VIỆN TRÍ TUỆ NHÂN TẠO
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

HỆ THỐNG ĐỀ XUẤT PHIM DỰA TRÊN ITEM COLLABORATIVE FILTERING & HADOOP MAPREDUCE

Giảng viên hướng dẫn: TS. Trần Hồng Việt
CN. Đỗ Thu Uyên

Nhóm sinh viên thực hiện: Đỗ Xuân Cảnh
Trần Hùng Đức
Lê Văn Đức

MỤC LỤC

01 | Tổng quan về Big Data và Hadoop MapReduce

Big Data là thuật ngữ chỉ việc xử lý các tập hợp dữ liệu khổng lồ và phức tạp, vượt quá khả năng xử lý của các công cụ và phương pháp truyền thống.

Đặc trưng của Big Data:

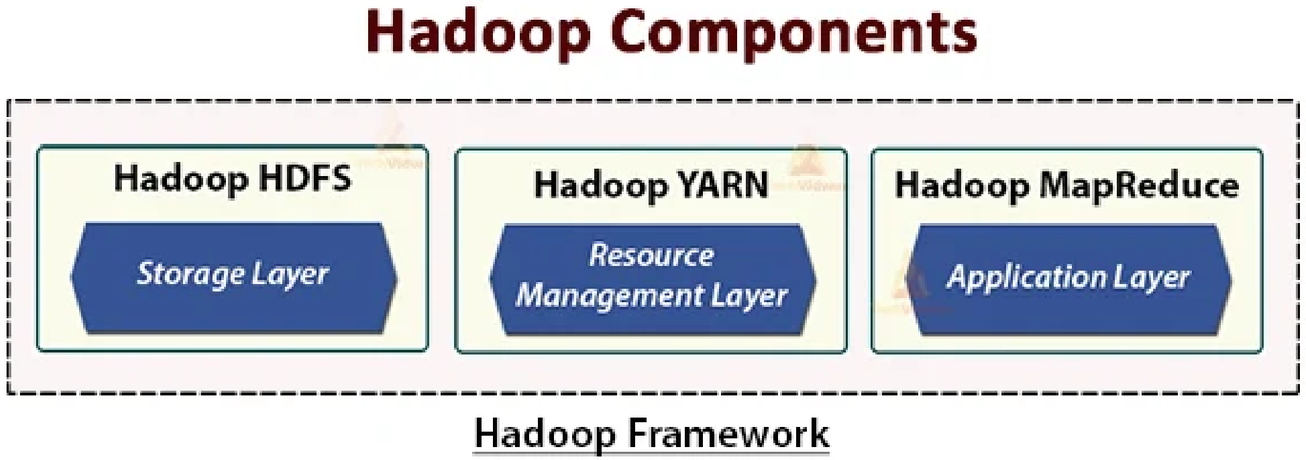
- Volume: Khối lượng dữ liệu lớn cần phải xử lý
- Velocity: Dữ liệu được tạo ra và xử lý với tốc độ cao
- Variety: Dữ liệu có nhiều dạng khác nhau, từ văn bản, hình ảnh, video đến các dạng dữ liệu phi cấu trúc
- Veracity: Đảm bảo tính chính xác và tin cậy của dữ liệu
- Value: Mang tiềm năng kinh tế khổng lồ và việc khai thác đúng cách sẽ giúp tạo ra những thông tin quý giá, thúc đẩy lợi ích kinh doanh và ra quyết định hiệu quả

01 | Tổng quan về Big Data và Hadoop MapReduce

Hadoop là một công nghệ phân tán và mã nguồn mở được sử dụng phổ biến để xử lý và lưu trữ khối dữ liệu lớn trên các cụm máy tính phân tán, được thiết kế để xử lý và lưu trữ dữ liệu lớn một cách hiệu quả.

Hadoop gồm ba thành phần chính:

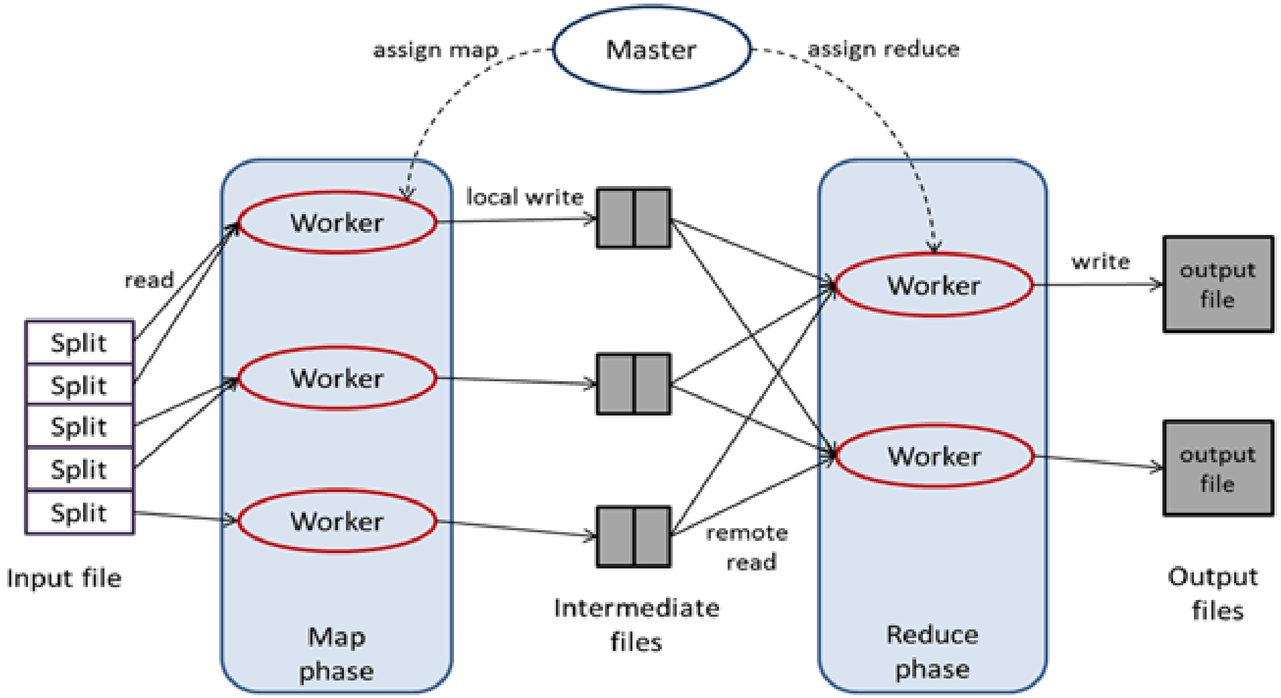
- HDFS: Hệ thống file phân tán, lưu trữ dữ liệu lớn với độ tin cậy cao.
- YARN: Quản lý tài nguyên và lên lịch ứng dụng trên Hadoop.
- MapReduce: Mô hình xử lý dữ liệu lớn qua hai pha Map và Reduce.



MapReduce là một framework dùng để viết các ứng dụng xử lý song song một lượng lớn dữ liệu có khả năng chịu lỗi cao xuyên suốt hàng ngàn cụm máy tính.

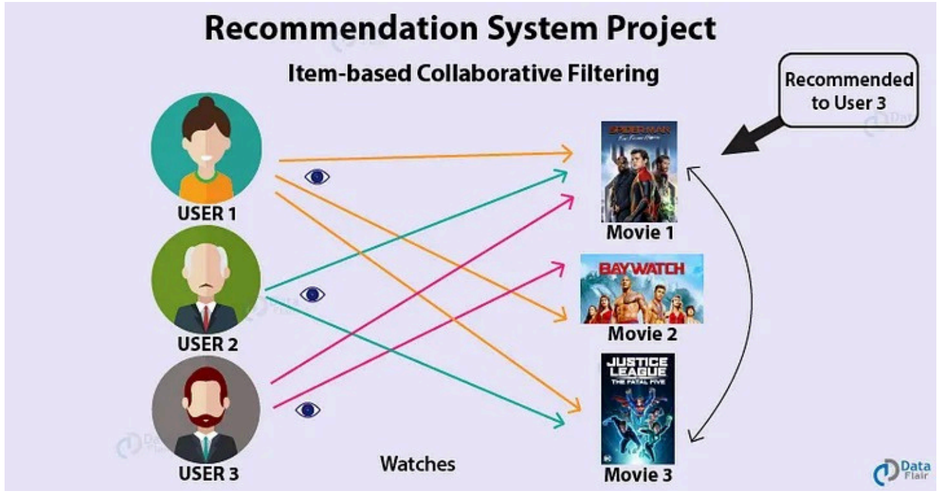
MapReduce thực hiện 2 chức năng chính đó là:

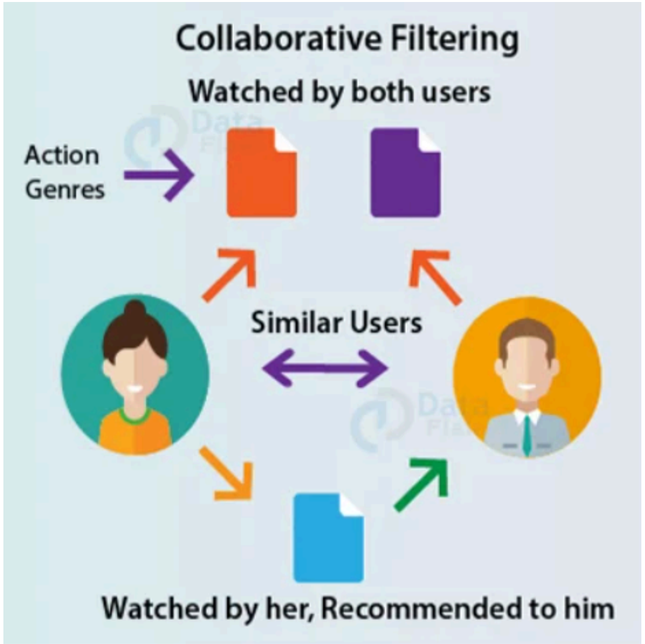
- Map: Thực hiện đầu tiên, có chức năng tải, phân tích dữ liệu đầu vào và chuyển đổi thành tập dữ liệu theo cặp key/value.
- Reduce: Nhận kết quả đầu ra từ tác vụ Map, kết hợp dữ liệu lại với nhau thành tập dữ liệu nhỏ hơn, tạo ra kết quả cuối cùng.



02 | Bài toán gợi ý phim

- Thuật toán: Item-Based Collaborative Filtering
- Ý tưởng: Dự đoán và gợi ý những phim mà người dùng có thể thích dựa trên các phim họ đã xem và đánh giá.
  - Input:
    - Tên phim, tên người dùng và đánh giá của người dùng
  - Output:
    - Top danh sách phim gợi ý



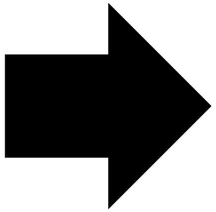


Input

	Phim 1	Phim 2	Phim 3	Phim 4
Đỗ Cảnh	3	3	?	4
Hùng Đức	3	?	5	2
Lê Đức	5	?	4	?
Văn A	?	3	?	1

Bước 1: Xây dựng ma trận đồng xuất hiện

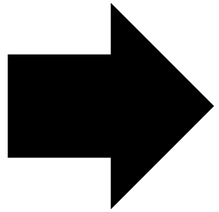
	Phim 1	Phim 2	Phim 3	Phim 4
Đỗ Cảnh	3	3	?	4
Hùng Đức	3	?	5	2
Lê Đức	5	?	4	?
Văn A	?	3	?	1



	Phim 1	Phim 2	Phim 3	Phim 4
Phim 1	3	1	2	2
Phim 2	1	2	0	2
Phim 3	2	0	2	1
Phim 4	2	2	1	3

Bước 2: Chuẩn hóa

	Phim 1	Phim 2	Phim 3	Phim 4
Phim 1	3	1	2	2
Phim 2	1	2	0	2
Phim 3	2	0	2	1
Phim 4	2	2	1	3



	Phim 1	Phim 2	Phim 3	Phim 4
Phim 1	0.375	0.2	0.4	0.25
Phim 2	0.125	0.4	0	0.25
Phim 3	0.25	0	0.4	0.125
Phim 4	0.25	0.4	0.2	0.375

03 Thuật toán Item Collaborative Filtering

	Phim 1	Phim 2	Phim 3	Phim 4
Đỗ Cảnh	3	3	?	4
Hùng Đức	3	?	5	2
Lê Đức	5	?	4	?
Văn A	?	3	?	1

×

	Phim 1	Phim 2	Phim 3	Phim 4
Phim 1	0.375	0.2	0.4	0.25
Phim 2	0.125	0.4	0	0.25
Phim 3	0.25	0	0.4	0.125
Phim 4	0.25	0.4	0.2	0.375

=

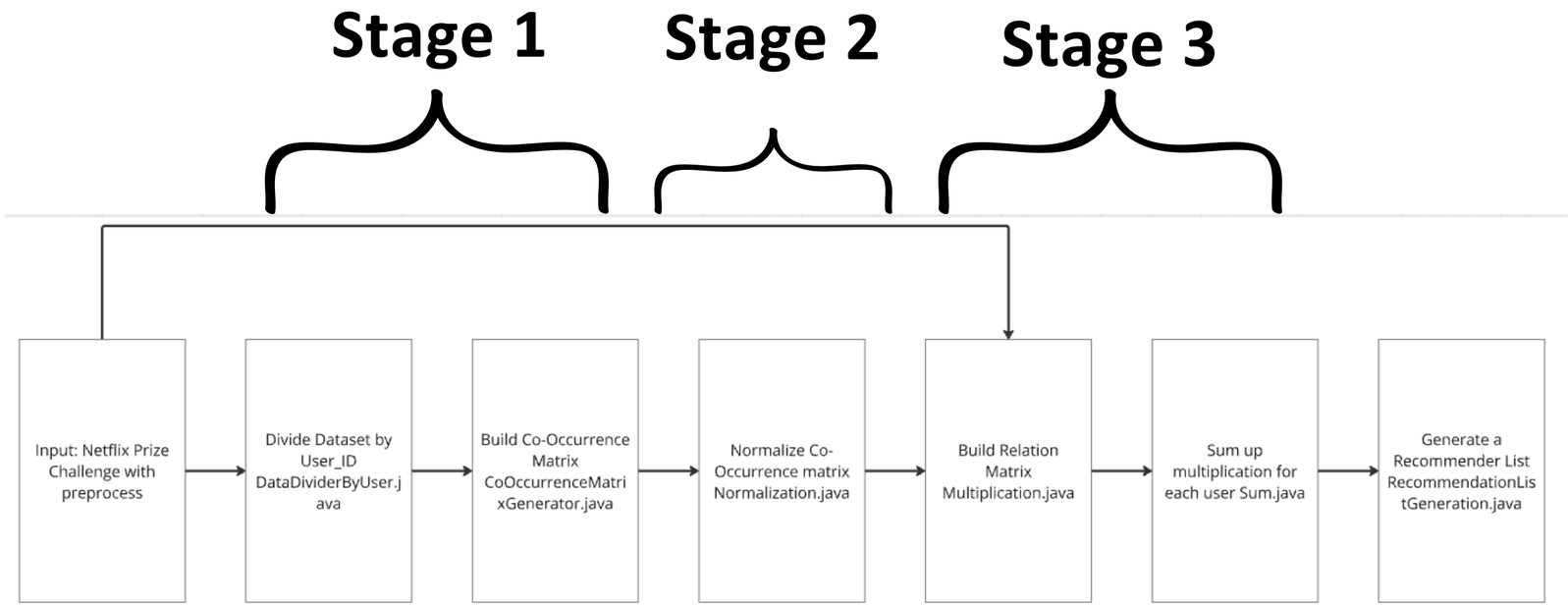
	Phim 1	Phim 2	Phim 3	Phim 4
Đỗ Cảnh	2.5	3.4	2.0	3.0
Hùng Đức	2.875	1.4	3.6	2.125
Lê Đức	2.875	1.0	3.6	1.75
Văn A	0.625	1.6	0.2	1.125

Bước 3: Xây dựng ma trận đánh giá

04 Item Collaborative Filtering kết hợp MapReduce

user	movie	rating
1	1	3
1	2	3
2	1	4
3	2	3
4	1	1

Sample data



Data Divide by User ID

Input

user	movie	rating
1	1	3
1	2	3
1	4	4
2	3	5
4	2	3
4	4	1
3	4	2

Map

Tạo cặp key-value với key là userID và value là chuỗi movieID:rating

Reduce

Gộp dữ liệu đánh giá phim cho mỗi người dùng thành một chuỗi duy nhất

Output

user	movie:rating
1	1:3, 2:3, 4:4
2	1:3, 3:5, 4:2
3	1:5, 3:4
4	2:3, 4:1

Build Co-Occurrence Matrix

Input

user	movie:rating
1	1:3, 2:3, 4:4
2	1:3, 3:5, 4:2
3	1:5, 3:4
4	2:3, 4:1

Map

Với mỗi cặp phim trong danh sách, Mapper tạo ra một cặp key-value, trong đó Key là cặp phim và Value là 1, đại diện cho việc hai phim đã xuất hiện cùng nhau.

Reduce

Tính tổng số lần cặp phim xuất hiện.

Output

movie:movie	relation
1:1	3
1:2	1
1:3	2
1:4	2
2:1	1
2:2	2

Normalize co-occurrence matrix

Input

movie:movie	relation
1:1	3
1:2	1
1:3	2
1:4	2
2:1	1
2:2	2

Map

Tách cặp phim và mức độ tương quan từ dòng dữ liệu, sau đó ghi ra context với key là movie_1 và value là movie_2

Reduce

Chuẩn hóa mỗi đơn vị của ma trận đồng xuất hiện.

Output

movie	movie=relation
1	1=0.375
2	1=0.125
3	1=0.25
4	1=0.25
1	2=0.2
2	2=0.4

04 Item Collaborative Filtering kết hợp MapReduce

Input

user	movie	rating
1	1	3
1	2	3
1	4	4
2	3	5
4	2	3
4	4	1
3	4	2

movie	movie=relation
1	1=0.375
2	1=0.125
3	1=0.25
4	1=0.25
1	2=0.2
2	2=0.4

Multiplication

Cooccurrence Map

Tách cặp phim và mức độ tương quan từ dòng dữ liệu và gửi chúng đến reducer với key là movie_2 và value là movie_1=relation

Rating Map

Tách cặp user, movie, và rating từ dòng dữ liệu và gửi chúng đến reducer với key là movie và value là user

Reduce

Tính toán tích của mỗi mục trong ma trận đồng xuất hiện với ma trận đánh giá.

Output

user:movie	relation*rating
1:1	1.125
1:2	0.6
1:3	1.2
1:4	0.75
2:1	1.125
2:2	0.6

Sum

Input

user:movie	relation*rating
1:1	1.125
2:1	1.125
3:1	1.875
1:2	0.6
2:2	0.6
3:2	1.0
1:3	1.2

Map

Tách cặp user:movie và giá trị relation*rating và gửi chúng đến reducer với key là user:movie và value là giá trị tương ứng.

Reduce

Tính tổng của các giá trị được gửi từ mapper cho mỗi key

Output

user:movie	relation*rating
1:1	2.5
1:2	3.4
1:3	2.0
1:4	3.0
2:1	2.875
2.2	1.4

Recommendation

Input

user:movie	recommend_score
1:1	2.5
1:2	3.4
1:3	2.0
1:4	3.0
2:1	2.875
2.2	1.4

Map

Key là user:movie và value là giá trị recommend đã tính ở bước Sum. Mapper tách cặp user:movie và recommend từ dòng dữ liệu, sau đó gửi chúng đến reducer với key là user và value là movie:recommend

Reduce

Tìm ra k movies có recommend_score cao nhất cho mỗi người dùng. Reducer nhận các giá trị recommend_score.
Sử dụng PriorityQueue để lấy k movies có recommend_score cao nhất với mỗi người dùng.Duyệt qua danh sách các phim và đánh giá, thêm vào hàng đợi
Kết quả là một chuỗi key-value, trong đó key là user và value là movie.

Output

User	Movie:Relation
1	3:2.0
1	1:2.5
1	4:3.0
1	2:3.4
2	2:1.4
2	4:2.125
2	1:2.875

05 | Thử nghiệm

Input

```
1488844,1,3
822109,1,5
885013,1,4
30878,1,4
823519,1,3
893988,1,3
124105,1,4
1248029,1,3
1842128,1,4
2238063,1,3
1503895,1,4
2207774,1,5
2590061,1,3
2442,1,3
543865,1,4
1209119,1,4
804919,1,4
1086807,1,3
1711859,1,4
372233,1,5
1080361,1,3
1245640,1,3
558634,1,4
2165002,1,4
1181550,1,3
```

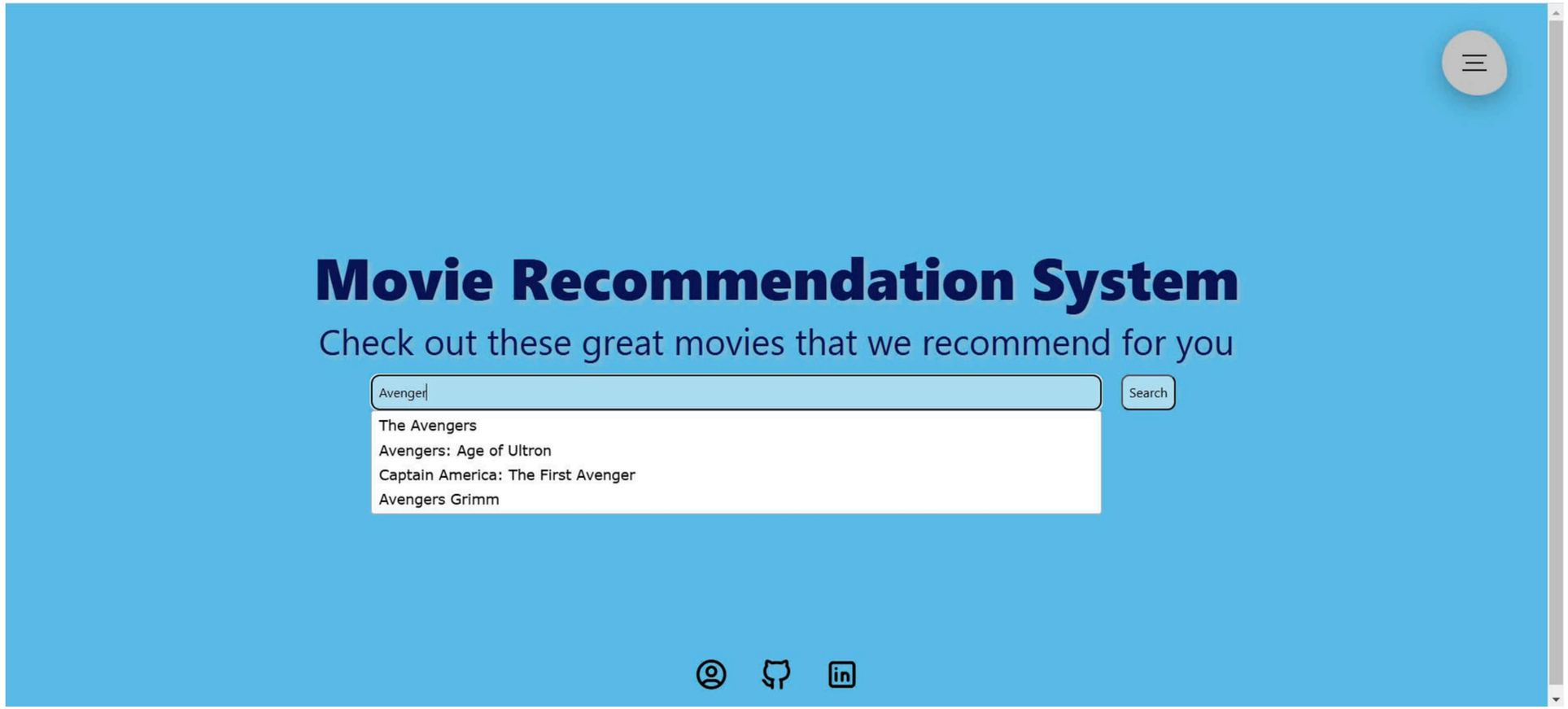
```
1,Dinosaur Planet
2,Isle of Man TT 2004 Review
3,Character
4,Paula Abdul's Get Up & Dance
5,The Rise and Fall of ECW
6,Sick
7,8 Man
8,What the #$*! Do We Know!?
9,Class of Nuke 'Em High 2
10,Fighter
11,Full Frame: Documentary Shorts
12,My Favorite Brunette
13,Lord of the Rings: The Return of
14,Nature: Antarctica
15,Neil Diamond: Greatest Hits Live
16,Screamers
17,7 Seconds
18,Immortal Beloved
19,By Dawn's Early Light
20,Seeta Aur Geeta
```

Output

output > RecommendName > part-r-00000 > data

```
7   What the #$*! Do We Know!?
7   Sick
7   Full Frame
7   Immortal Beloved
7   Character
307 What the #$*! Do We Know!?
307 Sick
307 Full Frame
307 Immortal Beloved
307 Character
424 Immortal Beloved
424 My Favorite Brunette
424 Character
424 Screamers
424 Inspector Morse 31
462 7 Seconds
462 Never Die Alone
462 Chump Change
462 Strange Relations
462 Screamers
491 7 Seconds
491 Never Die Alone
491 Chump Change
491 Strange Relations
491 Screamers
685 The Rise and Fall of ECW
685 Isle of Man TT 2004 Review
685 8 Man
685 Class of Nuke 'Em High 2
685 Nature
695 What the #$*! Do We Know!?
```

05 | Thử nghiệm



06 | Kết luận

- Hiểu về thuật toán Item-based Collaborative Filtering
- Triển khai ý tưởng và giải pháp cho việc sử dụng Hadoop MapReduce trong việc triển khai thuật toán Item-based Collaborative Filtering
- Xây dựng sơ đồ thuật toán và triển khai thành công chương trình demo
- Hạn chế:
  - Chạy Pseudo Distributed Mode
  - Không thể xử lí data > 100k dòng
- Future work:
  - Mở rộng với bộ data lớn hơn
  - Sử dụng thêm thuật toán khác