

推荐算法（系统）

推荐系统出现的原因

➤随着信息技术和互联网的发展，人们逐渐从信息匮乏的时代走入了信息过载（information overload）的时代。在这个时代，无论是信息消费者还是信息生产者都遇到了很大的挑战。

- 无明确需求
- 信息过载

什么是推荐系统

- 推荐的出现

随着电子商务规模的不断扩大，商品个数和种类快速增长，顾客需要花费大量的时间才能找到自己想买的商品。这种浏览大量无关的信息和产品过程无疑会使淹没在信息过载问题中的消费者不断流失。

- 当用户无法准确描述自己的需求时，搜索引擎的筛选效果将大打折扣。

- 在此背景下，推荐系统出现了，推荐系统的任务就是解决搜索引擎的问题，主动发现用户当前或潜在需求，并主动推送信息给用户，从而实现信息提供商与用户的双赢。

什么是推荐系统

- 推荐系统是通过建立用户与信息产品之间二元关系，利用已有的选择过程或相似性关系挖掘每个用户潜在感兴趣的对象，进而进行个性化推荐，其本质就是信息过滤，解决信息过载的问题。
- 为了解决信息过载问题，无数科学家已经提出了很多解决方案，其中最具代表性的就是分类目录和搜索引擎，这也就催生了两大互联网公司雅虎和谷歌。
- 和搜索引擎一样，推荐系统也是帮助用户快速发现有用信息的工具。和搜索引擎不同的是，推荐系统不需要用户提供明确的需求，而是通过分析用户的历史行为给用户的兴趣建模，从而主动给用户推荐能够满足他们兴趣和需求的信息。
- 因此，从某种意义上说，推荐系统和搜索引擎对于用户来说是两种互补的工具。搜索引擎满足了用户有明确目的的主动查找需求，而推荐系统在用户没有明确目的的时候帮助他们发现感兴趣的新内容。

什么是推荐系统

- 要了解推荐系统是如何工作的，可以先回顾一下现实社会中用户面对很多选择时做决定的过程。以看电影为例，一般来说，我们可能用如下方式决定最终看什么电影。
 1. 向朋友咨询。我们也许会打开聊天工具，找几个经常看电影的好朋友，问问他们有没有什么电影可以推荐。甚至，我们可以打开微博，发表一句“我要看电影”，然后等待热心人推荐电影。这种方式在推荐系统中称为社会化推荐（social recommendation），即让好友给自己推荐物品。
 2. 我们一般都有喜欢的演员和导演，有些人可能会打开搜索引擎，输入自己喜欢的演员名，然后看看返回结果中还有什么电影是自己没有看过的。比如我非常喜欢周星驰的电影，于是就去豆瓣搜索周星驰，发现他早年的一部电影我还没看过，于是就会看一看。这种方式是寻找和自己之前看过的电影在内容上相似的电影。推荐系统

什么是推荐系统

可以将上述过程自动化，通过分析用户曾经看过的电影找到用户喜欢的演员和导演，然后给用户推荐这些演员或者导演的其他电影。这种推荐方式在推荐系统中称为基于内容的推荐（content-based filtering）。

3. 我们还可能查看排行榜，比如著名的IMDB 电影排行榜，看看别人都在看什么电影，别人都喜欢什么电影，然后找一部广受好评的电影观看。这种方式可以进一步扩展：如果能找到和自己历史兴趣相似的一群用户，看看他们最近在看什么电影，那么结果可能比宽泛的热门排行榜更能符合自己的兴趣。这种方式称为基于协同过滤（collaborative filtering）的推荐。

■ 从上面3 种方法可以看出推荐算法的本质是通过一定的方式将用户和物品联系起来，而不同的推荐系统利用了不同的方式。

什么是推荐系统

下图展示了联系用户和物品的常用方式，比如利用好友、用户的历史兴趣记录以及用户的注册信息等。推荐系统常用的3种联系用户和物品的方式通过这一节的讨论，我们可以发现推荐系统就是自动联系用户和物品的一种工具，它能够在信息过载的环境中帮助用户发现令他们感兴趣的信息，也能将信息推送给对它们感兴趣的用户。下一节将通过推荐系统的实际例子让大家加深对推荐系统的了解。



图1-2 推荐系统常用的3种联系用户和物品的方式

推荐系统和搜索引擎的异同

➤相同点:

- 都是一种帮助用户快速发现有用信息的工具

➤不同点:

- 搜索引擎需要用户主动提供准确的关键词来寻找信息
- 推荐系统不需要用户提供明确的需求，而是通过分析用户的历史行为给用户的兴趣建模

➤从某种意义上说，推荐系统和搜索引擎对于用户来说是两个互补的工具

- 搜索引擎满足了用户有明确目的时的主动查找需求
- 推荐系统能够在用户没有明确目的的时候帮助他们发现感兴趣的新内容

个性化的推荐系统应用

■ 个性化推荐系统的应用和搜索引擎不同，个性化推荐系统需要依赖用户的行为数据，因此一般都是作为一个应用存在于不同网站之中。在互联网的各类网站中都可以看到推荐系统的应用，而个性化推荐系统在这些网站中的主要作用是通过分析大量用户行为日志，给不同用户提供不同的个性化页面展示，来提高网站的点击率和转化率。广泛利用推荐系统的领域包括电子商务、电影和视频、音乐、社交网络、阅读、基于位置的服务、个性化邮件和广告等。尽管不同的网站使用不同的推荐系统技术，但总地说来，几乎所有的推荐系统应用都是由前台的展示页面、后台的日志系统以及推荐算法系统3部分构成的。

1.2.1 亚马逊的推荐化列表

- 推荐结果的标题、缩略图以及其他内容属性
- 推荐结构平均分
- 推荐理由
- 亚马逊根据用户的历史行为给用户做推荐，因此如果它给你推荐了一本金庸小说，大多是你曾经在亚马逊上对武侠小说方面的书给过喜欢的反馈，基于物品的推荐。

亚马逊 amazon.cn

全部分类

浏览 全部商品分类

曹正凤的亚马逊 Z秒杀 礼品卡 我要开店 海外购 帮助 In English

您好, 曹正凤 我的帐户 购物车

我的亚马逊 我的浏览记录 为我推荐 **改善为我推荐** 我的个人主页 了解更多

您的亚马逊

图书

图书	作者	评分	价格	推荐理由
Applied Multivariate Statistical Analysis	Richard A. Johnson, Dean W. Wichern	★★★★★ (25)	¥ 1,150.00	为什么推荐这些商品?
应用多元统计分析	沃尔夫冈·哈德勒(Wolfgang Hurdle)	★★★★★ (4)	¥ 65.00 ¥ 52.00	为什么推荐这些商品?
多元统计分析	何晓群	★★★★★ (11)	¥ 39.00 ¥ 29.30	为什么推荐这些商品?
例解回归分析	查特吉 (Samprit Chatterjee)	★★★★★ (11)	¥ 69.00 ¥ 52.50	为什么推荐这些商品?
统计推断	George Casella	★★★★★ (16)	¥ 66.00 ¥ 55.50	为什么推荐这些商品?
时间序列分析及应用	克萊尔(Jonathan D. Cryer)	★★★★★ (35)	¥ 48.00 ¥ 24.00	为什么推荐这些商品?

查看 图书 中的所有推荐商品

1.2.2 电影和视频网站

- 在电影和视频网站中，个性化推荐系统也是一种重要的应用，它能帮助用户在浩瀚的视频库中找到自己感兴趣的内容。
- Netflix是电影和视频网站个性化推荐系统，其推荐结果展示页面包含了以下几个部分
 1. 电影的标题和海报。
 2. 用户反馈模块——包括Play（播放）、评分和Not Interested（不感兴趣）3种。
 3. 推荐理由——因为用户曾经喜欢过别的电影。
- 基于物品的推荐

Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos				
1	BellKor's Pragmatic Chaos	0.8567	10.06	2009-07-26 18:18:28
2	The Ensemble	0.8567	10.06	2009-07-26 18:38:22
3	Grand Prize Team	0.8582	9.90	2009-07-10 21:24:40
4	Opera Solutions and Vandelay United	0.8588	9.84	2009-07-10 01:12:31
5	Vandelay Industries I	0.8591	9.81	2009-07-10 00:32:20
6	PragmaticTheory	0.8594	9.77	2009-06-24 12:06:56
7	BellKor in BigChaos	0.8601	9.70	2009-06-13 08:14:09
8	Dace	0.8612	9.59	2009-07-24 17:18:43
9	Feeds2	0.8622	9.48	2009-07-12 13:11:51
10	BigChaos	0.8623	9.47	2009-04-07 12:33:59
11	Opera Solutions	0.8623	9.47	2009-07-24 00:34:07
12	BellKor	0.8624	9.46	2009-07-26 17:19:11
Progress Prize 2008 - RMSE = 0.8627 - Winning Team: BellKor in BigChaos				
13	xiangliang	0.8642	9.27	2009-07-15 14:53:22
14	Gravity	0.8643	9.26	2009-04-22 18:31:32
15	Cee	0.8651	9.18	2009-06-21 19:24:53

1.2.3 社交网络

- 最近几年，互联网最激动人心的产品莫过于以Facebook 和 Twitter为代表的社交网络应用。在社交网络中，好友们可以互相分享、传播信息。社交网络中的个性化推荐技术主要应用在3个方面：
 1. 利用用户的社交网络信息对用户进行个性化的物品推荐；
 2. 信息流的会话推荐；
 3. 给用户推荐好友。
- Facebook 最宝贵的数据有两个，一个是用户之间的社交网络关系，另一个是用户的偏好信息。因此Facebook 推出了一个推荐API，称为Instant Personalization。该工具根据用户好友喜欢的信息，给用户推荐他们的好友最喜欢的物品。很多网站都使用了Facebook 的API 来实现网站的个性化。

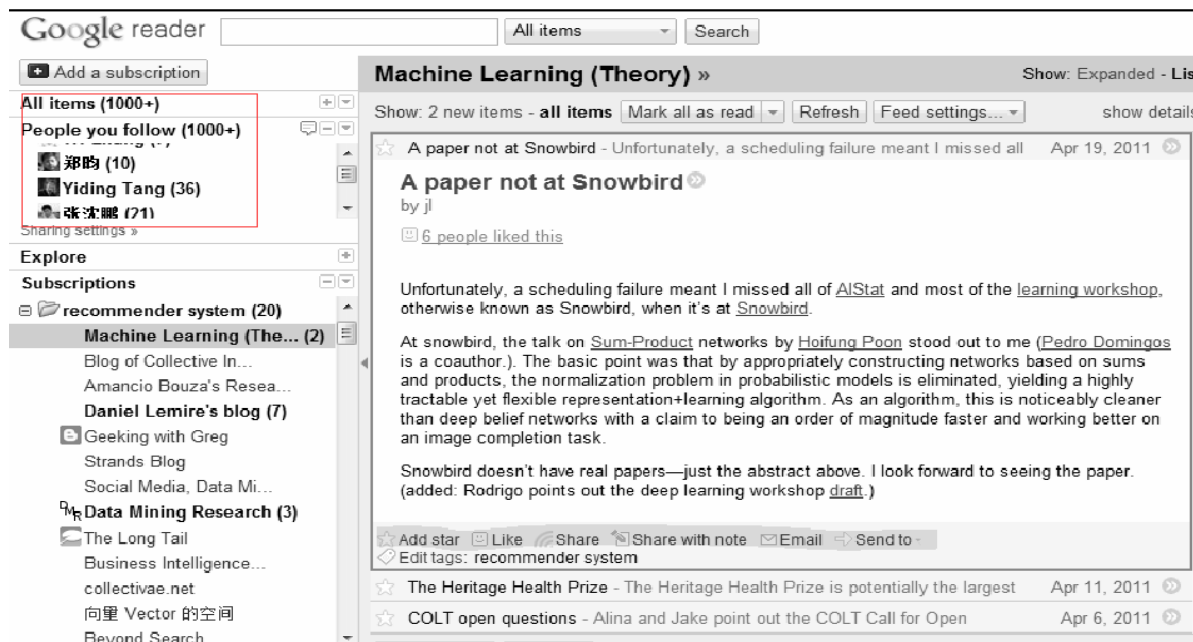
1.2.4 个性化阅读

■ 阅读文章是很多互联网用户每天都会做的事情。个性化阅读同样符合前面提出的需要个性化推荐的两个因素：首先，互联网上的文章非常多，用户面临信息过载的问题；其次，用户很多时候并没有必须看某篇具体文章的需求，他们只是想通过阅读特定领域的文章了解这些领域的动态。

■ 目前互联网上的个性化阅读工具很多，国际知名的有Google Reader，国内有鲜果网等。同时，随着移动设备的流行，移动设备上针对个性化阅读的应用也很多，其中具有代表性的有Zite 和Flipboard。

1.2.4 个性化阅读

- Google Reader 是一款流行的社会化阅读工具。它允许用户关注自己感兴趣的人，然后看到所关注用户分享的文章。
- 如下图所示，如果单击左侧的People you follow（你关注的人），就可以看到其他用户分享的文章。



1.2.4 个性化阅读

- 就像订报纸、杂志一样，您可以从近300万个博客、新闻网站、报刊杂志、电子书、网络小说中订阅任何喜欢的内容。
- 比如某明星的博客、南方周末、三联生活周刊、华尔街日报、新浪新闻等，鲜果将把订阅的新内容第一时间推送给您。



1.2.5 基于位置的服务

■ 在中关村闲逛时，肚子饿了，打开手机，发现上面给你推荐了几家中关村不错的饭馆，价格、环境、服务、口味都如你所愿，这几乎就是基于位置的个性化推荐系统最理想的场景了。随着移动设备的飞速发展，用户的位置信息已经非常容易获取，而位置是一种很重要的上下文信息，基于位置给用户推荐离他近的且他感兴趣的服务，用户就更有可能去消费。基于位置的服务往往和社交网络结合在一起。比如大众点评网等等。

1.2.6 个性化广告

■ 广告是互联网公司生存的根本。很多互联网公司的盈利模式都是基于广告的，而广告的CPC、CPM 直接决定了很多互联网公司的收入。

■ 目前，很多广告都是随机投放的，即每次用户来了，随机选择一个广告投放给他。这种投放的效率显然很低，比如给男性投放化妆品广告或者给女性投放西装广告多半都是一种浪费。因此，很多公司都致力于广告定向投放（Ad Targeting）的研究，即如何将广告投放给它的潜在客户群。个性化广告投放目前已经成为了一门独立的学科——计算广告学——但该学科和推荐系统在很多基础理论和方法上是相通的，比如它们的目的都是联系用户和物品，只是在个性化广告中，物品就是广告。

■ 个性化广告投放和狭义个性化推荐的区别是，个性化推荐着重于帮助用户找到可能令他们感兴趣的物品，而广告推荐着重于帮助广告找到可能对它们感兴趣的用戶，即一个是以用户为核心，而另一个以广告为核心。目前的个性化广告投放技术主要分为3种。

1.2.6 个性化广告

1. 上下文广告 通过分析用户正在浏览的网页内容，投放和网页内容相关的广告。代表系统是谷歌的AdSense。
 2. 搜索广告 通过分析用户在当前会话中的搜索记录，判断用户的搜索目的，投放和用户目的相关的广告。
 3. 个性化展示广告 我们经常在很多网站看到大量展示广告（就是那些大的横幅图片），它们是根据用户的兴趣，对不同用户投放不同的展示广告。雅虎是这方面研究的代表。
-
- 在个性化广告方面最容易获得成功的无疑是Facebook，因为它拥有大量的用户个人资料，可以很容易地获取用户的兴趣，让广告商选择自己希望对其投放广告的用户。在Facebook 的广告系统界面，允许广告商选择自己希望的用户群，然后Facebook 会根据广告商的选择告诉他们这些限制条件下广告将会覆盖的用户数量。

1.2.7 推荐系统模块

1. 用户建模模块
2. 推荐对象建模模块
3. 推荐算法模块

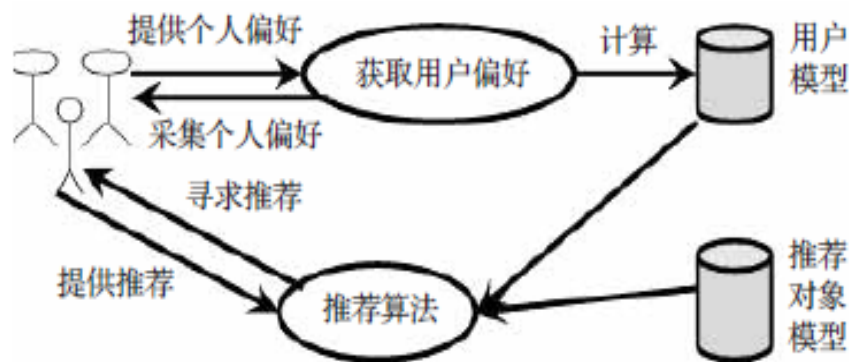


图1 推荐系统通用模型

■一个完整的推荐系统由3个部分组成：收集用户信息的**行为记录**模块，分析用户喜好的**模型分析**模块和**推荐算法**模块。

1.3 推荐系统的发展历程

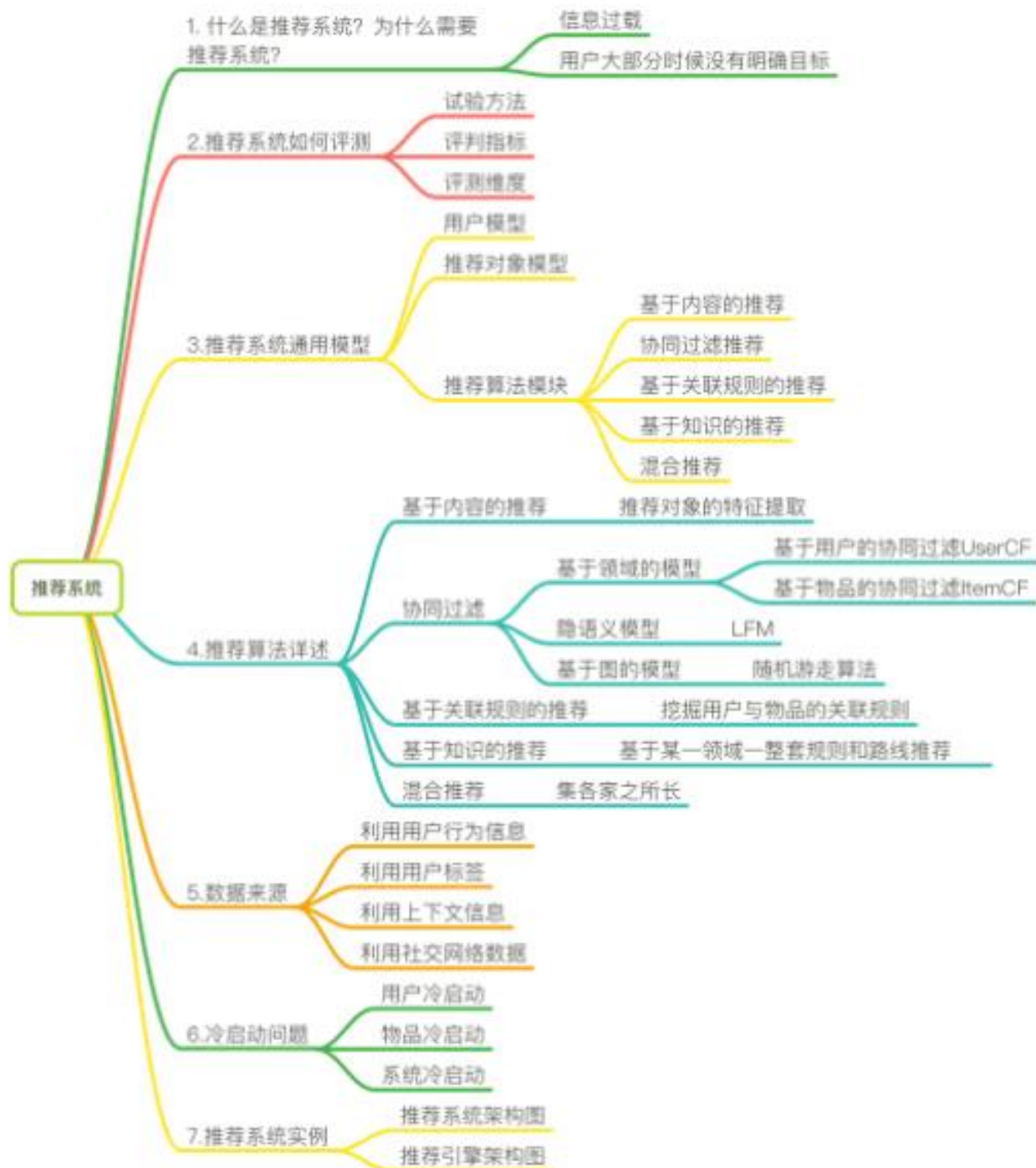
- 1995年3月，卡耐基.梅隆大学的RobertArmstrong等人在美国人工智能协会上提出了个性化导航系统Web Watcher；斯坦福大学的MarkoBalabanovic等人在同一会议上推出了个性化推荐系统LIRA；
- 1995年8月，麻省理工学院的Henry Lieberman在国际人工智能联合大会（IJCAI）上提出了个性化导航智能体Letizia；
- 1996年，Yahoo 推出了个性化入口My Yahoo；
- 1997年，AT&T实验室提出了基于协作过滤的个性化推荐系统PHOAKS和Referral Web；
- 1999年，德国Dresden技术大学的Tanja Joerding实现了个性化电子商务原型系统TELLIM；
- 2000年，NEC研究院的Kurt等人为搜索引擎CiteSeer增加了个性化推荐功能；
- 2001年，纽约大学的Gediminas Adoavicius和Alexander Tuzhilin实现了个性化电子商务网站的用户建模系统；

1.3 推荐系统的发展历程

- 2001年，IBM公司在其电子商务平台Websphere中增加了个性化功能，以便商家开发个性化电子商务网站。
- 2003年，Google开创了AdWords盈利模式，通过用户搜索的关键词来提供相关的广告。AdWords的点击率很高，是Google广告收入的主要来源。2007年3月开始，Google为AdWords添加了个性化元素。不仅仅关注单次搜索的关键词，而是对用户一段时间内的搜索历史进行记录和分析，据此了解用户的喜好和需求，更为精确地呈现相关的广告内容。
- 2007年，雅虎推出了SmartAds广告方案。雅虎掌握了海量的用户信息，如用户的性别、年龄、收入水平、地理位置以及生活方式等，再加上对用户搜索、浏览行为的记录，使得雅虎可以为用户呈现个性化的横幅广告。

1.3 推荐系统的发展历程

- 2009年，Overstock（美国著名的网上零售商）开始运用ChoiceStream公司制作的个性化横幅广告方案，在一些高流量的网站上投放产品广告。Overstock在运行这项个性化横幅广告的初期就取得了惊人的成果，公司称：“广告的点击率是以前的两倍，伴随而来的销售增长也高达20%至30%。”
- 2009年7月，国内首个推荐系统科研团队北京百分点信息科技有限公司成立，该团队专注于推荐引擎技术与解决方案，在其推荐引擎技术与数据平台上汇集了国内外百余家知名电子商务网站与资讯类网站，并通过这些B2C网站每天为数以千万计的消费者提供实时智能的商品推荐。
- 2011年9月，百度世界大会2011上，李彦宏将推荐引擎与云计算、搜索引擎并列为未来互联网重要战略规划以及发展方向。百度新首页将逐步实现个性化，智能地推荐出用户喜欢的网站和经常使用的APP。



主要的推荐算法

- 根据推荐算法的不同，推荐系统可以分为以下几类：
 - 关联分析
 - 点进率预测算法
 - 协同过滤（collaborative filtering）系统；
 - 基于内容（content-based）的推荐系统；
 - 混合（hybrid）推荐系统；

4.推荐算法详述



1、关联分析

×关联规则挖掘：

+在交易数据、关系数据或其他信息载体中，查找存在于项目集合或对象集合之间的频繁模式、关联、相关性、或因果结构。

×应用：

+购物篮分析、交叉销售、产品目录设计、 loss-leader analysis、聚集、分类等。

×举例：

+规则形式： “**Body → Head [support, confidence]**”.

+buys(x, “diapers”) → buys(x, “beers”) [0.5%, 60%]

+major(x, “CS”) ^ takes(x, “DB”) → grade(x, “A”) [1%, 75%]

1、关联分析

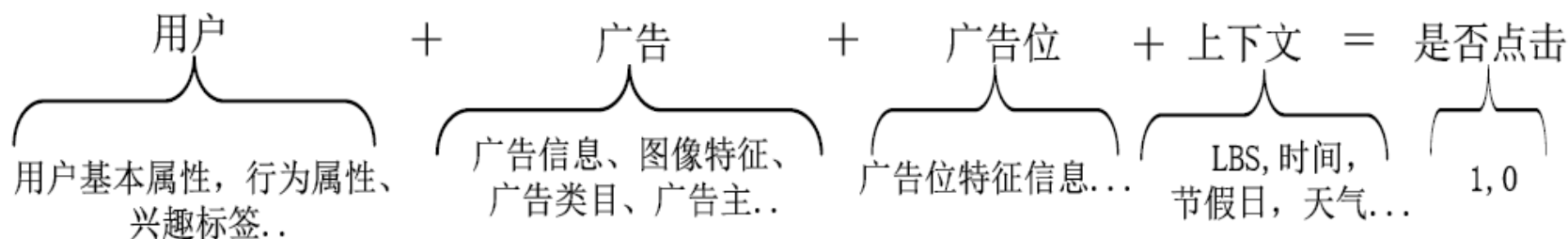
- 给定: (1)交易数据库 (2)每笔交易是: 一个项目列表 (消费者一次购买活动中购买的商品)
- 查找: 所有描述一个项目集合与其他项目集合相关性的规则
 - E.g., *98% of people who purchase tires and auto accessories also get automotive services done*
- 应用
 - * \Rightarrow 护理用品 (商店应该怎样提高护理用品的销售?)
 - 家用电器 \Rightarrow * (其他商品的库存有什么影响?)
 - 在产品直销中使用附加邮寄

2、点击率预测模型

$$CPM = pCTR \times CPC$$

广告主每次点击出价
代表广告主获取流量愿意付出的成本
(采用二阶拍卖机制)

预估点击率
代表用户点击
广告的意愿



3、协同过滤

- 在现今的推荐技术和算法中，最被大家广泛认可和采用的就是基于协同过滤的推荐方法。
- 要理解什么是协同过滤 (Collaborative Filtering, 简称 CF), 首先想一个简单的问题，如果你现在想看个电影，但你不知道具体看哪部，你会怎么做？大部分的人会问问周围的朋友，看看最近有什么好看的电影推荐，而我们一般更倾向于从口味比较类似的朋友那里得到推荐。这就是协同过滤的核心思想。
- 换句话说，就是借鉴和你相关人群的观点来进行推荐。
- 协同过滤就是指用户可以齐心协力，通过不断地和网站互动，使自己的推荐列表能够不断过滤掉自己不感兴趣的物品，从而越来越满足自己的需求。

3、协同过滤的实现

收集数据

- 这里的数据指的都是用户的历史行为数据，比如用户的购买历史，关注，收藏行为，或者发表了某些评论，给某个物品打了多少分等等，这些都可以用来作为数据供推荐算法使用，服务于推荐算法。

找到相似用户和物品

- 计算用户间以及物品间的相似度。
- 有多种计算相似度的方法（欧几里德距离、皮尔逊相关系数、Cosine 相似度等等）

进行推荐

- 基于用户的协同过滤推荐
- 基于物品的协同过滤推荐

3、协同过滤算法分类

- 协同过滤算法大致可分为两类，基于物品的与基于用户的；
- 区分很简单，若你挖掘的关系是物品与物品间的，就是基于物品的协同过滤算法，
- 若你挖掘的关系是用户与用户间的，就是基于用户的协同过滤算法。

■ 基于用户的 CF 的基本思想

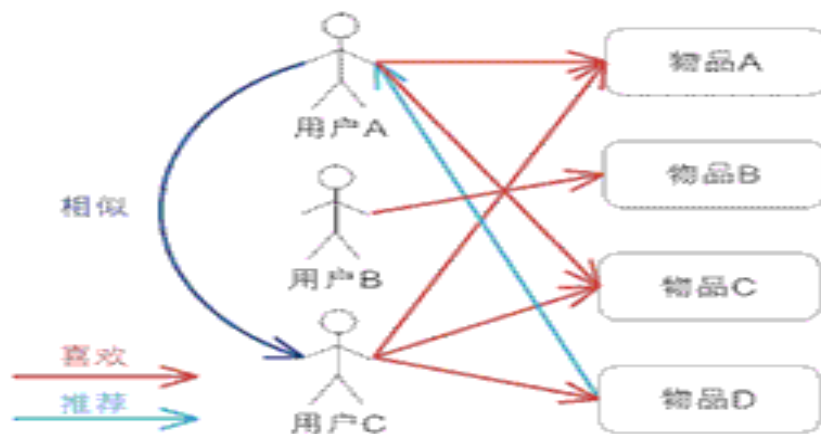
- 基于用户对物品的偏好找到相邻邻居用户，然后将邻居用户喜欢的推荐给当前用户。
- 计算上，就是将一个用户对所有物品的偏好作为一个向量来计算用户之间的相似度，找到 K 邻居后，根据邻居的相似度权重以及他们对物品的偏好，预测当前用户没有偏好的未涉及物品，计算得到一个排序的物品列表作为推荐。

3、协同过滤算法分类

■ 例子

- 用户对于用户 A，根据用户的历史偏好，这里只计算得到一个邻居 - 用户 C，然后将用户 C 喜欢的物品 D 推荐给用户 A。

用户/物品	物品A	物品B	物品C	物品D
用户A	√		√	推荐
用户B		√		
用户C	√		√	√



3、协同过滤算法分类

■ 基于物品的 CF 的基本思想

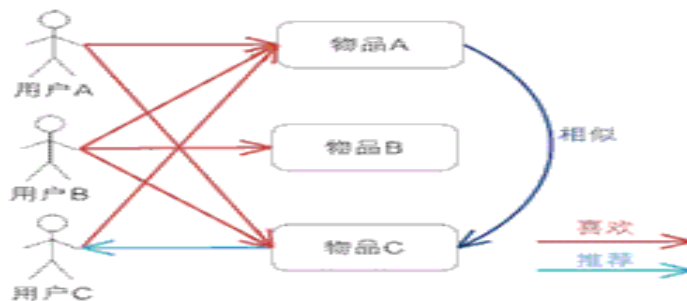
- 在计算邻居时采用物品本身，而不是从用户的角度，即基于用户对物品的偏好找到相似的物品，然后根据用户的历史偏好，推荐相似的物品给他。
- 从计算的角度看，就是将所有用户对某个物品的偏好作为一个向量来计算物品之间的相似度，得到物品的相似物品后，根据用户历史的偏好预测当前用户还没有表示偏好的物品，计算得到一个排序的物品列表作为推荐。

3 协同过滤算法分类

■ 例子

- 对于物品 A，根据所有用户的历史偏好，喜欢物品 A 的用户都喜欢物品 C，得出物品 A 和物品 C 比较相似，而用户 C 喜欢物品 A，那么可以推断出用户 C 可能也喜欢物品 C。

用户/物品	物品A	物品B	物品C
用户A	√		√
用户B	√	√	√
用户C	√		推荐



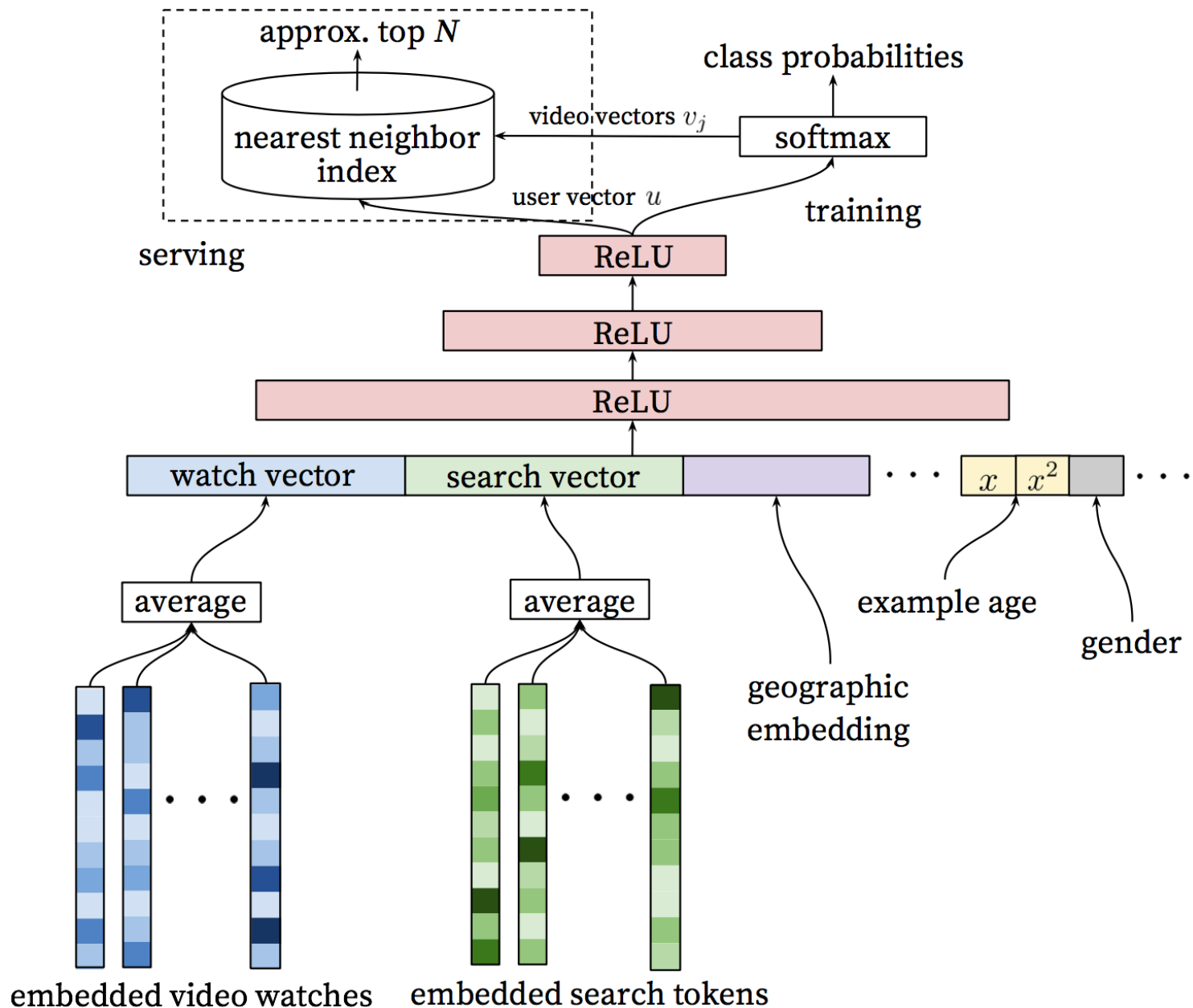
4、基于内容的推荐系统

- 根据用户过去喜欢的产品，为用户推荐和他过去喜欢的产品相似的产品。例如，一个推荐饭店的系统可以依据某个用户之前喜欢很多的烤肉店而为他推荐烤肉店。
- 核心思想：
分别对用户和产品建立**配置文件**，通过分析已经购买（或浏览）过的内容，建立或更新用户的配置文件。系统可以比较用户与产品配置文件的相似度，并直接向用户推荐与其配置文件最相似的产品。基于内容的推荐算法的根本在于**信息获取**和**信息过滤**。（用户的配置文件构建与更新是最为核心的部分之一）

4、基于内容的推荐系统

■ 基于内容的推荐过程

1. Item Representation: 为每个item抽取出一些特征（也就是item的内容了）来表示此item;
2. Profile Learning: 利用一个用户过去喜欢（及不喜欢）的item的特征数据，来学习出此用户的喜好特征（profile）；
3. Recommendation Generation: 通过比较上一步得到的用户profile与候选item的特征，为此用户推荐一组相关性最大的item。



对比

■ 例子——个性化阅读

- 对于个性化阅读来说，一个item就是一篇文章。根据上面的第一步，我们首先要从文章内容中抽取出代表它们的属性。常用的方法就是利用出现在一篇文章中词来代表这篇文章，而每个词对应的权重往往使用信息检索中的tf-idf来计算，一篇抽象的文章可以使用具体的一个向量表示了。
- 第二步就是根据用户过去喜欢什么文章来产生此用户喜好的 profile了，最简单的方法可以把用户所有喜欢的文章对应的向量的平均值作为此用户的profile。比如某个用户经常关注与推荐系统有关的文章，那么他的profile中“CB”、“CF”和“推荐”对应的权重值就会较高。
- 在获得了一个用户的profile后，CB就可以利用所有item与此用户profile的相关度对他进行推荐文章了。一个常用的相关度计算方法是cosine。最终把候选item里与此用户最相关（cosine值最大）的N个item作为推荐返回给此用户。

对比

■ 基于内容的和协同过滤的区别

- 举个简单的小例子，我们已知道
用户u1喜欢的电影是A, B, C
用户u2喜欢的电影是A, C, E, F
用户u3喜欢的电影是B, D
我们要解决的问题是：决定对u1是不是应该推荐F这部电影？
- 基于内容的做法：要分析F的特征和u1所喜欢的A、B、C的特征，需要知道的信息是A（战争片），B（战争片），C（剧情片），如果F（战争片），那么F很大程度上可以推荐给u1，这是基于内容的做法，你需要对item进行特征建立和建模。
- 协同过滤的办法：那么你完全可以忽略item的建模，因为这种办法的决策是依赖user和item之间的关系，也就是这里的用户和电影之间的关系。我们不再需要知道ABCF哪些是战争片，哪些是剧情片，我们只需要知道用户u1和u2按照item向量表示，他们的相似度比较高，那么我们可以把u2所喜欢的F这部影片推荐给u1。

5、混合推荐

- 各种推荐方法都有各自的优缺点，在实际应用中可以针对具体问题采用推荐算法的组合进行推荐
- 目的 是通过组合不同的推荐策略，达到扬长避短的目的，从而产生更符合用户需求的推荐

推荐系统的实验方法

■ 在介绍推荐系统的指标之前，首先看一下计算和获得这些指标的主要实验方法。在推荐系统中，主要有3种评测推荐效果的实验方法，即离线实验（offline experiment）、用户调查（user study）和在线实验（online experiment）。

■ 下面将分别介绍这3种实验方法的优缺点。

1. 离线实验

离线实验的方法一般由如下几个步骤构成：

- 通过日志系统获得用户行为数据，并按照一定格式生成一个标准的数据集；
- 将数据集按照一定的规则分成训练集和测试集；
- 在训练集上训练用户兴趣模型，在测试集上进行预测；
- 通过事先定义的离线指标评测算法在测试集上的预测结果。

1.4.1 推荐系统的实验方法

- 从上面的步骤可以看到，推荐系统的离线实验都是在数据集上完成的，也就是说它不需要一个实际的系统来供它实验，而只要有一个从实际系统日志中提取的数据集即可。这种实验方法的好处是不需要真实用户参与，可以直接快速地计算出来，从而方便、快速地测试大量不同的算法。它的主要缺点是无法获得很多商业上关注的指标，如点击率、转化率等，而找到和商业指标非常相关的离线指标也是很困难的事情。

2. 用户调查

- 离线实验的指标和实际的商业指标存在差距，比如预测准确率和用户满意度之间就存在很大差别，高预测准确率不等于高用户满意度。因此，如果要准确评测一个算法，需要相对比较真实的环境。最好的方法就是将算法直接上线测试，但在对算法会不会降低用户满意度不太有把握的情况下，上线测试具有较高的风险，所以在上线测试前一般需要做一次称为用户调查的测试。

1.4.1 推荐系统的实验方法

- 用户调查需要有一些真实用户，让他们在需要测试的推荐系统上完成一些任务。在他们完成任务时，我们需要观察和记录他们的行为，并让他们回答一些问题。最后，我们需要通过分析他们的行为和答案了解测试系统的性能。用户调查是推荐系统评测的一个重要工具，很多离线时没有办法评测的与用户主观感受有关的指标都可以通过用户调查获得。比如，如果我们想知道推荐结果是否很令用户惊喜，那我们最好直接询问用户。但是，用户调查也有一些缺点。首先，用户调查成本很高，需要用户花大量时间完成一个个任务，并回答相关的问题。有些时候，还需要花钱雇用测试用户。因此，大多数情况下很难进行大规模的用户调查，而对于参加人数较少的用户调查，得出的很多结论往往没有统计意义。因此，我们在做用户调查时，一方面要控制成本，又要保证结果的统计意义。此外，测试用户也不是随便选择的。需要尽量保证测试用户的分布和真实用户的分布相同，比如男女各半，以及年龄、活跃度的分布都和真实用户分布尽量相同。

1.4.1 推荐系统的实验方法

- 此外，用户调查要尽量保证是双盲实验，不要让实验人员和用户事先知道测试的目标，以免用户的回答和实验人员的测试受主观成分的影响。
 - 用户调查的优缺点也很明显。它的优点是可以获得很多体现用户主观感受的指标，相对在线实验风险很低，出现错误后很容易弥补。缺点是招募测试用户代价较大，很难组织大规模的测试用户，因此会使测试结果的统计意义不足。此外，在很多时候设计双盲实验非常困难，而且用户在测试环境下的行为和真实环境下的行为可能有所不同，因而在测试环境下收集的测试指标可能在真实环境下无法重现。
- ### 3. 在线实验
- 在线实验在完成离线实验和必要的用户调查后，可以将推荐系统上线做AB测试，将它和旧的算法进行比较。

1.4.1 推荐系统的实验方法

- 一般来说，一个新的推荐算法最终上线，需要完成上面所说的3个实验。
 - 首先，需要通过离线实验证明它在很多离线指标上优于现有的算法。
 - 然后，需要通过用户调查确定它的用户满意度不低于现有的算法。
 - 最后，通过在线的AB测试确定它在我们关心的指标上优于现有的算法。
- 介绍完3种主要的实验方法后，下一节将开始介绍推荐系统常用的实验指标，这些指标大部分都可以通过本节介绍的3种实验方法获得。



1.4.2 评测指标

- 本节将介绍各种推荐系统的评测指标。这些指标可用于评价推荐系统各方面的性能。这些指标有些可以定量计算，有些只能定性描述，有些可以通过离线实验计算，有些需要通过用户调查获得，还有些只能在线评测。下面将讨论各个不同的指标。

1. 用户满意度

用户作为推荐系统的重要参与者，其满意度是评测推荐系统的最重要指标。但是，用户满意度没有办法离线计算，只能通过用户调查或者在线实验获得。用户调查获得用户满意度主要是通过调查问卷的形式。用户对推荐系统的满意度分为不同的层次，比如，如下一个论文推荐系统设计的问卷调查：

1.4.2 评测指标

- 推荐的论文都是我非常想看的。
 - 推荐的论文很多我都看过了，确实是符合我兴趣的不错论文。
 - 推荐的论文和我的研究兴趣是相关的，但我并不喜欢。
 - 不知道为什么推荐这些论文，它们和我的兴趣丝毫没有关系。
-
- 由此可以看出，这个调查问卷不是简单地询问用户对结果是否满意，而是从不同的侧面询问用户对结果的不同感受。比如，如果仅仅问用户是否满意，用户可能心里认为大体满意，但是对某个方面还有点不满，因而可能很难回答这个问题。因此在设计问卷时需要考虑到用户各方面的感受，这样用户才能针对问题给出自己准确的回答。在在线系统中，用户满意度主要通过一些对用户行为的统计得到。比如在电子商务网站中，用户如果购买了推荐的商品，就表示他们在一定程度上满意。因此，我们可以利用购买率度量用户的满意度。此外，有些网站会通过设计一些用户反馈界面收集用户满意度。
 - 更一般的情况下，我们可以用点击率、用户停留时间和转化率等指标度量用户的满意度。

1.4.2 评测指标

2. 预测准确度

预测准确度度量一个推荐系统或者推荐算法预测用户行为的能力。这个指标是最重要的推荐系统离线评测指标，该指标可以通过离线实验计算，计算该指标时需要有一个离线的数据集，该数据集包含用户的历史行为记录。然后，将该数据集通过时间分成训练集和测试集。最后，通过在训练集上建立用户的行为和兴趣模型预测用户在测试集上的行为，并计算预测行为和测试集上实际行为的重合度作为预测准确度。例如，评分预测，很多提供推荐服务的网站都有一个让用户给物品打分的功能，那么，如果知道了用户对物品的历史评分，就可以从中习得用户的兴趣模型，并预测该用户在将来看到一个他没有评过分的物品时，会给这个物品评多少分。预测用户对物品评分的行为称为评分预测。

1.4.2 评测指标

- 预测准确度的一个经典度量方法就是度量系统的预测得分和用户的实际得分的平均绝对误差MAE。

$$MAE = \frac{1}{c} \sum_{a=1}^c |v_{ia} - r_{ia}|$$

其中c为系统中用户i打分产品的个数， v_{ia} 为用户实际得分， r_{ia} 为系统的预测得分。

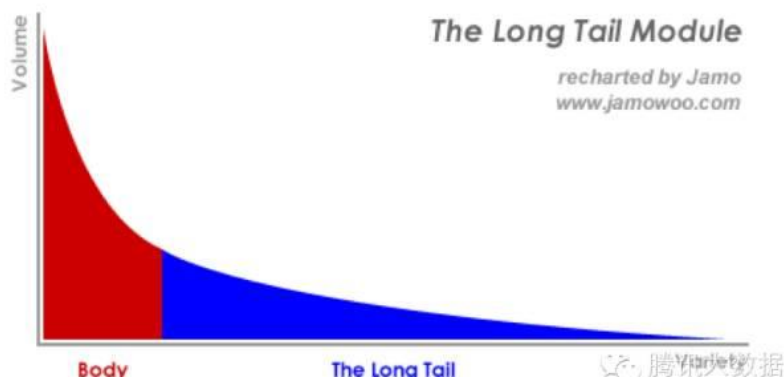
- 均方根误差（RMSE），它是预测值与真值偏差的平方和观测次数n比值的平方根。均方根误差能够很好地反映出测量的精密度。

$$MSE = \sqrt{\frac{1}{n_t} \sum_{(i,a)} |v_{ia} - r_{ia}|^2}$$

1.4.2 评测指标

3. 覆盖率

覆盖率（coverage）描述一个推荐系统对物品长尾的发掘能力。覆盖率有不同的定义方法，最简单的定义为推荐系统能够推荐出来的物品占总物品集合的比例。覆盖率是一个内容提供商会关心的指标。以图书推荐为例，出版社可能会很关心他们的书有没有被推荐给用户。覆盖率为100%的推荐系统可以将每个物品都推荐给至少一个用户。此外，从上面的定义也可以看到，热门排行榜的推荐覆盖率是很低的，它只会推荐那些热门的物品，这些物品在总物品中占的比例很小。一个好的推荐系统不仅需要有比较高的用户满意度，也要有较高的覆盖率。



1.4.2 评测指标

4. 多样性

用户的兴趣是广泛的，在一个视频网站中，用户可能既喜欢看《猫和老鼠》一类的动画片，也喜欢看成龙的动作片。那么，为了满足用户广泛的兴趣，推荐列表需要能够覆盖用户不同的兴趣领域，即推荐结果需要具有多样性。多样性推荐列表的好处用一句俗语表述就是“不在一棵树上吊死”。尽管用户的兴趣在较长的时间跨度中是一样的，但具体到用户访问推荐系统的某一刻，其兴趣往往是单一的，那么如果推荐列表只能覆盖用户的一个兴趣点，而这个兴趣点不是用户这个时刻的兴趣点，推荐列表就不会让用户满意。反之，如果推荐列表比较多样，覆盖了用户绝大多数的兴趣点，那么就会增加用户找到感兴趣物品的概率。因此给用户的推荐列表也需要满足用户广泛的兴趣，即具有多样性。

1.4.2 评测指标

5. 新颖性

新颖的推荐是指给用户推荐那些他们以前没有听说过的物品。在一个网站中实现新颖性的最简单办法是，把那些用户之前在网站中对其有过行为的物品从推荐列表中过滤掉。比如在一个视频网站中，新颖的推荐不应该给用户推荐那些他们已经看过、打过分或者浏览过的视频。但是，有些视频可能是用户在别的网站看过，或者是在电视上看过，因此仅仅过滤掉本网站中用户有过行为的物品还不能完全实现新颖性。新颖度的最简单方法是利用推荐结果的平均流行度，因为越不热门的物品越可能让用户觉得新颖。因此，如果推荐结果中物品的平均热门程度较低，那么推荐结果就可能有比较高的新颖性。

1.4.2 评测指标

6. 惊喜度

它的基本意思就是，如果推荐结果和用户的历史兴趣不相似，但却让用户觉得满意，那么就可以说推荐结果的惊喜度很高，而推荐的新颖性仅仅取决于用户是否听说过这个推荐结果。

7. 信任度

如果你有两个朋友，一个人你很信任，一个人经常满嘴跑火车，那么如果你信任的朋友推荐你去某个地方旅游，你很有可能听从他的推荐，但如果是那位满嘴跑火车的朋友推荐你去同样的地方旅游，你很有可能不去。这两个人可以看做两个推荐系统，尽管他们的推荐结果相同，但用户却可能产生不同的反应，这就是因为用户对他们有不同的信任度。对于基于机器学习的自动推荐系统，同样存在信任度（trust）的问题，如果用户信任推荐系统，那就会增加用户和推荐系统的交互。

1.4.2 评测指标

- 提高推荐系统的信任度主要有两种方法。首先需要增加推荐系统的透明度（transparency），而增加推荐系统透明度的主要办法是提供推荐解释。只有让用户了解推荐系统的运行机制，让用户认同推荐系统的运行机制，才会提高用户对推荐系统的信任度。其次是考虑用户的社交网络信息，利用用户的好友信息给用户做推荐，并且用好友进行推荐解释。这是因为用户对他们的好友一般都比较信任，因此如果推荐的商品是好友购买过的，那么他们对推荐结果就会相对比较信任。

8. 实时性

实时性在很多网站中，因为物品（新闻、微博等）具有很强的时效性，所以需要在物品还具有时效性时就将它们推荐给用户。比如，给用户推荐昨天的新闻显然不如给用户推荐今天的新闻。因此，在这些网站中，推荐系统的实时性就显得至关重要。

1.4.2 评测指标

- 推荐系统的实时性包括两个方面。首先，推荐系统需要实时地更新推荐列表来满足用户新的行为变化。比如，当一个用户购买了iPhone，如果推荐系统能够立即给他推荐相关配件，那么肯定比第二天再给用户推荐相关配件更有价值。很多推荐系统都会在离线状态每天计算一次用户推荐列表，然后于在线期间将推荐列表展示给用户。这种设计显然是无法满足实时性的。与用户行为相应的实时性，可以通过推荐列表的变化速率来评测。如果推荐列表户有行为后变化不大，或者没有变化，说明推荐系统的实时性不高。实时性的第二个方面是推荐系统需要能够将新加入系统的物品推荐给用户。这主要考验了推荐系统处理物品冷启动的能力。关于如何将新加入系统的物品推荐给用户，将在后面的章节进行讨论，而对于新物品推荐能力，我们可以利用用户推荐列表中有多大比例的物品是当天新加的来评测。

1.4.2 评测指标

9. 健壮性

任何一个能带来利益的算法系统都会被人攻击，这方面最典型的例子就是搜索引擎。搜索引擎的作弊和反作弊斗争异常激烈，这是因为如果能让自己的商品成为热门搜索词的第一个搜索果，会带来极大的商业利益。推荐系统目前也遇到了同样的作弊问题，而健壮性指标衡量了一个推荐系统抗击作弊的能力。比如最著名的就是行为注入攻击（profile injection attack）。众所周知，绝大部分推荐系统都是通过分析用户的行为实现推荐算法的。比如，亚马逊有一种推荐叫做“购买商品A的用户也经常购买的其他商品”。它的主要计算方法是统计购买商品A的用户购买其他商品的次数。那么，我们可以很简单地攻击这个算法，让自己的商品在这个推荐列表中获得比较高的排名，比如可以注册很多账号，用这些账号同时购买A和自己的商品。

1.4.2 评测指标

还有一种攻击主要针对评分系统，比如豆瓣的电影评分。这种攻击很简单，就是雇用一批人给自己的商品非常高的评分，而评分行为是推荐系统依赖的重要用户行为。

算法健壮性的评测主要利用模拟攻击。首先，给定一个数据集和一个算法，可以用这个算法给这个数据集中的用户生成推荐列表。然后，用常用的攻击方法向数据集中注入噪声数据，然后利用算法在注入噪声后的数据集上再次给用户生成推荐列表。最后，通过比较攻击前后推荐列表的相似度评测算法的健壮性。如果攻击后的推荐列表相对于攻击前没有发生大的变化，就说明算法比较健壮。在实际系统中，提高系统的健壮性，除了选择健壮性高的算法，还有以下方法。设计推荐系统时尽量使用代价比较高的用户行为。比如，如果有用户购买行为和用户浏览行为，那么主要应该使用用户购买行为，因为购买需要付费，所以攻击购买行为的代价远远大于攻击浏览行为。在使用数据前，进行攻击检测，从而对数据进行清理。

1.4.2 评测指标

10. 商业目标

很多时候，网站评测推荐系统更加注重网站的商业目标是否达成，而商业目标和网站的盈利模式是息息相关的。一般来说，最本质的商业目标就是平均一个用户给公司带来的盈利。不过这种指标不是很难计算，只是计算一次需要比较大的代价。因此，很多公司会根据自己的盈利模式设计不同的商业目标。不同的网站具有不同的商业目标。比如电子商务网站的目标可能是销售额，基于展示广告盈利的网站其商业目标可能是广告展示总数，基于点击广告盈利的网站其商业目标可能是广告点击总数。因此，设计推荐系统时需要考虑最终的商业目标，而网站使用推荐系统的目的除了满足用户发现内容的需求，也需要利用推荐系统加快实现商业上的指标。

1.4.2 评测维度

■ 上一节介绍了很多评测指标，但是在评测系统中还需要考虑评测维度，比如一个推荐算法，虽然整体性能不好，但可能在某种情况下性能比较好，而增加评测维度的目的就是知道一个算法在什么情况下性能最好。这样可以为融合不同推荐算法取得最好的整体性能带来参考。

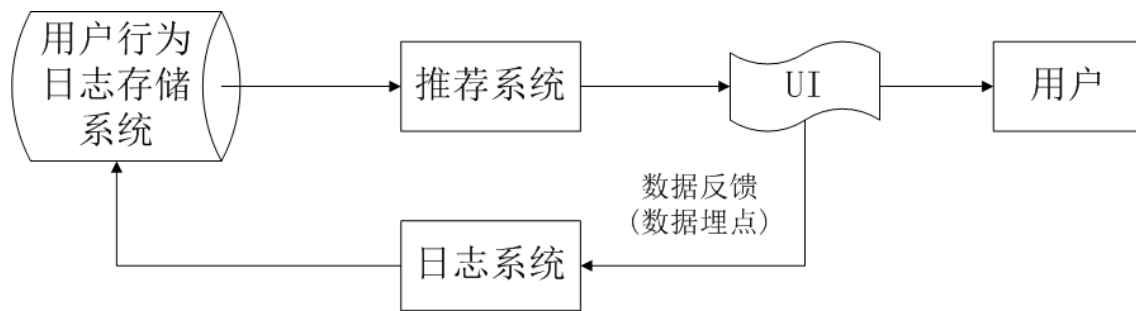
■ 一般来说，评测维度分为如下3种。

1. 用户维度 主要包括用户的人口统计学信息、活跃度以及是不是新用户等。
2. 物品维度 包括物品的属性信息、流行度、平均分以及是不是新加入的物品等。
3. 时间维度 包括季节，是工作日还是周末，是白天还是晚上等。

■ 如果能够在推荐系统评测报告中包含不同维度下的系统评测指标，就能帮我们全面地了解推荐系统性能，找到一个看上去比较弱的算法的优势，发现一个看上去比较强的算法的缺点。

1.9 推荐系统架构

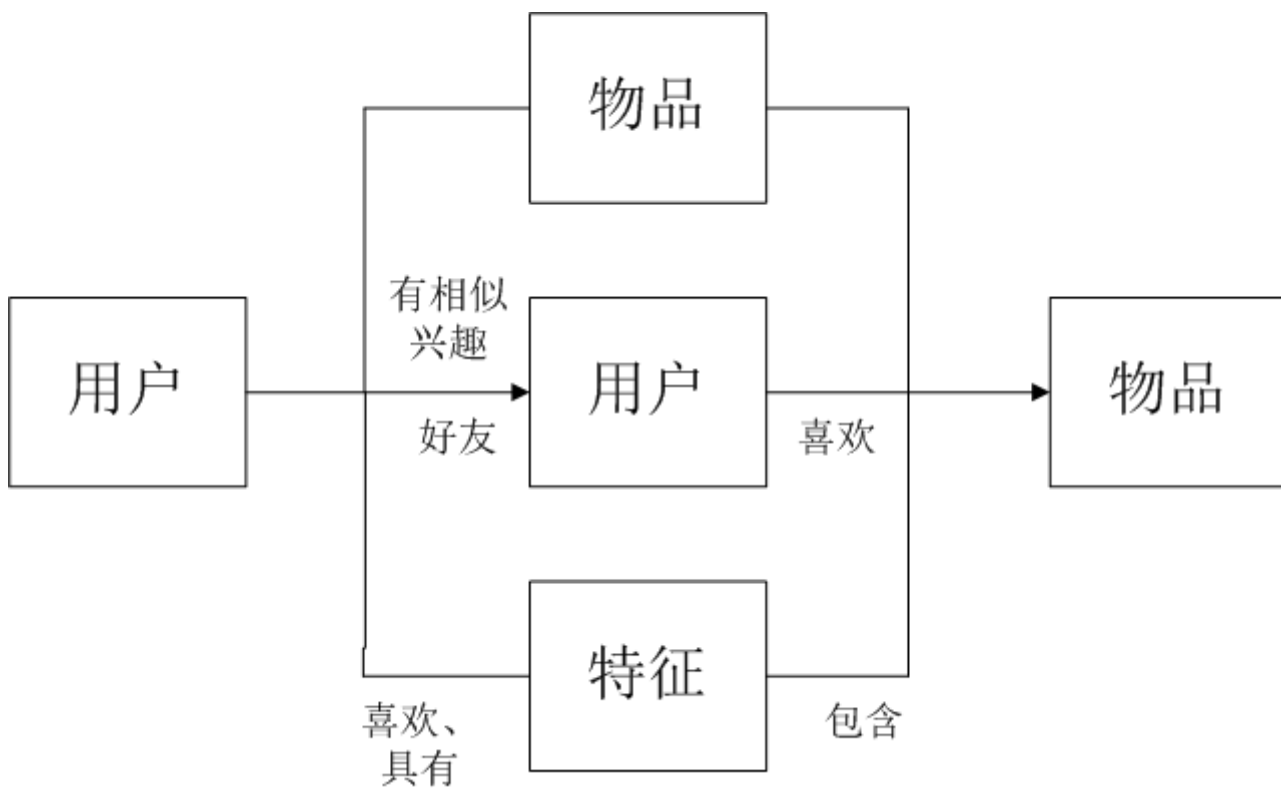
- 从互联网收集并整理了推荐系统的架构，其中包括一些大公司的推荐系统框架（数据流存储、计算、模型应用），可以参考这些资料，取长补短，最后根据自己的业务需求，技术选型来设计相应的框架。



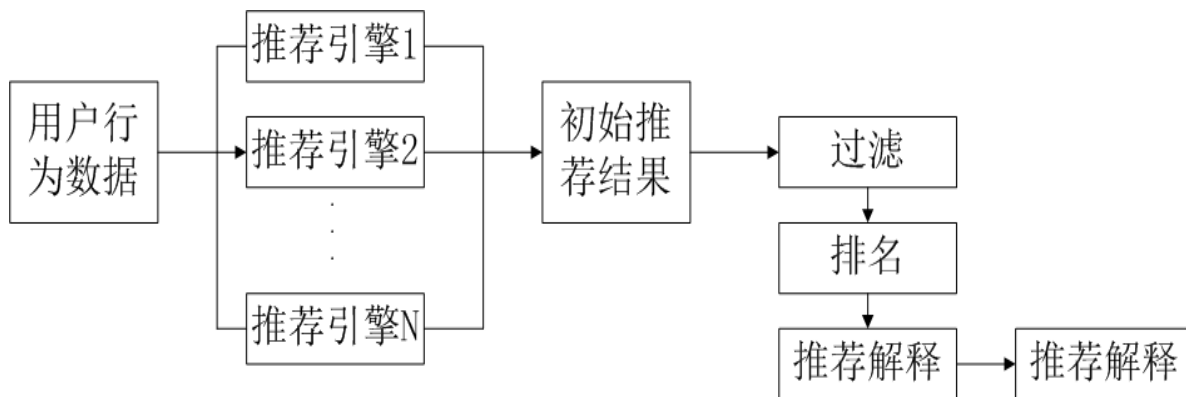
- 界面UI那一块包含3块东西：1) 通过一定方式展示推荐物品(物品标题、缩略图、简介等)；2) 给的推荐理由；3) 数据反馈改进个性化推荐；关于用户数据的存放地方：1)数据库/缓存用来实时取数据；2) hdfs（Hadoop分布式文件系统）文件上面。

1.9 推荐系统架构

- 抽象出来的三种推荐方式



1.9 推荐系统架构



- 推荐引擎的构建来源于不同的数据源(也就是用户的特征有很多种类, 例如统计的、行为的、主题的)+不同的推荐模型算法, 推荐引擎的架构可以试多样化的(实时推荐的+离线推荐的), 然后融合推荐结果(人工规则+模型结果), 融合方式多样的, 有线性加权的或者切换式的等。

1.9 推荐系统架构

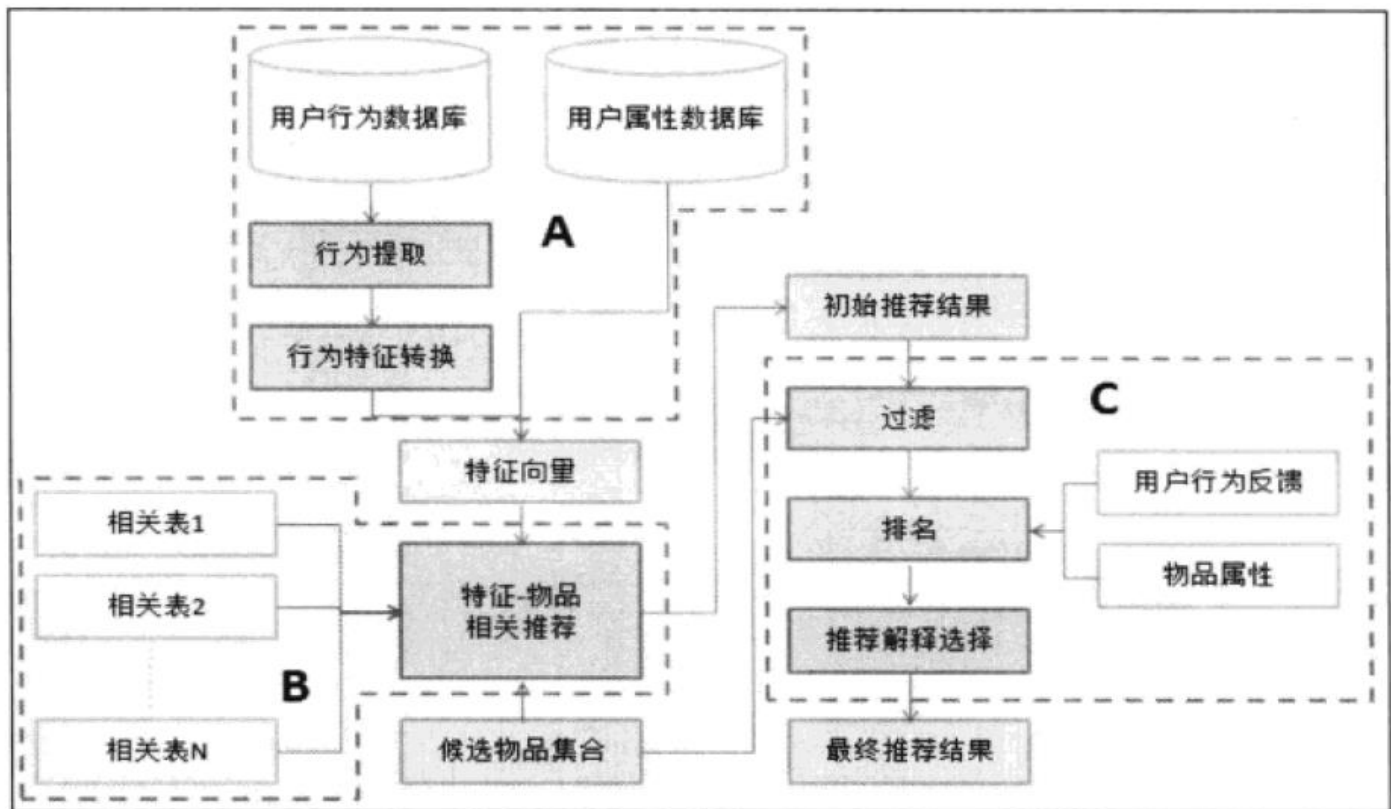


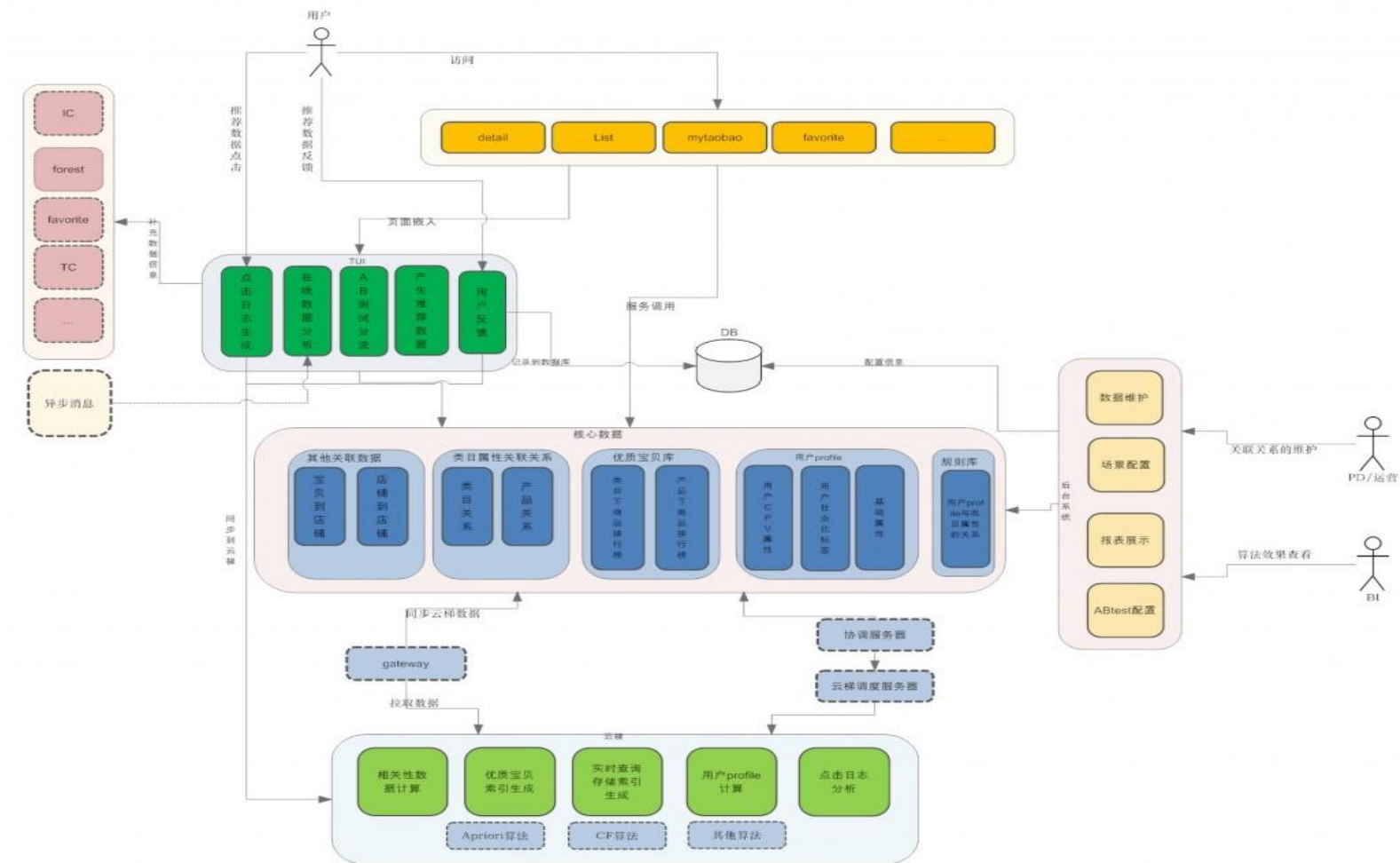
图7-9 推荐引擎的架构图

- A模块负责用户各类型特征的收集，B模块的相关表是根据上图中的推荐引擎来生成的，B模块的输出推荐结果用来C模块的输入，中间经过过滤模块（用户已经产生行为的物品，非候选物品，业务方提供的物品黑名单等），排名模块也根据预设定的推荐目标来制定，最后推荐解释的生成（这是可能是最容易忽视，但很关键的一环，微信的好友推荐游戏，这解释已经胜过后台的计算作用了）。

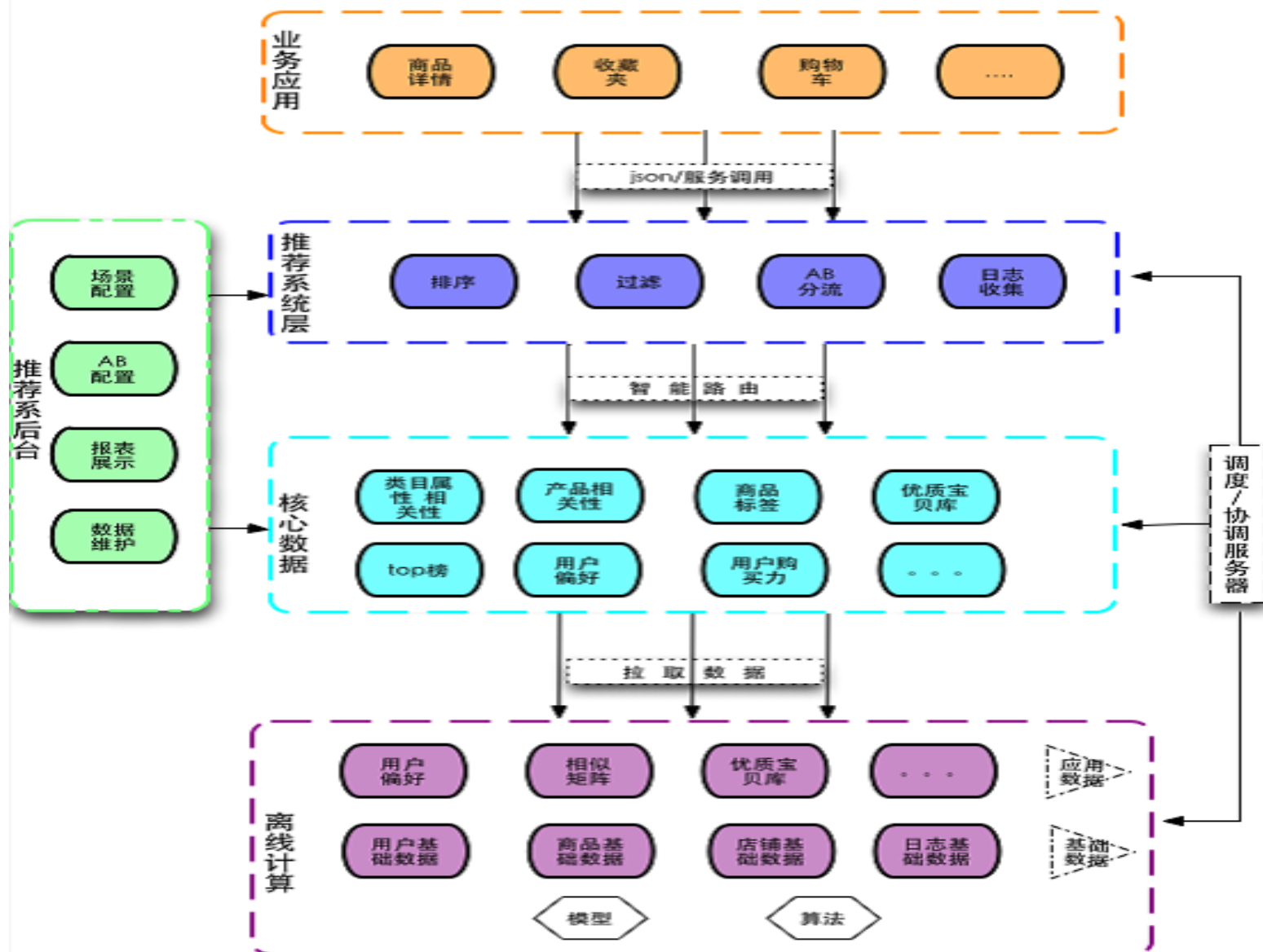
1.9 推荐系统架构

■ 淘宝的推荐系统(详细跟简单版)

推荐引擎系统结构

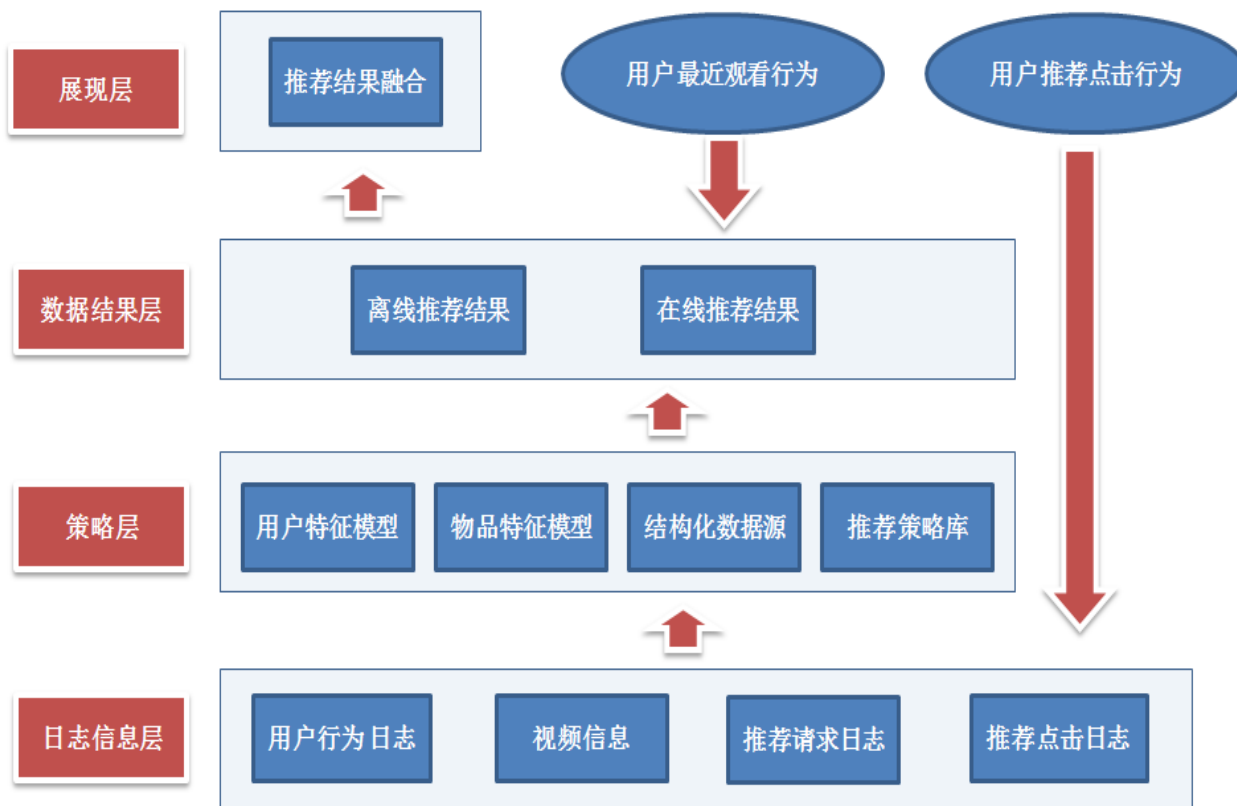


1.9 推荐系统架构



1.9 推荐系统架构

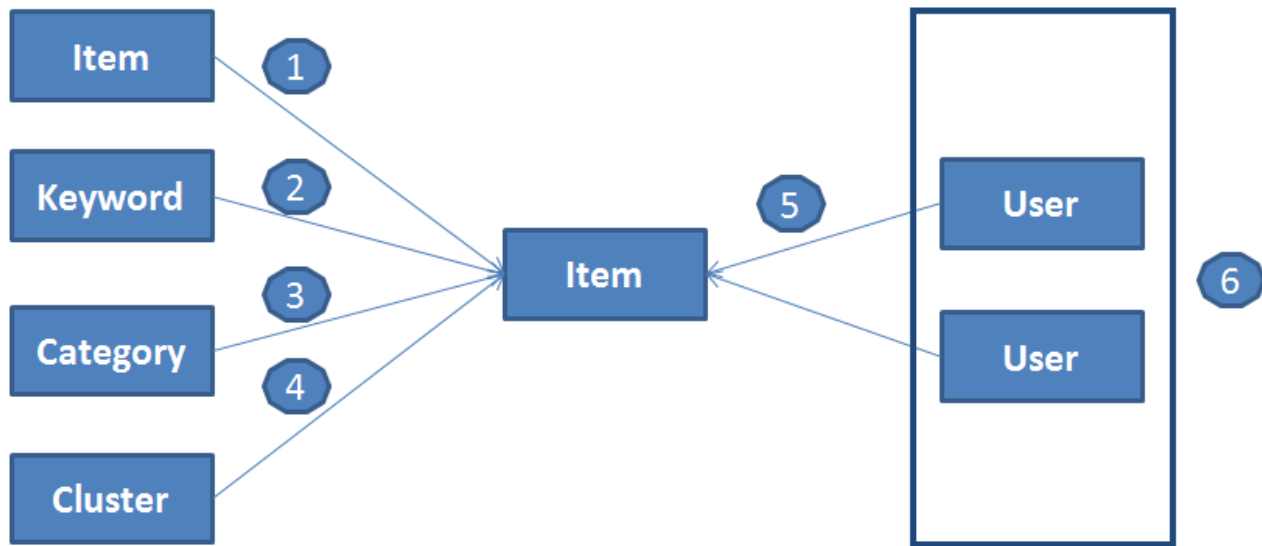
- 总结：淘宝的推荐系统，描述了推荐引擎搭建的整体架构，包括离线的分布式计算和存储、监控、数据统计和分析、实验平台等。给我们搭建推荐引擎提供了很好的建议。整体流程大致这样。通过后台的分布式计算，将算法产生的算法结果数据存储进入一种介质中，首推hbase。然后，通过一种叫做云梯的机制将算法结果推入中间层介质中，供推荐系统在线部分调用。在线部分提供引擎和实验分流，用户的行为将存储进入hadoop中，数据统计分析平台由hive来搭建，主要用来分析和统计hadoop中的用户行为log。这张图不仅讲了，推荐系统的架构流程，也讲了跟这个平台有关系的人，是怎么介入的，我觉得提供的信息可很好的参考。



1.9 推荐系统架构

- 模型前数据准备(理解数据源, 用户, 物品)

理解关系 – 结构化数据源



- | | |
|---------------|--------------|
| ① 物品与物品之间的关联 | ④ 聚类与物品之间的关联 |
| ② 关键词与物品之间的关联 | ⑤ 用户与物品之间的关联 |
| ③ 类目与物品之间的关联 | ⑥ 用户与用户之间的关联 |

1.9 推荐系统架构

理解用户 – 用户特征模型

使用设备特征	PC、Phone、Pad
人口统计学特征	性别、年龄段
访问时间特征	上次登陆时间、一天访问时间的分布
访问频次特征	一天访问次数、一周访问次数、一周点击推荐次数
关注物品特征	关注物品的热度、关注物品的类目
其他行为特征	搜索、收藏、顶踩、分享

1.9 推荐系统架构

理解物品 – 物品特征模型

物品特征模型

物品内容特征（视频、图像、文本等）

物品标签

物品所属类目、聚类

物品流行度、时效性

物品社会特征（适合男性、女性还是老年、儿童等）

物品价值评价

1.9 推荐系统架构

- 模型策略

Association Rules	计算物品相关性
classify	预测用户性别、年龄段
cluster	物品聚类、用户分群
dimension reduction	计算相似用户
time series analysis	计算用户兴趣模型转移

1.9 推荐系统架构

- 其他需要考虑的场景

关注点

个性化推荐排序与搜索排序的差别

基于行为的推荐与基于内容的推荐的差别

分时分场景推荐

推荐算法A/B测试

人工数据审查（阅读数据）

从 bad case 中学习

对比协同过滤，深度学习召回模型的优势

更全面的行爲表达

在模型中结合点击，收藏，搜索等多种行为，更全面地表示用户行为偏好

可添加画像特征

可加入性别，地域等用户画像相关的特征

包含行为顺序

模型中考虑用户的行为顺序

组合复杂特征

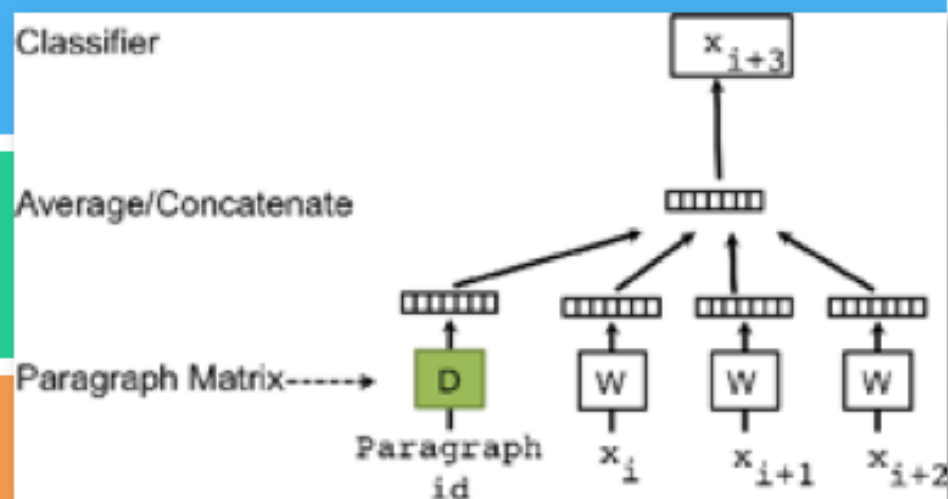
神经网络进行更复杂的特征组合，挖掘更深层次的关联关系

基于神经网络的文本语义分析模型

利用自然语言处理技术，得到能表示每篇文章主题内容的语义向量

根据语义向量，推荐用户喜欢的主题的文章

只需要分析文章内容，不需要用户行为，能很好的推荐新文章，解决内文章的冷启动问题



基于行为

- 推荐跟用户喜欢的内容相似的内容
- 推荐相似用户喜欢的内容
- 使用深度学习模型，根据用户行为数据训练模型

基于用户画像

- 利用已有的关注关系
- 基于标签构建用户画像
- 根据用户的兴趣偏好，召回相关的内容

冷启动&多样性

- 运用自然语言处理技术，对新文章进行语义分析，解决新文章的冷启动问题
- 运用UCB探索用户的潜在兴趣算法，保证推荐结果的多样性

多源数据

- 热门
- 编辑精选
- 最新
- 本地化内容

数据根基

推荐系统架构

