中国科学技术大学

硕士学位论文

基于用户画像的手机主题推荐系统

作者姓名: 胡磊

学科专业: 信息安全专业

导师姓名: 周武旸 教授

张四海 博士

完成时间: 二〇一五年十二月

University of Science and Technology of China

A dissertation for master's degree

The Phone Theme Recommendation System Based on User Profile

Author :	Lei Hu
Speciality :	Information Security
Supervisor :	Prof. Wuyang Zhou
	Dr. Sihai Zhang
Finished Time :	December 12th, 2015

中国科学技术大学学位论文原创性声明

本人声明所呈交的学位论文，是本人在导师指导下进行研究工作所取得的成果。除已特别加以标注和致谢的地方外，论文中不包含任何他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的贡献均已在论文中作了明确的说明。

作者签名：_____　　签字日期：_____

中国科学技术大学学位论文授权使用声明

作为申请学位的条件之一，学位论文著作权拥有者授权中国科学技术大学拥有学位论文的部分使用权，即：学校有权按有关规定向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅，可以将学位论文编入《中国学位论文全文数据库》等有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。本人提交的电子文档的内容和纸质论文的内容相一致。

保密的学位论文在解密后也遵守此规定。

□公开　□保密（____ 年）

作者签名：_____　　导师签名：_____

签字日期：_____　　签字日期：_____

摘 要

信息爆炸使得用户很难有效的从海量的数据中快速获取自己需要的信息，推荐系统凭借精准定位和” 千人千面” 的个性化服务受到互联网企业的青睐和研究者的重视。本论文讨论了如何构建一个基于用户画像模块和用户兴趣探索模块的手机主题推荐系统，并详细介绍了用户画像建模和用户兴趣探索。

在小米科技的工作经历让笔者意识到传统的个性化推荐系统面临着诸多挑战，其中最根本的问题是如何根据企业的商业目标和业务特点来优化推荐系统，具体到手机主题行业，推荐系统需要解决社交化、长尾性、冷启动、动态推荐等一系列综合问题。由此，笔者提出并实现了一种基于用户画像和用户兴趣探索的手机主题个性化推荐系统。本文的主要工作和贡献如下:

- 实现了推荐系统的用户画像模块：利用信息检索（IR）技术从用户注册信息获取到用户的人口属性、职业、地理位置、性别等信息并标签化，不同标签的来源，标签的传递路径，转发关系，标签的本身，以及标签与用户之间的共现关系决定了这个标签对应的权重，权重越高则认为该标签的可信度越高。AB 测试显示结合了用户画像的推荐系统提升了约 8% 的点击转换率。

- 实现了推荐系统的用户兴趣探索模块：为了迎合用户不断变化着的兴趣，用户兴趣探索使用了特征提取技术和用户满意度量化技术，对每个用户维护一个能动态变化的短期兴趣标签向量空间。首先，利用用户兴趣特征向量和商品特征向量计算出用户-商品的相关分数。然后，利用用户行为（购买，评分，点赞，划屏频率等）量化用户满意度。一次成功的用户兴趣标签探索，首先应该有很低的相关分数和很高的满意度，其次兴趣标签应该是一个小众兴趣标签。实验表明示结合了用户兴趣探索的推荐系统能显著提升推荐商品的多样性。

- 系统地研究了时效性对推荐系统的影响：用户画像针对的是用户的静态信息，代表了用户的长期兴趣，用户兴趣探索针对的是用户的动态信息，代表了用户的短期兴趣，对于不同的行业俩者对用户行为的影响程度也不同。本文提出了基于时间的线性衰减模型能有效融合用户的长、短期兴趣。

- 设计了融合用户画像和用户兴趣探索的原型推荐系统: 论文在总结本人在用户画像、用户兴趣探索工作的基础上设计了动态推荐系统。该系统能够实时反馈用户的最新行为, 并根据用户行为的变化自动探索出用户新的兴趣, 从而不断改善用户在推荐系统中的体验。

关键词： 推荐系统 长尾效应 动态兴趣 用户画像建模 用户兴趣探索

ABSTRACT

Keywords: recommender systems, long tail, dynamic, user profile, user interests exploration

目 录

表格索引

插图索引

算法索引

第一章　绪论

1.1　研究背景与意义

自互联网诞生以来，用户寻找信息的方法经历了几个阶段。早期的用户主要靠直接记住感兴趣网站的网址来寻找内容，直接促使 Yahoo! 提出了分类目录系统，将网站分门别类方便用户查询。但随着信息越来越多，分类目录也只能记录少量的网站，于是产生了搜索引擎。以 Google 为代表的搜索引擎可以让用户通过关键词找到自己需要的信息，但是，搜索引擎需要用户主动的提供显式关键词来寻找信息，因此它不能解决用户的更多的潜在需求，当用户无法精准描述自己的需求时，搜索引擎就无能为力了，于是又催生出推荐系统 [7]。以亚马逊电商官网为代表的推荐系统是一种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提供明确的需求，而是通过分析用户的历史行为来给用户画像建模从而主动给用户推荐出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是两个互补的工具。搜索引擎满足用户显式的需求，而推荐系统能够在用户没有明确目的的时候帮助他们发现潜在的需要。随着物联网和用户终端设备的发展，人们逐渐从信息的匮乏时代走进了信息的过载 (Information overload) 时代。无论是作为信息消费者的普通用户，还是作为信息生产者的提供商面临着数据爆炸时代的挑战。作为用户，如何从充斥着大量噪声的大数据中找到自己感兴趣的信息是一件非常耗时费力的事情，笔者曾有过这样的一种购物体验：在淘宝商城购买一台笔记本电脑，花费了一上午的时间才浏览、比较完所有的 thinkpad 品牌商家店面，如图 1.1。而近年来淘宝的交易额增长规模巨大，2005 年淘宝交易额为 80 亿，2010 年为 4000 亿，而到 2015 年淘宝双十一单日交易额就为 912 亿元，可见未来几年内笔者的这种关键字搜索 + 逐条浏览的购物方式已经不再具有可行性。而作为提供商，如何让自己生产的信息不埋没在大数据洪流中而受到潜在用户的充分关注，这也是其所要解决的一个课题，很多企业已经或者正在开发适合本公司的推荐系统 (Recommender System) 来解决这一矛盾。

推荐系统广泛应用于电子商务领域，通过分析用户的数据，帮助用户找到喜欢和感兴趣的商品，然后推荐给他们。推荐系统的最大优点在于它能收集用户的兴趣信息并根据用户的不同偏好，主动的为用户做出个性化推荐，而且此推荐信息是动态更新的，也就是说随着时间的推移，用户的兴趣在逐渐改变，推荐系统的推荐结果也会随之改变。因此，推荐系统大大的提高了网站的用户体验，方便了用户对资源信息的查询。推荐系统的主要任务就是联系用户和信息，一方面协助用户发现自己潜在感兴趣的信息从而提升用户的满意度，另一方面让信息针对性的展现在只对它有兴趣的用户面前从而提升商品的转化率，于是实现了消费者和生产者的双赢。

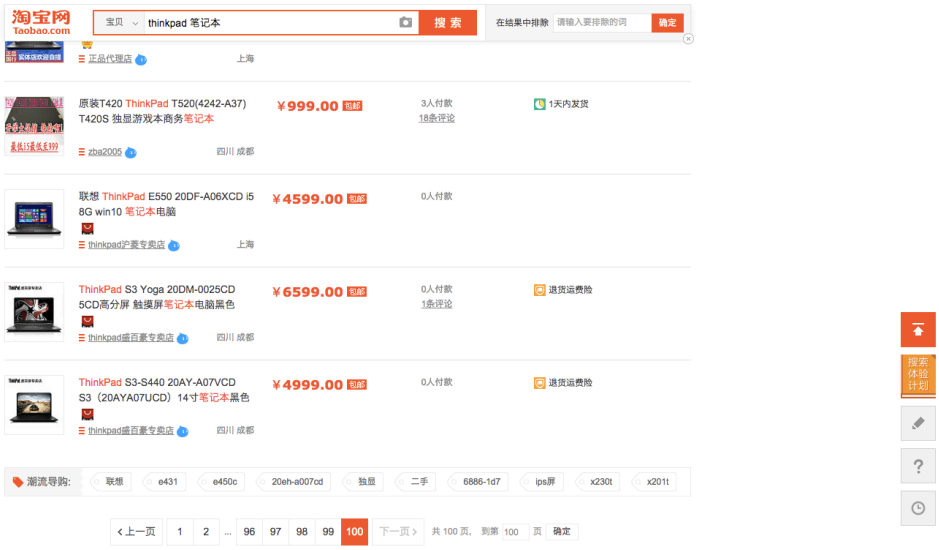

图 **1.1**　淘宝购物页面

1.1.1　推荐系统的产生与发展

随着科学技术与信息传播的迅猛发展，人类社会进入了一个全新的大数据时代，互联网和物联网无处不在的影响着人类生活的方方面面，并颠覆性改变了人们的生活方式，互联网用户既代表了网络信息的消费者，也代表了网络内容的生产者。尤其是随着 Web 2.0 时代的到来，社交化网络媒体的异军突起，互联网中的信息量呈指数级增长，而由于用户的辨别能力有限，使得其在庞大且复杂的互联网信息中找寻有用信息的成本巨大，这就是所谓的“信息过载问题。搜索引擎和推荐系统的出现为用户解决“信息过载提供了非常重要的技术手段。搜索引擎是被动的，用户在搜索互联网中的信息时需要在搜索引擎中输入关键词，搜索引擎根据输入在系统后台进行信息匹配，将与用户查询相关的信息展示给用户。但是当用户无法精确描述自己需求时，搜索引擎就无能为力了。推荐系统是主动的，用户不需要提供明确的需求，而是通过分析用户的历史行为来对用户进行分析，从而主动给用户推荐可能满足他们兴趣和需求的信息。因此搜索引擎和推荐系统是两个互补的技术手段。

推荐系统概念是 1995 年在美国人工智能协会（AAAI）上由 CMU 大学的教授 Robert Armstrong 首先提出并推出了推荐系统的原型系统——Web Watcher。随后推荐系统的研究工作开始慢慢壮大。第一个正式商用的推荐系统是 1996 年 Yahoo 网站推出的个性化入口 MyYahoo。21 新世纪推荐系统的研究与应用随着电子商务的快速发展而风起云涌，各大电子商务网站都开发、部署了推荐系统，有报告称 Amazon 网站中 35% 的营业额来自于自身的推荐系统。2006 年美国的 DVD 租赁公司 Netflix 在网上公开设立了一个推荐算法竞赛并公开了真实网站中的一部分数据，包含用户对电影的评分。Netflix 竞赛有效地推动了学术界和产业界对推荐算法的兴趣，很多有效的算法在此阶段被提了出来。近几年随着社会

化网络的发展，推荐系统在工业界广泛应用并且取得了显著进步。比较著名的推荐系统应用有：淘宝网的电子商务推荐系统、Youtube 的视频推荐系统、网易云音乐推荐系统以及 Facebook 好友推荐系统。

自推荐系统诞生后学术界对其关注的兴趣度也越来越大。从 1999 年开始美国计算机学会每年召开电子商务研讨会以来，发表的与推荐系统相关的论文数以千计。ACM 信息检索专业组在 2001 年开始把推荐系统作为该会议的一个独立研究主题。同年召开的人工智能联合大会也将推荐系统作为一个单独的主题。目前为止数据库、数据挖掘、人工智能、机器学习方面的重要国际会议（如 KDD、AAAI、ICML 等）都有大量与推荐系统相关的研究成果发表。同时第一个以推荐系统命名的国际会议 ACM Recommender Systems Conference 于 2007 年首次举办。在近几年的数据挖掘及知识发现国际会议举办的竞赛中，连续两年的竞赛主题都是推荐系统。2011 年的 KDD CUP 竞赛中，两个竞赛题目分别为音乐评分预测和识别音乐是否被用户评分。2012 年的 KDD CUP 竞赛中，两个竞赛题目分别为腾讯微博中的好友推荐和计算广告中的点击率预测。

1.1.2　推荐系统与电子商务

近几年随着电子商务蓬勃发展，推荐系统在互联网中的优势地位也越来越明显。在国外比较著名的电子商务网站有 Amazon 和 eBay，其中 Amazon 平台中采用的推荐算法是非常成功的。在国内比较典型的电子商务平台网站有淘宝网、网页云音乐、爱奇艺 PPS 等。在这些电子商务平台中，网站提供的商品数量不计其数，网站中的用户规模也非常巨大。据不完全统计天猫商城中的商品数量已经超过了 5000 万。在商品数量如此庞大的电商网站中，如果用户仅仅根据自己的购买意图输入关键字查询只会得到很多用户很难区分的相似结果，也不便用户做出选择。因此推荐系统作为能够根据用户兴趣为用户推荐商品的主要途径，从而为用户在购物的选择中提供建议的需求非常明显。目前比较成功的电子商务网站中，都不同程度地利用推荐系统在用户购物的同时为用户推荐一些商品，从而提高商品的销售额。另一方面，随着以智能手机为代表的物联网推动了移动互联网的发展。在用户在连入移动互联网的过程中，其所处的地理位置信息可以非常准确地被获取，并由此出现了大量的基于用户位置信息的网站。国外比较著名的有 Uber 和 Coupons。国内著名的有滴滴出行和美团网。例如，在美团网这种基于位置服务的网站中，用户可以根据自己的当前位置搜索餐馆、酒店、影院、旅游景点等信息服务。同时，可以对当前位置下的各类信息进行点评，为自己在现实世界中的体验打分，分享自己的经验与感受。当用户使用这类基于位置的网站服务时，同样会遭遇“信息过载问题。推荐系统可以根据用户的位置信息为用户推荐当前位置下用户感兴趣的内容，为用户提供符合其真正需要的内容，提升用户对网站的满意度。

随着社交网络的深入人心，用户在互联网中的行为不再局限于获取信息，更

多的是与网络上的其他用户进行互动。国外著名的社交网络有 Facebook、Twitter 等，国内的社交网络有微信、米聊等。在社交网站中用户不再是单个的个体，而是与网络中的很多人具有了错综复杂的社交关系链。社交网络中最重要的资源就是用户与用户之间的这种联系。社交网络中用户间的关系是多维度的，建立社交关系的因素可能是在现实世界中是亲人、同学、同事、朋友关系，也可能只是网络中的虚拟朋友，比如都是有着共同爱好的会员成员。在社交网络中用户与用户之间的联系紧密度反映了用户之间的信任关系，用户不在是一个个体存在，其在社交网络中的行为或多或少地会受到其他用户关系的影响。因此推荐系统在这类社交网站中的研究与应用应该考虑用户社交的影响。

现如今推荐系统在很多领域得到了广泛的应用，如出租车推荐、商品推荐、美餐推荐、电影推荐和音乐推荐，几乎囊括了人类的吃住行穿四大领域。不同领域的推荐系统具有不同的数据密度，对推荐系统的可扩展性以及推荐结果的相关性、流行性、新鲜性、多样性和新颖性具有不同的需求。

1.2　推荐系统定义

尽管实际需求不尽相同，一个完整的推荐系统通常都包括数据建模、用户建模、推荐引擎和用户接口 4 个部分，数据建模模块负责对拟推荐的物品数据进行准备，将其表示成有可以分析的数据格式，确定要推荐的候选物品集合，并对物品进行 ETL、分类、聚类等预处理，数据建模模块是推荐系统的数据基础，也是最耗系统存储资源的部分。用户建模模块负责分析用户的行为数据并获得用户的潜在喜好。用户的行为信息包括购买、试用、下载、浏览、收藏、停留时间和评论等。推荐引擎模块利用后台的推荐算法，实时地从候选物品集合中筛选出用户感兴趣的物品，排序后以 top N 的形式向用户提供推荐服务。推荐引擎是推荐系统的核心部分，也是最耗系统计算资源和时间的部分。用户接口模块承担展示推荐结果、收集用户反馈等功能。用户接口除了应具有布局合理、界面美观、使用方便等基本要求外，还要方便用户主动提供反馈。主要有两种类型的接口：Web 端和移动端。接下来的章节会详细介绍用户建模和推荐引擎。

1.2.1　用户建模

用户模型反映了用户潜在的兴趣偏好。用户兴趣的反馈可分为显性反馈和隐性反馈。显性反馈包含用户定制和用户评分两种方式，其中，用户定制是指用户对系统所列问题的答复，如年龄、性别、职业等基本人口信息。评分又分为布尔评分和多维评分。例如在 YahooNews 中采用布尔评分：喜欢和不喜欢。多维评分可以更详细地描述对某个产品的喜欢程度，如滴滴出行中用户对司机提供的服务满意程度可评价为 1 5 分。

很多时候用户不能够精准地提供个人偏好或者不愿意显性提供个人兴趣，更不愿意花费时间、精力维护个人的信息。所以隐性反馈往往能够正确地体现用户

的偏好以及偏好的变化趋势。常用的隐性反馈信息有：浏览、点击、停留时长、点击时间、地点、收藏、评论内容、搜索内容和点击顺序等。在协同过滤推荐方法中常常利用用户的隐性反馈作为用户对产品的评分。例如小米手机主题应用中用户试用过的主题记为喜欢，评分为 3；浏览过的主题记为感兴趣，评分为 2，未浏览过的主题评分为 1。一点资讯中用户点击了新闻标题评分为 0.8 分，阅读完全文则评分上升到 1 分；若用户跳过了系统推荐的新闻，则从系统预测评分中减去 0.2 分作为最终评分。

用户的兴趣可分为长期兴趣和短期兴趣。长期兴趣反映用户的真实的兴趣，短期兴趣常与热点话题相结合且经常改动，从最近的用户行为中学习到的短期兴趣模型可快速反映用户兴趣的变化趋势。常用的模型有向量空间模型、隐式马尔科夫模型和基于分类器的模型等。由于用户的兴趣常受物品本身周期性、热点事件、突发事件的影响，随意性很大。所以需要用较短的时间频率来更新用户模型。

1.2.2　推荐引擎

从数学的角度来说，推荐模块的本质过程就是在给定的约束条件下，让用户的利益最大化的过程。对于一个推荐系统来说，将其模型化后，主要涉及到的变量集合有：系统中的用户集合 U，系统中的产品项目集合 I，用户集合和项目集合对应的偏好关系 R，一般为评价集合，映射过程用效用函数 f 表示，即有：

$$f : U \times I \to R \tag{1.1}$$

对于任意目标用户 u，推荐系统的目的就是在项目空间 I 中搜索项目子集 i，使得满足：

$$N_u = max(f(u, N)|u \in U, N \in I) \tag{1.2}$$

推荐引擎的推荐方法可大致分为基于内容的推荐和基于协同过滤的推荐俩种。基于内容的推荐方法的原理是根据用户以往喜欢的物品，选择其他类似的物品作为推荐结果。例如现在有一部新电影与用户曾经看过的某部电影有相同演员或者标签类似，则用户有可能喜欢这部新电影。通常使用用户模型的标签向量空间来描述用户的兴趣偏好，同样对于每个物品的内容进行特征提取，作为物品模型的标签向量空间。然后计算用户模型的标签向量空间和候选物品模型的标签向量空间两者之间的相似度，相似度高的 top N 候选物品就可作为推荐结果推送给目标用户。1992 年提出的协同过滤技术是目前个性化推荐系统中应用最为成功和广泛的技术。国外的商业网站 Amazon 和国内的网易云音乐网站，都采用了协同过滤技术。其本质是基于用户或商品关联分析的技术，即利用用户所在群体的共同喜好来向单个用户进行推荐。协同过滤利用了用户的历史行为数据将用户聚为一类，协同过滤推荐通过计算相似用户，假设其他相似用户喜好的物品，当前用户也会喜欢。基于用户的协同过滤推荐通常包括两个步骤：根据用户

行为数据找到和目标用户兴趣相似的用户集，找到这个集合中用户喜欢的且目标用户没有购买过的物品推荐给目标用户。实际使用中协同过滤技术面临的制约，一是数据稀疏问题，二是冷启动问题。基于用户是协同过滤需要利用用户和用户或者物品与物品之间的关联性进行推荐。最流行的基于邻居关系的协同过滤方法：首先找出与指定用户评价历史数据相近的邻居，根据这些邻居的行为来预测用户行为或者找出与查询物品类似的物品。这样做的前提假设是两个用户在一组物品上有相似的评价，那么他们对其他的物品也倾向于有相似的评价。协同过滤算法的关键是找寻用户的最近邻居。当数据稀疏时，用户购买过的物品很难重叠导致协同推荐的效果就不是很有效，解决办法是二度邻居的行为也可以对当前用户的决策行为构成影响。另外一些解决稀疏问题的方法是添加缺省值，或者采用迭代补全的方法先补充部分数值，在此基础上再进一步补充其他数值。此外还可以利用迁移学习方法弥补数据稀疏。在真实应用中由于商品数量规模巨大，数据稀疏的问题更加突出。数据稀疏性使协同过滤方法的效果受到制约。如何甄别出与数据稀疏程度相适应的推荐算法是非常有价值的研究课题。常用的协同过滤方法有两类：基于内存的方法和基于模型的方法，前者主要是通过用户与物品之间的关系直接导出推荐结果，后者需要找到一个合适的参数化模型间接导出推荐结果。

- 基于内存的协同过滤鉴别出与查询用户相似的用户，然后将这些用户对物品评分的均值作为该用户评分结果的估计值。与此类似，基于物品的协同过滤鉴别出与查询物品类似的物品，然后将这些物品的评分均值作为该物品预测结果的估计值。常用的计算加权平均值的算法有皮尔逊系数、矢量余弦、MSD。

- 基于模型的方法通过适合训练集的参数化模型来预测结果。它包括聚类方法、贝叶斯分类器、回归方法。基于聚类方法的基本思想是将相似的用户聚合成类，有助于解决数据稀疏性和计算复杂性问题。贝叶斯的基本思想是给定用户 A 其他的评分和其他用户评分情况下，计算每个可能评分值的条件概率，然后选择一个最大概率值的评分作为预测值。基于回归方法的基本思想是先利用线性回归模型学习物品之间评分的关系，然后根据这些关系预测用户对物品的评分。

基于内容的推荐方法和基于协同过滤的推荐方法各有其优缺点。现有的系统大部分是一种混合系统，它结合不同算法和模型的优点并克服它们的缺点，从而得到了较好的推荐效果。最近一类成功的基于模型的方法是基于低秩矩阵分解的方法将评价矩阵分解为若干个低秩的子矩阵，这些子矩阵的乘积能对原始矩阵进行某种程度的复原，从而可以评估出缺失值。基于低秩矩阵分解的方法从评分矩阵中抽取一组潜在的因子，并通过这些因子向量描述用户和物品。在电影领域，这些自动识别的因子可能对应一部电影的常见标签。矩阵分解能够对

两类变量进行交互关系的预测。如果将因子分解模型应用到一个新的任务，针对新问题往往需要在原有因子分解基础上推导演化，实现新的模型和学习算法。例如 SVD++、STE、FPMC 和 TimeSVD++ 模型都是针对特定问题在原有因子分解模型基础上做的改进，因此普通的因子分解模型具有较差的泛化能力。在模型优化学习算法方面，虽然对基本矩阵分解模型的学习已经有很多算法，如梯度下降、交替最小二乘法和变分贝叶斯，但是对于更多的业务模型，最多且最常用的模型是结合了基于内容的推荐方法和基于协同过滤的推荐方法的组合模型。

1.3　大数据时代下的推荐系统

虽然推荐系统已经被成功运用在很多大型系统、网站，但是在当前大数据的时代下，推荐系统的面临的场景越来越复杂，推荐系统不仅需要解决传统的数据稀疏、冷启动和动态兴趣问题，还面临由大数据引发的更多、更复杂的实际问题，例如数以亿计的用户数目和海量用户同时访问推荐系统所造成的性能压力，使传统的基于单节点架构的推荐系统不再适用。同时 Web 服务器处理系统请求在大数据集下变得越来越多，Web 服务器响应速度缓慢制约了当前推荐系统为大数据集提供推荐。基于实时模式的推荐在大数据集下也面临着严峻考验，用户难以忍受超过秒级的推荐结果返回时间。传统推荐系统的单一数据库存储技术在大数据集下变得不再适用，急需一种对外提供统一接口、对内采用多种混合模式存储的存储架构来满足大数据集下各种数据文件的存储。并且传统推荐系统在推荐算法上采取的是单机节点的计算方式也不能满足海量用户行为数据的计算需求。大数据本身具有的复杂性、不确定性也给推荐系统带来诸多新的挑战，传统推荐系统的时间效率、空间效率和推荐准确度都遇到严重的瓶颈。

1.3.1　推荐系统的关键技术

分布式文件系统。传统的推荐系统技术主要处理小文件存储和少量数据计算，大多是面向服务器的架构，中心服务器需要收集用户的浏览记录、购买记录、评分记录等大量的交互信息来为单个用户定制个性化推荐。当数据规模过大，数据无法全部载入服务器内存时，就算采用外存置换算法和多线程技术，依然会出现 I/O 上的性能瓶颈，致使任务执行效率过低，产生推荐结果的时间过长。对于面向海量用户和海量数据的推荐系统，基于集中式的中心服务器的推荐系统在时间和空间复杂性上无法满足大数据背景下推荐系统快速变化的需求。大数据推荐系统采用基于集群技术的分布式文件系统管理数据。建立一种高并发、可扩展、能处理海量数据的大数据推荐系统架构是非常关键的，它能为大数据集的处理提供强有力的支持。Hadoop 的分布式文件系统架构是其中的典型。与传统的文件系统不同，数据文件并非存储在本地单一节点上，而是通过网络存储在多台节点上。并且文件的位置索引管理一般都由一台或几台中心节点负责。客户端从集群中读写数据时，首先通过中心节点获取文件的位置，然后与集群中

的节点通信，客户端通过网络从节点读取数据到本地或把数据从本地写入节点。在这个过程中由 HDFS 来管理数据冗余存储、大文件的切分、中间网络通信、数据出错恢复等，客户端根据 HDFS 提供的接口进行调用即可，非常方便。

分布式计算框架。集群上实现分布式计算的框架很多，Spark 作为推荐算法并行化的依托平台，既是一种分布式的计算框架，也是一种新型的分布式计算编程模型，是一种常见的开源计算框架。其基于内存的 MapReduce 算法的核心思想是分而治之，把对大规模数据集的操作，分发给一个主节点管理下的各个分节点共同完成，然后通过整合各个节点的中间结果，得到最终结果。计算框架负责处理并行编程中分布式存储、工作调度、负载均衡、容错均衡、容错处理以及网络通信等复杂问题，把处理过程高度抽象为两个函数：map 和 reduce。map 负责把任务分解成多个任务，reduce 负责把分解后多任务处理的结果汇总起来。

推荐算法并行化。大型企业所需的推荐算法要处理的数据量非常庞大，从 TB 级别到 PB 级甚至更高，腾讯 Peacock 主题模型分析系统需要进行高达十亿文档、百万词汇、百万主题的主题模型训练，仅一个百万词汇乘以百万主题的矩阵，其数据存储量已达 3TB。面对如此庞大的数据，若采用传统串行推荐算法，时间开销太大。当数据量较小时，时间复杂度高的串行算法能有效运作，但数据量极速增加后，这些串行推荐算法的计算性能过低，无法应用于实际的推荐系统中。因此，面向大数据集的推荐系统从设计上就应考虑到算法的分布式并行化技术，使得推荐算法能够在海量的、分布式、异构数据环境下得以高效实现。

1.3.2　推荐系统面临的问题

特征提取问题。推荐系统的推荐对象种类丰富，例如新闻、博客等文本类对象，视频、图片、音乐等多媒体对象以及可以用文本描述的一些实体对象等。如何对这些推荐对象进行特征提取一直是学术界和工业界的热门研究课题。对于文本类对象，可以借助信息检索领域已经成熟的文本特征提取技术来提取特征。对于多媒体对象，由于需要结合多媒体内容分析领域的相关技术来提取特征，而多媒体内容分析技术目前在学术界和工业界还有待完善，因此多媒体对象的特征提取是推荐系统目前面临的一大难题。此外推荐对象特征的区分度对推荐系统的性能有非常重要的影响。目前还缺乏特别有效的提高特征区分度的方法。

数据稀疏问题。现有的大多数推荐算法都是基于用户—物品协同过滤矩阵数据，数据的稀疏性问题主要是指用户—物品评分矩阵的稀疏性，即用户与物品的交互行为太少。一个大型网站可能拥有上亿数量级的用户和物品，用户评分数据总量在面对增长更快的“用户—物品评价矩阵”时，仍然表现出稀疏性，推荐系统研究中的经典数据集 MovieLens 的稀疏度仅 4.5%，Netflix 百万大赛中提供的音乐数据集的稀疏度是 1.2%。这些都是已经处理过的数据集，实际上真实数据集的稀疏度都远远低于 1%。例如，Bibsonomy 的稀疏度是 0.35%，Delicious 的稀疏度是 0.046%，淘宝网数据的稀疏度甚至仅在 0.01% 左右。根据经验，数据

集中用户行为数据越多，推荐算法的精准度越高，性能也越好。若数据集非常稀疏，只包含极少量的用户行为数据，推荐算法的准确度会大打折扣，极容易导致推荐算法的过拟合，影响算法的性能。

冷启动问题。冷启动问题是推荐系统所面临的最大问题之一。冷启动问题总的来说可以分为 3 类：系统冷启动问题、新用户问题和新物品问题。系统冷启动问题指的是由于数据过于稀疏，“用户—物品评分矩阵”的密度太低，导致推荐系统得到的推荐结果准确性极低。新物品问题是由于新的物品缺少用户对该物品的评分，这类物品很难通过推荐系统被推荐给用户，用户难以对这些物品评分，从而形成恶性循环，导致一些新物品始终无法有效推荐。新物品问题对不同的推荐系统影响程度不同：对于用户可以通过多种方式查找物品的网站，新物品问题并没有太大影响，如电影推荐系统等，因为用户可以有多种途径找到电影观看并评分；而对于一些推荐是主要获取物品途径的网站，新物品问题会对推荐系统造成严重影响。通常解决这个问题的途径是激励或者雇佣少量用户对每一个新物品进行评分。新用户问题是目前对现实推荐系统挑战最大的冷启动问题：当一个新的用户使用推荐系统时，他没有对任何项目进行评分，因此系统无法对其进行个性化推荐；即使当新用户开始对少量项目进行评分时，由于评分太少，系统依然无法给出精确的推荐，这甚至会导致用户因为推荐体验不佳而停止使用推荐系统。当前解决新用户问题主要是通过结合基于内容和基于用户特征的方法，掌握用户的统计特征和兴趣特征，在用户只有少量评分甚至没有评分时做出比较准确的推荐。

1.3.3　推荐系统开源项目介绍

工欲善其事，必先利器，关于大数据，有很多令人兴奋的事情，但如何分析、利用它也带来了很多困惑。好在开源观念盛行的今天，有一些在大数据领域领先的免费开源技术可供利用。

- Apache Hadoop：Hadoop 是一个由 Apache 基金会所开发的分布式系统基础架构，是一种用于分布式存储和处理商用硬件上大型数据集的开源框架，可让各企业迅速从海量结构化和非结构化数据中获得洞察力。Hadoop 的框架最核心的设计就是 HDFS 和 MapReduce。HDFS 为海量的数据提供了存储，则 MapReduce 为海量的数据提供了计算。HDFS 有高容错性的特点，并且设计用来部署在低廉的硬件上；而且它提供高吞吐量来访问应用程序的数据，适合那些有着超大数据的应用程序。MapReduce 本身就是用于并行处理大数据集的软件框架，其根源是函数性编程中的 map 和 reduce 函数。它由两个可能包含有许多实例的操作组成。Map 函数接受一组数据并将其转换为一个键/值对列表，输入域中的每个元素对应一个键/值对。

- Hive：Hive 是建立在 Hadoop 上的数据仓库基础构架。它提供了一系列的

工具，可以用来进行数据提取转化加载，这是一种可以存储、查询和分析存储在 Hadoop 中的大规模数据的机制。Hive 定义了简单的类 SQL 查询语言，称为 HQL，它允许熟悉 SQL 的用户查询数据。同时，这个语言也允许熟悉 MapReduce 开发者的开发自定义的 mapper 和 reducer 来处理内建的 mapper 和 reducer 无法完成的复杂的分析工作，十分适合数据仓库的统计分析。

- Spark：Spark 是加州大学伯克利分校所开源的类 Hadoop 的通用并行框架，Spark 拥有 Hadoop 所具有的优点；但不同于 Hadoop 的是 Job 中间输出结果可以保存在内存中，从而不再需要读写 HDFS，因此 Spark 能更好地适用于数据挖掘与机器学习等需要迭代的 MapReduce 的算法。

- Kafka：Kafka 是一种高吞吐量的分布式发布订阅消息系统，它可以处理消费者规模的网站中的所有用户行为流数据。这种用户行为流数据是在现代网络上的许多社会功能的一个关键因素。这些数据通常是由于吞吐量的要求而通过处理日志和日志聚合来解决。对于像 Hadoop 的一样的日志数据和离线分析系统，但又要求实时处理的限制，Kafka 一个可行的解决方案。其目的是通过 Hadoop 的并行加载机制来统一线上和离线的消息处理，也是为了通过集群机来提供实时的消费。

1.4 论文结构

本文的其余正文内容由以下章节组成:

- 第二章首先介绍了推荐系统基本概念和算法模型，包括数据挖掘算法 [6] 和信息提取技术 [5] 的应用，然后详细介绍了用户画像和用户兴趣探索。

- 第三章主要讨论了如何设计一个实际的动态长尾推荐系统，以及动态推荐系统的各个主要模块设计和需要遵守的设计原则。然后根据手机主题业务特点介绍了用户兴趣动态性，最后讨论了推荐系统的评测指标。

- 第四章主要讨论了用户画像建模，包括用户画像的数据来源，用户标签权重计算，以及用户画像建模方式。接下来从不同维度分解用户画像标签属性，最后列举了用户画像在实际生产中的应用场景，包括解决用户冷启动、用户兴趣多样性的问题，并给出了相关的实验结果及分析。

- 第五章主要讨论了如何利用用户兴趣探索跟踪用户动态并挖掘用户小众兴趣，从而提升推荐系统的长尾效应，文中给出了相关的实验结果及分析。

- 第六章是论文的结束语和展望，在对目前工作简要总结的基础上，提出了推荐系统下一步研究的任务和方向。

第二章　推荐系统综述

2.1　引言

当今的时代是信息过载 (Information overload) 的时代 [18]。对于一个用户来讲互联网上充斥着大量对其无用的信息，如何从这些信息里找到用户感兴趣的信息，并把这些信息推送给用户是推荐系统面临的主要问题。推荐系统通过对用户的历史行为进行挖掘，对用户画像进行建模预测用户未来的行为。

推荐系统的研究和很多早期的研究相关，比如认知科学 (cognitive science)[14]，信息检索 (information retrieval) 和预测理论 [16]。随着互联网的兴起，研究人员开始研究如何利用用户对物品行为数据来预测用户的兴趣并给用户做推荐 [17]。推荐系统开始成为一个比较独立的研究问题。到 2006 年为止推荐系统的研究主要集中在基于邻域的协同过滤算法，目前工业界应用最广泛、最知名的算法应该就是亚马逊开发并使用的协同过滤算法 [19]。

近年来很多研究人员意识到推荐的时效性和多样性对于用户的体验度非常重要，而长尾效应对提高商品销售量有非常大的帮助。创建用户兴趣画像则是其中比较有效的解决途径之一，因为度量用户对物品的喜好不仅取决于用户的喜好和物品的属性，也取决于用户所处的环境，或者称做上下文 (Context)，上下文信息有很多类型，其中时间是一种重要的上下文信息，用户在不同的时间可能喜欢不同的物品，物品在不同的时间也有不同的流行度。因此推荐系统应该是一个动态系统，随着时间的变化会给用户不同的推荐结果 [20]。

2.2　推荐系统的算法模块

2.2.1　协同过滤算法

协同过滤的根本原理是，人们可以从和自己有相同品味、习性的人群那里获得高质量的推荐。协同过滤算法主要研究如何聚类具有相似兴趣特征的人群并基于此做出推荐，因为算法本身是基于用户社交群体，因此往往会涉及到大规模的用户行为数据的计算。协同过滤的应用领域也很广：电子商务，金融信贷，搜素引擎，互联网企业，网络社区等需要对用户提供个性化体验的服务商。因为中国现有的人口国情，协同过滤算法往往需要面对亿万级用户和海量的用户-主题交互数据。作为输入数据，一个用户是以一个 N 维度的向量来表示，N 代表所有的主题数量。向量内容可以为正也可为负，分别表示了用户喜欢、讨厌该主题的程度。对于热门主题，给其打分的用户会很多，其分数应该乘以一个因子 u 得到有效的分数，u 代表所有给其打分的用户个数的倒数，大多数用户向量是稀疏的。在协同过滤算法中关键性的一步就是要选择测量的距离，描述集合相似度算法有欧氏距离、闵可夫斯基距离、汉明距离等，这里选择余弦距离公式 (cosine

similiarity)，公式描述如下，其中 $similarity_{uv}$ 代表用户 u 与 v 之间的兴趣相似度，N(u) 表示用户 u 曾经喜欢过的物品集合，N(v) 表示用户 v 曾经喜欢过的物品集合。

$$similarity_{uv} = \frac{|N(u) \cdot N(v)|}{||N(u)|| \cdot ||N(v)||} \tag{2.1}$$

然后利用相似度算法把用户分类成独立的集合，每个用户有且只属于其中的一个集合，对于每个集合，取这个集合最受欢迎的 top K 个主题，作为推荐内容推荐给集合的所有用户。大多数情况下协同过滤算法面都临着一个问题：最坏情况下需要遍历所有的用户和所有的主题，算法计算复杂度为 O(MN)，M 是用户数 N 是主题数，解决方法可以借助一种简单的降维思想加以解决：通过去掉那些非常冷门的主题对 N 做降维，通过去掉那些非常不活跃的用户对 M 做降维，计算维度下降的代价是降低了推荐系统的准确性。

User-Based CF 更多的是挖掘用户之间的社交属性，而 Item-Based CF 更多的是挖掘物品之间的特征属性。根据电子商务业务的特殊性这里选择用 Item-Based CF 算法，原因包括：

- 现有电子商务的矩阵计算一台服务器即可承受；相反，百万级别的用户，计算量需要用一个 spark 集群完成计算，从经济效益的角度讲不可取。
- 电子商务一般来讲周上线不足 100 多款，周更新不到 5%，且可以利用增量计算方法更新 item-item 矩阵；相反，主题用户周增加量为十万数量级，加上用户兴趣、社交、互动的动态变化，增量计算无能为力，导致每次更新 user-user 矩阵的时候计算量很大。
- 如果用 ItemCF，会只推荐与相似领域的主题的东西给用户，这样在有限的推荐列表中就可能包含了一定数量本领域不热门的 item，所以 ItemCF 推荐长尾的能力比较强，代价是推荐多样性不足，但是对整个系统而言，因为不同的用户的主要兴趣点不同，所以系统的 coverage 也会很大。
- ItemCF 的算法还可以为推荐结果做出理性的解释。如一个用户之前购买过魔兽世界的主题包，推荐系统会给其推荐魔兽争霸主题包并附上说明：因为用户曾经买过类似的主题包, 并且评价分数不错。

2.2.2　聚类模型算法

聚类分析（Cluster analysis）是对于统计数据分析的一门技术，和分类算法一个主要的区别就是聚类不需要人工参与打标签，基于聚类和 SlopeOne 预测的协同过滤方法，也可以在一定程度上解决传统协同过滤算法用户评分矩阵稀疏和冷启动问题，在降低用户评分矩阵稀疏性的同时提高目标用户最近邻居的查

询速度。聚类是把相似的对象通过静态分类的方法分成不同的组别或者更多的子集（subset），这样让在同一个子集中的成员对象都有相似的一些属性，聚类结果不仅可以揭示数据间的内在联系与区别，还可以为进一步的数据分析与知识发现提供重要依据。在结构性聚类中关键性的一步就是要选择测量的距离。一个简单的测量就是使用曼哈顿距离，它相当于每个变量的绝对差值之和。该名字的由来起源于在纽约市区测量街道之间的距离就是由人步行的步数来确定的。聚类模块可以是对用户兴趣属性相似度做聚类，也可以对用户社交属性相似度做聚类，或者俩种兼有。

在现实社会中人们的兴趣和选择往往受到身边亲朋好友的影响。在互联网中随着诸如国内的腾讯，国外的 Twitter 等社会网络网站的兴起，如何利用用户的社会属性做推荐是近几年推荐领域比较热门的研究问题。基于社会网络的推荐算法被称为社会化推荐 (Social Recommendation)。近几年在工业界已经有了很多社会化推荐系统。最简单的社会化过滤算法是基于邻域的算法 (Neighborhood-based Method)。给定用户 u, 令 F(u) 为用户 u 的好友集合,N(u) 为用户 u 喜欢的物品集合。那么用户 u 对物品 i 的喜好程度定义为用户 u 的好友中喜欢物品 i 的好友个数，如公式 2.2。

$$P_{vi} = \sum_{v \in F(v) i \in N(u)} 1 \tag{2.2}$$

聚类算法在许多领域受到广泛应用，包括机器学习，数据挖掘，模式识别，图像分析以及生物信息，电子商务推荐使用了 k-means 聚类算法，k-means 算法表示以空间中 k 个点为中心进行聚类，对最靠近他们的对象归类。

input : k
output: k 个集合

1 **while** *true* **do**
2 　选择聚类的个数 k。
3 　任意产生 k 个聚类，然后确定聚类中心，或者直接生成 k 个中心。
4 　对每个点确定其聚类中心点。
5 　再计算其新的聚类中心。
6 　如果新旧聚类中心没有变化，跳出循环。
7 **end**
8 **return** k 个中心点

算法 **2.1:** k means

2.2.3 SlopeOne 算法

由 Daniel Lemire 和 Anna Maclachlan 于 2005 年发表的论文中提出，该算法的特点就是实现简单而高效。举例，如表 2.1所示主题 2 和 1 之间的平均评分差值为 (2+(-1))/2=0.5. 因此，主题 1 的评分平均比主题 2 高 0.5。同样的，主题 3 和 1 之间的平均评分差值为 3。因此，如果我们试图根据小敏对主题 2 的评分来预

测她对主题 1 的评分的时候，我们可以得到 2+0.5 = 2.5。同样，如果我们想要根据她对主题 3 的评分来预测她对主题 1 的评分的话，我们得到 5+3=8。

表 **2.1**　SlopeOne 示例

顾客	主题 1	主题 2	主题 3
小明	5	3	2
小磊	3	4	未知
小敏	未知	2	5

为减少过拟合的发生而实现基于 SlopeOne 的协同过滤算法，该方法运用更简单形式的回归表达式 (f(x)=x+b) 和单一的自由参数，而不是一个主题评分和另一个主题评分间的线性回归 (f(x)=ax+b)。该自由参数只不过就是两个主题评分间的平均差值，在某些实例当中它比线性回归的方法更有效。基于聚类算法和 SlopeOne 预测的协同过滤方法还可以很有效的解决数据稀疏性和推荐系统冷启动问题，实现步骤如下：

- 根据余弦相似度公式计算主题 i 和主题 j 的属性相似性，记为 sims(i，j)。
- 根据现有主题的既得评分，组成前述的评分矩阵，计算两个主题之间的评分相似性，记为 simr(i，j)。
- 将前两步所得结果进行线性组合，组合结果作为最终的综合相似性，记为 sim(i，j)：sim(i，j) = αsims(i，j)+(1-α)simr(i，j)。
- 对 sim(i，j) 按从大到小排序。取相似度最大的前 k_n 个主题作为邻居主题，从而得到目标主题的邻居主题集 I = i_1，i_2，$i_3 \cdots k_n$，在这个邻居主题集的基础上，对目标用户运用加权 SlopeOne 算法进行预测，将预测评分填入空缺的评分矩阵。

2.3　推荐系统用户画像模块

2.3.1　用户画像定义

Alan Cooper（交互设计之父）最早提出了用户画像的概念:Personas are a concrete representation of target users。用户画像是真实用户的虚拟代表，是建立在一系列真实数据之上的目标用户画像。通过用户历史行为去了解用户，根据他们的目标、行为和观点的差异，将他们区分为不同的类型，然后每种类型中抽取出典型特征，赋予名字、照片、一些人口统计学要素、兴趣标签等描述，就形成了一个人物原型 (personas)，图 4.1所示为一个典型的用户画像，标签面积越大代表其权重越高。一些大公司很喜欢用 personas 用做研究，比如阿里，腾讯，微软等，刻画每个用户，是任何一家社交类型的服务都需要面对的问题，不同的公司针对各自业务会有不同的需求，构建用户画像的动机和目标也会存在一定差异。用户

画像定义使用标签来量化用户特性属性，以达到描述用户的目的。用户画像的难点就是数据源，因为要拿到足够多足够全的数据很不容易，所以用户画像的建模需要与业务相结合，与此同时用户画像是动态更新的，因为人是不断变化的。用户画像的核心工作是为用户打标签，目的之一是为了让人能够理解并且方便计算机处理，如可以做分类统计：喜欢红酒的用户有多少？喜欢红酒的人群中，男、女比例是多少？也可以做数据挖掘工作：利用关联规则计算，喜欢红酒的人通常喜欢什么运动品牌？利用聚类算法分析，喜欢红酒的人年龄段分布情况？用户画像包含着的标签为自动化计算提供了一种便捷的方式，使得计算机能够程序化处理与人相关的信息，甚至通过算法、模型能够“理解”人。当计算机具备这样的能力后，无论是搜索引擎、推荐引擎、广告投放等各种应用领域，都能进一步提升精准度，提高用户信息获取的效率。

2.4　用户画像数据来源

电子商务用户画像的信息来源可以有如下几种方式:

- 显式用户行为。显式方法主要是通过获取用户注册信息中的有关的兴趣和偏好或允许用户自己定义和修改用户画像来实现，一般获取的是用户相对静态和稳定的属性，例如：性别、年龄区间、地域、受教育程度、学校、公司等。主题应用商店本身就有比较完整的用户注册引导、用户信息完善任务、认证用户审核等，在收集和清洗用户属性的过程中，需要注意的主要是标签的规范化以及不同来源信息的交叉验证。

- 隐式用户行为。隐式方法则是通过跟踪用户的行为和交互来评估和推测用户画像，一般获取的是用户更加动态和易变化的兴趣特征，首先，用户兴趣会受到环境、热点事件、季节等方面的影响，一旦这些因素发生变化，用户的兴趣容易产生迁移；其次，用户的行为多样且碎片化，不同行为反映出来的兴趣差异较大。

- 第三方应用数据。一些功能性应用如微信、微博提供的第三方免注册登陆 API 接口，可以直接获取第三方应用账号提供的用户基本数据。

- 自然语言处理技术。利用自然语言处理技术提取用户购买评价、评论语句中的关键词，作为用户画像标签的一部分。

在个性化服务的用户画像建模中，最常用的方式是将以上几种或多种方法结合起来，通过显式方式来获取静态用户信息如姓名、性别、职业等；通过隐式方式来获取动态用户信息如用户兴趣、爱好等；通过第三方登陆接口获取用户的分享、动态信息等；通过自然语言处理技术分析用户的当前心态、满意度、消费心情等。

2.4.1 用户画像构建

一个标签通常是人为规定的高度精炼的特征标识，如年龄段标签：25 35 岁，地域标签：北京。标签有两个重要特征：语义化和短文本，人能很方便地理解每个标签含义。这也使得用户画像模型具备实际意义。能够较好的满足业务需求。如，判断用户偏好。同时，每个标签通常只表示一种含义，标签本身无需再做过多文本分析等预处理工作，这为利用机器提取标准化信息提供了便利。人制定标签规则，并能够通过标签快速读出其中的信息，机器方便做标签提取、聚合分析。所以，用户画像和用户标签为我们展示了一种朴素、简洁的描述用户信息的方法。构建用户画像是为了还原用户信息，因此数据来源于所有与用户相关的数据。对于与用户相关数据的分类，一般采用一种封闭性的分类思想。如，世界上分为两种人，一种是懂计算机的人，一种是不懂计算机的人；客户分三类，高价值客户，中价值客户，低价值客户；产品生命周期分为，投入期、成长期、成熟期、衰退期，所有的子分类将构成了类目空间的全部集合。这样的分类方式，有助于后续不断枚举并迭代补充遗漏的信息维度。不必担心架构上对每一层分类没有考虑完整，造成维度遗漏留下扩展性隐患。另外，不同的分类方式根据应用场景，业务需求的不同，也许各有道理，按需划分即可。

本文将用户数据划分为静态信息数据、动态信息数据两大类。静态信息数据是指用户相对稳定的信息，如图所示，主要包括人口属性、商业属性等方面数据。这类信息，自成标签，如果企业有真实信息则无需过多建模预测，更多的是数据清洗工作，因此这方面信息的数据建模不是本篇文章重点。动态信息数据是指用户不断变化的行为信息，广义上讲，一个用户打开网页，点击了一个链接，购买了一个杯子等都属于用户行为。当行为集中到互联网，乃至电商，用户行为就会聚焦很多。用户行为可以被看作用户动态信息的唯一数据来源。用户画像的目标是通过分析用户行为，最终为每个用户打上标签，以及该标签的权重。其中标签表征了用户对该内容有兴趣、偏好、需求等等。权重表征了用户的兴趣、偏好指数，也可能表征用户的需求度，可以简单的理解为可信度，概率。

下面内容将详细介绍如何根据用户行为，构建模型产出标签、权重。一个事件模型包括：时间、地点、人物三个要素。每一次用户行为本质上是一次随机事件，可以详细描述为：什么用户，在什么时间，什么地点，做了什么事。

- 什么时间：时间包括两个重要信息，时间戳 + 时间长度。时间戳，为了标识用户行为的时间点，通常采用精度到秒的时间戳即可。浏览器时间精度，准确度最多也只能到毫秒。时间长度为了标识用户在某一页面的停留时间。

- 什么地点：用户的接触点。对于每个用户接触点。潜在包含了两层信息：网址和内容。网址定位了一个互联网页面地址，或者某个产品的特定页面。可以是 PC 上某电商网站的页面 url，也可以是手机上的微博，微信等应用某个功能页面，某款产品应用的特定画面。内容可以是单品的相关信息：类

别、品牌、描述、属性、网站信息等等。其中网址决定了权重；内容决定了标签。

- 什么事：用户行为类型，对于电商有如下典型行为：浏览、添加购物车、搜索、评论、购买、点击赞、收藏等等。不同的行为类型对于接触点的内容产生的标签信息，具有不同的权重。

综合上述分析，用户画像的数据模型，可以概括为下面的公式：用户标识 + 时间 + 行为类型 + 接触点（网址 + 内容），某用户因为在什么时间、地点、做了什么事。用户标签的权重还可能随时间的增加而衰减，因此定义时间为衰减因子 r，行为类型、网址决定了权重，内容决定了标签，进一步转换为公式，标签权重 = 衰减因子 × 行为权重 × 网址子权重。

2.4.2　用户画像标签维度

一个用户可以从多个方面去刻画，也就是说用户画像可以从多个维度来考虑和构建。作为虚拟电子商品交易平台，电子商务市场的用户在平台上通过某些行为（点击、浏览、购买）生产或获取信息，也通过其它一些行为（如转发、评论、赞）将信息传播出去，信息的传播是通过用户之间的社交关系所进行的，并且在生产、消费、传播信息的过程中对信息的选择和过滤体现了用户在兴趣方面的倾向性。由此，我们可以将用户画像按照图 4.4所示的四个维度进行划分，即属性维度、兴趣维度、社交维度和行为维度。用户属性和用户兴趣是传统用户画像中包含的两个维度。前者刻画用户的静态属性特征，例如用户的身份信息（性别、年龄、受教育程度、学校等），后者则用于刻画用户在信息筛选方面的倾向（例如用户的购买能力、兴趣标签、能力标签等）。社交维度是从社交关系及信息传播的角度来刻画用户的。在社区中用户不在仅仅是一个个体，用户和用户之间的社交关系构成了一张网络，信息在这张网络中高速流动，但是这种流动并不是无差别的，信息的起始点，所经历的关键节点以及这些节点构成的关系圈都是影响信息流动的重要因素。行为维度是一个比较新的研究方向，目的是发现影响用户属性、信息变化的行为因素，分析典型用户群体的行为模式。一方面可以通过行为模式的复用来促进用户在电子商务应用平台的成长；另一方面也有利于平台认识用户，和发现新的或异常的用户行为。

属性维度: 属性维度属于传统用户画像的范畴，即对用户的信息进行标签化。一方面，标签化是对用户信息进行结构化，方便计算机的识别和处理；另一方面，标签本身也具有准确性和非二义性，也有利于人工的整理、分析和统计。用户属性指相对静态和稳定的人口属性，例如：性别、年龄区间、地域、受教育程度、学校、公司等信息的收集和建立主要依靠产品本身的引导、调查、第三方提供等，在此基础上需要进行补充和交叉验证。

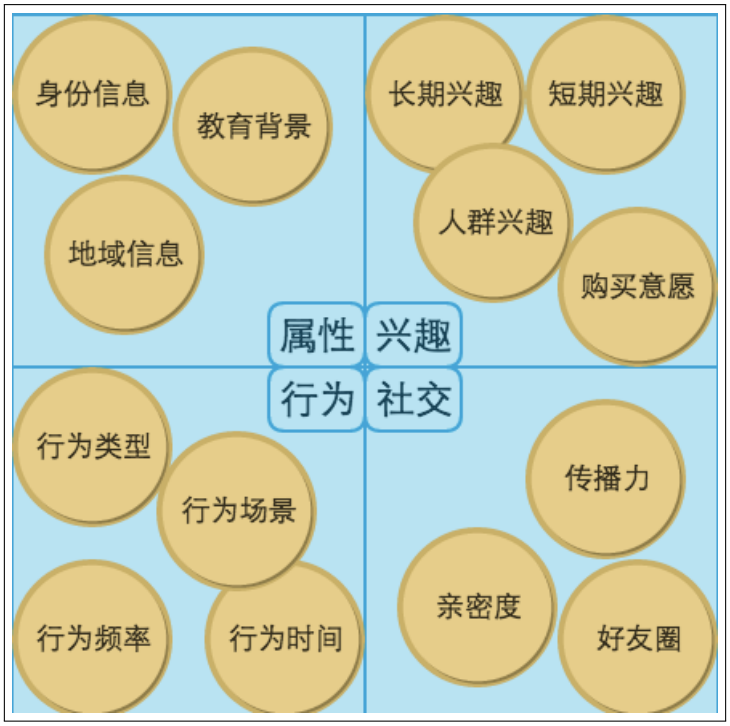

图 **2.1**　用户画像维度划分

- 标签来源：不是所有的词都适合充当用户标签，这些词本身应该具有区分性和非二义性；此外，还需要考虑来源的全面性，除了用户主动提供的兴趣标签外，用户在使用的过程中的行为，构建的用户关系等也能够反应用户的兴趣，因此也要将其考虑在内。

- 权重计算：得到了用户的兴趣标签，还需要针对用户给这些标签进行权重赋值，用来区分不同标签对于该用户的重要程度。

兴趣维度: 由于用户兴趣维度的重要性，因此有一个独立于用户画像模块的兴趣探索模块，下一章节将会详细介绍到。用户兴趣是更加动态和易变化的特征，首先兴趣受到人群、环境、热点事件、行业等方面的影响，一旦这些因素发生变化，用户的兴趣容易产生迁移；其次，用户的行为多样且碎片化，不同行为反映出来的兴趣差异较大，在用户画像建模的过程中，主要考虑如下几个方面：

- 时效性：随着时间的变化，用户的兴趣会发生转移，有些兴趣会贯穿用户使用社交媒体的全过程，而有些兴趣则是受热点时间、环境因素等的影响。

- 长尾性：对于电商领域来讲，那些冷门的用户兴趣的总和可以和那些为数不多的大众化兴趣所占的市场份额相匹配或胜出。

- 兴趣和购买意愿的区分：用户具有某方面的兴趣，只代表了他愿意接受这方面的信息，并不能代表他具有购买相关内容的意愿。例如对于一些只看不买的用户，我们认为其购买意愿很小，因此对其会尽可能多的展示免费主题。

社交维度: 如果将主题应用平台的用户视作节点，用户之间的关系视作节点之间的边，那么这些节点和边将构成一个社交的网络拓扑结构，或称作社交图谱。消费信息就是在这个图谱上进行传播。从社交的维度建立用户画像，需要从不同的角度细致和全面地描述这个消费图谱的特征，反应影响信息传播的各层面上的因素，寻找节点之间的关联度，以及刻画图谱本身的结构特征。其中包括:

- 用户个体对消费信息传播的影响：不同用户在信息传播过程中的重要性不一样，影响大的用户对于信息的传播较影响小的用户更具有促进作用。
- 量化用户关系紧密度：存在社交关联的用户，关系越近的用户之间越容易产生相同的消费行为。
- 寻找相似的用户：消费中非对等的关系本身可以认为是一种认证，用户基于兴趣、消费态度等原因反应到线上的一种关联。那么在消费维度上的相似用户至少能反应他们在某种因素上的一致性。
- 识别关系圈: 从关系图谱的本身的结构出发，从中发掘关联紧密的群体，有助于促销广告的精准投放和主题包的推广。以上关于关系建模的任务可以看作是逐步深入的，从“个体”–>“关联”–>“相似”–>“群体”的逐渐深入。

行为维度: 分析用户的行为，建立行为模式有两个任务：针对典型个体行为进行时序分片，分析用户成长的相关因素；针对典型群体的行为进行统计，为其构建通用的用户画像。

- 典型个体的行为时序分析。所谓典型个体是指某段时间内，成长比较突出的用户。例如从一个新用户从新注册到点击过百、浏览过千需要有一个积累过程，有些用户积累较快，有些较慢，而这些积累较快的用户可以作为典型个体；或者某些用户在某一阶段消费有限，但在某时刻消费激增，无论是消费金额还是数量都变化很大，这种也可以作为典型个体。针对典型个体，需要挖掘与其用户成长相关的行为因素。基本方法是对时间进行分片，获取用户在不同时间片上的行为统计，以及在各个时间分片上的用户成长指标（点击量、购买量、点击转换比等）。在此基础上针对用户行为的统计量的变化，利用关联性分析或回归来分析用户成长与哪些因素有关。
- 典型群体行为模式分析。针对典型个体，从用户的基本信息、人口信息、兴趣维度，可以将相似的典型用户划分为同一的群体，称作典型群体，针对典型群体中的用户按照成长程度进行划分，按不同的成长阶段统计用户行为，即建立了该典型群体的行为模型。例如，对于“年龄在 20 30 岁，女性，付费用户”这样的典型群体，从日点击量、月消费额等维度将其划分到初创、成长、快速提升、成熟等阶段，针对不同成长阶段内的行为组合进行统计，结果构成该群体的行为模式。如图 4.5

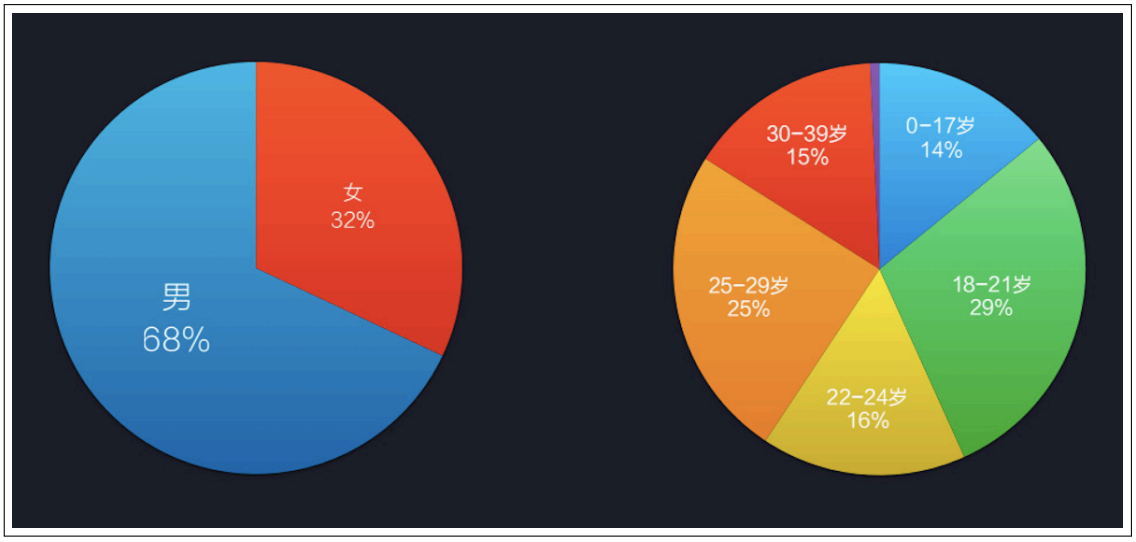

图 **2.2**　电子商务市场用户群体分布

2.4.3　用户画像应用场景

优化电子商务市场供求: 改变了原有的先设计、再销售的传统模式。第三方主题设计师在设计一款新产品前，会先设定好主题类型，然后通过用户画像平台中分析该用户群体的偏好，有针对性的设计产品，从而改变原先新产品高失败率的窘境，增强销售表现。如设计一款智能手表主题，面向 28-35 岁的年轻男性，通过在平台中进行分析，发现属性 =“金属”、风格 =“硬朗”、颜色 =” 深灰色”、价格区间 =“中等”的偏好比重最大，那么就给新产品的设计提供了非常客观有效的决策依据。

提高新人留存率: 工商管理有一个理论叫做，维护一个老用户的成本是获取新用户成本的五分之一甚至更低。所以如果能够把一些已经流失的用户召回来，这时候成本比拉一个新用户低得多，你做的事也会带来更大的价值。鉴于此公司启动了一个项目叫” 用户画像之拉新”，首先利用用户画像得出最近一个月没有登录过的用户数据，然后根据浏览时长分档，这是因为用户需要花自己的时间成本才能留下的最有价值的标签，之后利用用户静态标签，像姓名、职业、年龄、地域分布做进一步的细分，最后针对不同类型的用户提供不同的优惠活动。ABtest 显示，与传统一刀切的推荐相比，基于用户画像的拉新留存率提高了约 50%。

用户消费等级分群: 大至用户终端品牌、机型、操作系统，细至屏幕分辨率、屏幕尺寸，用户画像记录了每一个用户群体的详细终端特征。哪一类人群最容易被这款应用吸引，愿意为这款应用付费? 开发者经常考虑的问题可以从用户画像找到答案。每一个用户群的价格分布、增值业务费用分布以及流量费用，包括用户详细的消费特征，比如付费频率，丰富了推荐系统的数据依据。

用户流失预警: 一般情况用户在消费过程中会经历对如下几个期间: 新鲜期，沉迷期，消退期，离开四个阶段，如何能够延长用户在应用的停留周期是需要解

决的问题之一。用户画像可以辅助推荐系统进行流失用户特征分析，通过决策数算法，分析流失用户特征，建立不同原因流失的用户模型，然后通过这些特征得到当前在应用活跃用户中匹配流失概率高的用户数据。

反作弊: 用户画像会对用户的消费能力、空闲时间、信用评级等维度进行打分；利用反作弊模型通过业务方访问收集数据，供安全部门参考。

2.5　推荐系统用户兴趣探索

现实世界的一切事物都处在变化之中。用户的兴趣、物品的属性都是在不断的变化，一个系统中每天会有大量的新用户新物品加入；时间作为一种重要的上下文信息 (Context)，不同的时间用户也会有不同的兴趣，比如用户在白天和晚上的兴趣可能不同，周末和工作日的兴趣可能不同，不同的季节用户的兴趣也会有所不同。因此，合理的利用时间信息，对推荐的精准度和用户的满意度将会有很大的提升。而传统的推荐系统在设计时并没有主动的考虑到时间因素，推荐系统的动态效应表现在:

- 用户偏好随时间变化 (User bias shifting): 用户可能在某一天只对他喜欢的物品评分，某一天可能只对他不喜欢的物品评分。因此用户某一天的平均分是随时间变化的。
- 物品偏好随时间变化 (Item bias shifting): 物品的受欢迎程度也是随时间变化的。一款主题包在刚上线的时候的因为用户关注度小平均评分会很高，随着时间的推移，越来越多的用户参与到评分中，会使其慢慢接近真实的评分。
- 用户兴趣随时间变化 (User preference shifting): 用户在不同的时候可能有不同的兴趣，比如小孩都喜欢动漫主题包，但当他长大了可能喜欢汽车主题包。
- 季节效应: 用户行为会受季节效应的影响。主题推荐中主要的季节效应有暑期的效应，以及一些纪念日的效应 (比如国庆纪念日前后，抗日题材的主题包会受到较多的关注)。

为保持推荐系统的动态特性，工业界一般用数据追加的方式进行增量计算。推荐系统利用 hadoop 集群可以在 2 个小时内完成最近 24 小时数据的增量计算并将结果追加到现有的计算结果中，耗费的这 2 个小时可以用更少的时间进行增量计算并做数据追加。

2.5.1　用户行为数据存储

电子商务用户行为数据的特点包括: 用户基数庞大。以电子商务网站淘宝网为例，注册用户往往以千万计，活跃用户达百万计；用户规模增长快。每个用户

的行为数量较小。即使是活跃用户，每天最多也只能产生上百条行为记录，每年不超过十万条；用户行为的计算较为复杂。计算用户的两次登录间隔天数、反复购买的商品、累积在线时间，这些都是针对用户行为的计算，通常具有一定的复杂性；用户行为数据格式不规整，字段丢失率较高。根据用户行为数据的这些特点采用基于 Hadop 分布式的架构。

2.5.2　用户行为数据预处理

Hive 是建立在 Hadoop 上的数据仓库基础架构。它提供了一系列的工具，用来进行数据提取、转换、加载，是一种可以存储、查询和分析存储在 Hadoop 中的大规模数据机制。可以把 Hadoop 下结构化数据文件映射为一张成 Hive 中的表，并提供类 sql 查询功能，除了不支持更新、索引和事务，sql 其它功能都支持。可以将 sql 语句转换为 MapReduce 任务进行运行，作为 sql 到 MapReduce 的映射器。提供 shell、JDBC/ODBC、Thrift 等接口。优点是成本低可以通过类 sql 语句快速实现简单的 MapReduce 统计。作为一个数据仓库，Hive 的数据管理按照使用层次可以从元数据存储、数据存储和数据交换三个方面介绍。数据预处理是数据挖掘过程中一个重要步骤，当原始数据存在不一致、重复、含噪声、维度高等问题时，更需要进行数据的预处理，以提高数据挖掘对象的质量，最终达到提高数据挖掘所获模式知识质量的目的。

随着电子商务市场交易规模的逐步增大，积累下来的业务数据和用户行为数据越来越多，这些用户数据往往是电子商务平台最宝贵的财富。目前在电子商务推荐系统中大量地应用到了机器学习和数据挖掘技术，例如个性化推荐、搜索排序、用户画像建模等等，为企业创造了巨大的价值。数据预处理主要工作是：

- 从原始数据，如文本、图像或者应用数据中清洗出特征数据和标注数据
- 对清洗出的特征和标注数据进行处理，例如样本采样，样本调权，异常点去除，特征归一化处理等过程。最终生成的数据主要是供模型直接使用。

根据不同业务数据的预处理方式也不同，一般来讲原始服务器日志数据脏数据的形成原因包括：缩写词不统一，数据输入错误，不同的惯用语，重复记录，丢失值，不同的计量单位，过时的编码等。相应的，数据预处理内容包括数据清理、数据集成、数据变换、数据归约、数据离散化。数据清理包括格式标准化、异常数据清除、错误纠正、重复数据的清除。对于电子商务用户数据来讲，引起空缺值的原因主要是用户设备异常造成的，有些时候是因为与其他已有数据不一致而被删除或数据的改变没有进行日志记载。根据数据空缺情况的不同有不同的处理方式：

- 忽略元组。当一个记录中有多个属性值空缺、特别是关键信息丢失时, 已不能反映真实情况, 它的效果非常差。

- 去掉属性。缺失严重时, 已无挖掘意义。
- 人工填写空缺值。但是工作量大且可行性低。
- 默认值。比如使用 unknown 或-∞。
- 使用属性的平均值填充空缺值。
- 预测最可能的值填充空缺值。使用贝叶斯公式或判定树这样的基于推断的方法。

2.5.3 用户行为建模

实体域。当我们想基于用户行为分析来建立用户兴趣模型时，我们必须把用户行为和兴趣主题限定在一个实体域上。个性化推荐落实在具体的推荐中都是在某个实体域的推荐。对于手机主题应用市场来说，实体域包括所有的主题，背景图片，铃声，闹铃等。用户行为。浏览，点击，下载，试用，购买，评论等都可是用户行为。本文所指的用户行为都是指用户在某实体域上的行为。比如用户在手机铃声产生的行为。用户兴趣。用户的兴趣维度，同样是限定在某实体域的兴趣，通常以标签 + 权重的形式来表示。比如，对于手机主题，用户兴趣向量可以是「动漫，0.6」,「体育，0.1」,「情感，0.7」等分类标签。值得一提的是，用户兴趣只是从用户行为中抽象出来的兴趣维度，并无统一标准。而兴趣维度的粒度也不固定，如「体育」,「电影」等一级分类，而体育下有「篮球」,「足球」等二级分类，篮球下有「NBA」,「CBA」,「火箭队」等三级分类。我们选取什么粒度的兴趣空间取决于具体业务模型。

实际应用中，在社交网络用户的行为一般是主动进行的，例如，自行定义或选择标签，浏览页面，使用站内产品或第三方 APP，发表博文或对其他博文内容的点赞或收藏，关注其他用户并将其关注的对象划分到自行设置的各用户组内等。而上述这些社交网络用户的行为能够在一定程度上反映出用户的兴趣。因此，社交网络中，可以根据用户的这些网络行为来进行用户的兴趣挖掘。该阶段是用户行为数据进行建模，以抽象出用户的标签，这个阶段注重的应是大概率事件，通过数学算法模型尽可能地排除用户的偶然行为。基于用户标签的兴趣挖掘方法。具体地，可以根据标签的具体内容，将标签归类到相应的兴趣类别后，再根据用户的自定义标签及其所属的兴趣类别，分析出用户的兴趣。

2.6 本章小结

本章首先从学术研究和商业应用两个角度介绍了推荐系统常用的算法，包括协同过滤、聚类算法等若干种不同的推荐算法。然后重点介绍推荐系统用户画像，包括用户画像的构建和用户画像构建标签维度的划分和用户画像诸多应用

场景。最后介绍了用户兴趣探索，包括用户行为数据的存储，用户行为数据的预处理和用户行为建模。

第三章　动态推荐系统设计

3.1　前言

推荐系统的形式化定义如下：设 C 是所有用户的集合，S 是所有可以推荐给用户的主题的集合。实际上，C 和 S 集合的规模通常很大，如上百万的顾客以及上万款手机主题。设效用函数 u() 可以计算主题 s 对用户 c 的推荐度（如提供商的可靠性（vendor reliability）和产品的可得性（product availability），即 $u = C \times S \rightarrow R$，R 是一定范围内的全序的非负实数，推荐要研究的问题就是找到推荐度 R 最大的那些主题 S*，如式 3.1

$$\forall c \in C, S^* = argmax_{s \in S} u(c, s) \tag{3.1}$$

除了推荐系统自身如冷启动、数据的稀疏性等问题，还有一个关注点就是推荐系统的时间效应问题。比较常见的时间效应问题主要反映在用户兴趣的变化、物品流行度的变化以及手机主题的季节效应, 这些问题都可以利用用户画像解决。本章节主要介绍如何搭建一个具有长尾性、实时性的动态推荐系统。动态推荐形态由以下几个模块组成：用户画像模型，用户兴趣探索模块，推荐主题模块，推荐算法模块。通用的推荐系统模型流程是：首先，推荐系统把用户画像模型中兴趣需求信息和推荐主题模型中的特征信息匹配，然后使用相应的推荐算法进行计算筛选，找到用户可能感兴趣的推荐主题，最后推荐给用户。

用户画像模块对应着用户长期兴趣，用户兴趣探索对应着用户短期动态兴趣。短期兴趣的特点是临时、易变；长期兴趣的特点是长久、稳定；用户的短期兴趣可能会转化为长期兴趣，所以需要在推荐时综合考虑长期兴趣和短期兴趣。考虑到推荐系统的时间效应问题，将输入数据集归结为一个四元组，即用户，物品，行为，时间，通过研究用户的历史行为来预测用户将来的行为。需要解决以下俩个问题：动态评分预测、时效性的影响。首先，动态评分预测问题。数据集可以选用比较直观的显性反馈数据集，即 (用户，物品，评分，时间)，研究是这样一个问题，给定用户 u，物品 i，时间 t，预测用户 u 在时间 t 对物品 i 的评分 r。对于该类问题，与时间无关的评分预测问题算法主要有以下几种：用户兴趣的变化，如年龄增长，从儿童长成青少年壮年；生活状态的变化，由以前的小学生到大学生；社会事件的影响如俩会等。此外还有季节效应问题，一些在春季很流行的，在夏季节未必就很流行。该问题的解决有待进一步思考。对于时效性的影响，每个在线系统都是一个动态系统，但它们有不同的演化速率。比如说，新闻更新很快，但音乐，电影的系统演化的却比较慢。

本章首先介绍用户画像和兴趣探索模块，其中兴趣探索模块需要根据业务的演化速率来调整迭代深度。然后介绍推荐主题模块，之后介绍推荐算法模块和指标体系，最后做总结。

3.2　用户画像和兴趣探索模块

目前基于用户画像的推荐，主要用在基于内容的推荐，从最近的 RecSys 大会（ACM Recommender Systems）上来看，不少公司和研究者也在尝试基于用户画像做 Context-Aware 的推荐（情境感知，又称上下文感知）。利用用户的画像，结合时间、天气等上下文信息，给用户做一些更加精准化的推荐是一个不错的方向。一个好的推荐系统要给用户提供个性化的、高效的、动态准确的推荐，那么推荐系统应能够获取反映用户多方面的、动态变化的兴趣偏好，推荐系统有必要为用户建立一个用户兴趣探索模型，该模型能获取、表示、存储和修改用户兴趣偏好，能进行推理，对用户进行分类和识别，帮助系统更好地理解用户特征和类别。推荐系统根据用户画像进行推荐，所以用户画像对推荐系统的质量有至关重要的影响。建立用户画像模型之前需要考虑问题有：模型的输入数据有哪些，如何获取模型的输入数据；如何考虑用户的兴趣及需求的变化；建模的对象是谁以及如何建模；模型的输出是什么。用户画像模型的输入数据主构成包括：

- 用户属性，分为社会属性和自然属性，包括用户最基本的如用户的姓名、年龄、职业、收入、学历等信息。用户注册时的对自然属性和社会属性进行初始建模。

- 用户手工输入的信息：是用户主动输出给系统的信息，包括用户在搜索引擎中打出的关键词，用户评论中发布的感兴趣的主题、频道。还有一类重要的信息就是用户反馈的信息，包括用户自己对推荐结果的满意程度；用户标注的浏览页面的感兴趣、不感兴趣或感兴趣的程度等。

- 用户的浏览行为和浏览内容：用户浏览的行为和内容体现了用户的兴趣和需求，它们包括浏览次数、频率、停留时间等，浏览页面时的操作（收藏、保存、复制等）、浏览时用户表情的变化等。服务器端保存的日志记录了用户的浏览行为和内容。

3.2.1　用户行为的权重排序

用户显式行为数据记录了用户在平台上不同的环节的各种行为，这些行为一方面用于候选集触发算法中的离线计算（主要是点击、浏览），另外一方面，这些行为代表的用户兴趣强弱不同，因此在训练重排序模型时针对不同的行为设定了不同的权重值，以更细地刻画用户的行为强弱程度。此外，用户的购买、试用等行为还作为重排序模型的交叉验证特征值，用于模型的离线训练和在线预测。负反馈数据反映了当前的结果可能在某些方面不能满足用户的需求，因此在后续的候选集触发过程中需要考虑对特定的因素进行过滤或者降权，提高用户体验；同时在重排序的模型训练中，A/B 测试结果作为负例参与模型训练。用户画像是刻画用户属性的元数据，其中有些是直接获取的基础数据，有些是经过挖

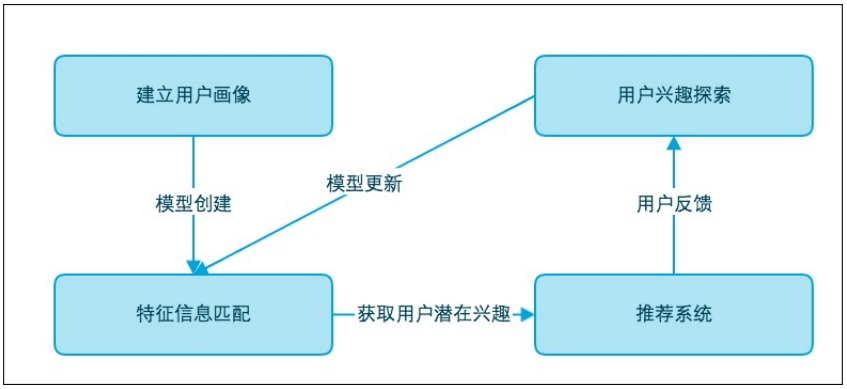

图 **3.1**　用户画像的使用

掘的二次数据，这些属性一方面可以用于候选集触发过程中对标签进行加权或降权，另外一方面可以作为重排序模型中的用户维度特征。通过对数据的挖掘可以提取出一些关键词，然后使用这些关键词给主题打标签，用于主题的个性化展示。

3.2.2　用户行为的获取方式

对于用户的一些人口属性信息采用了显式方式直接获取，对于用户一些明显的兴趣偏好采用了隐式获取，对于用户潜在的兴趣偏好则通过关联技术启发式获取。显式获取用户兴趣偏好的方法是简单而直接的做法，能准确地反映用户的需求，同时所得的信息比较具体、全面、客观，结果比较可靠。缺点就是数量稀少，原因用户不太愿意花时间来向商家表达自己的喜好，并且这种方法灵活性差，答案存在异质性，当用户兴趣改变时需要用户手动更改系统中用户兴趣。同时该方法对用户不是很人性化。隐式获取法是指系统通过记录用户行为数据，通过权重排序获取用户的兴趣偏好，用户的很多动作都能暗示用户的喜好，包括查询、浏览页面和文章、标记书签、反馈信息、滑屏等。隐式的跟踪可以在建立用户画像基本数据的同时不打扰用户的正常消费活动。这种方法的缺点就是跟踪的结果未必能正确反映用户的兴趣偏好。上述获取兴趣偏好的方法有时受用户教育背景、职业和习惯等因素的限制，用户有时意识不到自己的兴趣主题，因此能为用户提供启发式信息，如领域术语抽取和相似度物品聚类，可以实现领域知识的复用，为用户间的协同提供支持，提高用户兴趣获取质量。用户的兴趣和需求会随着时间和情景发生变化，用户画像模块要考虑到用户长期兴趣偏好和短期兴趣偏好，还要考虑兴趣的变化。因此结合用户画像和用户兴趣探索的动态建模将是团队下一个研究方向，如图 3.1所示。

当前开发阶段的用户画像更新采用了时间窗方法和遗忘机制来反映用户兴趣的变化。更新机制是天级别，所以无法及时跟踪用户兴趣的变化并给出推荐结果，just-in-time 型有更强学习效率和动态变化适应能力的建模也是未来的重要研究方向。

3.3　推荐主题模块

推荐主题采用了单用户建模和群组建模，单用户建模针对个体用户进行建模，群组建模是针对一类用户进行建模。动态推荐系统对 Top 20% 的活跃用户采用了单用户建模，这样做的的目的有俩个：1，活跃用户的用户行为数据相比较其他用户更多，不存在数据稀疏性问题；2，活跃用户往往消费金额多，单独对其建模有助于提供高质量的推荐服务，有利于提升手机主题转化率。于此同时对剩余用户采用群组建模，这是因为只有对不活跃用户聚类后才可以得到足够多用来建模的数据。

和用户画像一样，对手机主题进行描述之前要考虑：提取手机主题的什么特征，如何提取，提取的特征用于什么目的，主题的特征描述和用户画像之间有关联。提取到的每个主题特征标签对推荐结果会有什么影响。主题的特征描述向量空间能否自动更新。推荐主题的向量空间中的主题特征和用户画像中的兴趣标签进行推荐计算，获得推荐主题的推荐权重，所以推荐主题的向量空间与用户画像密切相关，所有要用同样的标签集来表达用户的兴趣偏好和推荐主题。推荐系统推荐主题包括众多的领域，比如体育、动漫、科技还有诸如音乐、电影等。不同的主题，特征也不相同，动态推荐系统主要采用了基于内容的方法和基于分类的方法两大类方法。基于内容的方法是从主题本身抽取相关信息来表示主题，具体方法是用加权关键词矢量，该方法通过对标注主题的标签进行统计分析得出的特征向量，通过计算每个标签的信息增量（Information gain），即计算每个特征在主题中出现前后的信息熵之差。在完成主题特征提取后，还需要计算每个特征的权值，权值大的对推荐结果的影响就大。基于分类的方法是把推荐主题放入不同类别中，这样可以把同类主题推荐给对该类主题感兴趣的用户了。文本分类的方法使用了 k 最近邻方法（KNN）。主题的类型一部分由第三方设计师自己预先定义，同时利用聚类算法自动产生一些类型。实验表明聚类的精度非常依赖于主题的数量，而且由自动聚类产生的类型可能对用户来说是毫无意义的，因此可以有选择的进行手工选定的类型来分类主题，在没有对应的候选类型或需要进一步划分某类型时，才使用聚类产生的类型。推荐系统推荐给用户的主题首先不能与用户购买过的主题重复，其次也不能与用户刚刚看过的主题不是太形似或者太不相关，这就是所谓的模型过拟合问题（可扩展性问题）。出现这一问题的本质上来自数据的不完备性，解决的主要的方法是引入随机性，使算法收敛到全局最优或者逼近全局最优。针对这一问题考察了被推荐的主题的相关性和冗余性，要同时保证推荐的多样性，又不能与用户看过的主题重复或毫不相关。推荐系统中出现新的主题时，推荐系统尤其是协同过滤系统中，新主题出现后必须等待一段时间才会有用户浏览和评价，而在此之前推荐系统是无法对此主题进行推荐，这就是推荐系统研究的另一个难点和重点——冷启动问题。

3.4　推荐算法模块

现有的推荐算法类型很多，但是各有各的局限，因此动态推荐系统采用了组合推荐算法，即融合了协同过滤推荐，基于内容推荐和基于关联规则推荐组合推荐算法，他们的主要优缺点对比如所示。

表 **3.1**　推荐系统主要算法比较

推荐方法	优点	缺点
基于内容推荐	推荐结果直观，容易解释；不需要领域知识	稀疏问题；新用户问题；复杂属性不好处理；要有足够数据构造分类器
协同过滤推荐	新异兴趣发现、不需要领域知识；随着时间推移性能提高；推荐个性化、自动化程度高；能处理复杂的非结构化对象	稀疏问题；可扩展性问题；新用户问题；质量取决于历史数据集；系统开始时推荐质量差；
基于规则推荐	能发现新兴趣点；不要领域知识	规则抽取难、耗时；产品名同义性问题；个性化程度低；
基于效用推荐	无冷开始和稀疏问题；对用户偏好变化敏感；能考虑非产品特性	用户必须输入效用函数；推荐是静态的，灵活性差；属性重叠问题；
基于知识推荐	能把用户需求映射到产品上；能考虑非产品属性	知识难获得；推荐是静态的

3.4.1　推荐算法

基于内容推荐。基于内容的推荐（Content-based Recommendation）是信息过滤技术的延续与发展，它是建立在对手机主题的标签信息上作出推荐的，而不需要依据用户对手机主题的评价意见，需要用机器学习的方法从关于内容的特征描述的事例中得到用户的兴趣资料。手机主题是通过相关的特征的属性来定义，系统基于用户评价对象的特征，学习用户的兴趣，考察用户资料与待预测手机主题的相匹配程度。用户的资料模型取决于所用学习方法，采用了综合决策树、神经网络和基于向量的组合方法。基于内容的用户资料是需要有用户的历史数据，用户资料模型可能随着用户的偏好改变而发生变化。基于内容推荐方法的优点是：不需要其它用户的数据，没有冷开始问题和稀疏问题。能为具有特殊兴趣爱好的用户进行推荐。能推荐新的或不是很流行的手机主题，没有产品问题。通过列出推荐手机主题的内容特征，可以解释为什么推荐那些手机主题。

协同过滤推荐。利用用户的历史喜好信息计算用户之间的距离，然后利用目标用户的最近邻居用户对评价的加权评价值来预测目标用户对特定手机主题的喜好程度，系统从而根据这一喜好程度来对目标用户进行推荐。协同过滤是基于这样的假设：为一用户找到他真正感兴趣的内容的好方法是首先找到与此用户有相似兴趣的其他用户，然后将他们感兴趣的内容推荐给此用户。协同过滤正是

把这一思想运用到手机推荐系统中来，基于其他用户对某一类手机主题的评价来向目标用户进行推荐。基于协同过滤的推荐系统可以说是从用户的角度来进行相应推荐的，而且是自动的，即用户获得的推荐是系统从购买模式或浏览行为等隐式获得的，不需要用户努力地找到适合自己兴趣的推荐信息，如填写一些调查表格等。和基于内容的过滤方法相比，协同过滤具有如下的优点：能够过滤难以进行机器自动内容分析的信息。共享其他人的经验，避免了内容分析的不完全和不精确，并且能够基于一些复杂的，难以表述的概念（如信息质量、个人品味）进行过滤。有推荐新信息的能力。可以发现内容上完全不相似的信息，用户对推荐信息的内容事先是预料不到的。这也是协同过滤和基于内容的过滤一个较大的差别，基于内容的过滤推荐很多都是用户本来就熟悉的内容，而协同过滤可以发现用户潜在的但自己尚未发现的兴趣偏好。能够有效的使用其他相似用户的反馈信息，较少用户的反馈量，加快个性化学习的速度。

基于关联规则推荐。基于关联规则的推荐（Association Rule-based Recommendation）是以关联规则为基础。关联规则挖掘可以发现不同手机主题在销售过程中的相关性。管理规则就是在一个交易数据库中统计购买了手机主题集 X 的交易中有多大比例的交易同时购买了手机主题集 Y，其直观的意义就是用户在购手机主题买某些手机主题的时候有多大倾向去购买另外一些手机主题。算法的第一步关联规则的发现最为关键且最耗时，是算法的瓶颈，所有采用离线进行。其次，手机主题名称的同义性问题也是关联规则的一个难点。

组合推荐。由于各种推荐方法都有优缺点，手机主题推荐采用了组合推荐方式。研究和应用最多的是基于内容的推荐和协同过滤推荐的组合。最简单的做法就是分别用基于内容的方法和协同过滤推荐方法去产生一个推荐预测结果，然后用某方法组合其结果。组合推荐一个最重要原则就是通过组合后要能避免或弥补各自推荐技术的弱点。在组合方式上使用了如下几种组合思路：加权（Weight）：加权多种推荐技术结果。变换（Switch）：根据问题背景和实际情况或要求决定变换采用不同的推荐技术。混合（Mixed）：同时采用多种推荐技术给出多种推荐结果为用户提供参考。特征组合（Feature combination）：组合来自不同推荐数据源的特征被另一种推荐算法所采用。层叠（Cascade）：先用一种推荐技术产生一种粗糙的推荐结果，第二种推荐技术在此推荐结果的基础上进一步作出更精确的推荐。特征扩充（Feature augmentation）：一种技术产生附加的特征信息嵌入到另一种推荐技术的特征输入中。

3.5　动态推荐系统底层架构

3.5.1　基于 Spark 的实时计算

随着电子商务的高速发展和普及应用，个性化推荐的推荐系统已成为一个重要研究领域。个性化推荐算法是推荐系统中最核心的技术，在很大程度上决定

了电子商务推荐系统性能的优劣，决定着是否能够推荐用户真正感兴趣的信息，而面对用户的不断提升的需求，推荐系统不仅需要正确的推荐，还要实时地根据用户的行为进行分析并推荐最新的结果。实时推荐系统的任务就是为每个用户，不断地、精准地推送个性化的服务，甚至到达让用户体会到推荐系统比他们更了解自己的感觉。

MapReduce 为大数据挖掘提供了有力的支持，但是复杂的挖掘算法往往需要多个 MapReduce 作业才能完成，多个作业之间存在着冗余的磁盘读写开销和多次资源申请过程，使得基于 MapReduce 的算法实现存在严重的性能问题。大处理处理后起之秀 Spark 得益于其在迭代计算和内存计算上的优势，可以自动调度复杂的计算任务，避免中间结果的磁盘读写和资源申请过程，因此 Spark 能更好地适用于推荐系统迭代的模型计算。表 3.2为 MR 和 spark 的横向比较。相对

表 **3.2**　MR 和 spark 对比

过程	MapReduce	Spark
collect	在内存中构造了一块数据结构用于 map 输出的缓冲	没有在内存中构造一块数据结构用于 map 输出的缓冲，而是直接把输出写到磁盘文件
sort	map 输出的数据有排序	map 输出的数据没有排序
merge	对磁盘上的多个 spill 文件最后进行合并成一个输出文件	在 map 端没有 merge 过程，在输出时直接是对应一个 reduce 的数据写到一个文件中，这些文件同时存在并发写，最后不需要合并成一个
copy 框架	jetty	netty 或者直接 socket 流
对于本节点上的文件	仍然是通过网络框架拖取数据	不通过网络框架，对于在本节点上的 map 输出文件，采用本地读取的方式
copy 过来的数据存放位置	先放在内存，内存放不下时写到磁盘	一种方式全部放在内存；另一种方式先放在内存
merge sort	最后会对磁盘文件和内存中的数据进行合并排序	对于采用另一种方式时也会有合并排序的过程

于 MapReduce，Spark 在以下方面优化了作业的执行时间和资源使用。DAG 编程模型。通过 Spark 的 DAG 编程模型可以把七个 MapReduce 简化为一个 Spark 作业。Spark 会把该作业自动切分为若干个 Stage，每个 Stage 包含多个可并行执行的 Tasks。Stage 之间的数据通过 Shuffle 传递。最终只需要读取和写入 HDFS 一次。减少了中间的 HDFS 的读写。Spark 作业启动后会申请所需的 Executor 资源，所有 Stage 的 Tasks 以线程的方式运行，共用 Executors，相对于 MapReduce 方式，Spark 申请资源的次数减少了近 90%。Spark 引入了 RDD（Resilient Distributed Dataset）模型，中间数据都以 RDD 的形式存储，而 RDD 分布存储于 slave 节点

的内存中，这就减少了计算过程中读写磁盘的次数。RDD 还提供了 Cache 机制，使得分布式机器能共享只读数据。

推荐系统是基于协同过滤算法的实时推荐系统以及 ALS（交替最小二乘法）的并在 Spark Streaming 框架实现的。协同过滤推荐就是基于用户喜好信息，训练一个推荐模型，然后根据实时的用户喜好的信息进行预测，进行推荐。对于一个 users products rating 的评分数据集,ALS 会建立一个 user*product 的 m*n 的矩阵（其中，m 为 users 的数量，n 为 products 的数量）。这个矩阵的每一行代表一个用户 (u1,u2,⋯,u9)、每一列代表一个产品 (v1,v2,⋯,v9)。用户的打分在 1-9 之间。但是在这个数据集中，并不是每个用户都对每个产品进行过评分，所以这个矩阵往往是稀疏的，用户 i 对产品 j 的评分往往是空的，首先将这个稀疏矩阵通过一定的规律填满，这样就可以从矩阵中得到任意一个 user 对任意一个 product 的评分，具体步骤：假设 m*n 的评分矩阵 R，可以被近似分解成 $U*V^T$，U 为 m*d 的用户特征向量矩阵，V 为 n*d 的产品特征向量矩阵，d 为 user/product 的特征值的数量，对于电影类型的手机主题，可以从 d 个角度进行评价，如主角，铃声，背景，特效 4 个角度来评价，那么 d 就等于 4。矩阵 V 由 n 个 product*d 个特征值组成。对于矩阵 U，假设对于任意的用户 A，该用户对一款手机主题的综合评分和主题的特征值存在一定的线性关系，综合评分 =(a1*d1+a2*d2+a3*d3+a4*d4)，其中 a_i 为用户 A 的特征值，d_i 为之前所说的主题的特征值。ALS 算法认为 m*n 的评分矩阵 R 可以被近似分解成 $U*V^T$, 得到目标函数：

$$L(U,V)=\sum_{i,j\in R}(a_{ij}-u_i^T v_j)^2 \tag{3.2}$$

其中 a 表示评分数据集中用户 i 对产品 j 的真实评分，另外一部分表示用户 i 的特征向量和产品 j 的特征向量，但是这里之前问题还是存在，就是用户和产品的特征向量都是未知的，这个式子存在两个未知变量解决的办法是交替的最小二乘法, 为了防止过度拟合，需要加上正则化参数。固定 V 对 U 求导得到公式：

$$U_t=R_t V_{ut}(V_{ut}^T V_{ut}+\lambda n_{ut} I)^{-1}, i\in[1,m] \tag{3.3}$$

其中 R_t 表示用户 i 评过的手机主题的评分向量，V_{ut} 表示用户 i 评过的手机主题的特征向量组成的特征矩阵。n_{ut} 表示用户 i 评过的手机主题数量。同理，固定 U，可以得到求解 V_jd 的公式：

$$V_j=R_j^T U_{mj}(U_{mj}^T U_{mj}+\lambda n_{mj} I)^{-1} \tag{3.4}$$

R_j 表示评过手机主题 j 的用户向量，U_{mj} 表示评过手机主题 j 的用户特征向量组成的矩阵，m_{mj} 表示评过电影 j 的用户的数量。

首先用一个小于 1 的随机数初始化 V，根据式 3.3求 U，此时就可以得到初始的 UV 矩阵了，根据计算得到的 U 和式 3.4重新计算并覆盖 V，反复进行以上两步的计算，直到差平方和小于一个预设的数，或者迭代次数满足要求则停止。

3.6　量化评估推荐系统

推荐系统使用的评测方法包括离线实验，用户调研和在线实验。离线实验预先收集的用户选择和项目评分数据集，通过模拟用户的行为进而评估算法的效率，其目的在于对推荐系统执行用户调研或者在线评估之前过滤掉性能较差的算法。用户调研是为一组用户提供测试任务集并记录观察其行为，收集量化测量数据。可以当用户与系统进行交互时观察他们的行为，并且可以收集很多量化的测量值，如提交反馈之前用户花费的时间。

在线实验可以是推荐系统的评估主要方式。这种类型的实验通常是在离线实验和用户调研之后完成的，并且系统已经准备好在生产环境中使用。然而使用这种实验方法时也应该考虑到这样的风险：较差的推荐质量或者设计可能会打消实际用户再次使用系统的积极性。因此，系统在使用此策略之前进过了仔细评估。推荐系统的评估考虑了各种不同的方面，包括功能性的和非功能性的。功能性方面是指推荐系统所用算法的性能精度。其准确性测量包括对推荐系统预测评分或者 top-N 项目推荐列表的准确性度量。要测量离线实验的准确性，先选择一个相关数据集，数据集分为两部分：80% 作为训练集，20% 作为评分测试集。先从训练集中学习然后对测试集的评分进行预测。预测评分和实际评分的差异形成了准确性测量的依据。然后利用平均绝对误差（MAE）测量测试数据集全部预测评分和实际评分之间的误差。另一个功能性评价指标是覆盖率。它测量了推荐系统产生推荐项目的占比。目录覆盖率测量的是推荐系统做出的推荐项目占全部可用项目个数的比值。可能会存在推荐系统潜在可以推荐的项目。预测准确性并不总能够确保用户的满意程度。用户感性指标也是考虑的因素。

评测指标包括统计性指标：准确率 (Precision)、召回率 (Recall)、F 值 (F-Measure) 等，也包括用户感性指标：准确度、覆盖度、新颖度、惊喜度、信任度、透明度等。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。

3.6.1　统计性指标

用来评价结果的质量的统计性指标包括准确率、召回率和 F 值。其中精度是检索出相关文档数与检索出的文档总数的比率，衡量的是检索系统的查准率；召回率是指检索出的相关文档数和文档库中所有的相关文档数的比率，衡量的是检索系统的查全率。正确率、召回率和 F 值是在鱼龙混杂的环境中，选出目标的重要评价指标，其定义为：正确率 = 提取出的正确信息条数 / 提取出的信息条数，两者取值在 0 和 1 之间，数值越接近 1，正确率就越高。召回率 = 提取出的正确信息条数 / 样本中的信息条数，两者取值在 0 和 1 之间，数值越接近 1，召回率就越高。F 值综合了正确率和召回率的结果，见公式 5.1。当 F 值较高时则

能说明试验方法比较有效。

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3.5}$$

3.6.2　用户感性指标

通过不定期推送用户问卷，开发组可以用最小成本获得真实用户对推荐系统的新颖性、惊喜度和信任度评价。

推荐系统的新颖性。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。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。评测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。

推荐系统的惊喜度。惊喜度（serendipity）是最近这几年推荐系统领域最热门的话题，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。

推荐系统的信任度。信任度只能通过问卷调查的方式，询问用户是否信任推荐系统的推荐结果。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，因此如果推荐的主题是好友购买过的，那么他们对推荐结果就会相对比较信任。

3.6.3　其他系统性指标

推荐系统的覆盖度。覆盖度描述了推荐系统对物品长尾的发掘能力，一般通过所有推荐物品占总物品的比例和所有物品被推荐的概率分布来计算，比例越大、概率分布越均匀则覆盖率越大。

推荐系统的多样性。多样性能显著影响用户的体验。用户的兴趣是广泛的，用户可能既喜欢看《猫和老鼠》动漫的主题包，也喜欢看成龙电影的主题包。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，多样性描述了推荐列表中物品两两之间的不相似性，因此多样性和相似性是对应的。

推荐系统的实时性。有些主题包具有很强的时效性，比如圣诞节、情人节主题包，所以需要在物品还具有时效性时就将它们推荐给用户。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。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。

推荐系统的健壮性，健壮性是指推荐系统对数据异常的可控性，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。

推荐系统还是看目的是如何的，从用户角度讲是为了更好的理解用户，减少用户查找内容的时间和次数，从产品本身角度讲，是增加单位面积单位时间内的点击数或者说内容有效。从业务角度的衡量：衡量点击和打开率，这说明用户是否对内容感兴趣。衡量通过推荐系统替代用户主动搜索或者主动浏览的次数，可以通过横向与使用其他产品对比较，比如使用推荐系统提供内容的用户搜索次数和点击浏览目录次数明显下降。衡量推荐系统的满意度口碑，刨除因为页面位置效果等因素，衡量推荐系统一个重要的就是满意度的口碑问题，这个可以通过单个用户是否有重复使用的行为，曲线是否是一直上升的来衡量，如果一直有新用户访问，但一直没有老用户重复使用，说明用户满意度有问题。

3.7　总结

推荐系统经过了相当长的时间的发展，同时一些重点和难点问题得到了研究者的关注，相信是未来研究的热点问题。（1）用户兴趣偏好获取方法和推荐对象的特征提取方法的研究目前的推荐系统中实际上较少使用了用户和推荐对象的特征，即使使用很广泛的协同推荐使用的是用户的评分。主要是用户兴趣偏好的获取方法和推荐对象特征提取方法不是很适用，需要引入更精确适用的用户和对象特征。（2）推荐系统的安全性研究进行协同推荐时需要掌握用户的兴趣偏好等用户信息，但用户担心个人数据得不到有效保护而不愿暴露个人信息，这是协同推荐长期存在的一个问题。既能得到用户信息而提高推荐系统性能，又能有效保护用户信息将是未来推荐系统的一个研究方向。同时一些不法的用户为了提高或降低某些对象的推荐概率，恶意捏造用户评分数据而达到目的，这也是推荐系统存在的一个安全问题，被称为推荐攻击。检测并能预防推荐攻击也将是未来一个研究方向。（3）基于复杂网络理论及图方法的推荐系统研究复杂网络理论和图方法同协同推荐存在契合点，在文献中网络视频推荐问题转化为热量散播平衡态网络上的谱图分割问题，通过设计长尾发现的推荐策略引导用户发现潜在的感兴趣的网络视频。利用复杂网络理论和图方法进行推荐也是推荐系统研究的一个方向。（4）推荐的多维度研究目前的推荐研究都是基于用户-对象二维空间进行研究的，但是用户选择某个对象以及对对象的评分在不同的情况下会有所不同，也就是推荐使用的特征维度会有所不同，研究推荐的多维度也

是未来的一个研究方向。（5）稀疏性和冷启动研究稀疏性和冷启动问题是困扰推荐系统很长时间了，包括经典协同过滤算法和新出现的基于网络结构的推荐算法都存在该问题。有很多研究者对这一问题进行研究并提出解决办法，但该问题依然存在，还需要对其进行研究。（6）推荐系统性能评价指标的研究用户对算法准确度的敏感度、算法对不同领域的普适性、广义的质量评价方法等都是未来推荐系统性能评价要进行研究的目标。

第四章　用户画像建模

Alan Cooper（交互设计之父）最早提出了用户画像（persona）的概念:“Personas are a concrete representation of target users”。Persona 是真实用户的虚拟代表，是建立在一系列真实数据 (Marketing data，Usability data) 之上的目标用户画像。通过用户历史行为去了解用户，根据他们的目标、行为和观点的差异，将他们区分为不同的类型，然后每种类型中抽取出典型特征，赋予名字、照片、一些人口统计学要素、兴趣标签等描述，就形成了一个人物原型 (personas)，图 4.1所示为一个典型的用户画像，标签面积越大代表其权重越高。一些大公司很喜欢用 personas 做用研究，比如阿里，腾讯，微软等，刻画每个用户，是任何一家社交类型的服务都需要面对的问题，不同的公司针对各自业务会有不同的需求，构建用户画像的动机和目标也会存在一定差异。从手机主题应用商城的角度来讲，构建用户画像的目的包括:

- 完善及扩充用户信息。用户画像的首要动机就是了解用户，这样才能够提供更优质的服务。但是在实际中用户的信息提供得不尽完整，有些是因为平台的引导机制造成的，有时候又是用户不愿意或懒得提供，而且对于用户自行输入的内容又很难进行规范化此外，一些隐性或变化频繁的信息也需要通过用户的行为挖掘出来。

- 打造健康的主题设计生态圈。在掌握用户信息的基础上，平台就可以对自身的状况进行分析，从相对宏观的基础上把握主题市场的生态环境，挖掘设计作品的最大价值，帮助设计师提高收入，如图 4.2所示。例如通过对用户信息的聚类，能够对用户进行人群的划分，掌握不同人群的活跃程度、行为及兴趣偏好，热门主题的传播方式和流行引爆点等。

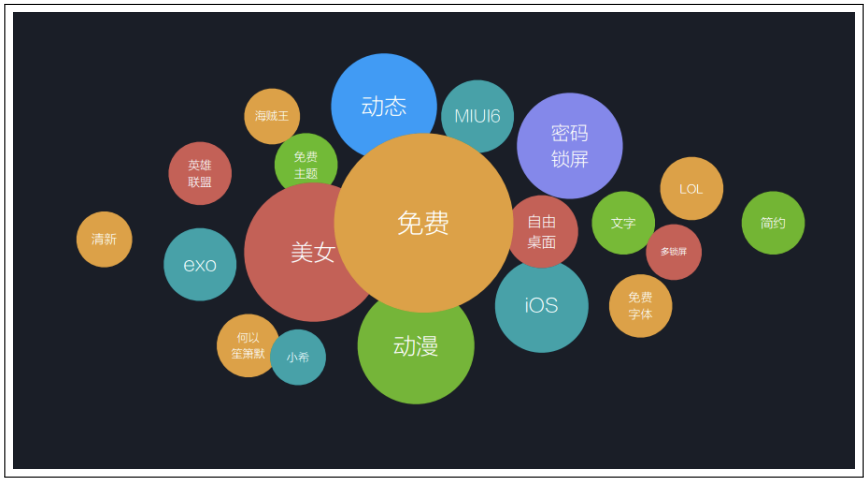

图 **4.1**　用户画像标签化

2015年Q1热销主题TOP10

序号	主题名称	销售金额（元）
1	iOS pro（好评返全款+超级自由桌面）	38万+
2	Forever love（自由桌面）	18万+
3	性感不是罪-琳	15万+
4	梵星Plus 动态星轨锁屏 密码锁屏 v5v6	14万+
5	美iOS(好评返现+强大锁屏+自由桌面）	14万+
6	I watch【至今最帅最酷的锁屏】	13万+
7	我们的爱（荧光闪耀）	9万+
8	ios8+win8(好评返现+双锁屏+自由桌面）	7万+
9	会动LOL英雄	7万+
10	【v6】喰種时代(iOS解锁+音乐界面+自由桌面)	6万+

图 **4.2**　2015 年 Q1 热销主题排行榜

- 支撑主题推荐系统的精准推荐。精准推荐的前提是对用户的清晰认知。以简单代金券发放为例，手机主题应用市场的历史数据呈现出两大类四种不同的消费习惯。代金券敏感型：发代金券才用、发代金券用的更多；代金券不敏感型：发不发都用，发代金券也不用。在推荐系统的用户画像系统中，上述四种群体会被分别冠以屌丝、普通、中产、土豪的标签。针对四类用户的运营策略也会全然不同，最直接的就是代金券的刺激频率以及刺激金额，而对“代金券”免疫的土豪群体，则更多地需要在优化服务上做文章。在实际场景中，影响用户对手机主题包的使用黏度的因素要远比代金券复杂得多，在这种情况下，利用用户画像可以对用户的“贴身跟踪”就能及时发现薄弱环节，因此从用户打开应用商店到退出使用，其间的每一步情况都被快的记录在案：哪一天退出的，哪一步退出的，退出之后“跳转”到什么软件等等。据此，用户画像也实现了用户另外一个纬度的归类，分清哪部分是忠实用户，哪部分可能是潜在的忠实用户，哪些则是已经流失的；更进一步来看流失的原因：因为代金券没有了流失？主题包质量不好流失？这些都是下一步精准推荐的依据。其实手机主题市场中的各项业务都与用户画像有着直接与间接的关系，无论是基于兴趣的推荐提升用户价值，精准的广告投放提升商业价值，还是针对特定用户群体的内容运营，用户画像都是其必不可少的基础支撑。直接地，用户画像可以用于兴趣匹配、关系匹配的推荐和投放；间接地，可以基于用户画像中相似的兴趣、关系及行为模式去推动用户兴趣和设计师的无缝对接。

- 主题市场安全领域的应用。随着手机主题市场的发展，商家会通过各种活动形式的补贴来获取用户、培养用户的消费习惯，但同时也催生一些通过

刷排行榜、刷红包的用户，这些行为距离欺诈只有一步之遥，但他们的存在严重破坏了市场的稳定，侵占了活动的资源。其中一个有效的解决方案就是利用用户画像沉淀方法设置促销活动门槛，即通过记录用户的注册时间、历史登陆次数、常用 IP 地址等，最大程度上隔离掉僵尸账号，保证市场的稳定发展。

用户画像的目的是将用户信息标签化，本文介绍针对主题应用商店本身的特点介绍用户画像的构建，该用户画像主要还是从电子商务的角度出发，完善用户信息和发掘用户兴趣，区分兴趣和购买意愿，并形式化、结构化表达出来。数据的来源也主要是主题平台本身，并没有采用更多的第三方数据。

4.1　用户画像的数据来源

手机主题用户画像的信息来源可以有如下几种方式:

- 用户注册信息和一些评论数据。当一个新用户注册时，系统会引导用户填写一些人口基本信息，包括电话号码，性别，职业等。这些信息一般是较为可信的。当用户发生消费行为时，可能会产生购买评价和对其他用户的互动信息，这些信息是不能直接使用的，需要采用自然语言处理技术转换为较短的文本标签。

- 用户行为数据，包括试用，购买，浏览，点击等。不同行为代表的权重也不尽相同，试用和购买的权重高一些，浏览和点击权重低一些。

- 第三方应用数据。当用户注册时可用选择利用微信、微博提供的第三方免注册登陆 API 接口，因此这些第三方接口也可以提供一些用户信息。

在个性化服务的用户画像建模中，推荐系统会将以上几种方法结合起来，通过显式方式来获取静态用户信息如姓名、性别、职业等；通过隐式方式来获取动态用户信息如用户兴趣、爱好等；通过第三方登陆接口获取用户的分享、动态信息等；通过自然语言处理技术分析用户的当前心态、满意度、消费心情。

4.2　标签权重计算

推荐本质上是一种个性化排序，因此在收集到一个用户可能存在的标签后，还需要给标签赋一定的权重，用来区分不同标签对于该用户的重要程度。一个标签对于特定用户的权重值可以大致表示为：标签权重 = (行为类型 + 时空上下文 + 长尾因子) × 时间衰减因子。举例，用户小磊昨天购买了一款 win8 风格的主题包，计算公式如表 4.1所示。

其中，用户行为类型包括浏览、添加购物车、搜索、评论、购买、点击赞、收藏等，不同的行为类型也具有不同的权重，我们定义购买权重计为 5，而浏览

表 **4.1**　标签权重计算公式

标签	win8 风格，比较大众化，长尾因子记为 1
时间	昨天，衰减因子为 0.95。
行为	购买行为，记为权重 5
上下文	用户通过关键字搜索进入，最近几天有多次浏览行为，记为权重 2+2
标签权重	(5+1+4)*0.95=9.5

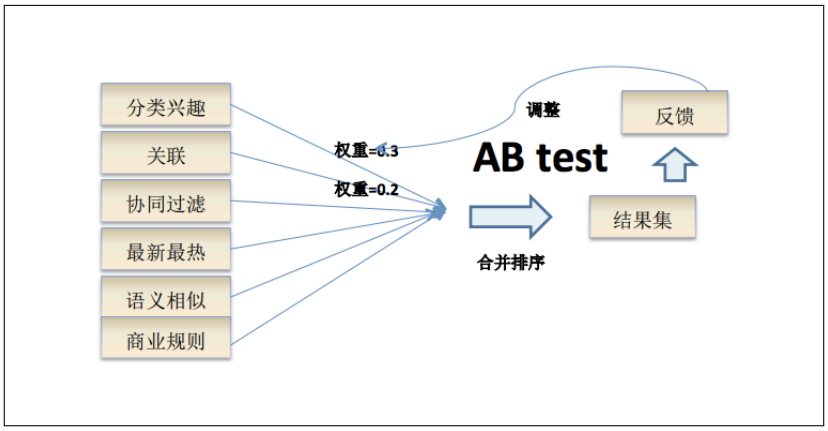

图 **4.3**　abtest 调整标签权重

仅仅为 1。空间上下文是指用户跳转入口方式，我们定义通过搜索入口权重高于排行榜入口。时间上下文是指用户之前是否接触过此类标签，接触频率等。长尾因子是指，如果标签本身是一个非常常见的词，那么它用于刻画用户的兴趣的区分性是比较差的，相反如果是一个长尾词，则区分性较强。出于这样的考虑，越是长尾词，标签的权重值会越高。标签的权重也随着时间的流逝而变化，用户的兴趣会发生转移，时间越久远，标签的权重应该相应的下降，距离当前时间越近的兴趣标签应该得到适当突出。出于这样的考虑，一般会在标签权重值上叠加一个时间衰减函数并体现不同的时效性。此外，针对用户的兴趣，还会设定一个较小的时间窗口来获取用户的短期兴趣，短期兴趣更新周期会较长期兴趣更短，兴趣更集中，但是能够比较及时地反应用户兴趣的变化。实际生产中标签权重计算需要人工参与调整，流程如图 4.3

4.3　用户画像建模方式

根据用户在建模过程中的参与程度，用户兴趣建模采用了户手工定制建模加自动用户建模。用户手工定制建模是指用户画像由用户自己手工输入或选择的用户建模方法，如用户手工输入感兴趣信息的关键词列表，或者是选择感兴趣的栏目等。在手机主题市场早期，用户手工定制建模是用户建模的主要方法。用户手工定制建模方法实现简单、效果也不错，但它存在以下几个问题: 完全依赖于用户，容易降低用户使用系统的积极性。即使用户乐意手工输入用户画像，

用户也难以全面，准确的罗列自己感兴趣的栏目或关键词，导致用户标签的质量有好有坏。当用户兴趣发生变化时，用户必须重新输入用户画像，这给用户带来了额外的负担。自动用户建模是指根据用户的浏览内容和浏览行为自动构建用户画像，自动用户建模由于无需用户主动提供信息不会显示干扰用户，有利于提高个性化服务系统的亲和度，因此，自动建模是用户画像主要的建模方式。自动用户画像建模如algorithm 4.1所示。

input : 结构化用户注册信息和用户浏览行为
output: 初始用户兴趣模型

1 从用户购买行为和反馈表单中提取兴趣标签及其对应的权重
2 生成显式兴趣标签表，如” 动漫”：”0.8”,” 汽车”：”0.4”,” 美少女”：”0.9”...
3 根据用户浏览行为获取用户兴趣标签获得隐式权重
4 生成隐式兴趣标签表，如” 免费”：”0.8”,” 特价”：”0.4”,” 热门”：”0.9”...
5 合并显式兴趣向量和隐式兴趣向量到当前用户画像
6 如有新数据，返回第一步，否则跳出循环。

算法 **4.1:** 自动用户画像建模算法

4.4　用户画像的维度分析

一个用户可以从多个方面去刻画，也就是说用户画像可以从多个维度来考虑和构建。作为虚拟电子商品交易平台，手机主题市场的用户在平台上通过某些行为（点击、浏览、购买）生产或获取信息，也通过其它一些行为（如转发、评论、赞）将信息传播出去，信息的传播是通过用户之间的社交关系所进行的，并且在生产、消费、传播信息的过程中对信息的选择和过滤体现了用户在兴趣方面的倾向性。由此，我们将用户画像按照图 4.4所示的四个维度进行划分，即属性维度、兴趣维度、社交维度和行为维度。用户属性和用户兴趣是传统用户画像中包含的两个维度。前者刻画用户的静态属性特征，例如用户的身份信息（性别、年龄、受教育程度、学校等），后者则用于刻画用户在信息筛选方面的倾向（例如用户的购买能力、兴趣标签、能力标签等）。社交维度是从社交关系及信息传播的角度来刻画用户的。在社区中用户不在仅仅是一个个体，用户和用户之间的社交关系构成了一张网络，信息在这张网络中高速流动，但是这种流动并不是无差别的，信息的起始点，所经历的关键节点以及这些节点构成的关系圈都是影响信息流动的重要因素。行为维度是一个比较新的研究方向，目的是发现影响用户属性、信息变化的行为因素，分析典型用户群体的行为模式。一方面可以通过行为模式的复用来促进用户在手机主题应用平台的成长；另一方面也有利于平台认识用户，和发现新的或异常的用户行为。

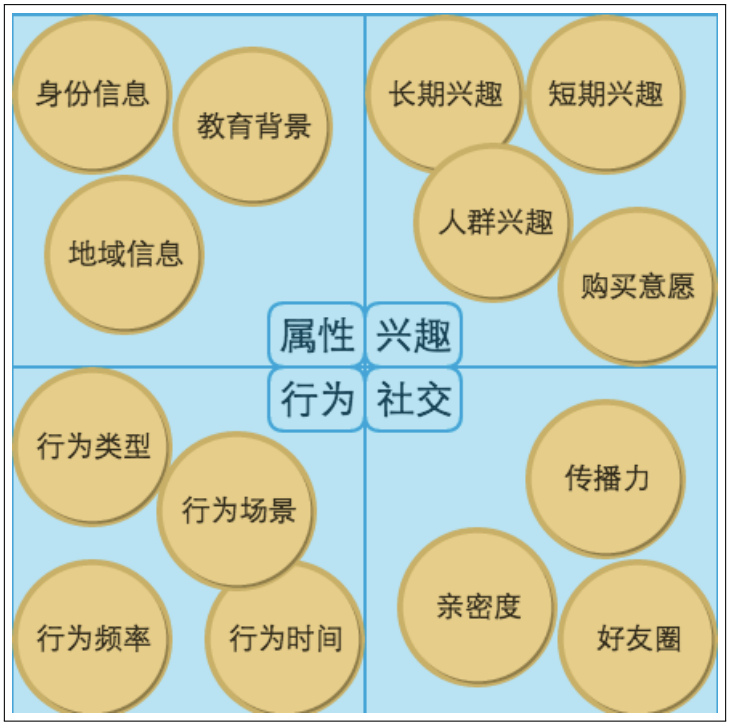

图 **4.4**　用户画像维度划分

4.4.1　属性维度

属性维度属于传统用户画像的范畴，即对用户的信息进行标签化。一方面，标签化是对用户信息进行结构化，方便计算机的识别和处理；另一方面，标签本身也具有准确性和非二义性，也有利于人工的整理、分析和统计。用户属性指相对静态和稳定的人口属性，包括性别、年龄区间、地域、受教育程度、学校、公司等信息的收集和建立主要依靠产品本身的引导、调查、第三方提供等，在此基础上需要进行补充和交叉验证。

- 标签来源：不是所有的词都适合充当用户标签，这些词本身应该具有区分性和非二义性；此外，还需要考虑来源的全面性，除了用户主动提供的兴趣标签外，用户在使用的过程中的行为，构建的用户关系等也能够反应用户的兴趣，因此也要将其考虑在内。

- 权重计算：得到了用户的兴趣标签，还需要针对用户给这些标签进行权重赋值，用来区分不同标签对于该用户的重要程度。

4.4.2　兴趣维度

由于用户兴趣维度的重要性，因此有一个独立于用户画像模块的兴趣探索模块。用户兴趣是更加动态和易变化的特征，首先兴趣受到人群、环境、热点事件、行业等方面的影响，一旦这些因素发生变化，用户的兴趣容易产生迁移；其

次，用户的行为多样且碎片化，不同行为反映出来的兴趣差异较大，在用户画像建模的过程中，主要考虑如下几个方面：

- 时效性：随着时间的变化，用户的兴趣会发生转移，有些兴趣会贯穿用户使用社交媒体的全过程，而有些兴趣则是受热点时间、环境因素等的影响。
- 长尾性：对于电商领域来讲，那些冷门的用户兴趣的总和可以和那些为数不多的大众化兴趣所占的市场份额相匹配或胜出。
- 兴趣和购买意愿的区分：用户具有某方面的兴趣，只代表了他愿意接受这方面的信息，并不能代表他具有购买相关内容的意愿。例如对于一些只看不买的用户，我们认为其购买意愿很小，因此对其会尽可能多的展示免费主题。

4.4.3 社交维度

如果将主题应用平台的用户视作节点，用户之间的关系视作节点之间的边，那么这些节点和边将构成一个社交的网络拓扑结构，或称作社交图谱。消费信息就是在这个图谱上进行传播。从社交的维度建立用户画像，需要从不同的角度细致和全面地描述这个消费图谱的特征，反应影响信息传播的各层面上的因素，寻找节点之间的关联度，以及刻画图谱本身的结构特征。其中包括：

- 用户个体对消费信息传播的影响：不同用户在信息传播过程中的重要性不一样，影响大的用户对于信息的传播较影响小的用户更具有促进作用。
- 量化用户关系紧密度：存在社交关联的用户，关系越近的用户之间越容易产生相同的消费行为。
- 寻找相似的用户：消费中非对等的关系本身可以认为是一种认证，用户基于兴趣、消费态度等原因反应到线上的一种关联。那么在消费维度上的相似用户至少能反应他们在某种因素上的一致性。
- 识别关系圈：从关系图谱的本身的结构出发，从中发掘关联紧密的群体，有助于促销广告的精准投放和主题包的推广。以上关于关系建模的任务可以看作是逐步深入的，从“个体”–>“关联”–>“相似”–>“群体”的逐渐深入。

4.4.4 行为维度

建立行为模式有两个任务：针对典型个体行为进行时序分片，分析用户成长的相关因素；针对典型群体的行为进行统计，为其构建通用的用户画像。

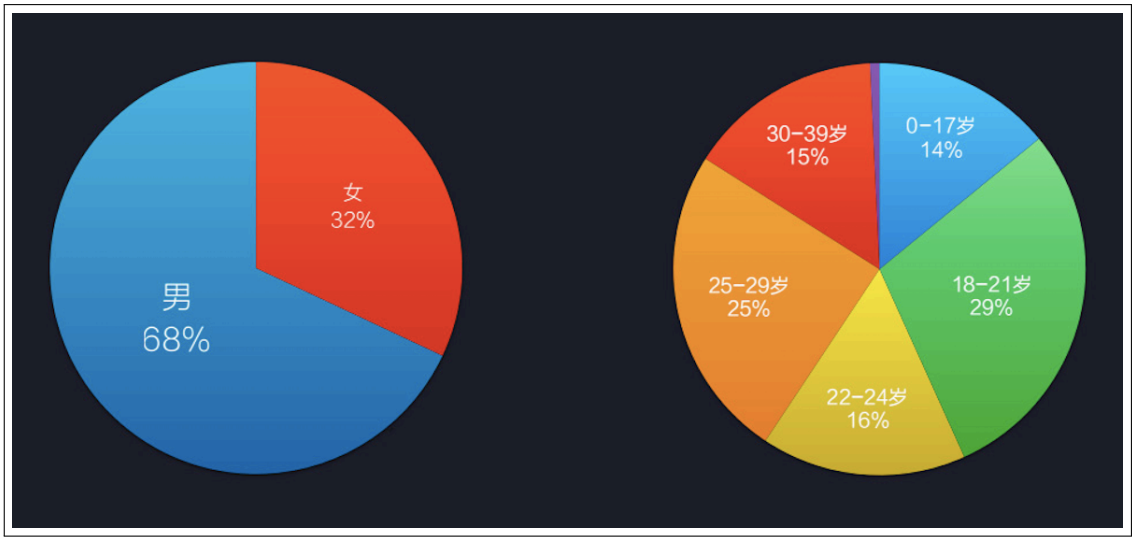

图 **4.5**　手机主题市场用户群体分布

- 典型个体的行为时序分析。所谓典型个体是指某段时间内，成长比较突出的用户。例如从一个新用户从新注册到点击过百、浏览过千需要有一个积累过程，有些用户积累较快，有些较慢，而这些积累较快的用户可以作为典型个体；或者某些用户在某一阶段消费有限，但在某时刻消费激增，无论是消费金额还是数量都变化很大，这种也可以作为典型个体。针对典型个体，需要挖掘与其用户成长相关的行为因素。基本方法是对时间进行分片，获取用户在不同时间片上的行为统计，以及在各个时间分片上的用户成长指标（点击量、购买量、点击转换比等）。在此基础上针对用户行为的统计量的变化，利用关联性分析或回归来分析用户成长与哪些因素有关。

- 典型群体行为模式分析。针对典型个体，从用户的基本信息、人口信息、兴趣维度，可以将相似的典型用户划分为同一的群体，称作典型群体，针对典型群体中的用户按照成长程度进行划分，按不同的成长阶段统计用户行为，即建立了该典型群体的行为模型。例如，对于“年龄在 20 30 岁，女性，付费用户”这样的典型群体，从日点击量、月消费额等维度将其划分到初创、成长、快速提升、成熟等阶段，针对不同成长阶段内的行为组合进行统计，结果构成该群体的行为模式。如图 4.5

4.5　用户画像应用场景

4.5.1　优化手机主题市场供求

改变了原有的先设计、再销售的传统模式。第三方主题设计师在设计一款新产品前，会先设定好主题类型，然后通过用户画像平台中分析该用户群体的偏好，有针对性的设计产品，从而改变原先新产品高失败率的窘境，增强销售表现。如设计一款智能手表主题，面向 28-35 岁的年轻男性，通过在平台中进行分

析，发现属性 =“金属”、风格 =“硬朗”、颜色 =“黑色”/” 深灰色”、价格区间 =“中等”的偏好比重最大，那么就给新产品的设计提供了非常客观有效的决策依据。

4.5.2 提高新人留存率

工商管理有一个理论叫做，维护一个老用户的成本是获取新用户成本的五分之一甚至更低。所以如果能够把一些已经流失的用户召回来，这时候成本比拉一个新用户低得多，你做的事也会带来更大的价值。鉴于此公司启动了一个项目叫” 用户画像之拉新”，首先利用用户画像得出最近一个月没有登录过的用户数据，然后根据浏览时长分档，这是因为用户需要花自己的时间成本才能留下的最有价值的标签，之后利用用户静态标签，像姓名、职业、年龄、地域分布做进一步的细分，最后针对不同类型的用户提供不同的优惠活动。ABtest 显示，与传统一刀切的推荐相比，基于用户画像的拉新留存率提高了约 50%。

4.5.3 用户消费等级分群

大至用户终端品牌、机型、操作系统，细至屏幕分辨率、屏幕尺寸，用户画像记录了每一个用户群体的详细终端特征。哪一类人群最容易被这款应用吸引，愿意为这款应用付费？开发者经常考虑的问题可以从用户画像找到答案。每一个用户群的价格分布、增值业务费用分布以及流量费用，包括用户详细的消费特征，比如付费频率，丰富了推荐系统的数据依据。

4.5.4 用户流失预警

一般情况用户在消费过程中会经历对如下几个期间：新鲜期，沉迷期，消退期，离开四个阶段，如何能够延长用户在应用的停留周期是需要解决的问题之一。用户画像可以辅助推荐系统进行流失用户特征分析，通过决策数算法，分析流失用户特征，建立不同原因流失的用户模型，然后通过这些特征得到当前在应用活跃用户中匹配流失概率高的用户数据。

4.5.5 反作弊

用户画像会对用户的消费能力、空闲时间、信用评级等维度进行打分；利用反作弊模型通过业务方访问收集数据，供安全部门参考。

4.6 总结

用户画像对于推荐系统来讲，主要如下几个方面的提升：提升推荐系统的精度，用户画像将用户的长期偏好融入到了推荐内容中，维护了推荐系统一致性。abtest 显示，融合了用户画像的推荐模型比单纯的推荐模型在点击转化率指标提高了约 2.8%，考虑到 300 万用户的基数，2.8% 的提升是一个很大的进步；用户

画像还解决新用户的冷启动问题，对于一个新注册用户来讲，推荐系统可以利用用户画像的静态信息，然后结合商品信息进行推荐；提高推荐系统的时效性，对用户行为的离线预处理，可以节约推荐系统的大部分计算时间。但是用户画像只是反映了用户长期的兴趣，所以无法动态的反映用户短期兴趣，因此我们引入了用户兴趣探索模块，将在下一章节件详细介绍。

第五章　用户兴趣探索

电子商务产品的设计往往是数据驱动的，即许多产品方面的决策都是把用户行为量化后得出的。但就商品而言，那些热门主题往往只代表了用户一小部分的个性化需求，只有通过对用户行为的充分分析，才能更好的挖掘出用户的兴趣，最终提升商品的销售量。现有的推荐算法注重用户或资源间的相似性的同时却忽略了用户兴趣的动态变化，从而导致系统在时间维度上有偏离用户需求的趋势。

为了更好的探索用户兴趣，手机主题推荐系统充分利用了用户画像和商品特征表。用户画像包括基本信息和兴趣特征向量, 商品特征向量表包括分类、标签、适用人群等，给定某用户行为，用户兴趣探索过程分为如下几个步骤：首先，利用用户历史行为 (评论, 停留时长, 评分, 点赞, 购买等) 建模量化用户满意度, 然后，利用用户兴趣特征向量与商品特征向量得出相关分数，如果商品与用户的相关分数很低, 但有很高的用户满意度, 说明是一次成功的用户兴趣探索, 更新用户画像。如果是热门商品, 大量的用户都会点击, 但商品与用户不是很相关, 则认为其探索效果是有限的, 反之如果是小众商品, 考虑到长尾效应, 则可以认为其是更成功的兴趣探索。这里涉及到的关键概念包括用户满意度的量化、小众标签的定向挖掘、用户兴趣的动态化。

本章内容首先介绍海量用户行为数据的存储方式。用户行为数据拥有区别于传统数据库数据的特点有，用户行为数据量巨大，常面临 TB 甚至 PB 级的数据；含有较多的噪音；多维聚合式查询。针对这些特征，用户行为数据采用 Hbase 数据集群存储和 hadoop 集群计算。然后，介绍用户兴趣探索的算法模型；然后，介绍如何通过用户行为的分析量化用户满意度。最后，介绍小众兴趣标签的挖掘。

5.1　用户行为数据的存储

手机主题用户行为数据的特点包括：用户基数庞大。手机主题注册用户达千万级，活跃用户达百万级；用户规模增长快。月新注册用户达 10 万数量级。每个用户的行为数量较小。即使是活跃用户，每天最多也只能产生上百条行为记录，每年不超过十万条；用户行为的计算较为复杂。计算用户的两次登录间隔天数、反复购买的商品、累积在线时间，这些都是针对用户行为的计算，通常具有一定的复杂性；用户行为数据格式不规整，字段丢失率较高。根据用户行为数据的这些特点，我们采用基于 Hadop 分布式的架构。

- 高可靠性。Hadoop 按位存储和处理数据的能力使其具有高可靠性。

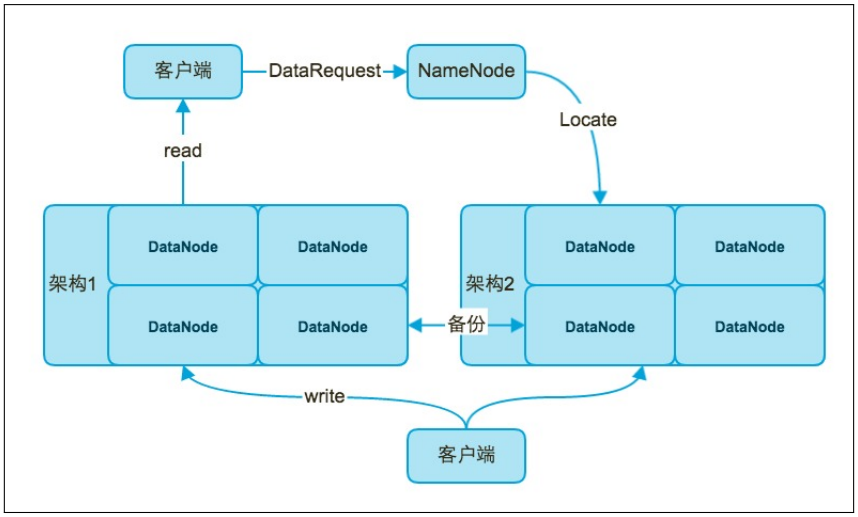

图 **5.1**　HDFS 体系结构

- 高扩展性。Hadoop 是在可用的计算机集簇间分配数据并完成计算任务的，这些集簇可以根据用户增长规模方便地扩展到数以千计的节点中。
- 高容错性。Hadoop 能够自动保存数据的多个副本，并且能够自动将失败的任务重新分配。
- 高效性。Hadoop 能够在节点之间动态地移动数据，并保证各个节点的动态平衡，因此处理速度非常快。
- 低成本。hadoop 是开源的，项目的软件成本因此会大大降低。

5.1.1　HDFS 的体系架构

HDFS 采用主从 (Master/Slave) 结构模型，一个 HDFS 集群是由一个 NameNode 和若干个 DataNode 组成的 (在最新的 Hadoop2.2 版本已经实现多个 NameNode 的配置)。NameNode 作为主服务器，管理文件系统命名空间和客户端对文件的访问操作。DataNode 管理存储的数据。HDFS 支持文件形式的数据。从内部来看，文件被分成若干个数据块，这若干个数据块存放在一组 DataNode 上。NameNode 执行文件系统的命名空间，如打开、关闭、重命名文件或目录等，也负责数据块到具体 DataNode 的映射。DataNode 负责处理文件系统客户端的文件读写，并在 NameNode 的统一调度下进行数据库的创建、删除和复制工作。NameNode 是所有 HDFS 元数据的管理者，用户数据永远不会经过 NameNode，HDFS 体系结构图如图 5.1所示。

5.1.2　Hive 数据管理

HDFS 为海量的数据提供了存储，则 Hive 支撑了海量的数据统计。Hive 是建立在 Hadoop 上的数据仓库基础架构。它提供了一系列的工具，用来进行数据

提取、转换、加载，是一种可以存储、查询和分析存储在 Hadoop 中的大规模数据机制。可以把 Hadoop 下结构化数据文件映射为一张成 Hive 中的表，并提供类 sql 查询功能，除了不支持更新、索引和事务，sql 其它功能都支持。可以将 sql 语句转换为 MapReduce 任务进行运行，作为 sql 到 MapReduce 的映射器。提供 shell、JDBC/ODBC、Thrift 等接口。优点是成本低可以通过类 sql 语句快速实现简单的 MapReduce 统计。

本小节主要介绍了 Hadoop 分布式计算平台最核心的分布式文件系统 HDFS 以及数据仓库工具 Hive。从体系架构到数据定义到数据存储再到数据处理，Hadoop 分布式存储、计算平台为海量用户行为的分析和用户兴趣探索提供了可能。接下来的章节先介绍用户行为数据的分析，包括数据预处理和异常数据监测，然后介绍用户兴趣探索模块，包括算法模型、用户满意度量化、小众兴趣标签的挖掘。

5.2　用户行为数据的的预处理

数据预处理是数据挖掘过程中一个重要步骤，当原始数据存在不一致、重复、含噪声、维度高等问题时，更需要进行数据的预处理，以提高数据挖掘对象的质量，最终达到提高数据挖掘所获模式知识质量的目的。

5.2.1　背景

随着手机主题市场交易规模的逐步增大，积累下来的业务数据和用户行为数据越来越多，这些用户数据往往是电子商务平台最宝贵的财富。目前在手机主题推荐系统中大量地应用到了机器学习和数据挖掘技术，例如个性化推荐、搜索排序、用户画像建模等等，为企业创造了巨大的价值。本节主要介绍在用户兴趣探索实践中的数据预处理与特征挖掘方法。

5.2.2　特征提取

用户兴趣探索的任务包括：探索用户的兴趣广度、兴趣深度、兴趣变动趋势。依据这些信息，推荐系统就能知道在面对某一个用户时要推荐哪几类型商品，每类商品所占的比例，未来几天推荐内容会有哪些变化。在确定了目标之后，接下来需要确定使用哪些数据来达到目标。提取哪些特征数据可能与用户是否点击购买相关，一方面可以借鉴一些业务经验，另一方面可以采用一些特征选择、特征分析等方法。从业务经验来判断，可能影响用户是否点击下单的因素有：

- 用户历史行为。对于老用户，之前可能有过点击、购买等行为。
- 用户实时兴趣。
- 用户满意度。上面的特征都是比较好衡量的，用户满意度可能是更复杂的一个特征，具体体现在用户评分、评价、购买后使用频率、时长等。

- 是否热门，商品评价人数，购买数等。

在确定好要使用哪些数据之后，还需要对使用数据的可用性进行评估，包括数据的获取难度，数据的规模，数据的准确率，数据的覆盖率等。

- 用户历史行为。只有老用户才会有行为，新用户是没有的。
- 数据获取难度。获取用户 id 不难，但是获取用户年龄和性别较困难，因为用户注册或者购买时，这些并不是必填项，即使填了也不完全准确。如果一些特征需要通过其他预测模型交叉验证的话，就存在着模型精度的问题。
- 数据覆盖率。数据覆盖率也是一个重要的考量因素，例如地理位置特征，并不是所有用户的距离我们都能获取到，PC 端的就没有地理位置，还有很多用户禁止使用它们的定位功能。
- 用户实时行为。如果用户刚打开 app，还没有任何行为，同样面临着一个冷启动的问题。
- 数据的准确率。有时候用户购买一款主题，不一定是其真心喜欢，可能是因为遇到限时半价、购买返现等活动。

5.2.3 特征获取方式

特征提取方式分为在线提取和离线提取。离线特征获取方案。离线可以使用海量的数据，借助于分布式文件存储平台，例如 HDFS 等，使用 Spark 等工具来计算海量的数据等。在线特征比较注重获取数据的延时，由于是在线服务，需要在非常短的时间内获取到相应的数据，对查找性能要求非常高，因此推荐系统输入流使用了 Kafka，Kafka 是一种分布式的，基于发布/订阅的消息系统。

5.2.4 用户行为数据预处理

原始服务器日志数据脏数据的形成原因包括：缩写词不统一，数据输入错误，不同的惯用语，重复记录，丢失值，不同的计量单位，过时的编码等。相应的，数据预处理内容包括数据清理、数据集成、数据变换、数据归约、数据离散化。

1）数据清理包括格式标准化、异常数据清除、错误纠正、重复数据的清除。对于手机主题用户数据来讲，引起空缺值的原因主要是用户设备异常造成的，有些时候是因为与其他已有数据不一致而被删除或数据的改变没有进行日志记载。根据数据空缺情况的不同有不同的处理方式：

- 忽略元组。当一个记录中有多个属性值空缺、特别是关键信息丢失时，已不能反映真实情况，它的效果非常差。

- 去掉属性。缺失严重时, 已无挖掘意义。
- 人工填写空缺值。但是工作量大且可行性低。
- 默认值。比如使用 unknown 或-∞。
- 使用属性的平均值填充空缺值。
- 预测最可能的值填充空缺值。使用贝叶斯公式或判定树这样的基于推断的方法。

2）数据集成就是将多个数据源中的数据整合到一个一致的存储中，需要注意以下几个情况：

- 模式集成。整合不同数据源中的元数据时的实体识别问题，比如匹配俩个表中的用户 ID，A.custId=B.customerNo。
- 检测/解决数值冲突。对现实世界中的同一实体, 来自不同数据源的属性值可能有所不同，如同表示停留时长，A 表单位是秒，B 表单位为毫秒。
- 多表之间的数据冗余。同一属性在不同的数据库中会有不同的字段名，有些时候冗余可以被相关分析检测出来，计算公式如所示，其中 $\overline{A}$ 和 $\overline{B}$ 表示为字段 A 和 B 的平均值，$\sigma_A \sigma_B$ 表示其的标准差。仔细将多个数据源中的数据集成起来, 能够减少或避免结果数据中的冗余与不一致性, 从而可以提高挖掘的速度和质量。

$$r_{A,B} = \frac{\sum(A - \overline{A})(B - \overline{B})}{(n-1)\sigma_A \sigma_B} \tag{5.1}$$

3）数据变换包括数据的平滑变换、数据聚集和数据规范化。所谓规范化是指将数据按比例缩放, 使之落入一个小的特定区间，推荐系统采用了最小-最大规范化。如**??**所示，原始数值范围为 [min,max]，通过公式映射到新区间 [newMin,newMax]，v' 表示属性 v 的公式映射。

5.3　用户兴趣探索的算法模型

用户兴趣探索就是不断学习用户所感兴趣的内容反馈给个性化推荐模型去加强推送相关内容，本节首先介绍用户兴趣模型的基本概念，然后介绍算法模型的组成结构：用户异常兴趣探测，用户小众兴趣标签的挖掘和用户满意度量化，用户兴趣衰减算法。

5.3.1 基本概念概述

实体域。当我们想基于用户行为分析来建立用户兴趣模型时，我们必须把用户行为和兴趣主题限定在一个实体域上。个性化推荐落实在具体的推荐中都是在某个实体域的推荐。对于手机主题应用市场来说，实体域包括所有的主题，背景图片，铃声，闹铃等。

用户行为。浏览，点击，下载，试用，购买，评论等都可是用户行为。本文所指的用户行为都是指用户在某实体域上的行为。比如用户在手机铃声产生的行为。

用户兴趣。用户的兴趣维度，同样是限定在某实体域的兴趣，通常以标签 + 权重的形式来表示。比如，对于手机主题，用户兴趣向量可以是「动漫，0.6」，「体育，0.1」，「情感，0.7」等分类标签。值得一提的是，用户兴趣只是从用户行为中抽象出来的兴趣维度，并无统一标准。而兴趣维度的粒度也不固定，如「体育」，「电影」等一级分类，而体育下有「篮球」，「足球」等二级分类，篮球下有「NBA」，「CBA」，「火箭队」等三级分类。我们选取什么粒度的兴趣空间取决于具体业务模型。

兴趣空间。在同一层次上兴趣维度的集合，比如手机主题中，可以用「热门」，「游戏」，「限时特价」，「科技」来构成一个程序员兴趣标签空间，也可以用「二次元」，「萝莉」，「魔幻」，「纯真」，「召唤兽」· · · · ·「法术」等构成一个动漫兴趣标签空间。

5.3.2 用户异常兴趣探测算法

本文所指的用户异常兴趣是指用户对小众标签主题表现出足够多的满意度。比如用户小磊每次都会浏览动漫、美少女主题，但是有一天却购买了一款汽车手机主题，那么程序可以检测到这种异常情况，然后将汽车标签更新到用户画像中，并作为个性化推荐的依据。事实上用户兴趣迭代过程可以在很短的时间内完成，基于 hive + MapReduce 平台的时长维度为天，而基于 kafka + spark 平台可以将时长维度降到小时级别。用户异常兴趣检测算法要从用户的行为和偏好中发现新的兴趣标签，并基于此给予推荐，工作内容包括收集用户的最新的行为数据是并分析得出异常标签，如algorithm 5.1所示

5.3.3 长尾标签抽取算法

标签集中度 (tagFocus) 是指，如果某个标签在一类主题中出现的频率高，其他主题类型很少出现，则认为此兴趣标签具有很好的类别区分能力。这是因为包含兴趣标签 t 的主题越少，也就是 n 越小，则说明标签 t 具有很好的兴趣区分，则其探索权重越大。如果某一类主题包 C 中包含兴趣标签 t 的个数为 tagInThemeNum，而其它类包含 t 的总数为 tagInOtherNum，则所有包含 t 的主题数 n=tagInThemeNum+tagInOtherNum，当 m 大的时候，n 也大，标签权重值会小，

```
Input: 用户画像数据 userProfile , 用户显示、隐式行为数据 logUsers
Output: 用户异常兴趣标签 newUsersTags
init newUsersTags;
for (user_i in logUsers) do
    for (tag_j in user_i) do
        if (tag_j.weight == 0) then
            //若标签权重已经为 0, 该用户兴趣标签将被删除
            remove tag_j;
        end
        else
            //成功探测到新用户兴趣标签
            temp.get(user_i).set(tag_j);
        end
    end
end
return temp;
```

算法 **5.1:** 用户异常兴趣探测

就说明该标签 t 类别区分能力不强。实际上，如果一个标签在一个类的主题中频繁出现，则说明该标签能够很好代表这类主题的特征，这样的标签应该给它们赋予较高的权重，并选来作为该类主题的特征向量以区别于其它类主题。热度 (tagPopular) 指的是某一个给定标签在用户画像中出现的频率。例如在 300 万用户总数中，十分之一的用户标签中有" 火影" 标签，那么其热度为 0.1，除此之外有些标签如" 精品"，" 气质" 等标签占了总词频的 80% 以上，而它对区分主题类型几乎没有用。我们称这种词叫"应删标签"。即应删除词的权重应该是零，也就是说在度量相关性是不应考虑它们的频率。标签集中度公式如式 5.2所示，我们很容易发现，如果一个标签只在很少的主题包中出现，我们通过它就容易锁定搜索目标，它的权重也就应该大。反之如果一个词在大量主题包中出现，我们看到它仍然不很清楚要找什么内容，因此它应该小。热度公式如式 5.3所示。长尾标签抽取算法如algorithm 5.2所示。

$$\mathrm{tagFocus} = \log \frac{|\mathrm{tagInThemeNum}|}{|\mathrm{tagInThemeNum} + \mathrm{tagInOtherNum}|} \tag{5.2}$$

$$\mathrm{tagPopular} = \log \frac{|\mathrm{peopleLikeTagNum}|}{|\mathrm{allPeople}|} \tag{5.3}$$

5.3.4　用户满意度量化算法

要从用户的行为和偏好中量化用户满意度，并基于此实现兴趣标签探索，如何收集用户的偏好行为成为用户兴趣探索效果最基础的决定因素。用户有很多方式向系统提供自己的偏好信息，而且不同的应用也可能大不相同。表 5.1列举

```
Input: 用户画像数据 userProfile , 用户显示、隐式行为数据 logUsers
Output: 长尾兴趣标签 longTailTags
init longTailTags;
for (user_i in logUsers) do
    for (tag_j in user_i) do
        weight_ij=tag_j.tagFocus/tag_j.tagPopular;
        if (weight_ij <= threshold) then
            //若标签权重小于阈值, 该用户兴趣标签将被删除
            remove tag_j;
        end
        else
            //成功探测到新用户兴趣标签
            longTailTags.get(user_i).set(tag_j);
        end
    end
end
return longTailTags;
```

算法 **5.2:** 长尾兴趣探测

的用户行为都是比较通用的，实际使用中提取的用户行为有数十种，根据不同行为反映用户喜好的程度将它们进行加权，得到用户对于物品的总体喜好。显式的用户反馈比隐式的权值大，但比较稀疏，毕竟进行显示反馈的用户是少数；而隐式用户行为数据是用户在使用应用过程中产生的，它可能存在大量的噪音和用户的误操作，可以通过经典的数据挖掘算法过滤掉行为数据中的噪音，这样可以使分析更加精确。然后是归一化操作，因为不同行为的数据取值可能相差很大，比如，用户的浏览数据必然比购买数据大的多，如何将各个行为的数据统一在一个相同的取值范围中，从而使得加权求和得到的总体喜好更加精确，就需要我们进行归一化处理使得数据取值在 [0,1] 范围中。算法如algorithm 5.2所示。

5.3.5　标签权重的线性衰减

标签权重的线性衰减算法结合了手机主题用户兴趣偏好变化频繁的特点，根据时间因素权重自动进行衰减，并准确反映用户兴趣的变化趋势。该模型是指用户对资源项目的评分仅代表评价当时的兴趣度，随着时间的推移，用户对该资源项目的评分将规律性地自动衰减，当项目评分衰减到 0 时，该资源项目将被兴趣模型所淘汰。评分衰减可以按照线性规律进行，如图 5.2所示。算法描述如algorithm 5.4所示。

表 **5.1**　用户行为和其权重

用户行为	类型	特征	作用	权重
评分	显式	整数量化的偏好，可能的取值是 [0, 5]	通过用户对物品的评分，可以精确的得到用户的满意度，但是噪声比较大，比如遇到好评返现活动	1
分享	显式	布尔量化的偏好，取值是 0 或 1	通过用户对物品的投票，可以精确的得到用户的喜好度，同时可以推理得到被转发人的兴趣取向(不太精确)	2
评论	显式	一段文字，需要进行文本分析，得到偏好	通过分析用户的评论，可以得到用户的情感：喜欢还是讨厌	1
赞/踩	显示	布尔量化的偏好，取值是 0 或 1	带有很强的个人喜好度	3
购买、试用	显式	布尔量化的偏好，取值是 0 或 1	用户的购买是很明确的说明这个项目它感兴趣。	3
点击流	隐式	包括滑屏频率，滑屏次数，屏停留时长，用户对物品感兴趣，需要进行分析，得到偏好	用户的点击一定程度上反映了用户的注意力，所以它也可以从一定程度上反映用户的喜好。	1
停留时长	隐式	一组时间信息，噪音大，需要进行去噪，分析，得到偏好	用户的页面停留时间一定程度上反映了用户的注意力和喜好，但噪音偏大，不好利用。比如说用户在浏览一个主题的时候，丢下手机和同学出去踢球去了，页面提留时长可能会很长	1

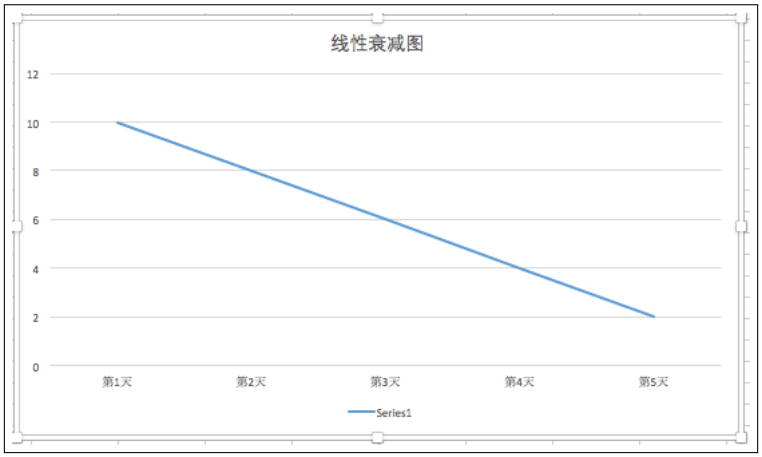

图 **5.2**　线性衰减模型

```
Input: 用户显示、隐式行为数据 logUsers
Output: 用户行为权重 userActionWeight
init userActionWeight;
for (user_i in logUsers) do
    for (action_j in user_i) do
        //获取用户此次行为的偏好权重并做归一化
        weight_ij=getWeigth(action_j); if
        (user_i exsit in userActionWeight) then
            //对用户行为做加权处理。
            double remaind=userActionWeight.get(user_i);
            userActionWeight.get(user_i).set(remaind+action_j * weight_j);
        end
        else
            userActionWeight.get(user_i).set(action_j * weight_j);
        end
    end
end
return userActionWeight;
```

算法 **5.3:** 用户满意度量化算法

```
Input: 用户画像模型中所有用户兴趣哈希表 users
Output: 更新后的所有用户兴趣哈希表 users
for (user_i in users) do
    //考虑到断电或意外关机等原因导致系统中断运行的特殊情况，算法
    加入了临时变量 temp
    temp.add(user_i.profile);
end
for (user_i in logUsers) do
    for (tag_j in user_i) do
        if (tag_j == 0) then
            //若标签权重已经为 0, 该用户兴趣标签将被删除
            remove tag_j;
        end
        else if (tag_j exsit in temp(user_i)) then
            //若存在新的评分值, 则更新为新标签权重
            temp.get(user_i).set(tag_j);
        end
        else
            //否则的话，将偏好值减少 0.5，进行衰减
            temp.get(user_i).set(tag_j-0.5);
        end
    end
end
return temp;
```

算法 **5.4:** 用户画像线性衰减

图 **5.3**　达尔文雀

5.4　用户兴趣探索评估方法

5.4.1　线下测试

在太平洋东部加拉帕戈斯（Galapagos）的一个小岛上有一种名叫达尔文雀的鸟，一部分生活在岛的西部，另一部分生活在岛的东部，由于生活环境的细微不同它们进化出了不同的喙，如图 5.3所示，这被认为是自然选择学说上的一个重要例证。同样一种鸟，究竟哪一种喙更适合生存呢？自然界给出了她的解决方案，让鸟儿自己变异（设计多个方案），然后优胜劣汰。具体到达尔文雀这个例子上，不同的环境中喙也有不同的解决方案。上面的例子包含了 A/B 测试最核心的思想：多个方案并行测试；每个方案只有一个变量（比如鸟喙）不同；以某种规则优胜劣汰。评判用户画像模型的效率高低，主要是看该模型带来的点击率、转换率等指标数据，其他统计量见表 5.2所示。理论上评测推荐系统的指标有用户满意度、预测准确度、覆盖率、多样性、新颖度、惊喜度、信任度、实时性、健壮性等。然而商业开发中，评测推荐结果只看重一个指标：点击转化率。能够提升商业价值，给业务带来更多利益的推荐系统，就是好的推荐系统。

A/B 测试对用户画像建模的作用有三个：特征提取，一些标签对用户的兴趣有强相关作用，如性别标签，有些标签是弱相关作用，如用户职业标签，A/B 测试需要筛选出强相关标签，过滤掉弱相关标签；权重量化，根据 A/B 测试实验显示，发现用户画像中的最近点击标签、最近关注标签所占权重比想象中的要大；标签组合，有些标签是冗余的，只需从中选一即可。A/B 测试具体实现步骤如下：

- 方案设计。实验之前需得到一个基准版本，然后把又争议的标签按照优先度列举出来决定是否实验。真正的 A/B 测试只应一次改动一个地方，这意味着标签选择、权重量化、标签组合要分开来测试。

- 确定数据评估方案。根据实验内容不同评估它们好坏的标准也不同，如果是标签选择那么衡量的主要指标是点击量，如果是权重量化那么衡量的主要指标是点击转换率。

表 **5.2**　A/B 测试主要评估指标

指标	描述
访客数	访客数就是指一天之内到底有多少不同的用户访问了你的网站。访客数要比 IP 数更能真实准确地反映用户数量。
浏览量	即 Page View，浏览量和访问次数是呼应的。用户访问网站时每打开一个页面，就记为 1 个 PV。同一个页面被访问多次，浏览量也会累积。
点击转化率	点击转化率计算公式: 点击转化率 = 成交笔数/浏览量 *100%，成交笔数影响着成交金额，所以点击转化率成为了衡量推荐系统效果的重要数据之一。
停留时长	停留时长是用户访问网站的平均停留时间，是衡量网站用户体验的一个重要指标。如果用户不喜欢主题包的内容，可能稍微看一眼就关闭页面了，那么停留时长就很短; 如果用户对页面的内容很感兴趣，停留时长就很长。
跳出率	跳出率是指访客来到网站后，只访问了一个页面就离开网站的访问次数占总访问次数的百分比，跳出率越低说明流量质量越好，用户对网站的内容越感兴趣。
其他指标	各种辅助性指标如点击量/用户，购买量/用户，下载量/用户等。

- 流量分配。为了试实验所得数据具备统计意义，能准确反映用户的真实行为，需要对流量设置一个下限。除此之外，为了使各个方案具有可比性，A、B 俩个方案的流量必须是相等的。
- 测试周期。根据所需测试的项目的不同测试周期也有所不同，如添加一个地理标签需要的测试周期以天为单位，如果涉及到多个标签的权重变动则需要测试周期以周为单位。
- 评估结果。适者胜出，其代表的数据作为下一轮回 A/B 测试的基准版本。
- 建立通用的数据评估题型。在经过各种类型 A/B 测试实验后，已经积累很多的评估指标，有必要把这些指标抽象出来形成一个通用的数据评估模型，减少以后实验的重复设计评估指标的时间。

5.5　总结

这一章主要研究了标签动态变化的对推荐系统的影响，实际中用户同时会受到社会因素和个人因素的影响, 但这两种因素在会产生不同强度的影响。在快速变化的系统中, 用户行为更加会受到社会因素的影响, 而在变化相对较慢的系统中, 用户行为则更加受到个人因素的影响。本章首先介绍了用户行为数据的存储方式以及基于此的用户行为数据的的预处理。然后介绍了用户兴趣探索的组

成内容，包括用户异常兴趣探测、长尾标签抽取、用户满意度量化、标签权重的线性衰减。最后给出了用户兴趣探索评估方法，包括离线和在线俩种。下一章节主要介绍如何把用户画像和兴趣探索融入到推荐系统中，从而搭建出一个具有长尾性、实时性的推荐系统。

第六章　结束语

如果说过去的十年是搜索技术大行其道的十年，那么个性化推荐技术将成为未来十年中最重要的革新之一。目前几乎所有大型的电子商务系统，如 Amazon、阿里、小米、滴滴等，都不同程度地使用了各种形式的推荐系统。一个好的推荐系统需要满足的目标有：个性化推荐系统必须能够基于用户之前的口味和喜好提供相关的精确的推荐，而且这种口味和喜欢的收集必须尽量少的需要用户的劳动。推荐的结果必须能够实时计算，这样才能够在用户离开网站前之前获得推荐的内容，并且及时的对推荐结果作出反馈。实时性也是推荐系统与通常的数据挖掘技术显著不同的一个特点。一个完整的推荐系统由三部分构成：用户画像模块，用户行为模块、推荐算法模块。用户画像模块记录了用户长期的信息，刻画用户的基础类型。用户行为模块负责记录能够体现用户喜好的行为，比如购买、下载、评分等。这部分看起来简单，其实需要非常仔细的设计。比如说购买和评分这两种行为表达潜在的喜好程度就不尽相同完善的行为记录需要能够综合多种不同的用户行为，处理不同行为的累加。推荐算法模块的功能则实现了对用户行为记录的分析，采用不同算法建立起模型描述用户的喜好信息，通过推荐模块实时的从内容集筛选出目标用户可能会感兴趣的内容推荐给用户。因此，除了推荐系统本身，为了实现推荐，还需要一个可供推荐的内容集。比如，对于手机主题推荐系统来说，所有上线主题就是这样的内容集。我们对内容集本身需要提供的信息要求非常低，在经典的协同过滤算法下，内容集甚至只需要提供 ID 就足够。而对于基于内容的推荐系统来说，由于往往需要对内容进行特征抽取和索引，我们就会需要提供更多的领域知识和标签属性。

推荐系统是一种联系用户和内容的信息服务系统，一方面它能够帮助用户发现他们潜在感兴趣的内容，另一方面它能够帮助内容供者将内容投放给对它感兴趣的用户。推荐系统的主要方法是通过分析用户的历史行为来预测他们未来的行为。因此，时间是影响用户行为的重要因素。关于推荐系统动态特性的研究相对比较少，特别是缺乏系统性的研究。对动态推荐系统的研究，无论是从促进用户兴趣模型的理论角度出发，还是从实际需求来看，都具有重要的意义，本文的研究工作正是在这一背景下展开。

6.1　研究工作总结

本文对推荐系统特别是与用户画像相关的动态推荐系统的相关工作做了总结和回顾之外，主要的工作包括以下几个方面：

- 设计出了基于用户画像的推荐模型：按照用户属性和行为特征对全部用户进行聚类和精细化的客户群细分，将用户行为相同或相似的用户归类到一个消费群体，这样就可以将推荐平台所有的用户划分为 N 个不同组，每个

组用户拥有相同或相似的行为特征，这样电商平台就可以按照不同组的用户行为对其进行个性化智能推荐。目前国内主流电商平台，在进行个性化智能推荐系统升级过程，都在逐步向 DNN 渗透和扩展，也是未来个性化智能推荐必经之路。在现有用户画像、用户属性打标签、客户和营销规则配置推送、同类型用户特性归集分库模型基础上，未来将逐步扩展机器深度学习功能，通过系统自动搜集分析前端用户实时变化数据，依据建设的机器深度学习函数模型，自动计算匹配用户需求的函数参数和对应规则，推荐系统根据计算出的规则模型，实时自动推送高度匹配的营销活动和内容信息。

- 设计出了考虑用户长期兴趣和短期兴趣的用户兴趣探索模型: 通过量化用户满意度和量化量化主题标签的流行度，最大化推荐系统的推荐多样性和长尾性，提高用户的惊喜度。

- 动态推荐系统的原型设计: 综合前几章的推荐系统各个模块的研究, 设计了一个实际的推荐系统的原型。该系统包含了用户画像和用户兴趣探索模型。能够实时根据用户行为变化的趋势, 实时的调整推荐结果排名, 从而不断改善用户在推荐系统中的体验。

6.2　对未来工作的展望

本文对推荐系统的用户画像和用户兴趣探索模型进行了较深入的研究, 但是针对用户兴趣变化的推荐模型的实现还有很多工作要做。本人认为用户兴趣动态推荐系统的实现有待解决的问题如下:

- 用户行为的离线和在线计算的分配：用户行为每天产生的数据量很大，哪些行为需要在线实时计算反馈，哪些行为只需要离线计算即可，需要根据具体业务的特点和用户习惯赋予每种行为一个权重，然后根据权重排名决定计算方式。因此, 用户行为的特征提取、分析将是我们将来工作的一个重要方面。

- 用户兴趣探索模型对推荐系统的影响: 本文的所有工作基本集中在高推荐系统的点击购买转换率上。但点击购买转换率并不是推荐系统追求的唯一指标。比如, 预测用户可能会去看, 从而给用户推荐速度与激情, 这并不是一个好的推荐。因为速度与激情的热度很高, 因此并不需要别人给他们推荐。上面这个例子涉及到了推荐系统的长尾度, 即用户希望推荐系统能够给他们新颖的推荐结果, 而不是那些他们已经知道的物品。此外, 推荐系统还有多样性等指标。如何利用时间信息, 在不牺牲转换率的同时, 提高推荐的其他指标, 是我们将来工作研究的一个重要方面。

- 推荐系统随时间的进化: 用户的行为和兴趣是随时间变化的, 意味着推荐系统本身也是一个不断演化的系统。其各项指标, 包括长尾度, 多样性, 点击率都是随着数据的变化而演化。如何让推荐系统能够通过利用实时变化的用户反馈, 向更好的方面发展是推荐系统研究的一个重要方面。

最后, 希望本文的研究工作能够对动态推荐系统的发展作出一定的贡献, 并真诚的希望老师们出宝贵的批评意见和建议。

参考文献

[1] O. Celma. *Music Recommendation and Discovery in the Long Tail*. Springer. 2010.

[2] Marko Balabanovi´c and Yoav Shoham. *Fab: content-based, collaborative recommendation*. Commun. ACM, 40:66–72, March 1997.

[3] Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, David M. Pennock. *Methods and Metrics for Cold-Start Recommendations*. New York City, New York: ACM. pp. 253–260. 2002.

[4] CTEX Sia, K.C., Zhu, S., Chi, Y., Hino, K., Tseng, B.L. *Capturing User Interests by Both Exploitation and Exploration*. Technical report, NEC Labs America. 2006.

[5] Jansen, B. J. and Rieh, S. *The Seventeen Theoretical Constructs of Information Searching and Information Retrieval*. Journal of the American Society for Information Sciences and Technology. 61(8), 2010.

[6] Han, Jiawei; Kamber, Micheline. *Data mining: concepts and techniques*. Morgan Kaufmann. p. 5. 2001.recmd-system

[7] Francesco Ricci and Lior Rokach and Bracha Shapira. *Introduction to Recommender Systems Handbook*. Springer, pp. 1-35. 2011.

[8] Robert K. Merton. *The Matthew Effect in Science*. Science, 159(3810):56– 63, January 1968.

[9] Junghoo Cho and Sourashis Roy. *Impact of search engines on page popularity*. In Proceedings of the 13th international conference on World Wide Web, WWW '04, pages 20–29, New York, NY, USA, ACM. 2004.

[10] Daniel M. Fleder and Kartik Hosanagar. *Recommender systems and their impact on sales diversity*. In Proceedings of the 8th ACM conference on Electronic commerce, EC '07, pages 192–199, New York, NY, USA, ACM. 2007.

[11] Henry Kautz, Bart Selman, and Mehul Shah. *Referral web: combining social networks and collaborative filtering*. Commun. ACM, 40:63–65, March 1997.

[12] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. *Evaluating collaborative filtering recommender systems*. ACM Trans. Inf. Syst., 22:5–53, January 2004.

[13] Kohavi, Ron, Longbotham, Roger. *Online Controlled Experiments and A/B Tests*. In Sammut, Claude; Webb, Geoff. 2015.

[14] Elaine Rich. *Readings in intelligent user interfaces*. chapter User modeling via stereotypes, pages 329–342. 1998.

[15] Anne-F. Rutkowski and Carol S. Saunders. *Growing pains with information overload*. Computer, 43:96–95, June 2010.

[16] J. Scott Armstrong, editor. *Principles of Forecasting - A Handbook for Researchers and Practitioners*. Kluwer Academic, 2001.

[17] Henry Kautz, Bart Selman, and Mehul Shah. *Referral web: combining social networks and collaborative filtering*. Commun. ACM, 40:63–65, March 1997.

[18] Greg Linden, Brent Smith, and Jeremy York. *Amazon.com recommendation- s: Item-to-item collaborative filtering*. IEEE Internet Computing, 7:76–80, January 2003.

[19] Anne-F. Rutkowski and Carol S. Saunders. *Growing pains with information overload*. Computer, 43:96–95, June 2010.

[20] Yehuda Koren. *Collaborative filtering with temporal dynamics*. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '09, pages 447–456, New York, NY, USA, 2009. ACM.

[21] Thomas Hofmann and Jan Puzicha. *Latent class models for collaborative filtering*. In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, IJCAI '99, pages 688–693, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[22] Bruce Krulwich. *Lifestyle finder: Intelligent user profiling using large-scale demographic data*. AI Magazine, 18(2):37–45, 1997.

[23] Mohsen Jamali and Martin Ester. *Trustwalker: a random walk model for combining trust-based and item-based recommendation*. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '09, pages 397–406, New York, NY, USA, ACM. 2009.

致 谢

人生就是一个关于成长的漫长故事。而在中科大求学作为本人人生体验的一部分，亦是这样的一段故事。在此的俩年半，俯仰之间，科大的“问道”、“学术”于此，让我经历了这样的三段成长：学于师友，安于爱好，观于内心。

“古之学者必有师，师者，所以传道、授业、解惑也”。师友的教诲不可能一直跟着自己，可是他们治学态度却融入了我的人生观。授课的华保健老师的严谨、郭燕老师的认真、丁菁老师的直率、席菁老师的踏实都曾触动我，并给予我前进方向上的指引。

本论文内容为数据挖掘在电商行业的工程实现，因此有一段真实的、贴近数据挖掘领域的实习经历尤为重要。感谢我在苏州国云数据公司实习的 CEO 马晓东学长，让我有机会一窥大数据行业的内幕；感谢我在小米实习的导师方流博士，感谢我在滴滴出行工作的机器学习研究院李佩博士，让我成为大数据挖掘工程师的梦想又更近了一步；感谢我的导师周武旸教授和张四海教授，指导我完成论文。向师友和书籍学习，是从外界汲取；只有回归到自己的内心和思绪才能沉淀．在每个夜幕深沉或是晨曦初露的时刻里，感受自己情绪的流动，反思自己的取舍得失，然后才有了融于师友和书籍时的奋进。这样的三段成长，如今已是一体，不断地相互印证与反馈！

“逝者如斯夫，不舍昼夜”。成长亦复如是，不断的和昨日的自己告别。但是，一路有你，真好！相会是缘，同行是乐，共事是福！

胡磊

2016 年 4 月 4 日