


# 微平台推荐系统介绍

PRESENTED BY 钱欣耀



# CONTENTS

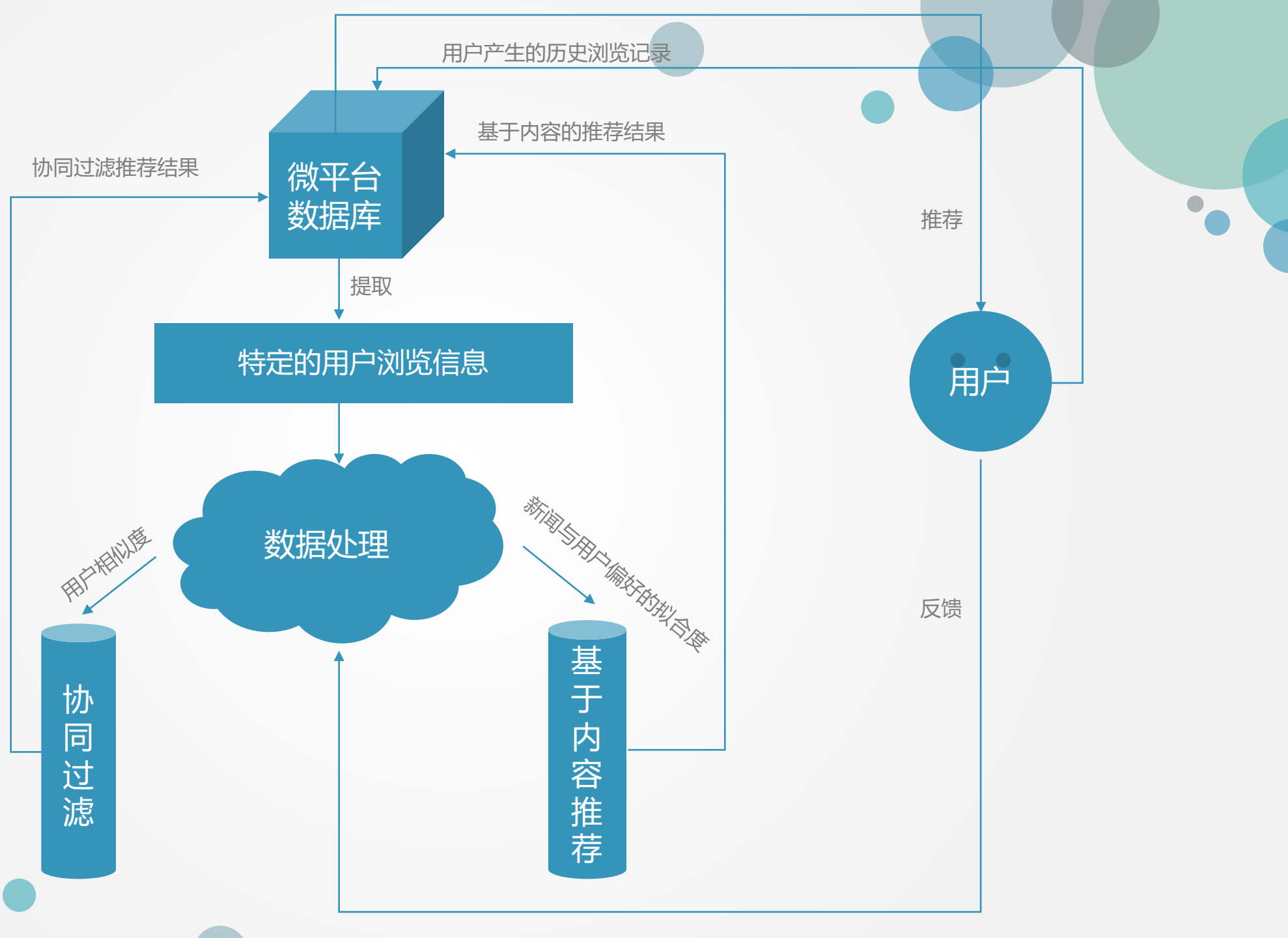
- 01 推荐系统整体框架
  - 02 推荐系统所用算法及所需数据
  - 03 推荐系统所需微平台配合
  - 04 推荐系统使用方法
- 



# 整体框架

The Framework

# 整体框架

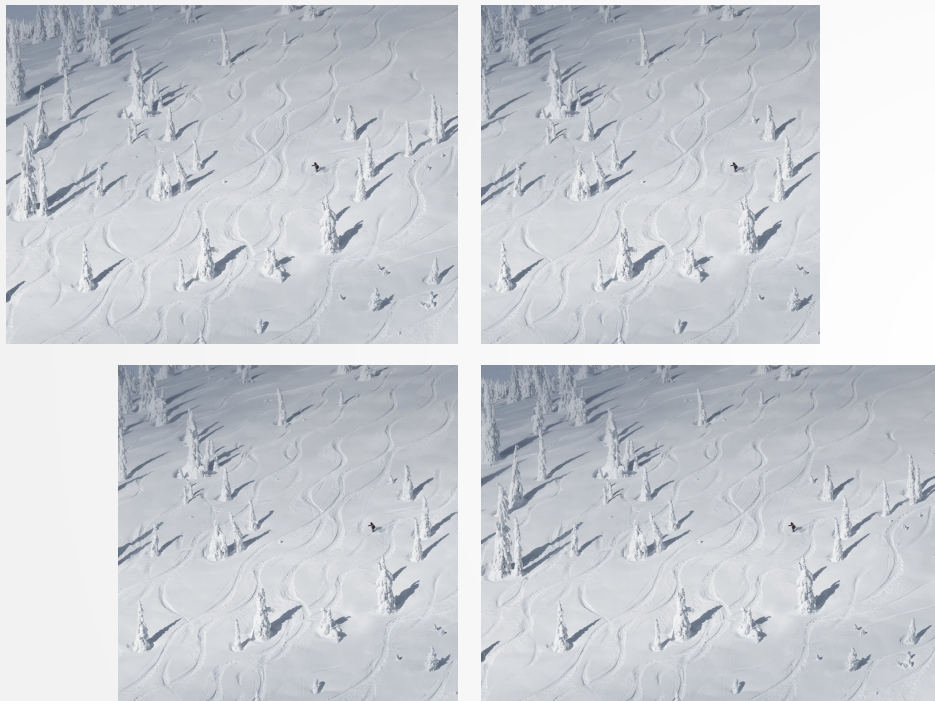




2

# 所用算法及所需数据

The Algorithm and Relevant Data

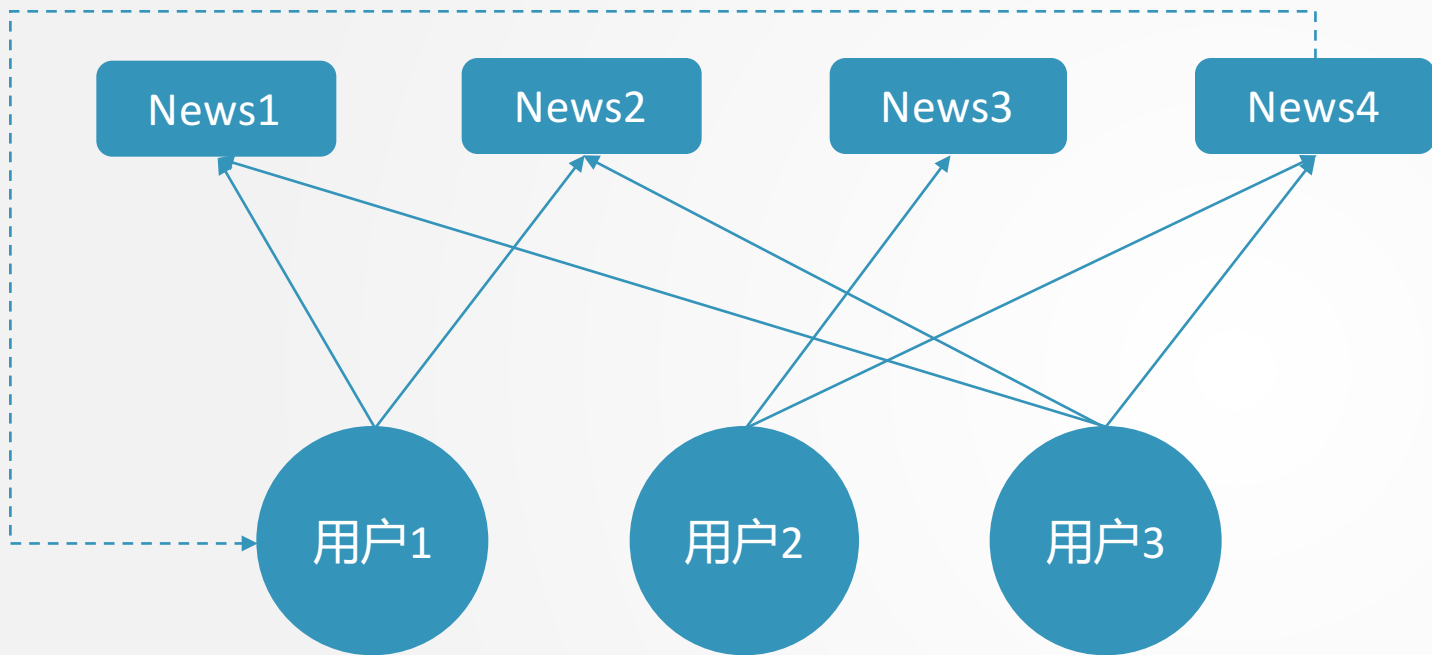


## 主流推荐算法

- 1、基于协同过滤的推荐v
- 2、基于内容的推荐v
- 3、基于热点新闻的推荐v
- 4、基于模型的推荐（后期可考虑加入）
- 5、基于静态数据的推荐

# 所用算法及所需数据

## 1、基于协同过滤的推荐原理



| 用户感兴趣（浏览过）的新闻 |                     |
|---------------|---------------------|
| 用户1           | {News1,News2}       |
| 用户2           | {News3,News4}       |
| 用户3           | {News1,News2,News4} |

显然，用户1与用户3的爱好更具有相似性。那么我们就可以把用户3看过，但用户1尚未看过的新闻（News4）推荐给用户1。

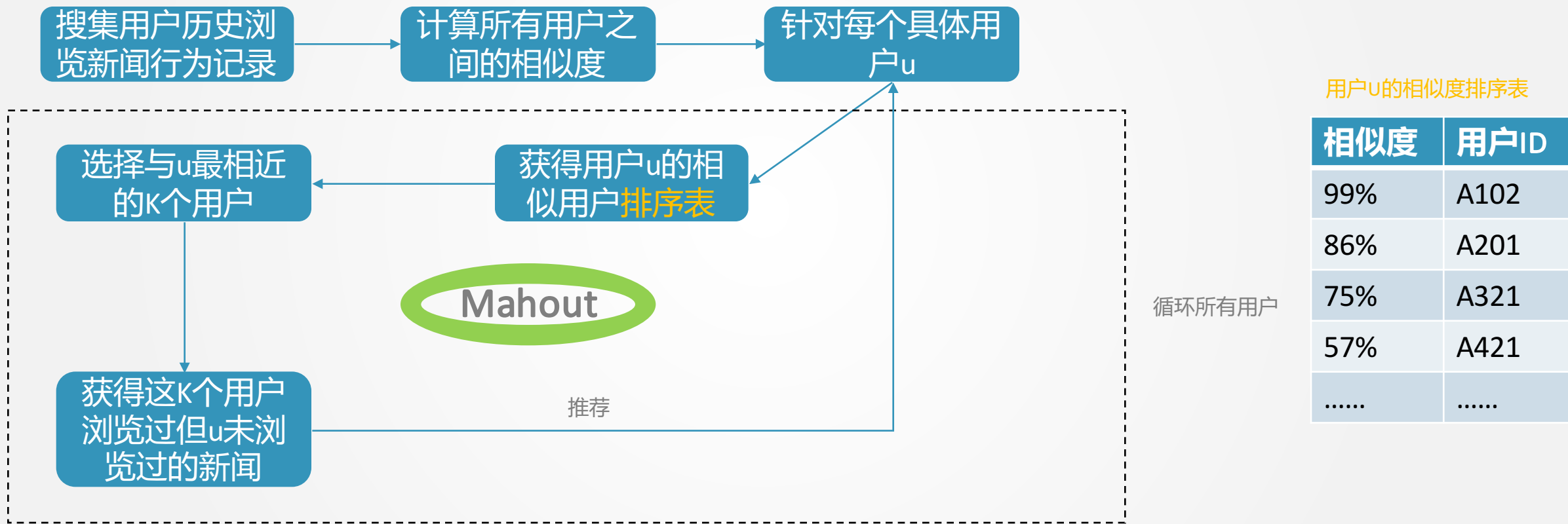
### 喜欢这首歌的人

-  高万星GWX♂ 45秒前听过
-  五岛岛主♂ 46秒前听过
-  Neil\_Carterwip♂ 51秒前听过
-  猪得天下♀ 57秒前听过
-  打鼓的胖汪♂ 1分钟前听过

“网易云音乐” 喜欢这首歌的人栏目

# 所用算法及所需数据

## 1、基于协同过滤的具体做法



Mahout : Apache基金会基于Java的数据挖掘与机器学习类库，提供了关于推荐系统所需的分类、用户相似度计算、紧邻用户计算等工具类



## 所用算法及所需数据

### 1、基于协同过滤的优势与缺陷

优势：

(1) 推荐效果较好，用户一般会对和自己爱好相似的人看的新闻产生兴趣。

(2) 思路清晰，实现较为简单

缺陷：

(1) 对于新用户/新新闻，存在“冷启动”问题

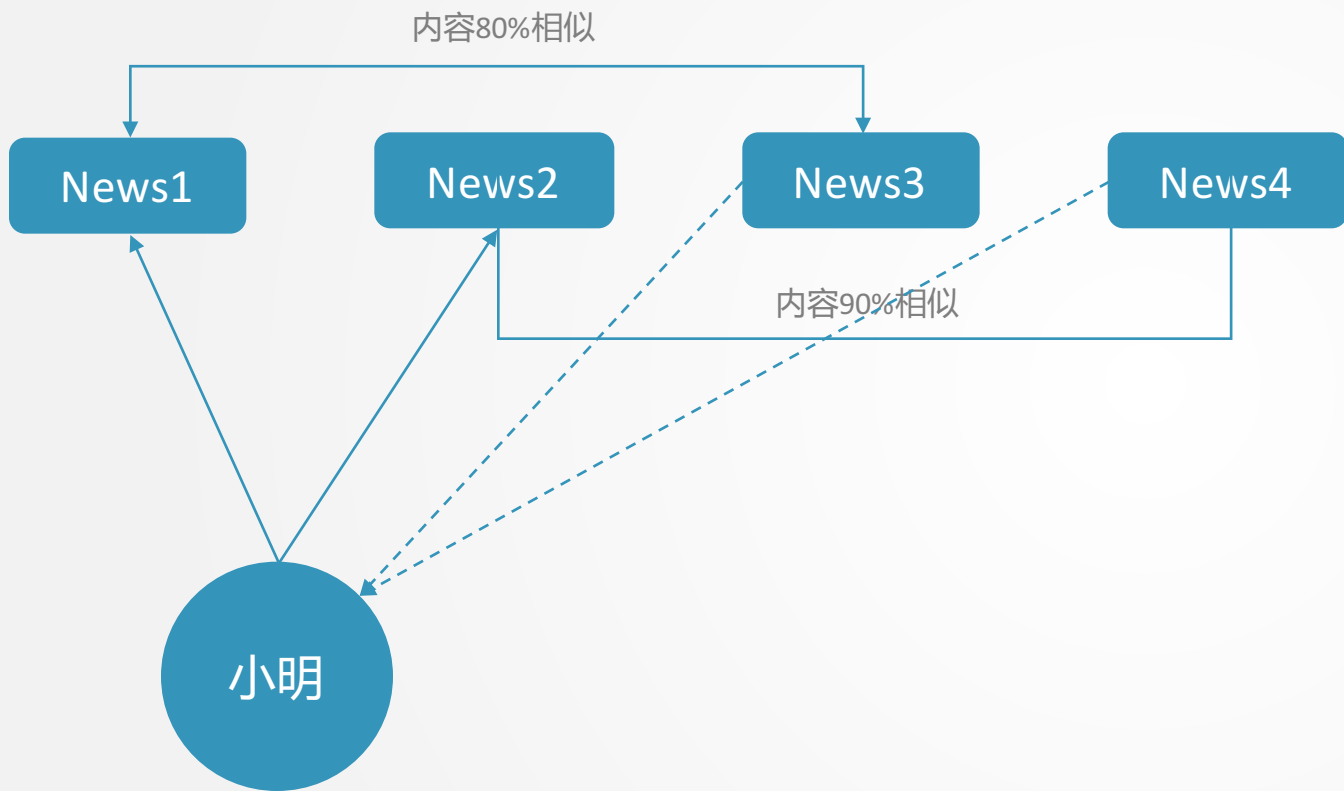
(2) 无法对用户的兴趣进行具体地把握

(3) 当活跃用户数较少时，计算效果不好

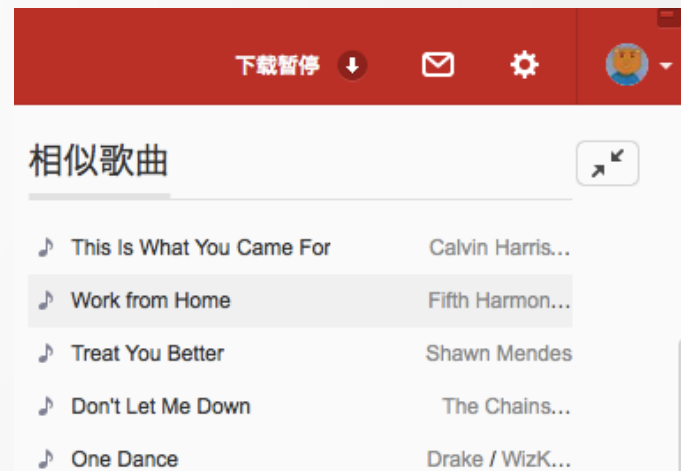
下面要介绍的“基于内容的推荐”方法，可以比较好地解决上述问题

# 所用算法及所需数据

## 2、基于内容的推荐原理



用户小明虽然只看过News1和News2，但是通过对比发现，News1和News3有80%的内容上的相似度，而News2和News4有90%的内容上的相似度，于是，有理由将News3和News4也都推荐给小明



例：“网易云音乐”的相似歌曲

## 所用算法及所需数据

### 2、基于内容的推荐原理—如何定义内容相似

新闻作为文本类的数据，本身可以从文本特征几个方面去提取它的特征信息，进而将不同的新闻间的特征信息进行比较

常见的特征信息有：新闻文本长度、新闻所属话题类型（社会、健康、国家政策）、来源（今日头条、知乎、36氪）、**关键词**（识意团队、中南财经政法大学）



威斯康辛州将重新计票，美国大选进入加时赛！希拉里能否翻盘？

例

新闻所属话题类型：国际政治

来源：今日头条

新闻关键字：美国大选、希拉里、威斯康辛州

关键词具有比较强的新闻核心内容表征能力！

# 所用算法及所需数据

## 2、基于内容的推荐原理—如何提取新闻关键词

TF-IDF算法 ( term frequency-inverse document frequency ) : TF-IDF是一种统计方法,用以评估一字词对于一个文件集或一个语料库中的其中一份文件的**重要程度**。字词的重要性随着它在文件中出现的次数成正比增加,但同时会随着它在语料库中出现的频率成反比下降。

如何理解“字词的重要性”,以及“成正比增加”与“反比下降”?

(1) “字词的重要性”:因为查找的是文本的关键词,所以要将文本中最“重要”,或者做最能体现文本内容独特性的那些词语找出来。

(2) “成正比增加”:如果一个词在文本中出现的次数越多,那么我们就越有理由认为该词就属于文本的关键词之一。

(3) “反比下降”:但是有些词如“中国”、“社会”、“媒体”等词,可能是在各个新闻里都容易出现的高频率词,针对这样的词,我们就需要以一种方式降低它对于单独文档内容的独特性贡献。即若一个词在整个语料库的所有文档里都有出现,那么在计算单个文档的关键词时,我们就会相应地调整该词属于文档关键词的可能性。

# 所用算法及所需数据

## 2、基于内容的推荐原理—用户偏好构建

如何知道用户喜欢哪些关键词呢？ --从用户历史浏览记录里挖掘

用户喜好关键词列表构建

(1) 在数据库中为每个用户维持一个关键词列表。这个关键词列表是针对每个微平台模块这一层面的。即每个模块享有一个大小相同，但内容独立的关键词列表。

如：用户小明的关键词列表：“知乎精选”:{金融、计算机科学与技术、美女.....}，“36氪”:{人工智能、VR.....}

(2) 用户浏览了某个模块的某个新闻News<sub>i</sub>。利用TF-IDF算法提取出News<sub>i</sub>的K个关键词即对应的TF-IDF值（关键程度），并将它们存入用户相应模块的关键词列表中。

如：用户小明看了“知乎精选”里的一个新闻《如何看待刘慈欣作品中透露出的对民主政治的无情嘲弄》，通过TF-IDF抽取该新闻的三个关键词：刘慈欣:100.23、三体:70.21、民主政治:96.02，于是在小明的“知乎精选”关键词列表中插入：{.....刘慈欣:100.23、三体:70.21、民主政治:96.02}。

而若小明之前的“知乎精选”列表里已经有了“三体”关键词及对应的TF-IDF值怎么办？ --将TF-IDF值进行叠加，表示用以加强用户对该关键词的感兴趣程度。

## 所用算法及所需数据

### 2、基于内容的推荐原理—用户偏好构建

问题：用户的喜好是会不断更新的？这种做法会不会导致推荐结果收敛到用户以前特别喜欢的几个关键词上？

考虑到这个问题，借用前人论文（《个性化推荐系统中用户兴趣建模研究》）中的思路：为关键词列表设置一个衰减系数 $\lambda$ ，定期对用户的喜好关键词的TF-IDF值进行更新，减少关键词的收敛倾向。

例：设置一个每天更新的 $\lambda=0.8$ （如何确定最优值）。

昨天小明的“知乎精选”关键词列表为：“{刘慈欣：100，三体80}”

今天更新为：“{刘慈欣：80，三体64}”

## 所用算法及所需数据

### 2、基于内容的推荐原理—新闻内容与用户喜好拟合度计算

有了用户的喜好关键词列表{keyword1:value1,keyword2:value2.....}，以及某条新闻的关键词列表{nkeyword1:nvalue1,nkeyword2:nvalue2.....}

只需要再做两个Map的键匹配与值的运算即可。若有相同的键，则值相乘，多个相同键的值乘积累加。若无相同的键，值记为0

例：小明的“知乎精选”关键词列表:{算法交易：100，网络游戏：200...}

“知乎精选”模块的某新闻的关键词列表:{网络游戏：100，真人扮演：80...}

那么该新闻与小明的喜好拟合度即为： $200*100=20000$

对于所有新进来的新闻计算该拟合度，将拟合度最高的N个新闻推送给用户。

## 所用算法及所需数据

### 3、考虑到若基于算法的可推荐新闻数太少—热点新闻推荐

为用户设定一个新闻推荐最小值 $N$ ，若通过两个算法生成的结果数加在一起小于 $N$ ，那么就用“热点新闻”作为余下的补充，推荐给用户。

所谓“热点新闻”即指：从用户浏览历史表newslogs中提取出的，**近期**被最多用户阅读的新闻。





3

# 所需微平台配合

Cooperating

## 所需微平台配合

推荐系统项目本身独立于微平台的mcip与app，只需要从微平台的数据库里增删改查相关信息，因此主要需要数据库的配合。（参考202.114.234.171:25432数据库连接）

具体需求：

- 1、在新闻模块中加入“推荐”模块。
- 1、将推荐新闻放在首页展示
- 2、增加newslogs表：记录用户的历史浏览信息。
- 3、增加recommend表：记录为用户生成的推荐新闻信息以及用户的反馈情况。
- 4、在users表中增加json类型字段“upreflist”，即用户的喜好关键词列表(ulabel)
- 5、在news表中增加json类型字段nkeywordlist，用以存储该新闻对应的关键词列表
- 在news表中增加字段narticle，用以存储该新闻的具体文字内容

## 所需微平台配合

推荐系统项目本身独立于微平台的mcip与app，只需要从微平台的数据库里增删改查相关信息，因此主要需要数据库的配合。（参考202.114.234.171:25432数据库连接）

| Name        | Type      | Length | Decimals | Dimen... | Allow Null                          | Key |
|-------------|-----------|--------|----------|----------|-------------------------------------|-----|
| nlid        | int8      | 0      | 0        | 0        | <input type="checkbox"/>            |     |
| nluserid    | varchar   | 20     | 0        | 0        | <input checked="" type="checkbox"/> |     |
| nnewsid     | int8      | 0      | 0        | 0        | <input checked="" type="checkbox"/> |     |
| nlttime     | timestamp | 6      | 0        | 0        | <input checked="" type="checkbox"/> |     |
| nprefer     | int8      | 0      | 0        | 0        | <input type="checkbox"/>            |     |
| nlonguserid | int8      | 0      | 0        | 0        | <input checked="" type="checkbox"/> |     |

| Name      | Type      | Length | Decimals | Dimen... | Allow Null                          | Key |
|-----------|-----------|--------|----------|----------|-------------------------------------|-----|
| ruserid   | varchar   | 20     | 0        | 0        | <input type="checkbox"/>            |     |
| rnewsid   | int8      | 0      | 0        | 0        | <input type="checkbox"/>            |     |
| rfeedback | int2      | 0      | 0        | 0        | <input checked="" type="checkbox"/> |     |
| rfbtime   | timestamp | 6      | 0        | 0        | <input checked="" type="checkbox"/> |     |



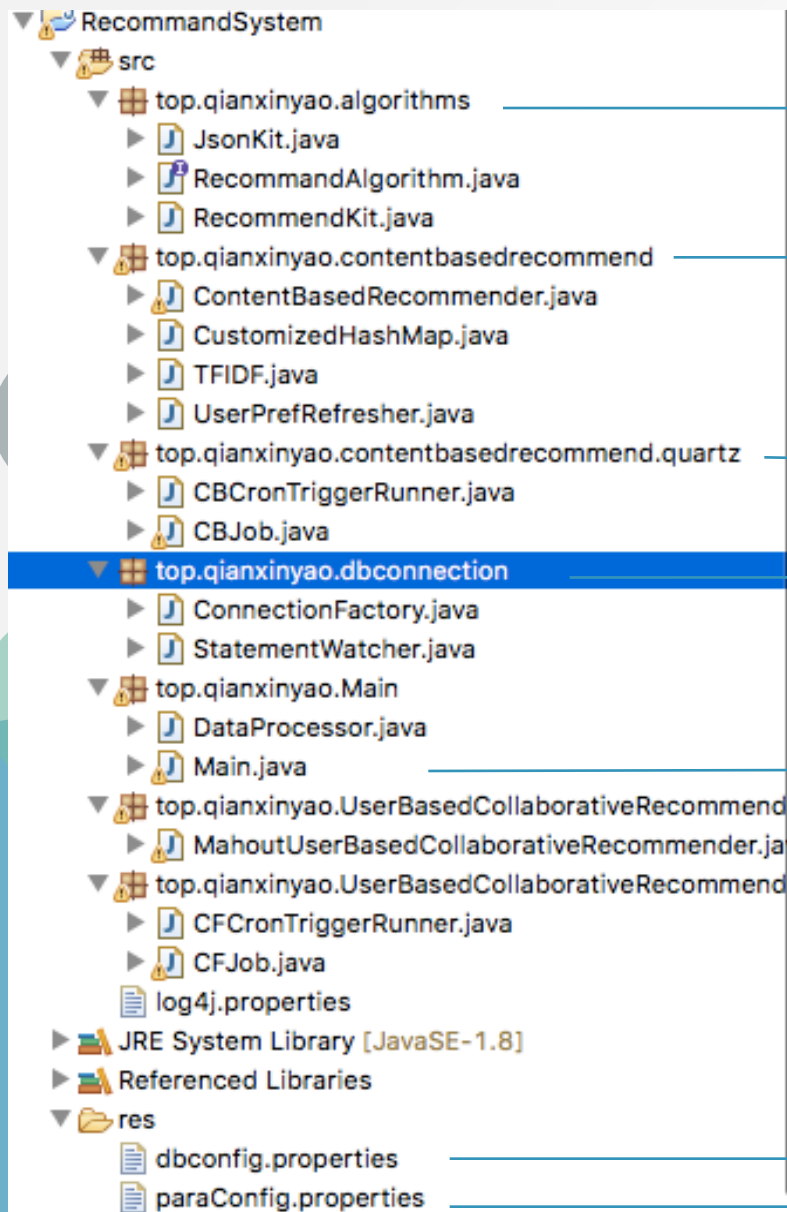
4

# 使用方法

The Approach to System

# 使用方法

## 项目架构



定义了推荐算法的接口以及一些数据处理的工具类

基于内容的推荐算法的相关类

基于内容的推荐算法下属的定时任务

推荐系统的数据库连接相关类

Main方法所在类，用于启动推荐系统

基于Mahout的协同过滤的相关类

基于内容的推荐算法下属的定时任务

数据库配置文件

系统参数配置文件

# 使用方法

推荐系统的主要工作是：

1、针对基于内容的推荐算法，推荐系统每日0点0分0秒执行一次用户喜好关键词列表的更新（关键词对应的喜好值 $\times$ 衰减系数 $\lambda$ ）。

2、完成用户关键词列表更新后，开始执行推荐。


（1）调用基于协同过滤的推荐的相关类，生成针对所有用户的N1条推荐结果，并将推荐结果以【用户ID，推荐新闻ID】的形式，插入到recommend表中。

（2）调用基于内容的推荐的相关类，生成针对所有用户的N2条推荐结果，并将推荐结果入库到recommend表中。

3、微平台给用户推送新闻时，加载recommend表中的记录。

（N1，N2均为可调节参数）

recommend表

| Name        | Type      | Length | Decimals | Dimen... | Allow Null                          | Key   |
|-------------|-----------|--------|----------|----------|-------------------------------------|---|
| ruserid     | varchar   | 20     | 0        | 0        | <input type="checkbox"/>            |   |
| rnewsid     | int8      | 0      | 0        | 0        | <input type="checkbox"/>            |   |
| rfeedback   | int2      | 0      | 0        | 0        | <input checked="" type="checkbox"/> |   |
| rtime       | timestamp | 6      | 0        | 0        | <input checked="" type="checkbox"/> |   |
| recommendid | int8      | 0      | 0        | 0        | <input type="checkbox"/>            |  |

（由于定时任务的存在，推荐系统会在一次启动后，一直保持运行。）



**THANK YOU  
FOR WATCHING**

PRESENTED BY Tom Qian