ liệu tốn khá nhiều thời gian nghiên cứu vì em không biết làm cách nào để mô hình hóa multi linear regression. Sau khi nghiên cứu trên google thì em đã thấy một cách để mô hình: sử dụng việc mô hình một giá trị không phụ thuộc vào một giá trị phụ thuộc (ở đây là từng cột so với điểm số).
- Học được những gì:
  - Đã học được khá nhiều thứ về việc xử lí và mô hình hóa dữ liệu qua các lỗi đã xảy ra trong quá trình làm đồ án.
  - Nắm vững hơn quá trình xử lí dữ liệu bằng pipeline.
  - Cách thu thập và chọn lọc thông tin để đưa vào xử lý.
  - Cách sử dụng github để làm việc nhóm.
  - Hiểu hơn được những phần cơ bản trong thuật toán học máy qua việc thực hành và tìm tòi, nghiên cứu.
  - Tăng cường khả năng đọc hiểu tiếng Anh hơn để có thể đọc các tài liệu chuyên ngành. Động lực để cố gắng học tốt tiếng Anh.
- Nếu có thêm thời gian thì sẽ làm gì:
  - Tìm cách tối ưu việc xử lí để độ lỗi nhỏ hơn.
  - Tìm cách tối ưu hơn để mô hình hóa multi linear regression.
  - Kiểm tra xem với các cách học máy khác có cách nào tối ưu hơn không.

7. TÀI LIỆU THAM KHẢO

- Những code có trong những bài tập 1-2-3 và những phần demo của môn học này.
- https://stackoverflow.com/questions/52404857/how-do-i-plot-for-multiple-linear-regression-model-using-matplotlib
- https://stackoverflow.com/questions/62408093/one-hot-encoding-multiple-categorical-data-in-a-column
- https://scikit-learn.org/stable/
- https://stackoverflow.com/questions/5306079/python-how-do-i-convert-an-array-of-strings-to-an-array-of-numbers