Recommender System with Mapreduce - 2

赵敏 老师

扫描二维码关注微信/微博
获取最新IT面试情报及权威解答

微信: ninechapter
知乎专栏: http://zhuanlan.zhihu.com/jiuzhang
微博: http://www.weibo.com/ninechapter
官网: www.jiuzhang.com

Recap

- What is recommender System
- What is Item CF
- How to implement Item CF with MR
  - Build co-occurrence matrix

Divide Data by User ID

User_id	Movie_id: Rating	Movie_id: Rating
1	1:10	2: 8
2	1: 8	2: 5.5
3	2: 4.5	8: 9
4	2: 6.5	8: 2.5

Build Co-Occurrence Matrix

Co-Occurrence Matrix

MovieA: MovieB	Relation
1: 1	2
1: 2	2
2: 1	2
2: 2	4
2: 8	2
8: 2	2
8: 8	2

Build Co-occurrence Matrix

九章算法

Input

User_id	Movie_id: Rating	Movie_id: Rating
1	1:10	2: 8
2	1: 8	2: 5.5
3	2: 4.5	8: 9
4	2: 6.5	8: 2.5

Output

MovieA: MovieB	Relation
1: 1	2
1: 2	2
2: 1	2
2: 2	4
2: 8	2
8: 2	2
8: 8	2

Build Co-occurrence Matrix: Mapper

User_id	Movie_id: Rating	Movie_id: Rating
1	1:10	2: 8
2	1: 8	2: 5.5
3	2: 4.5	8: 9
4	2: 6.5	8: 2.5

MovieA: MovieB	Relation
1:1	1
1:2	1
2:1	1
2:2	1
1:1	1
1:2	1
2:1	1
2:2	1

....

Build Co-occurrence Matrix: Reducer

MovieA: MovieB	Relation
1:1	1
1:2	1
2:1	1
2:2	1
1:1	1
1:2	1
2:1	1
2:2	1

....

Merge

MovieA: MovieB	Relation
1: 1	2
1: 2	2
2: 1	2
2: 2	4
2: 8	2
8: 2	2
8: 8	2

Implement recommender system with MapReduce

We have co-occurrence matrix now, what do we miss?

Implement recommender system with MapReduce

- Build rating matrix
- Multiply co-occurrence matrix and rating matrix
- Generate recommender list

Rating matrix

User Rating Matrix

```
1,10001,5.0
1,10002,3.0
1,10003,2.5
2,10001,2.0
2,10002,2.5
2,10003,5.0
2,10004,2.0
3,10001,2.0
3,10004,4.0
3,10005,4.5
3,10007,5.0
4,10001,5.0
4,10003,3.0
4,10004,4.5
4,10006,4.0
5,10001,4.0
5,10002,3.0
5,10003,2.0
5,10004,4.0
5,10005,3.5
5,10006,4.0
```

Multiply co-occurrence matrix and rating matrix

Co-Occurrence Matrix

MovieA: MovieB	Relation
1: 1	5
1: 2	3
1: 4	2
2: 1	3
2: 2	4
2: 3	2

User Rating Matrix

User_id	Movie_id: Rating
1	1:10
1	2: 8
2	1: 8
2	3: 4.5
4	2: 5.5
4	4: 6.5

Multiply co-occurrence matrix and rating matrix

Restore co-occurrence matrix input to a real matrix

Multiply co-occurrence matrix and rating matrix

九章算法

Co Matrix

	M1	M2	M3	M4	M5
M1	2/6	2/6	1/6	1/6	0
M2	2/1 1	4/1 1	2/1 1	2/1 1	1/1 1
M3	1/6	2/6	2/6	0	1/6
M4	1/5	2/5	0	2/5	0
M5	0	1/3	1/3	0	1/3

Rating Matrix

Movie	User B rating
M1	3
M2	7
M3	8
M4	0
M5	0

Result Matrix

User_id	movie:rating
userB	1: 4.66
userB	2: 4.54
userB	3: 5.5
userB	4: 3.4
userB	5: 5

Mapper

Co-Occurrence Matrix → HashMap:

[movie1 : {movie1, movie2, 8}{movie1, movie3, 5}{movie1, movie7, 6}]

[movie2 : {movie2, movie1, 8}{movie2, movie5, 9}{movie2, movie9, 10}]

- setup()?
- map()?

Multiply co-occurrence matrix and rating matrix

九章算法

Co Matrix

	M1	M2	M3	M4	M5
M1	2/6	2/6	1/6	1/6	0
M2	2/11	4/11	2/11	2/11	1/11
M3	1/6	2/6	2/6	0	1/6
M4	1/5	2/5	0	2/5	0
M5	0	1/3	1/3	0	1/3

Rating Matrix

Movie	User B rating
M1	3
M2	7
M3	8
M4	0
M5	0

Result Matrix

Movie	User B rating
M1	2/6*3 + 2/6*7 +...
M2	2/11*3 + 4/11*7 + ...
M3	1/6*3 + 2/6*7 +...
M4	1/5*3 + 2/5*7 +...
M5	0*3 + 1/3*7 +...

UserB: Movie1 = Co[M1][M1]* Rating[M1] + Co[M1][M2]* Rating[M2] + Co[M1][M3]* Rating[M3] + Co[M1][M4]* Rating[M4] + Co[M1][M5]* Rating[M5]

Mapper

[movie1 : {movie1, movie2, 8}{movie1, movie3, 5}{movie1, movie7, 6}]
[movie2 : {movie2, movie1, 8}{movie2, movie5, 9}{movie2, movie9, 10}]

User_id	Movie_id: Rating
1	1:10
1	2: 8
2	1: 8
2	3: 4.5
4	2: 5.5
4	4: 6.5

User_id	Movie_id:Score
1	movie2: 80
1	Movie3: 50
1	Movie7: 60

Normalize Co-occurrence Matrix

	M1	M2	M3	M4	M5
M1	2	2	1	1	0
M2	2	4	2	2	1
M3	1	2	2	0	1
M4	1	2	0	2	0
M5	0	1	1	0	1

归一化处理

	M1	M2	M3	M4	M5
M1	2/6	2/6	1/6	1/6	0
M2	2/11	4/11	2/11	2/11	1/11
M3	1/6	2/6	2/6	0	1/6
M4	1/5	2/5	0	2/5	0
M5	0	1/3	1/3	0	1/3

Reducer

User_id	Movie_id:Score
1	movie2: 80
1	Movie3: 50
1	Movie7: 60
1	Movie2: 30
1	Movie3: 20

User_id	Movie_id: Score
1	Movie2: 80+30
1	Movie3: 50+20
1	Movie7: 60

Generate Recommender List

- Match Movie ID with Movie Name

How to expand your project

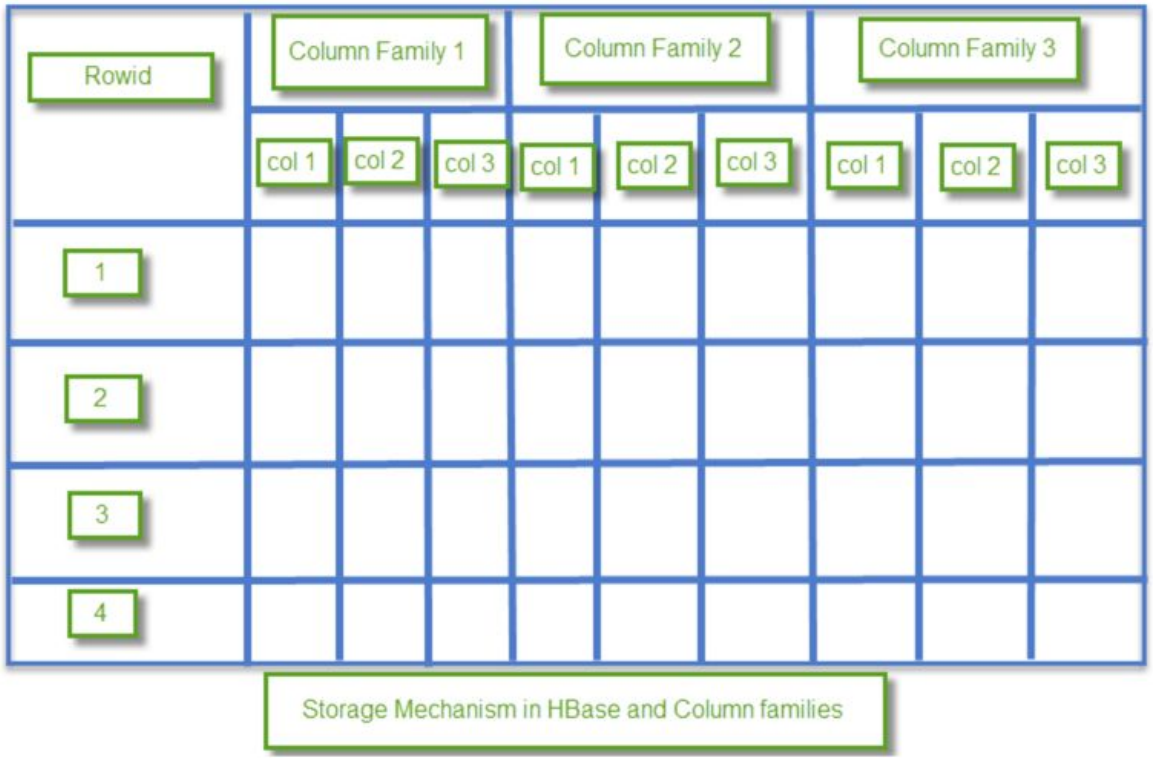
In phase 3, it can be OOM if co-occurrence matrix is very huge, how to improve the performance?

- Two input files

How to expand your project

- Use Mapper-Join in MR1
  - Mapper1: key = movie1 value = movie2: cooccurrence
  - Mapper2: key = movie1 value =user:rating
  - Reducer: key = user value = movie2: coocurrence*rating
- MR2
  - Mapper: key = user: movie* value = score
  - Reducer: key = user value = movie*: score

HBase



HBase


```
$ create 'recommender', 'user_info', 'movie_info'
$ put 'recommender','1','user_info:id', '1'
$ put 'recommender','1','user_info:name', 'Min Zhao'
$ put 'recommender','1','movie_info:recommender', '1072'
$ put 'recommender','1','movie_info:watched', '1009'
$ scan 'recommender'
```

What you have learned

- What is Recommender System
- Methods to implement recommender system
- Understand the theory behind ItemCF
- How to implement ItemCF in MR

Thanks~~