

# 实现方案

1. 从movies.txt文件中提取出每条评论的productID和userID

2. 过滤掉不是电影的productID

若某些product页面不存在导演、演员信息，或者该product的时长过短，则判断其不为电影，直接过滤。最终得到movie.csv文件。

3. 自定义规则，得到同一系列的电影的ID集合

将上述的movie.csv文件根据电影名进行排序，计算各电影名的编辑距离，若编辑距离小于某一阈值，则判断其为同一系列的电影，得到IP.csv文件。

4. 将所有电影的movieID和userID存入neo4j图数据库，而电影以及用户的其余信息存储在mysql数据库中。在neo4j中存储时将movieID和userID设为节点，将评论设为关系

(1) 将所有电影的信息单独存储为一个moviesId\_list.csv文件，并设置csv表头，记录csv第一列为movieId，第二列是节点的LABEL，均为movie

1	movieId:ID(node)	:LABEL
2	B00187MZH0	movie
3	B0006HC06E	movie
4	B000UINP1S	movie
5	B0013B34X0	movie
6	B003QP4CNC	movie
7	B00004CWJ9	movie

(2) 将所有用户的信息单独存储为一个userId\_list.csv文件，并设置csv表头，记录csv第一列为userId，第二列是节点的LABEL，均为user

1	userId:ID(node)	:LABEL
2	A1MDCM35WOMTX5	user
3	A2SSV5IV6HIB6M	user
4	A2LZKUARB5Y8C2	user
5	A39WT0YMCZISH4	user
6	AE5T5C6CPVT66	user
7	A3BTDYUGZ7RT00	user

(3) 将user与movie的评论关系存储为一个moviesId-userId.csv文件，设置表头，记录csv第一列是关系的终点，第二列是关系的起点，第三列是关系的类型，均为review

1	:END_ID(node)	:START_ID(node)	:TYPE
2	B003AI2VGA	A141HP4LYPWMSR	review
3	B003AI2VGA	A328S9RN3U5M68	review
4	B003AI2VGA	A1I7QGUDP043DG	review
5	B003AI2VGA	A1M5405JH9THP9	review
6	B003AI2VGA	ATXL536YX71TR	review
7	B003AI2VGA	A3QYDL5CDNYN66	review

(4) 进入数据库的bin文件夹，利用neo4j-admin import 工具将上述csv文件导入一个空的neo4j数据库

导入时设置上述moviesId\_list.csv, userId\_list.csv为nodes文件, 设置moviesId-userId.csv为relationship文件, 设置导入的id-type为STRING, 导入结果如下

```
1 PS C:\Users\cheng fu\AppData\Local\Neo4j\Relate\Data\dbmss\dbms-4fae42e5-dd19-4c26-80a1-f083e2027951\bin> .\neo4j-admin import --nodes
"\moviesId_list.csv" --nodes "\userId_list.csv" --relationships
"\moviesId-userId.csv" --ignore-missing-nodes --id-type=STRING
2 Neo4j version: 3.5.23
3 Importing the contents of these files into C:\Users\cheng
fu\AppData\Local\Neo4j\Relate\Data\dbmss\dbms-4fae42e5-dd19-4c26-80a1-
f083e2027951\data\databases\graph.db:
4 Nodes:
5   \moviesId_list.csv
6
7   \userId_list.csv
8 Relationships:
9   \moviesId-userId.csv
10
11 Available resources:
12   Total machine memory: 15.83 GB
13   Free machine memory: 7.36 GB
14   Max heap memory : 3.52 GB
15   Processors: 8
16   Configured max memory: 11.08 GB
17   High-IO: false
18
19 Import starting 2020-11-13 22:34:59.478+0800
20   Estimated number of nodes: 863.04 k
21   Estimated number of node properties: 863.04 k
22   Estimated number of relationships: 6.67 M
23   Estimated number of relationship properties: 0.00
24   Estimated disk space usage: 263.23 MB
25   Estimated required memory usage: 1.01 GB
26
27 InteractiveReporterInteractions command list (end with ENTER):
28   c: Print more detailed information about current stage
29   i: Print more detailed information
30
31 (1/4) Node import 2020-11-13 22:34:59.668+0800
32   Estimated number of nodes: 863.04 k
33   Estimated disk space usage: 46.91 MB
34   Estimated required memory usage: 1.01 GB
35   ..... 5% ?855ms
36   ..... 10% ?63ms
37   ..... 15% ?5ms
38   ..... 20% ?0ms
39   ..... 25% ?290ms
40   ..... 30% ?207ms
41   ..... 35% ?2ms
42   ..... 40% ?0ms
43   ..... 45% ?0ms
44   ..... 50% ?0ms
45   ..... 55% ?0ms
46   ..... 60% ?0ms
47   ..... 65% ?16ms
48   ..... 70% ?0ms
49   ..... 75% ?0ms
```

```

50 ..... 80% ?0ms
51 ..... 85% ?0ms
52 ..... 90% ?0ms
53 ..... 95% ?10ms
54 ..... 100% ?0ms
55
56 (2/4) Relationship import 2020-11-13 22:35:01.268+0800
57   Estimated number of relationships: 6.67 M
58   Estimated disk space usage: 216.31 MB
59   Estimated required memory usage: 1.01 GB
60 ..... 5% ?670ms
61 ..... 10% ?435ms
62 ..... 15% ?208ms
63 ..... 20% ?2ms
64 ..... 25% ?4ms
65 ..... 30% ?485ms
66 ..... 35% ?8ms
67 ..... 40% ?268ms
68 ..... 45% ?248ms
69 ..... 50% ?0ms
70 ..... 55% ?2ms
71 ..... 60% ?465ms
72 ..... 65% ?5ms
73 ..... 70% ?5ms
74 ..... 75% ?549ms
75 ..... 80% ?6ms
76 ..... 85% ?208ms
77 ..... 90% ?230ms
78 ..... 95% ?3ms
79 ..... 100% ?1ms
80
81 (3/4) Relationship linking 2020-11-13 22:35:05.071+0800
82   Estimated required memory usage: 1.00 GB
83 ..... 5% ?203ms
84 ..... 10% ?204ms
85 ..... 15% ?260ms
86 ..... 20% ?240ms
87 ..... 25% ?0ms
88 ..... 30% ?0ms
89 ..... 35% ?201ms
90 ..... 40% ?1ms
91 ..... 45% ?221ms
92 ..... 50% ?7ms
93 ..... 55% ?200ms
94 ..... 60% ?200ms
95 ..... 65% ?200ms
96 ..... 70% ?200ms
97 ..... 75% ?200ms
98 ..... 80% ?0ms
99 ..... 85% ?200ms
100 ..... 90% ?0ms
101 ..... 95% ?450ms
102 ..... 100% ?0ms
103
104 (4/4) Post processing 2020-11-13 22:35:08.738+0800
105   Estimated required memory usage: 1020.01 MB
106 - ..... 5% ?207ms
107 ..... 10% ?1ms

```

```
108 ..... 15% ?2ms
109 ..... 20% ?1ms
110 ..... 25% ?1ms
111 ..... 30% ?201ms
112 ..... 35% ?1ms
113 ..... 40% ?1ms
114 ..... 45% ?1ms
115 ..... 50% ?1ms
116 ..... 55% ?1ms
117 ..... 60% ?1ms
118 ..... 65% ?1ms
119 ..... 70% ?1ms
120 ..... 75% ?201ms
121 ..... 80% ?1ms
122 ..... 85% ?0ms
123 ..... 90% ?1ms
124 ..... 95% ?1ms
125 ..... 100% ?1ms
126
127
128 IMPORT DONE in 11s 181ms.
129 Imported:
130     862992 nodes
131     6670561 relationships
132     862992 properties
133 Peak memory usage: 1.04 GB
```

共导入169731部电影：

\$ match(n:movie) return count(n)

Table

count(n)

1

169731

Text

Code

Started streaming 1 records after 1 ms and completed after 2 ms.

693261个用户

\$ match(n:user) return count(n)

Table

count(n)

1

693261

Text

Code

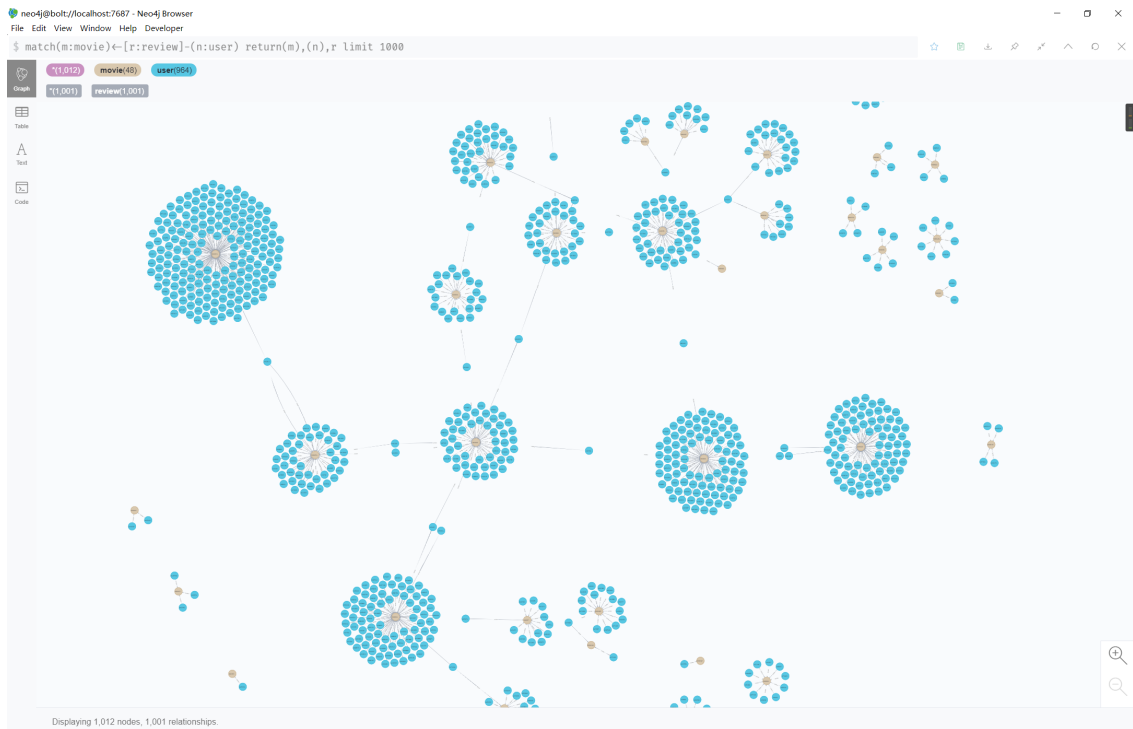
Started streaming 1 records after 15 ms and completed after 15 ms.

```
$ match(n:user)-[r:review]-(m:movie) return count(r)
```

	count(r)
1	6670561

Started streaming 1 records after 1 ms and completed after 1536 ms.

## 6670561个评论关系



### 5. 查询所有系列电影的评论人集合，得到最多的用户集合

运行neo4j.py文件，查询上述步骤提取出的各个系列的电影的评论用户集合，得到最多的用户集合数量为5737，将用户userId写出为result.txt中



## 遇到的问题

### match、merge、create语句速度过慢

在导入数据时，最初使用match、merge、create等语句进行数据插入，性能过慢，将数据全部导入完毕需要几百个小时.....最终改为了使用neo4j-admin import直接进行导入

### neo4j-admin import进行导入时jdk版本不兼容

```

1 警告: ERROR! Neo4j cannot be started using java version 1.8.0_261
2 警告: * Please use Oracle(R) Java(TM) 11, OpenJDK(TM) 11 to run Neo4j
   Server.
3 * Please see https://neo4j.com/docs/ for Neo4j installation instructions.
4 Invoke-Neo4jAdmin : This instance of Java is not supported
5 所在位置 C:\Users\cheng fu\AppData\Local\Neo4j\Relate\Data\dbmss\dbms-
   02c10639-b64e-4ad2-aba3-3c416e7fcec\bin
6 \neo4j-admin.ps1:13 字符: 7
7 + Exit (Invoke-Neo4jAdmin -Verbose:$Arguments.Verbose -CommandArgs $Arg ...
8 + ~~~~~
9 + CategoryInfo          : NotSpecified: (:) [Write-Error],
   WriteErrorException
10 + FullyQualifiedErrorId :
   Microsoft.PowerShell.Commands.WriteErrorException,Invoke-Neo4jAdmin

```

最初使用的是neo4j 4.1, jdk8, 提示版本不兼容, 将neo4j版本降级为3.5后解决