

ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



NHẬP MÔN
KHOA HỌC DỮ LIỆU

Đề tài:

Phân tích dữ liệu phim từ IMDB
và xây dựng mô hình gợi ý

Giảng viên hướng dẫn:

PGS. TS. Thân Quang Khoát

Sinh viên thực hiện:

Nguyễn Đức Anh 20172937

Nguyễn Minh Đăng 20172998

Võ Đức Quân 20173320

Nguyễn Văn Lương 20173249

Bùi Việt Dũng 20173045

HÀ NỘI, 12/2020

MỤC LỤC

1.	ĐẶT VẤN ĐỀ	1
2.	QUÁ TRÌNH THỰC HIỆN.....	1
2.1.	Thu thập dữ liệu	1
2.2.	Mô tả, đánh giá dữ liệu thô	4
2.2.1.	Theo các mốc đánh giá	5
2.2.2.	Theo năm ra mắt	7
2.2.3.	Theo thể loại	8
2.2.4.	Theo quốc gia.....	9
2.2.5.	TOP 10 phim theo điểm đánh giá	9
2.3.	Làm sạch, tiền xử lý dữ liệu.....	10
2.3.1.	Xử lý dữ liệu các trường “budget” và “gross”	10
2.3.2.	Xử lý dữ liệu trường “story_line”	10
2.3.3.	Xử lý dữ liệu trường “stars”	11
2.3.4.	Xử lý dữ liệu trường “director”, “language”, “country” và “genres”	12
2.4.	Xây dựng mô hình gợi ý	13
2.4.1.	Mục tiêu	13
2.4.2.	Content based.....	13
2.4.3.	Session-based.....	15
3.	KẾT LUẬN	20
	PHÂN CÔNG CÔNG VIỆC	21
	TÀI LIỆU THAM KHẢO	22

DANH MỤC HÌNH ẢNH

Hình 1. Các bước thực hiện	1
Hình 2. Ví dụ về định danh của phim trên IMDB.com	1
Hình 3. Các thông tin được thu thập	2
Hình 4. Cấu trúc một bản ghi được định nghĩa trong Golang	3
Hình 5. Chi tiết về sự thiếu dữ liệu ở các trường	4
Hình 6. Mô tả tổng quan dữ liệu trường rating	5
Hình 7. Phân bố phim theo hướng đánh giá chủ quan	6
Hình 8. Phân bố điểm đánh giá trung bình	6
Hình 9. Số lượng phim qua các năm	7
Hình 10. Doanh thu qua các năm	7
Hình 11. Số lượng phim theo thể loại	8
Hình 12. Điểm đánh giá trung bình theo thể loại	8
Hình 13. Số lượng phim theo quốc gia	9
Hình 14: Trước khi min max scale	14
Hình 15: Sau khi min-max scaler	14
Hình 16: Dữ liệu sau khi kết hợp dữ liệu thu thập được với bộ Movielens 1M	16
Hình 17: Kiến trúc mạng Skip-gram	17
Hình 18: So sánh model với size <100 , $=100$, >100 thì <100 mang lại kết quả tốt nhất	18
Hình 19: So sánh model với 3 size < 100	18
Hình 20: So sánh model với giá trị window chạy từ 5 đến 10	19
Hình 21: So sánh model với giá trị negative giảm dần từ 20 đến 11	19
Hình 22. Dữ liệu đầu vào của mô hình Session-based	20
Hình 23. Kết quả dự đoán của mô hình Session-based	20
Hình 24: Kết quả đánh giá model training được qua ndcg và hit-rate	20

1. ĐẶT VẤN ĐỀ

Trong thời đại kỷ nguyên số như hiện tại, Internet mỗi ngày sản sinh ra lượng dữ liệu vô cùng lớn và phong phú. Lượng dữ liệu này rất hữu ích nếu chúng ta có thể thu thập và xử lý chúng và hướng tới sử dụng cho nhiều mục đích khác nhau như nắm bắt mẫu và xu hướng xã hội từ dữ liệu, hay dự đoán hoặc gợi ý để cải thiện trải nghiệm người dùng trên các nền tảng thông tin số và mạng xã hội,...

Cùng với sự phát triển của kinh tế - xã hội, nhu cầu giải trí của con người cũng tăng cao, đặc biệt là ở bộ môn nghệ thuật thứ bảy – Điện ảnh. Trong những năm trở lại đây, ta có thể thấy Điện ảnh phát triển mạnh mẽ như thế nào với số lượng rạp chiếu phim tăng và hàng loạt nền tảng chiếu phim trực tuyến như Netflix, HBO Max,... Cùng với đó, khán giả cũng sẵn đón các thông tin về các bộ phim cũng như có nhu cầu đánh giá các bộ phim mà họ đã xem. Từ đó, dữ liệu về các bộ phim trở thành nguồn dữ liệu hữu ích cho các hệ thống gợi ý phim thương mại. Nhận thấy tầm quan trọng và sự thú vị trong dữ liệu về các bộ phim, chúng em tiến hành thu thập, xử lý, phân tích và xây dựng một hệ gợi ý phim cơ bản dựa trên đặc điểm của các bộ phim. Chúng em sử dụng dữ liệu được thu thập từ nền tảng thông tin điện ảnh trực tuyến IMDB.com.

2. QUÁ TRÌNH THỰC HIỆN

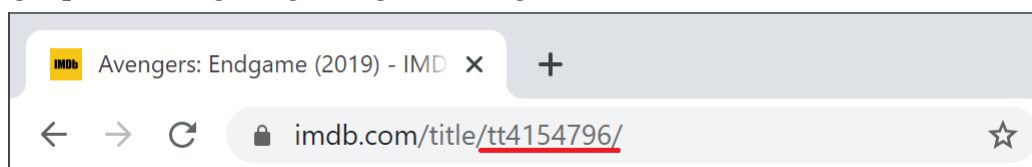
Các bước tiến hành lần lượt là: thu thập dữ liệu, tiền xử lý dữ liệu, trực quan hóa dữ liệu để có cái nhìn cụ thể với từng thuộc tính của các bộ phim và cuối cùng là xây dựng mô hình gợi ý đơn giản



Hình 1. Các bước thực hiện


2.1. Thu thập dữ liệu


Ứng dụng thu thập dữ liệu được viết bằng ngôn ngữ Golang sử dụng thư viện Colly, nền tảng hỗ trợ thu thập dữ liệu dành cho Golang. Golang là một ngôn ngữ lập trình với khả năng lập trình song song đơn giản nhưng mạnh mẽ.



Hình 2. Ví dụ về định danh của phim trên IMDB.com

Quá trình thu thập dữ liệu được thực hiện bằng cách tăng dần từ 1 và chuyển thành định danh của phim có dạng “tt...”. Sau đó, thực hiện gửi yêu cầu tới imdb. Bây giờ, thư viện Colly sẽ bóc tách các thành phần HTML nhận về để lấy các dữ liệu cần thiết.

[FULL CAST AND CREW](#) | [TRIVIA](#) | [USER REVIEWS](#) | [IMDbPro](#) | [MORE](#) 




A

Avengers:


Endgame

(2019)



8.4

791,939





Rate This

PG-13

3h 1min

Action, Adventure, Drama

26 April 2019 (USA)

1:06 | Trailer

119 VIDEOS | 988 IMAGES

After the devastating events of [Avengers: Infinity War](#) (2018), the universe is in ruins. With the help of remaining allies, the Avengers assemble once more in order to reverse Thanos' actions and restore balance to the universe.

Directors: [Anthony Russo, Joe Russo](#)

Writers: [Christopher Markus](#) (screenplay by), [Stephen McFeely](#) (screenplay by) | [14 more credits](#) »

Stars: [Robert Downey Jr., Chris Evans, Mark Ruffalo](#) [See full cast & crew](#) »

Storyline

[Edit](#)

After the devastating events of [Avengers: Infinity War](#) (2018), the universe is in ruins due to the efforts of the Mad Titan, Thanos. With the help of remaining allies, the Avengers must assemble once more in order to undo Thanos's actions and undo the chaos to the universe, no matter what consequences may be in store, and no matter who they face...

Country: [USA](#)

Language: [English](#) | [Japanese](#) | [Xhosa](#) | [German](#)

Box Office

Budget: [\\$356,000,000](#) (estimated)

Opening Weekend USA: \$357,115,007, 28 April 2019

Gross USA: \$858,373,000

Cumulative Worldwide Gross: [\\$2,797,800,564](#)

Hình 3. Các thông tin được thu thập

Cấu trúc của các bản ghi được định nghĩa trong Golang như sau

```
package model

type Movie struct {
    ID          string `json:"tconst" bson:"tconst"`
    Name        string `json:"name" bson:"name"`
    Year        string `json:"year" bson:"year"`
    Rating      string `json:"rating" bson:"rating"`
    RatingCount string `json:"rating_count" bson:"rating_count"`
    Runtime     string `json:"runtime" bson:"runtime"`
    Genres      string `json:"genres" bson:"genres"`
    Budget      string `json:"budget" bson:"budget"`
    Gross       string `json:"gross" bson:"gross"`
    Director    string `json:"director" bson:"director"`
    Stars       string `json:"stars" bson:"stars"`
    Country     string `json:"country" bson:"country"`
    Language    string `json:"language" bson:"language"`
    StoryLine   string `json:"story_line" bson:"story_line"`
}
```

Hình 4. Cấu trúc một bản ghi được định nghĩa trong Golang

Cụ thể:

- *ID*: Định danh của bộ phim trên IMDb
- *Name*: Tên bộ phim
- *Year*: Năm phát hành của bộ phim
- *Rating*: Điểm đánh giá trung bình bộ phim nhận được trên IMDb
- *RatingCount*: Số lượt đánh giá của bộ phim trên IMDb
- *Runtime*: Thời lượng của bộ phim
- *Genres*: Danh sách thể loại của bộ phim
- *Budget*: Chi phí sản xuất của bộ phim
- *Gross*: Tổng doanh thu tích lũy toàn cầu của bộ phim
- *Director*: Danh sách đạo diễn của bộ phim
- *Stars*: Danh sách các ngôi sao tham gia vào bộ phim
- *Country*: Danh sách địa điểm sản xuất bộ phim
- *StoryLine*: Tóm lược kịch bản bộ phim

2.2. Mô tả, đánh giá dữ liệu thô

Sau khi quá trình thu thập, dữ liệu thu được gồm **491739** bản ghi với **1** trường định danh và **13** trường dữ liệu. Dưới đây là mô tả phân trăm dữ liệu bị thiếu của các trường dữ liệu.

	field	num_null	percent_null
0	year	25	0.01
1	rating	234490	47.69
2	rating_count	234490	47.69
3	story_line	205305	41.75
4	genres	66566	13.54
5	country	6750	1.37
6	language	21486	4.37
7	budget	409413	83.26
8	gross	445732	90.64
9	runtime	143179	29.12
10	director	67050	13.64
11	stars	71505	14.54

Hình 5. Chi tiết về sự thiếu dữ liệu ở các trường

Theo đó, trường thông tin “year” có phần thiếu dữ liệu thấp nhất là **0.01%**. Trường “gross” có phần thiếu dữ liệu cao nhất, ứng với **90.64%**. Bên cạnh đó, ta có thể thấy phần trăm thiếu thông tin là như nhau với hai trường “rating” và “rating_count”. Tức là, chỉ khi nhận được đánh giá của người dùng thì thông tin của hai trường này mới có thể được cập nhật. Tổng quan, khi loại bỏ tất cả các bản ghi thiếu dữ liệu, chỉ còn lại **13165/491739** bản ghi đầy đủ dữ liệu, ứng với **0.02677%**.

2.2.1. Theo các mốc đánh giá

Rating của các bộ phim có kiểu dữ liệu là các số thực (float64) với giá trị nhỏ nhất là 1 và giá trị lớn nhất là 10. Trường thông tin “rating” là con số chung nhất để mọi người có thể đánh giá chất lượng của một bộ phim. Dưới đây là mô tả dữ liệu trường rating

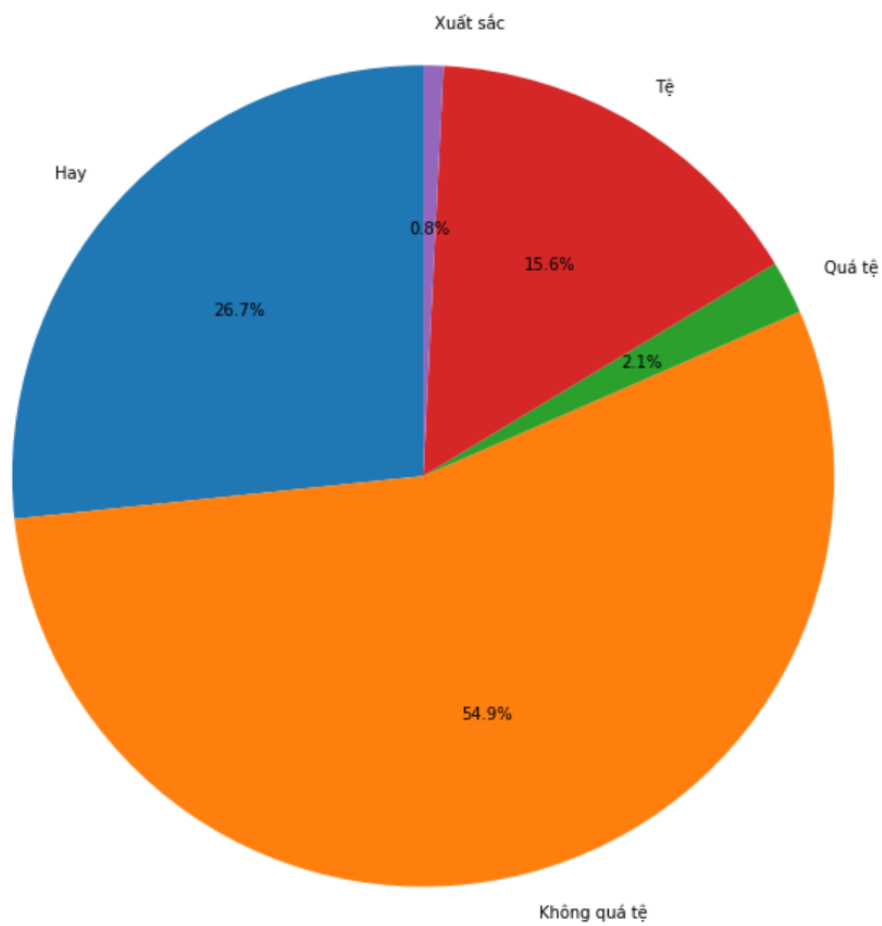
rating	
count	257249.000000
mean	6.128804
std	1.337768
min	1.000000
25%	5.400000
50%	6.300000
75%	7.000000
max	10.000000

Hình 6. Mô tả tổng quan dữ liệu trường rating

Với khoảng giá trị trải dài như vậy, chúng em thực hiện chia thành các khoảng để dễ dàng đánh giá chất lượng của bộ phim:

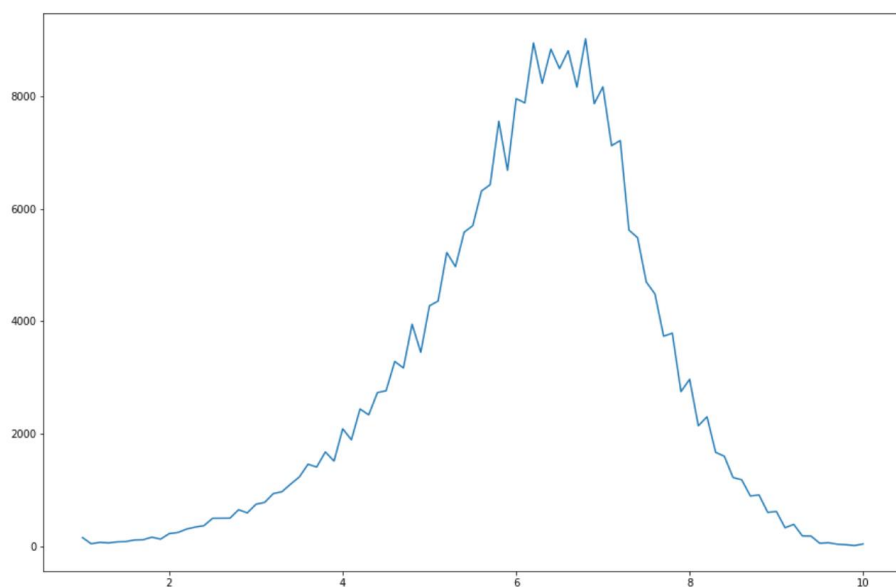
- Đánh giá “Quá tệ” với các bộ phim có đánh giá trung bình nhỏ hơn 3.
- Đánh giá “Tệ” với các bộ phim có đánh giá trung bình lớn hơn hoặc bằng 3 và nhỏ hơn 5.
- Đánh giá “Không quá tệ” với các bộ phim có đánh giá trung bình lớn hơn hoặc bằng 5 và nhỏ hơn 7.
- Đánh giá “Hay” với các bộ phim có đánh giá trung bình lớn hơn hoặc bằng 7 và nhỏ hơn 9.
- Đánh giá “Xuất sắc” với các bộ phim có đánh giá trung bình từ 9 trở lên.

Qua trực quan hóa, ta thấy phần lớn các phim nhận mức đánh giá “Không quá tệ” với **54.9%** tổng số phim (không bị thiếu trường “rating”). Tiếp đó là mức đánh giá “Hay” với **26.7%**. Bên cạnh đó, có rất ít phim “Xuất sắc” cũng như “Quá tệ”



Hình 7. Phân bố phim theo hướng đánh giá chủ quan

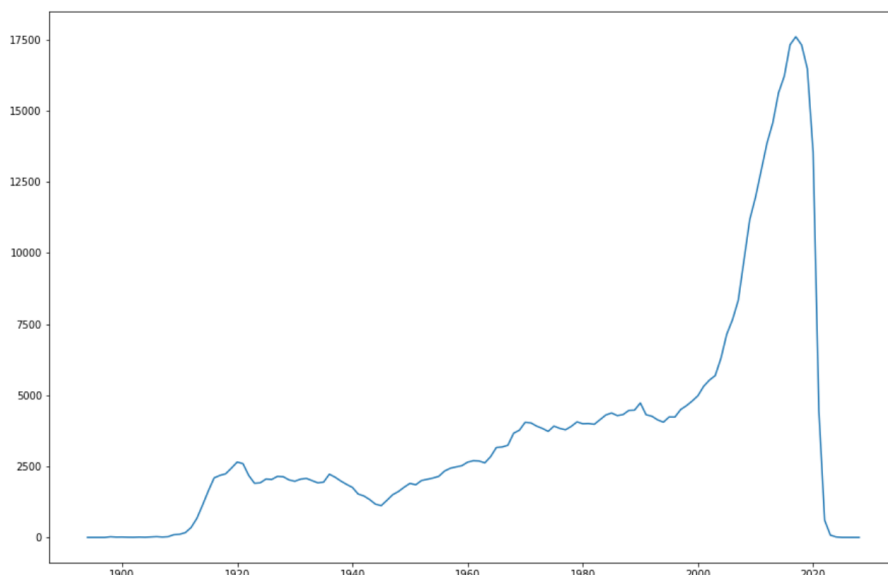
Điểm đánh giá IMDB chủ yếu thuộc khoảng **5 đến 8**



Hình 8. Phân bố điểm đánh giá trung bình

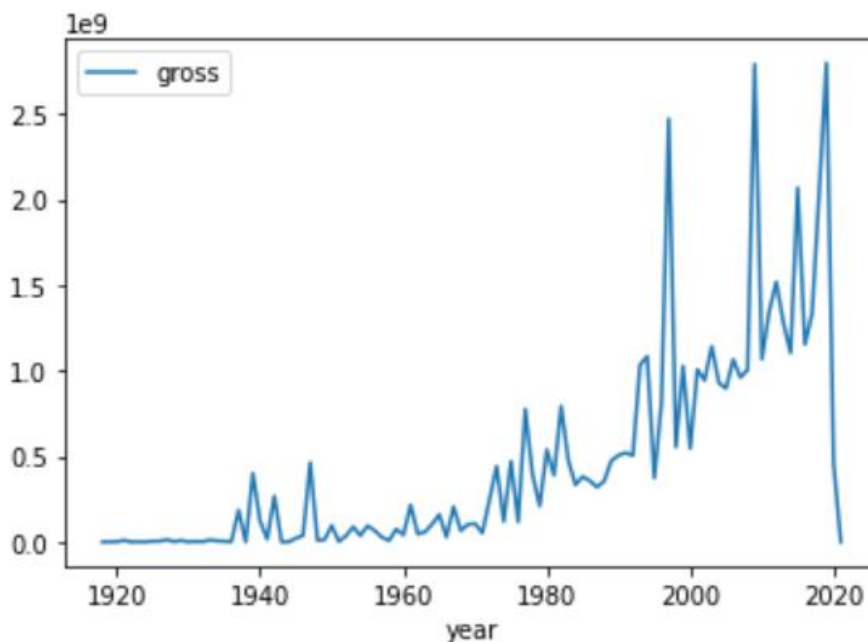
2.2.2. Theo năm ra mắt

Với sự phát triển của công nghệ và kỹ xảo, số lượng phim ra mắt khán giả ngày càng nhiều. Từ những năm 1960, ngành điện ảnh bắt đầu phát triển, nhưng chỉ khi tới những năm 2000, ngành điện ảnh đạt được mức tăng trưởng vượt bậc. Số lượng phim tăng một cách chóng mặt. Đỉnh cao nhất và vào năm 2017 với số lượng phim đạt gần **17500** phim. Bên cạnh đó, chúng ta có thể thấy rõ tác động của đại dịch COVID lên ngành công nghiệp điện ảnh như thế nào. Số lượng phim sụt giảm đáng kể.



Hình 9. Số lượng phim qua các năm

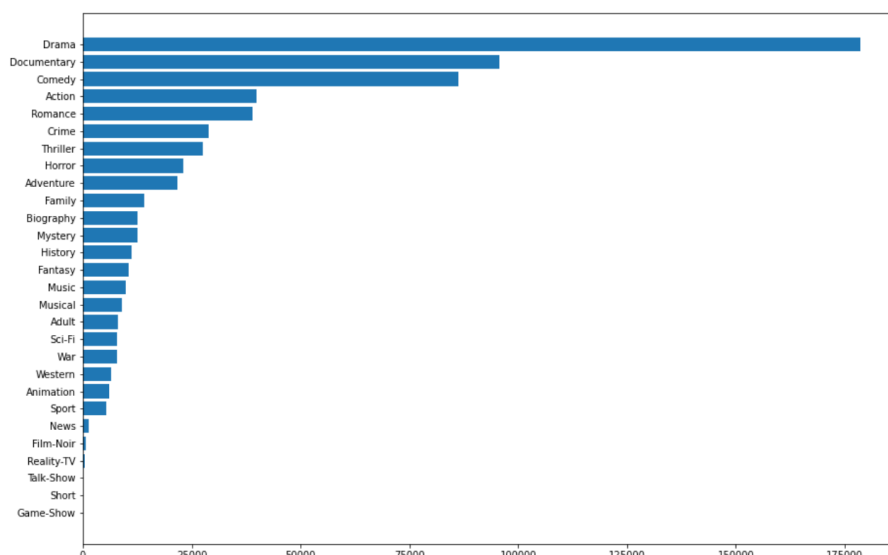
Doanh thu từ ngành điện ảnh không thật sự như chúng ta thấy ngày nay với các bom tấn tỷ đô. Doanh thu không ổn định.



Hình 10. Doanh thu qua các năm

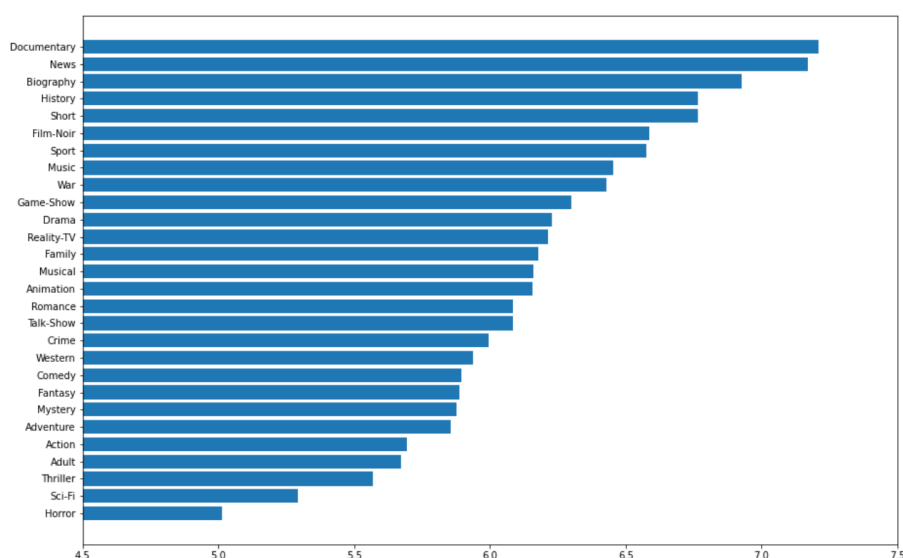
2.2.3. Theo thể loại

Các bộ phim trong bộ dữ liệu bao gồm 28 thể loại. Qua biểu diễn số lượng phim theo thể loại, ta thấy **Drama**, **Documentary** và **Comedy** là ba thể loại có số lượng phim vượt trội hơn nhiều so với các thể loại khác, cụ thể là **178617**, **95648** và **86248** phim. Còn các thể loại **Game-Show**, **Short**, **Reality-TV**, **Film-Noir** và **News** có số lượng không đáng kể.



Hình 11. Số lượng phim theo thể loại

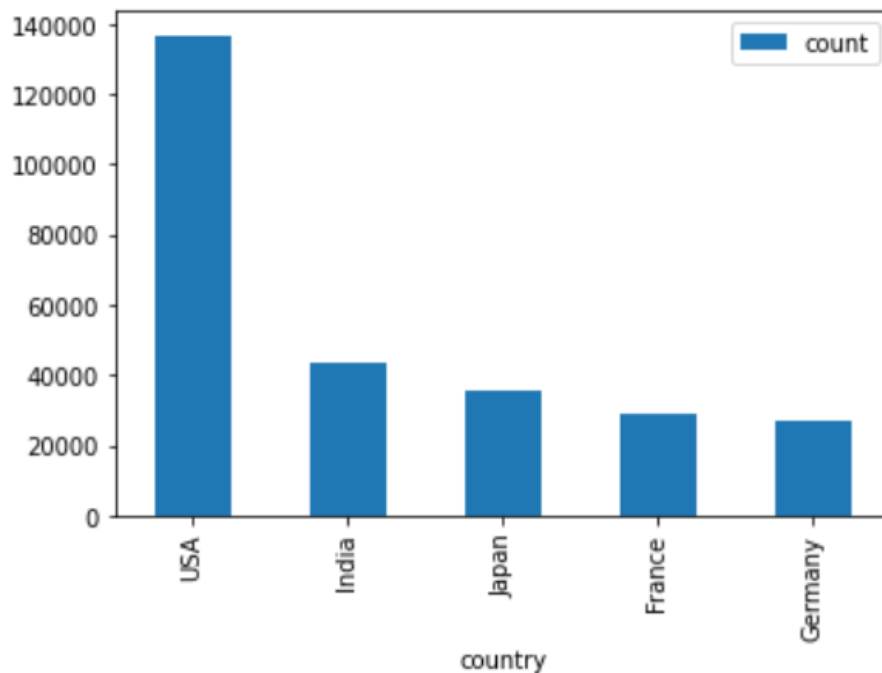
Tuy có số lượng thấp nhất nhưng điểm đánh giá trung bình của **Documentary** và **News** lại cao nhất, tương ứng là **7.2114** và **7.1719**. Điểm đánh giá trung bình thể loại thấp nhất thuộc về thể loại **Horror** và **Sci-Fi** với giá trị tương ứng là **5.0114** và **5.2918**



Hình 12. Điểm đánh giá trung bình theo thể loại

2.2.4. Theo quốc gia

Hollywood vẫn đang chứng tỏ mình là kinh đô điện ảnh của thế giới với số lượng phim sản xuất vượt trội so với các quốc gia khác. Theo sau đó là Ấn Độ, tuy nhiên, ta có thể thấy rõ sự khác biệt giữa vị trí thứ nhất và thứ hai.



Hình 13. Số lượng phim theo quốc gia

2.2.5. TOP 10 phim theo điểm đánh giá

Sau đây là bảng xếp hạng TOP 10 phim được đánh giá cao nhất trên toàn thế giới

	tconst	name	year	rating	rating_count
312164	tt0111161	The Shawshank Redemption	1994.0	9.3	2317694.0
180561	tt0068646	The Godfather	1972.0	9.2	1600459.0
104765	tt0050083	12 Angry Men	1957.0	9.0	681774.0
191781	tt0071562	The Godfather: Part II	1974.0	9.0	1117935.0
454071	tt0468569	The Dark Knight	2008.0	9.0	2279350.0
304013	tt0108052	Schindler's List	1993.0	8.9	1202286.0
425426	tt0167260	The Lord of the Rings: The Return of the King	2003.0	8.9	1626510.0
311443	tt0110912	Pulp Fiction	1994.0	8.9	1808257.0
148050	tt0060196	Il buono, il brutto, il cattivo	1966.0	8.8	682284.0
335938	tt0120737	The Lord of the Rings: The Fellowship of the Ring	2001.0	8.8	1643491.0

	tconst	name	year	rating	rating_count
89975	tt8483202	Thank You 5	2020.0	10.0	5.0
90338	tt10244450	ONE - The Documentary	2020.0	10.0	5.0
357526	tt6467090	Suffer for Good	2019.0	10.0	5.0
395146	tt1321834	Puppet Rampage	2008.0	10.0	5.0
409998	tt0160316	Girls Loving Girls	1996.0	10.0	5.0
416806	tt13368144	Discover Your Path Tour	2020.0	10.0	5.0
429900	tt3516936	Darkhan	2019.0	10.0	5.0
60347	tt8179214	The Straight and Narrow	2018.0	10.0	6.0
151601	tt9076274	Artist Depiction	2018.0	10.0	6.0
419903	tt3466518	Painting a Life: Documenting an Approach to Pa...	2014.0	10.0	6.0

Như đã trình bày ở trên, Mỹ đang là quốc gia sản xuất phim nhiều nhất trên thế giới. Bên trên là TOP 10 phim được đánh giá cao nhất nước Mỹ. Khi đối chiếu với TOP 10 thế giới, ta có thể thấy rằng 9/10 phim trong bảng xếp hạng do Mỹ sản xuất.

2.3. Làm sạch, tiền xử lý dữ liệu

Trong dữ liệu thu thập được, bên cạnh các trường dữ liệu kiểu số (“rating”, “rating_count”, “year”) cũng tồn tại các trường dữ liệu kiểu văn bản như trường “story_line”, “director” hay kiểu danh sách các phân tử dạng văn bản như “genres”, “stars”, “language” cần xử lý.

2.3.1. Xử lý dữ liệu các trường “budget” và “gross”

Các trường “budget” và “gross” lưu trữ dữ liệu kiểu văn bản có dạng là đơn vị tiền tệ rồi tới giá trị. Bên cạnh đó, hai trường này cũng chứa một lượng lớn đơn vị tiền tệ nên cần thiết phải quy đổi các đơn vị tiền tệ này sang USD để có thể đồng bộ. Chúng em tiến hành tách dữ liệu thành hai phần là đơn vị tiền tệ và giá trị. Sau đó, gán giá trị mới cho trường bằng giá trị quy đổi sang USD. Tỷ giá quy đổi tiền tệ cũng được thu thập bằng API qua trang web xe.com. Ngoài ra, một số đơn vị tiền tệ được coi là đã bị loại bỏ nếu không tồn tại trên hệ thống của trang web trên và được gán bằng NaN.

2.3.2. Xử lý dữ liệu trường “story_line”

Trường “story_line” mô tả tóm tắt nội dung của bộ phim. Tuy nhiên, kích thước của trường dữ liệu này không nhỏ cũng như không đồng nhất giữa các bản ghi nên ta chỉ trích xuất ra những đặc trưng tiêu biểu và gán giá trị mới cho trường đó bằng điểm trung bình của các đặc trưng.

Đầu tiên, ta sử dụng mô hình TF-IDF để trích xuất ra 5 đặc trưng tiêu biểu của trường dữ liệu này

```
data['story_line']
0      [american, life, story, tragic, western]
1      [colleen, liking, north, party, yoga]
2      [30, duty, report, serve, three]
3      [career, kuklinski, mob, much, persuade]
4      [academy, marriage, star, stars, staying]
...
10084   [academic, short, superstars, try, two]
10085   [bob, calvin, high, says, school]
10086   [crew, earth, josh, orb, spaceship]
10087   [disappeared, everybody, five, kids, wake]
10088   [abusive, addicted, family, father, flees]
Name: story_line, Length: 10089, dtype: object
```

Sau đó, ta sẽ tập hợp các trường này thành một từ điển với khóa là từ đó và giá trị là rating trung bình của từ đó

```
In [17]: sl_map
Out[17]: {'american': 6.3442477876106205,
          'life': 6.556346153846157,
          'story': 6.5,
          'tragic': 6.572727272727272,
          'western': 5.866666666666667,
          'colleen': 4.3,
          'liking': 4.3,
          'north': 6.3055555555555554,
          'party': 6.206976744186044,
          'yoga': 4.3,
          '30': 6.411111111111111,
          'duty': 6.0,
          'report': 6.65,
          'serve': 5.2,
          'three': 6.3304347826086955,
          'career': 6.38695652173913,
          'kuklinski': 6.8,
          'mob': 6.327999999999999,
          'much': 6.615999999999998,
          'persuade': 6.0}
```

Từ đây, ta đánh giá được thứ hạng của các từ và giá trị của trường “story_line” được gán bằng giá trị trung bình thứ hạng của các từ đặc trưng ở trên.

2.3.3. Xử lý dữ liệu trường “stars”

Trong trường dữ liệu stars, mỗi bản ghi bao gồm một danh sách đối tượng list bao gồm tên của 5 diễn viên nổi tiếng nhất. Để chuyển dạng dữ liệu này thành số, tương tự với cách mà chúng em đã làm với trường story line, chúng em sẽ tính ra điểm cho các ngôi sao dựa trên điểm rating trung bình. Cụ thể hơn, tên của tất các ngôi sao trong tập dữ liệu sẽ được trích xuất, từ đó chúng em có thể tính được điểm rating trung bình từ các bộ phim mà họ từng đóng. Như kết quả, một từ điển gồm tên diễn viên và số điểm trung bình của họ được tạo ra.

```
{'Elizabeth Olsen': 6.1,
'Maddie Hasson': 5.8,
'Tom Hiddleston': 6.6875,
'Harley Quinn Smith': 4.3,
'Johnny Depp': 6.786486486486487,
'Lily-Rose Depp': 4.3,
'Chris Klein': 5.400000000000001,
'Elijah Wood': 6.53125,
'Jon Bernthal': 6.76,
'Chris Evans': 6.641176470588236,
'James Franco': 6.078260869565218,
'Michael Shannon': 6.621428571428569,
'Adam Driver': 7.233333333333334,
'Julia Greer': 8.0,
'Scarlett Johansson': 6.846153846153846,
'Jonathan Lipnicki': 5.5,
'Richard E. Grant': 6.657142857142858,
'Rollo Weeks': 5.7,
'Anthony Chau-Sang Wong': 6.5,
```

Sử dụng từ điển trên, danh sách gồm tên của 5 ngôi sao nổi tiếng nhất của mỗi bản ghi được tính như sau: tổng điểm của các diễn viên trên tổng điểm số diễn viên trong danh sách.

2.3.4. Xử lý dữ liệu trường “director”, “language”, “country” và “genres”

Do mỗi bản ghi điều có duy nhất một giá trị theo dạng phân loại tại các trường “director”, “language”, “country” và “genres”, chúng em sẽ gán một giá trị số cho các dữ liệu phân loại áp dụng cho từng trường.

Trước khi xử lý:

	director	language	country	genres
0	Marc Abraham	English	USA	Biography,Drama,Music
1	Kevin Smith	English,French,German	USA	Action,Comedy,Fantasy
2	Bryan Gunnar Cole	English	USA	Drama
3	Ariel Vromen	English	USA	Biography,Crime,Drama
4	Noah Baumbach	English,Spanish	UK,USA	Comedy,Drama,Romance
...
10084	Olivia Wilde	English,Mandarin,Spanish	USA	Comedy
10085	Rawson Marshall Thurber	English	China,USA	Action,Comedy,Crime
10086	Brian Robbins	English	USA	Adventure,Comedy,Family
10087	David Moreau	French	Belgium,France	Adventure,Family,Fantasy
10088	Patrick Tam	Cantonese	Hong Kong	Drama

Sau khi xử lý:

	director	language	country	genres
0	2793	440	1177	223
1	2522	467	1177	33
2	599	440	1177	371
3	353	440	1177	211
4	3321	917	1172	261
...
10084	3360	842	1177	235
10085	3690	440	493	29
10086	568	440	1177	114
10087	1028	970	191	147
10088	3431	232	935	371

2.4. Xây dựng mô hình gợi ý

2.4.1. Mục tiêu

Trên thực tế, người dùng không thường xuyên là vô cùng phổ biến, vì vậy, đề xuất dựa theo phiên là hữu ích trong việc gợi ý cho tập người dùng ẩn danh, không đăng nhập hoặc định danh người dùng không được xác định vì một số lý do kỹ thuật hoặc quyền riêng tư.

2.4.2. Content based

a. Cơ sở lý thuyết

Nguyên tắc là tìm một số lượng training samples được huấn luyện được xác định trong khoảng cách gần nhất với điểm xác định. Số lượng mẫu có thể được xác định là một hằng số (K-Nearest Neighbors) hoặc thay đổi dựa trên mật độ của các điểm (radius-based neighbor learning).

Khoảng cách có thể được xác định thông qua nhiều loại thước đo khác nhau, trong đó khoảng cách Euclid được sử dụng nhiều nhất.

Dữ liệu huấn luyện được ghi nhớ thông qua một số cách như chuyển đổi qua một cấu trúc lập chỉ mục nhanh như Ball Tree hoặc KD Tree.

Có 3 thuật toán trong Nearest Neighbor bao gồm: Brute Force, K-D Tree, Ball Tree

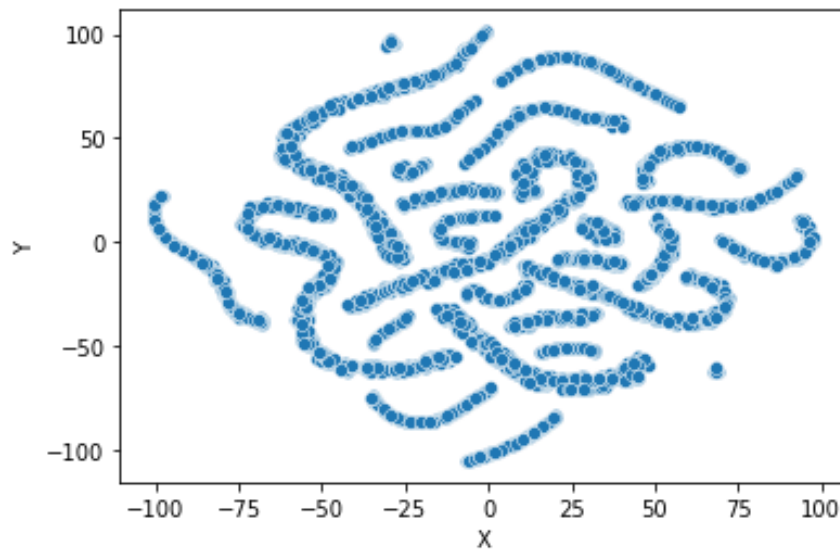
- Brute Force: Nó thực hiện tính toán tất cả các khoảng cách giữa các cặp điểm trong bộ dữ liệu. Với N mẫu và D chiều thì độ phức tạp thuật toán lên tới $O[DN^2]$. Do đó thuật toán này chỉ khả thi với những bộ dữ liệu nhỏ.
- K-D Tree: cấu trúc dữ liệu này sẽ cố gắng giảm số phép tính khoảng cách thông qua việc mã hóa thông tin khoảng cách được tính từ các mẫu. Nếu

điểm A ở xa với điểm B mà điểm B lại gần với điểm C thì A vs C sẽ rất x chúng ta không cần phải tính khoảng cách điểm đó nữa. Từ đó sẽ giảm độ thuật toán xuống $O[DN\log(N)]$ hoặc tốt hơn. Mặc dù K-D Tree hiệu quả trong việc tìm kiếm các neighbor ở có chiều thấp nhưng nó lại kém hiệu quả với các neighbor có số lượng chiều cao

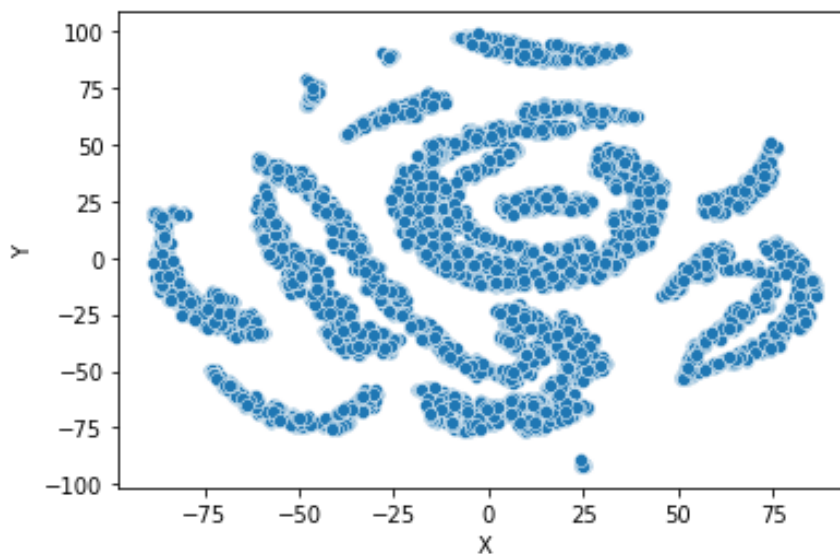
- Ball Tree: cấu trúc này giải quyết vấn đề tìm kiếm với các neighbor có số lượng chiều cao

Chọn Brute Force làm thuật toán cho bài toán do mẫu có 12 chiều và số lượng mẫu là 10089 mẫu

b. Chuẩn bị dữ liệu



Hình 14: Trước khi min max scale



Hình 15: Sau khi min-max scaler

c. Kết quả

Với dữ liệu training của mô hình

movie_id	year	rating_count	story_line	genres	country	language	budget	gross	runtime	director	stars	rating
1490785	0.000154	0.000478	0.000495	1.699738e-05	0.000090	0.000034	0.990878	0.134757	9.375234e-06	3.345273e-06	0.000411	6.0
3838992	0.000403	0.002443	0.000499	6.599801e-06	0.000235	0.000093	0.999970	0.007317	1.759947e-05	6.999789e-06	0.000653	4.0
768183	0.001003	0.001359	0.002648	1.854918e-04	0.000588	0.000220	0.999956	0.008329	4.599796e-05	1.987412e-05	0.003017	6.0
1491044	0.000183	0.006365	0.000720	1.920289e-05	0.000107	0.000040	0.910090	0.414361	9.646952e-06	3.658561e-06	0.000616	7.0
7653254	0.000109	0.012855	0.000392	1.402884e-05	0.000063	0.000049	0.999756	0.017936	7.363796e-06	2.273639e-06	0.000580	8.0
...
1489887	0.000079	0.003508	0.000192	9.187826e-06	0.000046	0.000033	0.234583	0.972090	3.987908e-06	1.477003e-06	0.000401	7.0
1489889	0.000009	0.000709	0.000028	1.302439e-07	0.000002	0.000002	0.224558	0.974460	4.805551e-07	1.772515e-07	0.000025	6.0
765476	0.000025	0.000485	0.000045	1.443645e-06	0.000015	0.000006	0.759813	0.650141	1.139720e-06	4.976776e-07	0.000053	5.0
3838034	0.000264	0.000184	0.000473	1.922206e-05	0.000025	0.000127	0.950119	0.311888	1.281470e-05	5.289929e-06	0.000197	5.0
768114	0.000729	0.000303	0.002939	1.348016e-04	0.000340	0.000084	0.937180	0.348826	4.396494e-05	1.507888e-05	0.002229	7.0

10089 rows × 12 columns

Với phim Yoga Hosers được hiển thị thì hệ thống sẽ gợi ý thêm 5 phim bên dưới thông qua model đã training

movie_id	title	distance
1	Yoga Hosers	0.000000
9361	South of Heaven, West of Hell	0.002206
5225	3 Blind Mice	0.002287
397	Gallowwalkers	0.002369
2377	Blood Feast	0.002456
903	A Heartbeat Away	0.002490

2.4.3. Session-based

Đề xuất dựa trên phiên cho phép gợi ý dựa trên phiên hiện tại của người dùng thay vì dựa trên lịch sử tương tác dài hạn.

a. Mô hình Word2Vec

Word2vec là một trong những mô hình đầu tiên về Word Embedding sử dụng mạng neural nhưng vẫn còn khá phổ biến ở thời điểm hiện nay nhờ có khả năng vector hóa từng từ dựa trên các từ mục tiêu và các từ context (văn cảnh). Về mặt toán học đây là việc ánh xạ từ một tập các từ (vocabulary) sang một không gian vector, với mỗi vector được biểu diễn bởi n số thực. Sau quá trình huấn luyện mô hình bằng thuật toán lan truyền ngược, trọng số được cập nhật liên tục rồi từ đó có thể tính được những khoảng cách quen thuộc như Euclidean, cosine... Những từ càng “gần” nhau về mặt khoảng cách thường là các từ hay xuất hiện cùng nhau trong văn cảnh, các từ đồng nghĩa, các từ thuộc cùng một trường từ vựng...

Áp dụng cho bài toán Session Based của chúng ta, với dữ liệu về ID phim và ID người dùng cùng với thời gian xác định thì ta có thể có được các phiên của người dùng tương ứng với một chuỗi các phim được cho là có mức độ quan tâm từ người dùng gần như tương

đương với nhau. Từ đó chúng ta có thể đưa danh sách các bộ phim (vocabulary) với các phiên xem như là các câu chứa phim mục tiêu và các phim xung quanh (context). Khi áp dụng với một tập dữ liệu đủ lớn gồm N người dùng thì mô hình sẽ có khả năng tổng quát hóa và gợi ý được bộ phim phù hợp nhất.

b. Phương án tiếp cận

Đối với bộ dữ liệu đã thu thập ta vẫn chưa có được dữ liệu về sự tương tác của người dùng. Nhận thấy bộ dữ liệu MovieLen – bộ dữ liệu lấy từ web site Movielens.org được cung cấp bởi GroupLens Research - có được thông tin về tương tác của người dùng nên quyết định của nhóm là hợp nhất với bộ thu thập được với bộ MovieLen và có được kết quả với 3706 phim và hơn 1 triệu tương tác người dùng kèm theo thời gian.

	user_id	movie_id	rating	timestamp
0	1	1193	5	978300760
1	1	661	3	978302109
2	1	914	3	978301968
3	1	3408	4	978300275
4	1	2355	5	978824291
...
1000204	6040	1091	1	956716541
1000205	6040	1094	5	956704887
1000206	6040	562	5	956704746
1000207	6040	1096	4	956715648
1000208	6040	1097	4	956715569

1000209 rows × 4 columns

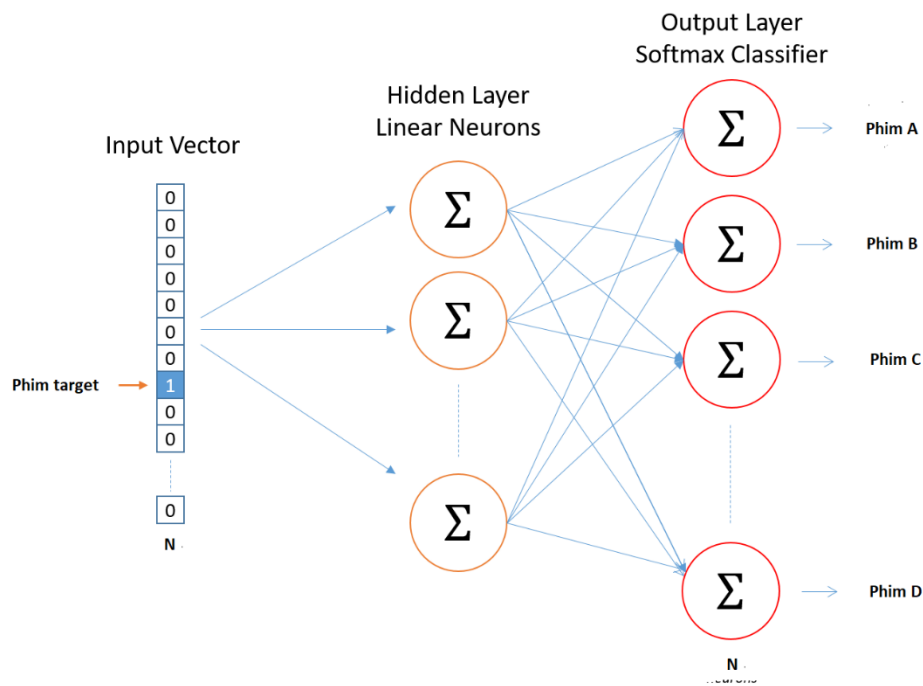
Hình 16: Dữ liệu sau khi kết hợp dữ liệu thu thập được với bộ MovieLen IM

Dữ liệu được chuyển sang dưới dạng các session để đưa vào mô hình để huấn luyện.

```
Session 1 ['858', '2384', '593', '1961', '2019', '1419',
Session 2 ['111', '282', '2067', '1230', '930', '947',
Session 3 ['3396', '920', '1210', '2146', '1387', '356',
Session 4 ['1882', '702', '1267', '2028', '3508', '3148',
Session 5 ['1721', '1883', '3438', '2376', '2428', '2683
```

Word2Vec có 2 hướng tiếp cận: Skip-gram và CBOW. Với Skip-gram thì sẽ dùng các phim mục tiêu để dự đoán các phim tương tự nó và CBOW thì ngược lại. Trong đó thì Skip-gram huấn luyện chậm hơn nhưng lại làm việc tốt hơn trên các tập dữ liệu nhỏ và do đặc trưng của mô hình nên khả năng vector hóa cho các từ ít xuất hiện tốt hơn CBOW

Để đưa phim vào để xử lý trong neural network chúng ta cần phải xây dựng 1 từ điển chứa N phim. Sau đó chuyển đầu vào của mô hình là một bộ phim dưới dạng one-hot vector có N phần tử với giá trị ở id phim đấy = 1 còn tất cả còn lại là = 0. Đầu vào được đi qua A neuron trong lớp ẩn (A có thể tùy chỉnh) và không sử dụng hàm kích hoạt ở đây. Output cũng là 1 single vector (cũng có N phần tử) được sử dụng hàm softmax ở lớp đầu ra nên cho ra một phân phối xác suất của tất cả các phim trong từ điển.



Hình 17: Kiến trúc mạng Skip-gram

c. *Cải thiện tham số*

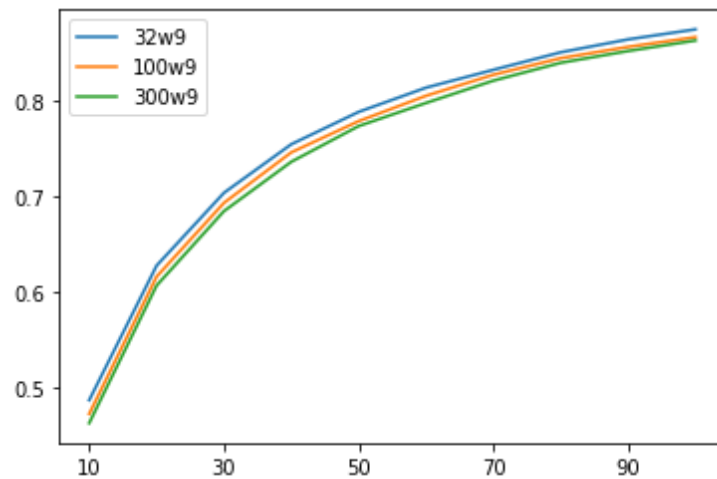
Với mô hình này chúng ta có được những tham số như sau:

- Size: Số lượng neural ở lớp ẩn
- Window: Kích thước của một khoảng các phim xung quanh được cho là liên quan với nhau
- Negative: Số lượng phim nhiễu sẽ được loại bỏ
- Sample: Ngưỡng để giảm lấy mẫu cho các phim có tần suất xuất hiện cao hơn giá trị này
- Ns_exponent: chỉ ra phim nào sẽ được ưu tiên khi lấy mẫu dựa theo tương quan tần suất xuất hiện của nó với phim context
- Worker = $(2 * multiprocessing.cpu_count() + 1)$: Số lượng worker thread để chạy training model
- Iter: Số lượng vòng lặp

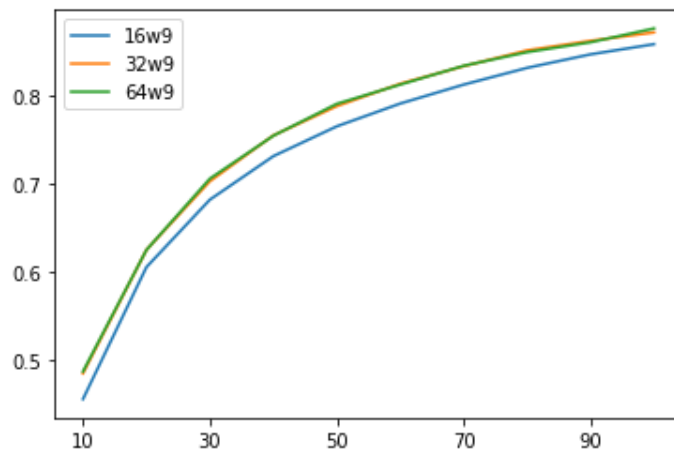
Qua thực nghiệm chúng em chọn tham số theo:

- Size: Số lượng neural ở lớp ẩn

Ta giữ nguyên window = 9 và lần lượt thay đổi giá trị của size



Hình 18: So sánh model với size <100 , $=100$, >100 thì <100 mang lại kết quả tốt nhất

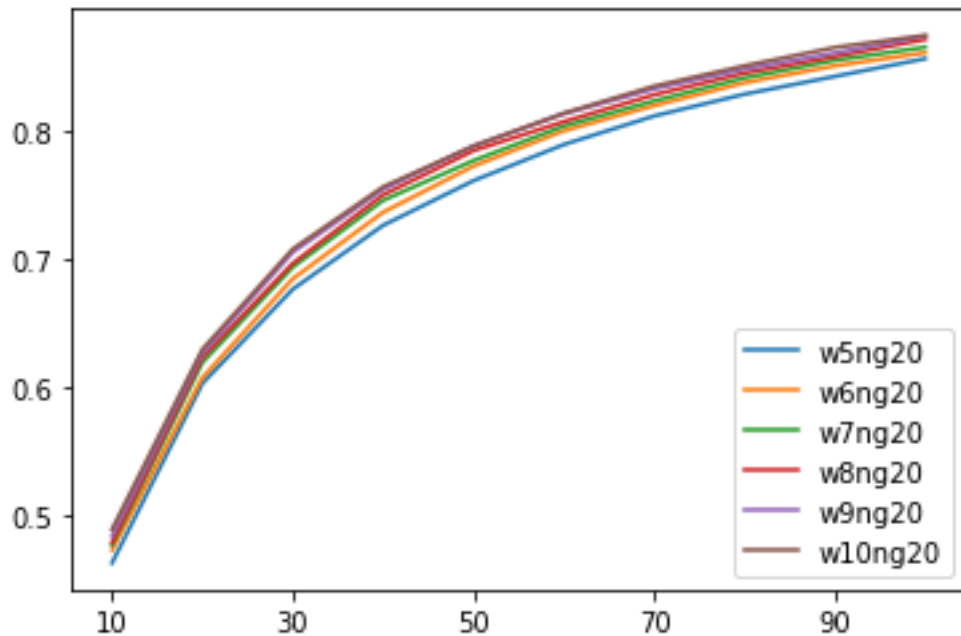


Hình 19: So sánh model với 3 size < 100

Ta được kết quả cho size = 64 có kết quả tốt hơn một chút so với size = 32 nên ta sẽ chọn size = 64

- Window:

Giữ nguyên giá trị negative = 20 và lần lượt thay đổi window từ giá trị 5 đến 10 ta có được so sánh như sau

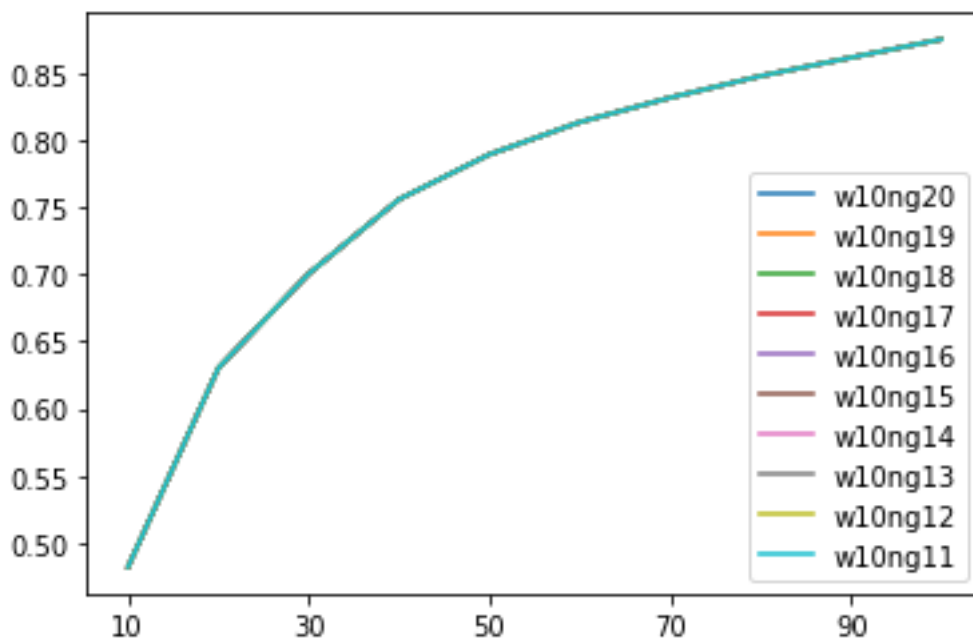


Hình 20: So sánh model với giá trị window chạy từ 5 đến 10

Từ đó ta được kết quả cho thấy với window = 9 cho ra kết quả cao nhất trong các giá trị so sánh

- Negative:

Thực hiện thay đổi giá trị negative với window được giữ nguyên = 10 thì cho ra được kết quả rằng negative không có ảnh hưởng tới độ chính xác mà chỉ có ảnh hưởng tới tốc độ training



Hình 21: So sánh model với giá trị negative giảm dần từ 20 đến 11

d. Kết quả

Từ dữ liệu đi vào bao gồm tiêu đề của phim, thể loại và số lượng truy cập phim:

clicked	title	genres
111	Taxi Driver (1976)	Drama Thriller
282	Nell (1994)	Drama
930	Notorious (1946)	Film-Noir Romance Thriller
1230	Annie Hall (1977)	Comedy Romance
2067	Doctor Zhivago (1965)	Drama Romance War

Hình 22. Dữ liệu đầu vào của mô hình Session-based

Sau khi đi vào mô hình đã huấn luyện chúng ta cho ra được kết quả gợi ý có độ tương đồng của các phim cao nhất

predict	title	genres	similar
53	Lamerica (1994)	Drama	0.928981
1071	For the Moment (1994)	Romance War	0.921626
2830	Cabaret Balkan (Bure Baruta) (1998)	Drama	0.917278

Hình 23. Kết quả dự đoán của mô hình Session-based

Đánh giá qua hàm ndcg và hit-rate:

	ndcg	hit-rate
score	0.08422	0.64

Hình 24: Kết quả đánh giá model training được qua ndcg và hit-rate

3. KẾT LUẬN

Qua bài tập lớn lần này, chúng em đã tích lũy được nhiều kỹ năng trong các tác vụ thu thập dữ liệu, trực quan hóa dữ liệu, xử lý dữ liệu, trích xuất các đặc trưng từ dữ liệu. Bên cạnh đó, chúng em cũng sử dụng các dữ liệu sau khi xử lý để ứng dụng đơn giản cho các mô hình dự đoán điểm đánh giá và mô hình gợi ý phim cho người dùng.

Qua quá trình thực hiện, các khó khăn là không thể tránh khỏi. Việc thao tác với dữ liệu như thế nào để có thể thấy được các giá trị tiềm ẩn của dữ liệu là một thách thức lớn. Nhưng đây cũng là mục đích chính của môn học “Nhập môn Khoa học dữ liệu”. Chúng em xin chân thành cảm ơn PGS. TS. Thân Quang Khoát đã tạo điều kiện cho chúng em được làm việc, học tập về lĩnh vực rất quan trọng trong cuộc sống.

Trong tương lai, chúng em sẽ cải thiện các kỹ năng để có thể thu thập được dữ liệu tương tác giữa người dùng với các bộ phim để có thể đánh giá được tốt hơn thái độ và xu hướng, sở thích của người dùng. Từ đó, việc gợi ý phim cũng như dự đoán điểm đánh giá của các bộ phim cũng sẽ trở nên hiệu quả hơn.

Phân công công việc

<u>Thành viên</u>	<u>Công việc</u>
Nguyễn Đức Anh	<ul style="list-style-type: none">- Tiền xử lý dữ liệu- Xây dựng mô hình dự đoán
Nguyễn Minh Đăng	<ul style="list-style-type: none">- Thu thập dữ liệu- Mô tả, trực quan hóa dữ liệu
Nguyễn Văn Lương	<ul style="list-style-type: none">- Tiền xử lý dữ liệu- Xây dựng mô hình gợi ý
Bùi Việt Dũng	<ul style="list-style-type: none">- Tiền xử lý dữ liệu- Xây dựng mô hình dự đoán
Võ Đức Quân	<ul style="list-style-type: none">- Xây dựng mô hình gợi ý- Hoàn thành file trình bày

Tài liệu tham khảo

- [1] Greenstein-Messica, A., Rokach, L., & Friedman, M. (2017). *Session-Based Recommendations Using Item Embedding. Proceedings of the 22nd International Conference on Intelligent User Interfaces - IUI '17.* doi:10.1145/3025171.3025197
- [2] Barkan, O., & Koenigstein, N. (2016). *ITEM2VEC: Neural item embedding for collaborative filtering. 2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP).* doi:10.1109/mlsp.2016.7738886
- [3] Phan Huy Hoang, *[Machine Learning] - Ứng dụng mô hình Word2vec cho bài toán session-based Recommender System?!*, viblo.asia
- [4] Bharat Raman, *Predict Movie Ratings via Machine Learning*, kaggle.com