# 最新工作：

数据：

将已有的新数据集导入数据库

数据清洗（现有数据格式、缺项等问题）

训练模型得到推荐数据，存入数据库

爬虫

java：

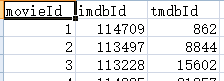
如何设计实现

前端：

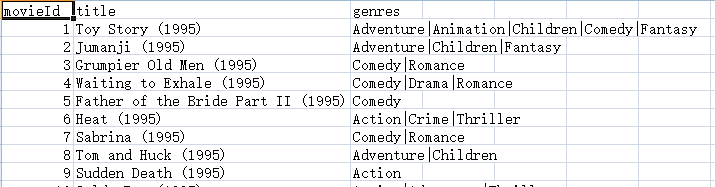
如何和java对接

# ml-latest-small文件格式：

## links：



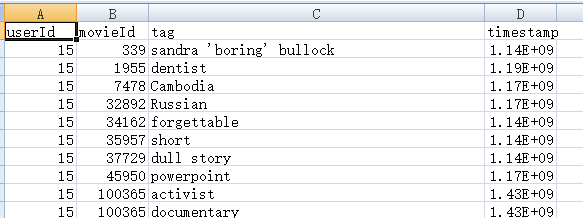
## movies：



## ratings

userid, movieId, rating, timestamp

tags



# 数据库设计

电影信息表（电影id，名称，）

用户信息表（用户id，密码，昵称，年龄等其他信息）

用户评分表（用户id，电影id，评分）

评论表（用户id，电影id，评论内容字符串）

搜索历史表（用户id，用户搜索过的字符串）

足迹表（用户id，用户点进去过的电影id）

猜你喜欢表(Uid,Mid1,Mid2,Mid3,Mid5,Mid5,Mid6)

类似电影表(喜欢这部电影的人也喜欢)(Mid, Mid1,Mid2,Mid3,Mid5,Mid5,Mid6)

所有id均为整型，从1开始递增。

电影类别是布尔类型

gender是枚举类型 F or M

age是小整型

职业是枚举类型

rating是枚举类型1 2 3 4 5

Movie

(Mid,Mname,Myear,Mdate,Mbrief,

Mdirector,Mwriter1,Mwriter2,Mwriter3,Mstar1,Mstar2, Mstar3, Mstar4, Mstar5,

Munknown, Maction , Madventure , Manimation ,Mchildren's , Mcomedy , Mcrime , Mdocumentary , Mdrama , Mfantasy ,Mfilm-Noir , Mhorror , Mmusical , Mmystery , Mromance , Msci-fi ,Mthriller , Mwar , Mwestern,

Mrating, Mpicture

)

User

(Uid,Upassword,Unickname,Ugender,Uage,Uoccupation)

)

Rating

(Rid,Uid,Mid,rating)

Comment

(Cid,Uid,Mid,comment)

SearchHistory

(Sid,Uid,content)

ViewHistory

(Vid,Uid,Mid)

YouMayLike

(Uid,Mid1,Mid2,Mid3,Mid5,Mid5,Mid6)

SimilarMovies

(Mid, Mid1,Mid2,Mid3,Mid5,Mid5,Mid6)

# 数据方面的工作：

获得最后的这些表（java，Access，sql ）

电影信息表：现有的文本数据u.item，爬虫得到的数据，（自己编一些数据），用scala、java和MySQL进行处理，最终得到电影信息表Access。

用户信息表：现有文本文件或爬虫数据，直接转成Access。 文本文件转Access

用户评分表：现有文本文件或爬虫数据，直接转成Access。

评论表：自己编一些或爬

猜你喜欢、类似电影

scala

scala：文件导入导出（训练模型优化）