

# VCM

电影推荐系统

- spring, spring mvc, mybatis, android, 爬虫, 最近邻居法

- <http://vcm.vatcore.com>

# crawler包(爬虫)

- DoubanMovieSubjectCollectionsCrawler

<https://movie.douban.com/subject/26718838/collections>

爬电影下的最近评价用户列表, 每页20个用户, 总共200个用户

- DoubanMovieSubjectCrawler

<https://movie.douban.com/subject/26718838/>

爬电影信息及movie\_cover

- DoubanMovieUserCollectCrawler

<https://movie.douban.com/people/soshuangwei/collect?start=0&sort=time&rating=all&filter=all&mode=grid>

爬用户看过的电影评分与短评

# controller.NewController.AbstractNewThread有两个方法

- **boolean addZombieUserRatingAndMovieByDoubanId(int doubanId)** //添加僵尸用户评分及电影(都是爬下来的), doubanId为豆瓣电影Id

```
userList = DoubanMovieSubjectCollectionsCrawler.crawler(doubanId); //爬某电影下最近评价的200个用户

while (userIterator.hasNext()) {

    userMapper.add(user); //添加用户到user表

    addRatingAndMovieByDoubanUsername(user, null); //添加用户评分及数据库缺失电影

}
```

- **boolean addRatingAndMovieByDoubanUsername(User user, User doubanUser)** //user为写入数据库的用户, doubanUser为爬的对象, 如果doubanUser为null, 则爬写对象都为user

```
ratingList = DoubanMovieUserCollectCrawler.crawler(user | doubanUser); //爬用户看过的电影评分与短评

for (Rating rating : ratingList) {

    if (movieMapper.findMovieByDoubanId(movie) == null) { //通过doubanId查询此电影是否在数据库中存在

        if (movieService.addMovieUseCrawlerByDoubanId(movie) == null) continue; //爬下电影信息及movie_cover, 并写入到movie表, 如果非电影则continue

    }

    ratingService.changeRating(rating); //添加或修改评分

}
```

- 如果爬虫线程只有一个, 则每个页面爬下需sleep(>2000), 如果有多个爬虫线程, 则sleep(>2000 \* 线程数)
- 默认构造方法会生成threadId, 管理员可以通过threadId查看线程进度, 如添加僵尸用户每完成一个用户进度+0.5%

# 有三个类继承AbstractNewThread

- BatchNewThread

调用addZombieUserRatingAndMovieByDoubanId(int doubanId)

- RandomBatchNewThread

从数据库中随机 (unhandle\_rating<10)得到doubanId, 调用addZombieUserRatingAndMovieByDoubanId(int doubanId) ,不停止线程则不停的随机并调用

- ImportDoubanUsernameThread

爬下指定豆瓣用户评分到指定用户(当前登录用户)下, 调用addRatingAndMovieByDoubanUsername(User user, User doubanUser)

批量新建(正在执行的任务有:0个)

尽量单个线程

当前任务状态:

强行终止所有执行任务(并不会瞬间停止)

强行停止

输入地址，抓取此页最近看过的用户及用户所有看过电影和评分

doubanId

批量增加

随机批量增加

输入doubanId抓取新建

doubanId

增加

将doubanUsername(id)的评分导入自己账户下

doubanUsername(id)

导入

普通用户只能使用最下面两个功能  
userId为1的用户可以使用所有

# handler包(常驻线程)

- 当启动tomcat时, 由spring将 `***Handler`类注入到`Handler`类, 读`Config`类的配置决定是否开启线程

## 更新电影的平均评分与评分用户总数线程

- `RatingsAndUsersHandler`

`movieService.calculationAndUpdateRatingsAndUsers();` //找到100部`unhandle_ratings`(未处理的评价总数, 处理过后归零) >0的电影遍历统计评价过的用户总数和评价均分, 将结果update到movie表

## 最近邻居线程(5个)

- `GenresRatingHandler`
- `FavoriteGenresHandler`与写配置文件线程
- `NearestNeighborHandler`与写配置文件线程

- GenresRatingHandler

```
        userList = userService.findHundredUsersIfRatingCountMoreThanXAndUnhandleRatingMoreThanY(); //找到100个rating_count(评价总数) >60并且unhandle_rating>30的用户

        for (User user : userList) {

            ratingService.selectAndAddGenresRatingListToGenresRatingByUserId(rating); //统计用户评价的每部电影类型下的平均评分和与总平均评分的相差, 比例, 类型占的比例,
            类型总量, 大于用户电影平均评分的类型并且此类型电影评价总数大于10会写入genres_rating表中(图中绿色的行)

            userService.makeUnhandleRatingsZeroByUserId(user); //unhandle_rating归零

        }
```

ratingCount: 评价过的电影总计

avgRating(s): 平均电影评分

count: 用户评价过的此类型电影总数

ratio: 此类型占有所有评价电影比例, 这个用户总共对416部电影评价了, 剧情类  
型的评价了189部, 则ratio = 189 / 416

avgRating(y): 类型的平均评分

avgDifference:类型的平均评分-电影的平均评分

avgRatio:类型的平均评分/电影的平均评分

weight: 类型权重, (ratio + avgRatio) \* avgRating, 由FavoriteGenresHandler操  
作

							ratingCount	avgRating(s)
							416	7.5962
genresId	nameZh	count	ratio(r)	avgRating(y)	avgDifference(y-s)	avgRatio(ar)(y/s)	weight{(r + ar) * y}	
1	剧情	189	0.45432693	7.8942	0.29799986	1.0392301	11.790437726224155	
10	动作	155	0.37259614	7.4065	-0.18970013	0.97502697	9.981170377795479	
13	喜剧	149	0.35817307	7.5839	-0.0123000145	0.9983808	10.287968728261689	
6	冒险	112	0.26923078	7.75	0.15380001	1.020247	9.993452683091164	
4	科幻	97	0.23317307	7.4433	-0.15290022	0.9798715	9.029054472010245	
14	爱情	89	0.2139423	7.3708	-0.22539997	0.97032726	8.729014111954427	
5	惊悚	76	0.1826923	7.3158	-0.2803998	0.96308684	8.382291306732732	
16	奇幻	69	0.16586539	7.8261	0.22989988	1.0302651	9.361036631113379	
3	犯罪	68	0.16346154	7.7353	0.13910007	1.0183117	9.141370890903033	
24	动画	56	0.13461539	8.1429	0.5467005	1.0719702	9.825106574872677	
12	家庭	43	0.103365384	7.7209	0.12470007	1.0164161	8.645720716267473	
2	悬疑	37	0.088942304	7.4054	-0.19080019	0.9748821	7.87804522126266	
8	传记	13	0.03125	7.2308	-0.36539984	0.951897	7.108939658419615	
11	灾难	13	0.03125	7.2308	-0.36539984	0.951897	7.108939658419615	
15	恐怖	11	0.026442308	6.7273	-0.8688998	0.88561386	6.135675602487153	
9	战争	11	0.026442308	7.4545	-0.14169979	0.98134595	7.51255778062049	
23	历史	9	0.021634616	7.7778	0.1816001	1.0239067	8.132011394588009	
19	音乐	7	0.016826924	8.5714	0.9751997	1.12838	9.816025765523861	
21	儿童	7	0.016826924	7.1429	-0.4533	0.94032544	6.836843603557265	
20	歌舞	5	0.012019231	8.4	0.8037996	1.105816	9.389815569794445	
26	古装	5	0.012019231	6.8	-0.7961998	0.89518446	6.168985253381837	
22	短片	5	0.012019231	8.0	0.4038	1.0531582	8.521419160068035	
7	同性	4	0.009615385	8.0	0.4038	1.0531582	8.502188391983509	
25	武侠	3	0.0072115385	6.6667	-0.9295001	0.8776362	5.899014279179953	
18	纪录片	3	0.0072115385	8.6667	1.0705004	1.1409258	9.950561985502478	
27	运动	2	0.0048076925	8.0	0.4038	1.0531582	8.463726852089167	
17	情色	2	0.0048076925	8.0	0.4038	1.0531582	8.463726852089167	
28	西部	1	0.0024038462	8.0	0.4038	1.0531582	8.444496082141995	



people(p) | mine(m)

ratingCount	avgRating(s)	ratingCount	avgRating(s)	mpRatingRatio(mpr)	[[m]s]/[[p]s]
416	7.5962	193	7.5855	0.99859136	

rating	count	rating	count
2	1	2	14
4	10	4	15
6	118	6	36
8	230	8	60
10	57	10	68

[p]original | [m]original | [p]weighted

[p]y \* r... 对[p]进行加权

genres	nameZh	count	ratio(r)	avgRating(y)	avgDifference(y-s)	avgRatio(an)(y/s)	weight((r + an) * y)
1	剧情	189	0.45432693	7.8942	0.28799988	1.0392301	11.7504377262241581
10	动作	155	0.37259614	7.4065	-0.18970013	0.97502697	9.981170377795479
13	喜剧	149	0.35817307	7.5839	-0.0123000145	0.9983808	10.287968728261689
5	冒险	112	0.26923076	7.75	0.15389001	1.020247	9.9934525683091164
4	科幻	97	0.23317307	7.4433	-0.15290022	0.9798715	9.029054472010245
14	爱情	89	0.2139423	7.3708	-0.22539997	0.97032726	8.729014111954427
5	惊悚	76	0.1826923	7.3158	-0.2803998	0.96308684	8.382291306732732
16	奇幻	69	0.16586539	7.8261	0.22989988	1.0302651	9.381036631113379
3	犯罪	68	0.16346154	7.7353	0.13910007	1.01831117	9.1413708990903033
24	动画	56	0.13461539	8.1429	0.6467905	1.0719702	9.823106574872677
12	悬疑	43	0.103365384	7.7209	0.12470007	1.0164161	8.646720716267473
2	悬疑	37	0.088942304	7.4054	-0.19080019	0.9748821	7.87804522126266
8	传记	13	0.03125	7.2308	-0.36539984	0.951897	7.108939658419615
11	灾难	13	0.03125	7.2308	-0.36539984	0.951897	7.108939658419615
15	恐怖	11	0.026442308	6.7273	-0.8688998	0.88561386	6.135675602487153
11	战争	11	0.026442308	7.4545	-0.14169979	0.98134595	7.51255778062049
23	历史	9	0.021634616	7.7778	0.1816001	1.0239067	8.132011394588009
19	音乐	7	0.016826924	8.5714	0.9751007	1.12838	9.816025765523861
21	儿童	7	0.016826924	7.1429	-0.4533	0.94032544	6.836843603557265
20	歌舞	5	0.012019231	8.4	0.8037996	1.105816	9.389815569794445
26	古装	5	0.012019231	6.8	-0.7961998	0.89518446	6.168985253381837
22	短片	5	0.012019231	8.0	0.4038	1.0531582	8.521419160068035
7	同性	4	0.009615385	8.0	0.4038	1.0531582	8.502188391983509
25	武侠	3	0.0072115385	6.6667	-0.9295001	0.8776362	5.899014279179953
18	纪录片	3	0.0072115385	8.6667	1.0705004	1.1409258	9.950561985502478
27	运动	2	0.0048076925	8.0	0.4038	1.0531582	8.463726852089167
17	情色	2	0.0048076925	8.0	0.4038	1.0531582	8.463726852089167
28	西部	1	0.0024038462	8.0	0.4038	1.0531582	8.444496082141995

/people/{userId}

可实时查看用户与目标用户(第一张)的状态

第一, 二张都是原始数据, 第三张为第一张相对第二张加权(用户电影平均评分/目标用户电影平均评分)得来

绿色行:会写入genres\_rating表

根据weight推测目标用户喜欢:剧情, 冒险, 动画  
用户喜欢:剧情, 爱情, 动画

这两个用户并不会成为邻居, 成为邻居的必要条件为喜欢的三个类型必须完全相同

成为邻居后, 要算出离用户最近的三个邻居, 用户的推荐电影从这三个用户中得出, 统计电影出现的喜爱类型数量, 从大到小排序, 再将电影列表返回给用户

邻居坐标系x轴为类型平均分, y轴为类型所占比例

- FavoriteGenresHandler

依赖 GenresRatingHandler处理后的结果, 处理genres\_rating表, 写入结果至favorite\_genres表, 内部有线程每10秒检查nextGenresRatingIdINI与nextGenresRatingId(下一个处理的id), 两者不同则写入磁盘更新nextGenresRatingId.ini, .ini文件只有在线程启动时读入, 其余时间都在写入备份, 配置文件路径由Config类指定

```
int count = genresRatingService.findAndAddFavoriteGenresByGenresRatingId(genresRating); //算出类型的weight((ratio + avgRatio) * avgRating), 找出weight最大的三个类型(推测出的用户喜欢类型)写入到favorite_genres表, 返回处理genres_rating的行数
```

```
if (count == 0) { ... } //到底了或卡住了(id不连续)...
```

- NearestNeighborHandler

依赖 FavoriteGenresHandler处理后的结果, 处理favorite\_genres表, 写入结果至nearest\_neighbor表, 写配置文件与处理流程与FavoriteGenresHandler 类似

```
int count = favoriteGenresService.findAndAddNearestNeighbor(favoriteGenres); //根据推测喜欢的三个类型找出相同用户(这些用户都为邻居), 算出用户与邻居(经过加权后)的距离, 邻居坐标系x轴为类型平均分, y轴为类型所占比例, 用户与邻居对应的三个类型的距离依次算出相加就是距离, 找出离用户最短距离的三个邻居, 将结果写入nearest_neighbor表
```

- RecommenderController | RecommendMobileController

```
goodMovieList = nearestNeighborService.findNearestNeighborGoodMovieListByUserId(nearestNeighbor); //找出三个最近邻居喜爱类型的电影列表, 对电影出现的类型次数进行计数, 每有一个喜欢类型, count++, 对map按值从大到小排序, 返回list
```

VCAM	name(movieId)	doubleant	O	演者	导演	片人	评论数	星级
FavoriteGenres								
genreIdnameZh								
1	剧情							
14	爱情							
24	动画							
Neighborhood								
neighborIddistance								
54	1.84914							
58	3.18924							
272	3.23463							
movieIdoubandrating/users/nameZh								
nameO								
releaseDate								
runtime								
4	1292052	9.6(35)		肖申克的救赎	The Shawshank Redemption	1994-09-10(多伦多电影节) 1994-10-14(美国)		142
519	1292328	9.0586(17)		东邪西毒	影射西毒	1994-09-17(中国香港)		100
1020	1309909	8.868(19)		阿飞正传	阿飞正传	1990-12-15(香港)		94
1032	204809	8.5(4)		杀手没有假期	In Bruges	2008-01-17(圣丹斯电影节) 2008-04-18(美国)		97
9	1295409	8.6(15)		东邪西毒	影射西毒	1994-02-02(香港)		108
1187	6791750	8.2857(7)		东邪西毒	影射西毒	2013-07-20(日本)		126
537	1302425	8.6(30)		东邪西毒	影射西毒	1999-02-13(香港)		85
539	25819229	7.333(9)		云中行走	The Walk	2016-01-22(中国香港) 2015-10-09(美国)		123
543	11680263	8.0(19)		功夫熊貓3	Kung Fu Panda 3	2016-01-29(美国) 2016-01-29(美国)		99
544	3233635	8.222(18)		功夫熊貓2	Kung Fu Panda 2	2011-05-26(中国香港) 2011-05-26(美国)		91
2393	3578925	8.0(7)		我家买了动物园	We Bought a Zoo	2011-12-23(美国)		124
35	1291668	9.0(8)		情人	L'amant	1992-01-22(法国) 1992-06-19(美国)		115
548	3203658	7.623(16)		热辣滚烫	Hot辣滚烫	2013-11-01(中国香港) 2012-09-08(多伦多电影节) 2012-09-28(美国)		91
37	1344447	8.1481(27)		死亡诗社	Dead Poets Society	1989-06-02(多伦多电影节) 1989-06-05(美国)		113
581	1291548	8.7(20)		死亡诗社	Dead Poets Society	1989-06-02(多伦多电影节) 1989-06-05(美国)		128
1079	20470074	2857(14)		贵姓之庭	貴姓之庭	2013-05-31(日本)		46
4141	1422927	8.0(2)		贵姓之庭	A Good Woman	2004-09-15		93
1880	3768423	7.75(8)		守护者联盟	Rise of the Guardians	2012-11-16(中国香港) 2012-11-21(美国)		97
50	1498818	8.444(8)		守护者联盟	Rise of the Guardians	2012-11-16(中国香港) 2012-11-21(美国)		143
1077	1474185	7.333(12)		人生遥控器	Click	2006-04-20(中国香港) 2006-06-23(美国)		107
1079	1309172	8.0(14)		初恋50次	50 First Dates	2004-06-25(中国香港) 2004-02-13(美国)		99
2618	1202737	8.5(4)		木偶奇遇记	Pocahontas	1990-02-07		96
60	25904044	258(31)		马达加斯加	La La Land	2017-02-16(中国香港) 2016-08-31(威尼斯电影节) 2016-12-25(美国)		128
574	5322596	9.25(16)		脱逃	Detachment	2011-04-25(圣丹斯电影节) 2011-06-25(美国)		97
1998	3178770	7.636(11)		马达加斯加3	Madagascar 3: Europe's Most Wanted	2012-06-06(美国) 2012-06-06(法国)		93
64	3173635	8.0(2)		天堂花开	Seraphite	2008-08-07		125
578	25919510	8.12(2)		死无对证	Dead Poets Society	2015-12-10(中国香港) 2015-11-11(台北金马影展)		109
580	2133223	8.352(17)		白日梦夜夜	The Secret Life of Walter Mitty	2013-12-25(美国)		114
71	1308817	7.8095(21)		两小无猜	Jean d'enfants	2003-09-17(法国)		93
72	3564223	7.652(23)		了不起的盖茨比	The Great Gatsby	2013-08-30(中国香港) 2013-05-01(纽约电影节) 2013-05-10(美国)		143
73	259781264	0(4)		月光骑士	Alfred	2016-11-30(中国香港) 2016-11-23(美国)		134
75	26648249	7.386(13)		月光骑士	Moonlight	2016-09-02(特柳斯电影节) 2016-09-10(多伦多电影节) 2016-10-21(美国)		111
76	2580443	9.067(15)		曼彻斯特海边的音乐声	Manchester by the Sea	2017-07(中国香港) 2016-09-08(多伦多电影节) 2016-11-18(美国)		137
607	1301168	8.133(15)		乱战	亂戰	2000-10-13(中国香港) 2000-05-19(蒙特利尔电影节) 2000-07-07(法国)		120
3168	1919245	8.5(12)		柯琳	Coraline	2009-02-06(美国)		96
57	1418634	8.6(20)		破晓时分	Breakdawn Mountain	2009-09-02(威尼斯电影节) 2009-01-06(加拿大) 2009-01-13(美国)		134
98	24757867	2222(26)		破晓时分	Breakdawn Mountain	2015-12-24(中国香港) 2015-09-09(威尼斯电影节)		137
1122	25921818	3158(19)		伊甸园	伊甸园	2016-10-28(中国香港)		111
1123	2632623	7.0(10)		伊甸园	伊甸园	2016-09-14(中国香港)		109
161	109944	9.476(42)		伊甸园	伊甸园	1994-09-14(美国)		133
1611	3698423	8.0(7)		伊甸园	伊甸园	2009-08-01(日本)		114
1131	1781126	7.5(8)		伊甸园	Zodiac	2007-03-02(美国)		157
108	1291549	8.9375(32)		伊甸园	Les chroniques	2004-10-16(中国香港) 2004-03-17(法国)		97
620	1296778	7.25(8)		伊甸园	Lolita	1997-09-27		137
109	1292001	8.7875(33)		海上钢琴师	La Leggenda del pianista	1998-10-26(意大利)		165
110	1291546	9.5284(34)		伊甸园	伊甸园	1993-01-01(香港)		171
111	1292720	9.2776(36)		伊甸园	Forrest Gump	1994-06-23(圣丹斯电影节) 1994-07-06(美国)		142
113	6786032	8.2(25)		伊甸园	伊甸园	2011-11-02(法国)		112
3166	2054933	7.1426(7)		伊甸园	Burn After Reading	2008-08-27(威尼斯电影节) 2008-09-12(美国)		96
117	1291561	9.4146(41)		伊甸园	半半半半半	2010-07-20(日本)		125
1141	2635560	7.623(16)		伊甸园	Fooland	2007-11-22		150
1142	3804891	7.8(20)		伊甸园	伊甸园	2013-12-03(中国香港)		117
630	1262233	8.5(20)		伊甸园	Accidental Orange	1971-11-15(圣丹斯电影节) 1972-01-13(美国)		136
118	26682004	2963(27)		伊甸园	伊甸园	2016-12-02(中国香港) 2016-08-20(日本)		106
631	1693195	8.5(8)		伊甸园	伊甸园	1987-08-01(日本)		107
632	1294638	8.5556(18)		伊甸园	E.T. - The Extra-Terrestrial	1982-05-26(蒙特利尔电影节) 1982-06-11(美国)		115
634	1291853	8.2(10)		伊甸园	The English Patient	1996-12-06(美国) 1997-03-14(法国)		162
636	1793929	8.381(21)		伊甸园	Dallas Buyers Club	2013-09-07(多伦多电影节) 2013-11-22(美国)		117
642	1292270	9.0(12)		伊甸园	Requiem for a Dream	2000-10-27		102
643	1291879	9.0(12)		伊甸园	伊甸园	1990-06-26(日本)		88
132	1296736	8.25(16)		伊甸园	The Pianist	2002-05-24(蒙特利尔电影节) 2002-09-25(法国)		150
133	1924663	9.0476(16)		伊甸园	Life of Pi	2012-11-22(中国香港) 2012-09-28(纽约电影节) 2012-11-21(美国)		127
134	2028677	7.0(10)		伊甸园	Hugo	2012-05-31(中国香港) 2011-11-23(美国)		127
647	1293460	8.5(12)		伊甸园	Singin in the Rain	1952-04-11(美国)		103
138	263543354	8.071(7)		伊甸园	伊甸园	2016-09-28(中国香港) 2016-09-28(美国)		131
139	25962329	9.046(42)		伊甸园	伊甸园	2016-03-04(中国香港) 2016-03-04(美国)		109
5772	1295467	9.0(2)		伊甸园	Romeo and Juliet	1968-10-19		138
141	259640715	9286(28)		伊甸园	伊甸园	2015-09-30(中国香港)		104
3217	1212766	8.5(4)		伊甸园	El italiano	2010-10-16(中国香港) 2007-10-11(西班牙)		105
637	1291569	8.680(28)		伊甸园	伊甸园	1984-07-14(香港)		102
639	1291545	8.75(24)		伊甸园	Big Fish	2003-12-10(多伦多电影节) 2004-01-09(美国)		125
661	1297359	8.623(17)		伊甸园	伊甸园	1997-07-12		134
665	1294539	8.666(18)		伊甸园	Breathless	1995-04-21		127
1178	3430271	8.666(18)		伊甸园	Desert Flower	2009-09-24		177
5275	1309220	8.4(5)		伊甸园	伊甸园	2005-10-20(香港)		100
669	1291841	9.428(21)		伊甸园	The Godfather	1972-03-15(蒙特利尔电影节) 1972-03-24(美国)		175
670	1292402	8.857(128)		伊甸园	Maiena	2000-10-27(意大利)		109
1183	259291517	3333(9)		伊甸园	Me Before You	2016-06-03(美国)		110
519	1292528	9.0476(21)		伊甸园	Transmigrating	1996-02-23(美国)		94
5791	25928148	0(1)		伊甸园	Roméo & Juliette De la suite à l'amour	2002-02-06(法国)		140
5792	269219528	0(1)		伊甸园	Mindens	2016-02-26(葡萄牙)		26
162	4058598	8.7273(22)		伊甸园	告白	2010-06-05(日本)		106
5795	263742057	0(2)		伊甸园	伊甸园	2016-09-23(中国香港) 2016-09-23(美国)		89
5796	1921583	9.0(4)		伊甸园	L'ukrosenie	2010-06-16(法国) 2010-12-25(美国)		80
5799	6731993	8.0(1)		伊甸园	Great Expectations	2012-11-30		128
2340	1416728	8.75(8)		伊甸园	花樣年華	1993-01-20		66
683	263233208	7692(26)		伊甸园	Hickock Ridge	2016-12-08(中国香港) 2016-11-04(美国)		139
172	2299578	8.333(18)		伊甸园	少男情事	2014-01-19(圣丹斯电影节) 2014-07-18(美国)		165
5804	20326557	10.0(1)		伊甸园	伊甸园	2016-09-28(中国香港) 2016-09-28(美国)		180
173	1292925	9.0(8)		伊甸园	My Own Private Idaho	1986-08-08(美国)		89
1709	1292792	8.0(7)		伊甸园	My Own Private Idaho	1991-09-12		104
5800	10756518	0(1)		伊甸园	Saving Mr. Banks	2013-12-13(美国)		125
5806	1438733	8.0(1)		伊甸园	Save Myself	2014-02-15		125
5808	3031832	8.0(1)		伊甸园	Save Myself	2009-12-26(美国)		180
179	2131688	8.5(26)		伊甸园	Save Myself	2014-09-26(纽约电影节) 2014-10-03(美国)		149
5811	1307135	10.0(1)		伊甸园	Save Myself	2012-03-13		87
181	257739328	8.1(25)		伊甸园	Whipash	2014-10-10(美国)		107
5813	25788944	0(1)		伊甸园	The Coltiber	2014-09-11(多伦多电影节) 2015-03-13(美国)		99
604	1291555	7.25(16)		伊甸园	伊甸园	2004-09-28(中国香港) 2004-05-20(蒙特利尔电影节) 2004-09-29(香港)		129
4282	1296185	8.666(13)		伊甸园	伊甸园	1999-12-08(蒙特利尔电影节) 2000-01-14(美国)		127

# 网页与android端的推荐页(userId=269)

## VCM

标记量必须大于60并且符合系统规则

- 肖申克的救赎 9.6(35)
- 东邪西毒 9.0588(17)
- 阿飞正传 8.8889(18)
- 杀手没有假期 8.5(4)
- 纵横四海 8.8(15)
- 起风了 8.2857(7)
- 喜剧之王 8.6(30)
- 云中行走 7.3333(9)
- 功夫熊猫3 8.0(19)
- 功夫熊猫2 8.2222(18)
- 我家买了动物园 8.0(7)
- 情人 9.0(8)
- 精灵旅社 7.625(16)
- 记忆碎片 8.1481(27)
- 死亡诗社 8.7(20)
- 言叶之庭 8.2857(14)
- 痴男怨女 8.0(2)
- 守护者联盟 7.75(8)
- 通天塔 8.4444(9)
- 人生遥控器 7.3333(12)
- 初恋50次 8.0(14)
- 木偶奇遇记 8.5(4)
- 爱乐之城 8.2581(31)
- 超脱 9.25(16)
- 马达加斯加3 7.6364(11)
- 花落花开 8.0(2)

搜索

推荐

我

# Android

Task接口有三个默认方法(java8特性)用来与服务端通讯, \*\*\*Task类都实现这个接口, 使用默认方法的版本必须 $\geq$ android7.0(api24)

控制器在controller.mobile下, 一般与类名对应, 除显示movie\_cover外, 服务端都用json给客户端传信息

# Activity

- **StartPageActivity**

起始页, 每次启动程序检查是否之前登录过并向服务端验证用户, 控制器为VerifyMobileController, 通过跳转MainActivity, 不通过清除之前的用户信息, 跳转到LoginAndRegisterActivity, 从没登录过的同样跳转. 跳转都会finish();

- **LoginAndRegisterActivity**

有两个Fragment: LoginFragment, RegisterFragment. 可左右滑动切换

- **MainActivity**

有三个Fragment: SearchFragment, RecommendFragment, MineFragment. 可左右滑动切换

- **MovieActivity**

有一个或多个MovieFragment, 数量等于接受List<Movie>的size(), 可左右滑动切换

# Fragment

- MineFragment

显示nickname, 登出按钮点下跳转LoginAndRegisterActivity并finish();

- LoginFragment

登录页, 登录成功跳转MainActivity

- RegisterFragment

注册页

- SearchFragment

搜索页, 从服务端接受List<Movie>, 点进一项跳转到MovieActivity并显示那项的具体信息

- RecommendFragment

推荐页, 与SearchFragment类似

- MovieFragment

电影信息与评价页, 显示电影信息(不用从服务端获取, 在List<Movie>里), movie\_cover(用doubanId向服务端获取), 用户的评分与短评. 并能修改评分与短评, 控制器为RatingMobileController

VCMVCM



记忆碎片 Memento

评分:8.1481(27)

上映时间:2000-09-05(威尼斯电影节) 2001-05-25(美国)

时长:113分钟

imdb:tt209144

douban:1304447

简介:

当记忆支离破碎后,你该怎样面对生活? 本片的主人公就遭遇了这样的事情。

你的评分:

当前评分:

你的短评



东邪西毒 東邪西毒

评分:9.0588(17)

上映时间:1994-09-17(中国大陆)

时长:100分钟

imdb:tt109688

douban:1292328

简介:

在恋人（张曼玉）负气嫁给兄长的当晚，欧阳锋（张国荣）黯然离开白陀山，走进沙漠某个小镇，成为一名杀手中介人。他的朋友、风流剑客黄药师（梁家辉）每年都来小镇与他畅饮，酒话里有关白陀山的消息令他忘却往事的念想渐次演变成自欺行为。

你的评分:10

当前评分:10

神一样的剧本

提交

电影页面(userId=269)