中国科学技术大学

硕士学位论文

基于手机主题推荐系统的用户画像模型

作者姓名: 胡磊
学科专业: 信息安全专业
导师姓名: 胡小宇
完成时间: 二〇一六年九月

University of Science and Technology of China

A dissertation for master's degree

The User Profile Based on Phone Theme Recommendation System

Author :	Lei Hu
Speciality :	Information Security
Supervisor :	Prof. Wuyang Zhou
	Dr. Sihai Zhang
Finished Time :	September 1, 2016

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

始于二十世纪九十年代的信息爆炸，使得用户越来越难有效的从茫茫多的数据中获取所需信息，因此，推荐系统凭借其精准的定位和” 千人千面” 的个性化服务受到人们的青睐和研究者的重视。本论文讨论了如何构建一个基于手机主题推荐系统的用户画像模块和用户兴趣探索模块。

传统的个性化推荐系统面临着诸多挑战，其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，推荐系统面临着包括社交化、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适用于手机主题的用户画像模型，实践证明其能在很大程度上提升推荐系统的推荐质量。本文的主要工作和贡献有:

- 实现了推荐系统的用户画像模块。利用信息检索（Information Retrieval）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，标签的本身，以及标签与用户之间的共现关系决定着这个标签的初始权重，然后根据用户行为构建相应的 A/B 测试产出标签的实际权重，权重越高则认为该标签对用户影响越大。A/B 测试显示推荐系统利用用户画像标签进行推荐能显著提升诸如点击转换率等重要指标。

- 实现了推荐系统的用户兴趣探索模块。用户兴趣探索通过特征提取技术和用户满意度量化算法，定期更新用户兴趣标签和标签对应的权重。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。用户兴趣探索能够实时更新用户的兴趣标签，帮助推荐系统持续满足用户的不断变化的需求。

- 利用时间因子衰减模型融合用户的长期兴趣和短期兴趣。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，衰减模型法的本质是利用自然遗忘规律拟合用户真实的兴趣衰减过程。

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

Information explosion in the new age let it's hard for users to get valuable infomation from the vast amounts of data, so the recommended system begin to go to the middle of the stage because it's precise forecast and Personalized service. So we here to discuss how to modeling users profile model and users interested exploration model for a android phone theme application recommended system.

There are so many weekness of the traditional recommended system, the most import one is how to sell more products, specific for android phone application, the recommended system need to solve Socializing problem, cool start problem, dynamic recommend based on timeline and so on. So the author proposed and implemented users profile model and users interested exploration model which include:

- Realized the use profile model of recommended system, we use information retrieval technology to get use basic information like occupation, location, gender from user registration information, different tag has different weight depending on the way they got, the path of they transfer and the relation between use and tags, the more weight of tag the high of credibility the tag has. A/B test show that recommended system can improve click conversion rate rapidly.

- Realized the users interested exploration model of recommended system, which using feature extraction technology and user satisfaction scoring algorithm, we maintain a dynamic interesting tags vector space for all user. first, we can get user-item-scores by product users interesting vector metric and items feature metric. Then get the users satisfaction based on users history actions like buying, rateing, clicking and so on. one successful exploration means it has low user-item-relation-scores and high user satisfaction, and the tag also is minority. Experiments show that with the users interested exploration model, the recommended system has more long-tail effect.

- Sucessfully put user long term interesting and short term interesting into one model using linear decay algorithm, users profile model contains static infomation of users, users interested exploration model contains dynamic infomation of users interesting, this papar come up with the strategy to balance the static infomation and the dynamic infomation.

Keywords: recommend system, long-tail, dynamic, user profile, user interest explore

目 录

表格索引

插图索引

第一章　绪论

1.1　研究背景与意义

互联网自二十世纪九十年代从诞生、发展，到现在已经演化为人类社会的必需品。但是，后互联网时代又是个性化时代 [1]，需要一种系统精确刻画每个用户的兴趣爱好并能不动声色的在主页上表现出来，我们把这种提供个性化服务的系统系统统称为推荐系统。推荐系统是一种比搜索引擎更人性化、个性化的系统服务，不需要用户主动提供关键词，因此它能满足用户的更多的潜在需求，尤其当用户自己都无法精准描述自身需求的时候 [2]。第一代推荐系统以亚马逊为代表，作为一个电子商品平台，一方面有数以万计的商品需要被用户了解、熟悉和购买，另一方面有数以亿计的用户无法找到称心如意的商品。推荐系统通过构建用户和商品之间的桥梁，每年为亚马逊贡献近三十个百分点的创收！由此可见推荐系统能帮助用户快速发现有用的商品信息，具体来讲，首先推荐系统通过分析用户的历史行为每个用户进行独一无二的画像建模 [3]，目的有俩个：1，熟悉每个用户和他们的潜在需求；2，把拥有相同品味的用户归为一类群体，这样一来所有人的需求总和就可能是其中一个人的潜在需求，方便企业卖出更多的商品。随着用户终端设备的普及，出现了诸如淘宝、美团、滴滴、今日头条等互联网平台，几乎包办了人们衣食住行的方方面面，人们因为可选择性太多而出现了“选择性困难”的症状，这其实就是信息过载时代的具体表现形式。也就是说，在这个数据爆炸时代，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，都面临着日益严峻的挑战，现代人每天面临着从各种不必要的数据中找到有用的商品，其实是在浪费生命。每一个有追求、有理想的现代人，真的需要好好的设计人生，以一种精要的方式摒弃不必要之事，而这也是林语堂先生所说的生之智慧。笔者曾有过这样的一种购物体经历：笔者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。对于传统的推荐系统，首先，需要积累足够多的商品信息，因为只有尽可能的在基于所有的商品大局观上，才有可能得出比较正确的商品推荐候选集合；其次，需要尽可能积累用户的行为样本数据，因为这些样本将会是推荐统计假设检验的唯一数据标准，统计学之所以常让人意外，就是因为人们只能得到部分样本，而部分样本只是包含了事情的部分信息，于是就有扭曲事情本质的趋势，因此，精度是推荐系统最重要的指标之一，后文会详细介绍如果通过用户画像和用户兴趣提升推荐系统的精度；最后就是甄别，哪些商品对哪些用户有着非同寻常的吸引力，其实对于一个用户来讲，平台上存在的绝大多数商品，包括数据、资源和他人观点，都没有什么价值，只有少数商品效果非凡，影响巨大，推荐系统的核心就是算法 [4]，通过算法甄别无意义的多数，只留下有意义的少数。总之，通过算法分析用户兴趣，分析商品特性，对用户跟所有商品的关

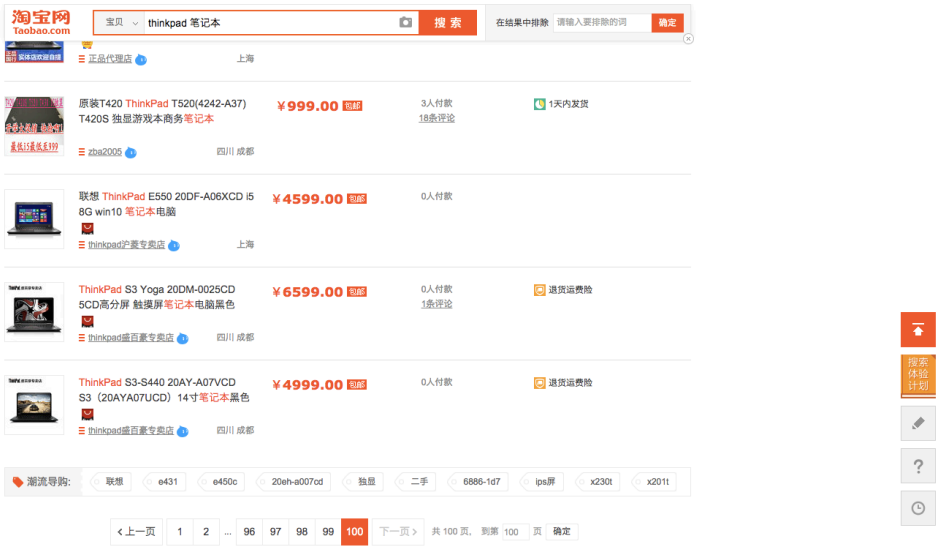

图 **1.1**　淘宝购物搜索图

联度打分、排序、取 topN，然后给出推荐结果。

但是，传统的推荐系统也有一些问题，典型的有数据稀疏问题、新用户问题、马太效应、实时推荐问题和用户兴趣变动问题。数据稀疏问题的本质就是商品信息数据过于膨胀，即使是骨灰级用户也没办法穷尽百分之一的商品，因此大多数的用户-商品相关值都是零，这不利于推荐系统做出正确的推荐结果；新用户问题又叫冷启动问题，指一个用户刚刚注册登录，推荐系统没有与此人相关的信息，于是就没有方法做出推荐；马太效应是指越热门的商品越有被推荐的趋势，这种情况其实不是一件好事，因为：1，商品营收不平衡会增加平台的风险性，如果平台大多数营收的贡献来自于极少类明星商品，一旦这类商品发生问题，平台也会有问题；2，根据 2/8 原则，冷门商品虽然营收少，但它们的基数大，潜力无限；实时推荐问题是指用户从浏览到购买这段时间一般很短，而推荐系统需要打时间差，在用户点击之后、购买之前的这段时间做出推荐，但这是很难实现的，客观原因是工程上需要一定计算时间的；用户兴趣变动问题是指用户兴趣是一个动态的过程，有可能随着季节周期性变动，有可能随着年龄发散性变化，推荐系统需要及时收集数据，保持对用户兴趣的最优拟合。

基于这些问题的存在，笔者基于推荐系统实现了用户画像模块和用户兴趣探索模块，用以解决传统推荐系统所面临的种种问题，并帮助推荐系统做出更好的推荐结果。用户画像模块其实就是回归了问题的本质：以人为本，用数据说话。通过分析、收集所有与用户有关的数据，为每个用户建立、维护一个独一无二的用户画像，用户画像的最大优点在于它能主动收集用户的基本人口数据、长期兴趣和短期兴趣，而且用户画像中的信息是动态更新的，也就是说随着时间的推移，用户的兴趣在逐渐改变，用户画像里的兴趣标签也会随之改变，最大程度上保证了用户兴趣的连续性和变化性；用户兴趣探索模块包括三个原则：1，用户和潜在感兴趣商品的关联度很低，这保证了探索的商品都是用户从前没有看

见过的；2，用户满意度很高，即通过量化用户行为，包括点击次数、滑屏次数、滑屏频率、滑屏时长、点赞、分享等行为，得出用户的满意度；3，潜在商品的标签是小众的、冷门的标签，因为热门商品是没必要也不需要做探索的。

1.2　推荐系统的简介

推荐系统的研究其实是个交叉学科，因为其跟很多早期的基础领域的研究相关，比如认知科学 [5]，信息检索 [6] 和预测理论 [7]。随着数据时代的到来，研究人员开始研究如何利用用户对商品的行为数据来预测用户的兴趣，同时为用户提供推荐服务 [8]。近些年来，推荐系统越来越开始成为一个专门的研究课题，到 2005 年左右为止推荐系统的研究还是集中在基于 user、item 的协同过滤算法，在工业界目前应用最具有影响力的算法应该就是亚马逊的协同过滤算法 [9]。推荐系统推荐有俩个原则：1，给用户的推荐的商品不能与用户购买过的商品重复；2，不能与用户刚浏览过的商品太相关，否则这些商品会让用户有种千篇一律的感觉。

1.2.1　推荐系统的产生与发展

随着计算机存储技术以摩尔定律指数增长，信息的传播也开始爆发式的迅猛发展起来，我们这个时代的人类社会进入了一个崭新的大数据信息时代：互联网和物联网几乎无处不在，影响人类的衣食住行等方方面面，因此颠覆性的更改了人们的生活方式，在典型的互联网共享经济平台，一个用户既代表了消费者，也代表了生产者，就如笔者在滴滴出行的角色，平时上下班打车，属于运力消费者，周末开车做网约车司机，则变身为运力生产者。但是不好的一方面也开始浮现，曾几何时笔者发现自己社交账号开始多起来，数量之多以至于没办法记住每个账号的密码。这就是 Web 2.0 时代的一个副作用—让人们疲于奔命的把生活浪费在刷各种动态、信息，忙于各种无脑点赞、分享，而没有时间思考。社交化网络媒体如微信、微博的异军突起，导致互联网中的信息数据中充满了广告和噪声，而普通用户一般缺少过滤、屏蔽噪声的主观意愿和技术能力，不仅使其信息检索的时间成本巨大，也会让其在茫茫多的数据海洋里迷失自我，这就是信息过载问题的根源所在 [10, 11]。作为非常重要的技术手段，推荐系统和搜索引擎为用户解决信息过载提供了不可或缺的保障。俩者的不同之处在于：搜索引擎是分散的、被动的，用户需要先输入关键词，搜索引擎根据关键字在服务器后台进行信息检索，利用算法获得最优的匹配信息并展示给用户。但我们更多时候遇到的问题是，并不能精确描述自己的需求，而这就是推荐系统的强项，因为推荐系统是主动收集用户平日中一点一滴的数据，以至于用户不需要提供明确的需求，推荐系统只是通过分析用户的历史行为数据就可以“猜出”用户的意图，因此，如果我们把推荐系统和搜索引擎看作为两个互补的技术手段，那么效果一定很棒。

推荐系统最开始的概念，应该是在 1995 年由美国人工智能协会 [12] 上的 Robert Armstrong 教授首先提出，不仅如此，Armstrong 教授还不遗余力的推广了一个推荐系统的原型系统。受其启发，推荐系统的研究工作开始起步并发展壮大。第一个商用推荐系统应该属于 Yahoo 网站的个性化入口 MyYahoo。到了 21 新世纪，随着电子商务的风起云涌，推荐系统的研究与应用开始水涨船高，包括 eBay、taobao、Amazon、youtube[13] 等各大电子商务网站都有了自己的推荐系统。2006 年美国的 Netflix[14] 在网上公开了一个推荐算法竞赛，设立了丰厚的奖金，选手通过利用 Netflix 公开了的真实网站中的一部分数据，包含用户对电影的评分，利用数据挖掘算法预测哪些用户会购买哪些电影。2014 年阿里举办了阿里大数据竞赛，笔者所在团队参加了初赛选拔并顺利进入决赛，阿里公开了其部分用户三个月的浏览、收藏、购买商品数据，选手可以利用阿里天池计算资源做出预测，阿里大数据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。总体说来，一个成功的个性化推荐系统的应用主要表现在以下几个方面：

(1) 将潜在用户转变为购买者：用户在浏览的同时并不意味着一定是要消费，也许只是看看，遇到合适就买，没有就算。个性化推荐系统的职责之一是能够洞察用户的潜意识，帮助用户找到其感兴趣的商品，从而促成购买过程。

(2) 提高平台的连带销售能力：有时候个性化推荐系统需要一点联想能力，如用户购买了手机，那么推荐手机壳就是一种明智的联想，会让用户产生“你懂我”的感觉。

(3) 提高客户对平台忠诚度：个性化推荐系统就是那个时时刻刻为用户着想的机器人，每一次推荐只是那么一点点，不是很多，但都很好，这种依赖就是俞军先生所说的体验壁垒，对于用户来讲，从滴滴出行换到优步，功能还是原来的功能，只是用户习惯的打车方式都变了，以至于无法接受优步。

1.2.2 推荐系统的应用

对于诸如 Linkedin 的社交网络，推荐系统改变用户扩展人脉的模式和方法，而这是基于一种假设：你的朋友的朋友有可能就是你的朋友。对于诸如淘宝的电商平台，搜索提供的静态体验，并不足以让用户产生购买欲望，推荐系统加强了交互，包括用户和商品、用户和用户，一个人的消费可能带动一群人模仿，成就了一种极致的营销模式 [15]。对于诸如滴滴出行的网约车平台，给定某一时刻、起始经纬度、终点经纬度、车型，根据推荐算法一定会有一个最优派单，使得司机和乘客所得的好处，远远大于平台的抽成费用，形成了我们所说的三方共赢。

随着网络社交的深入人心，用户不再满足于单纯的获取信息，而是与网络上的其他用户进行关注、聊天和互动。国外著名的社交网络就有 Twitter、Facebook 等，国内的社交网络有微信、微博等。在社交网站中用户不再是一个静止端点，

而是与他人有错综复杂关系的社交网络。对于微信来说，最重要的资源应该就是用户之间的关联。这其中的关系可能是多层次的、多维度的、按时间序列走的，关联的因素可能是是亲人、好友、同学、同事，也可能只是网络中的萍水之交，如都是 QQ 黄金会员。因此，用户之间的关联应该有一个权重，表明了用户之间的紧密度、信任度，一个用户的好友可能是另一个好友的亲戚，因此推荐系统有助于帮助用户挖掘潜在的熟人。

1.3　用户画像的简介

用户，指企业的潜在消费者，是构成现有用户的大部分群体的统称。画像，是对一个用户的可视化、客观的描述。用户画像就是能够客观、可视化地描述潜在消费者的模型。用户画像建模的关键工作就是为用户打上合适的标签，标签通常是人为规定，且具有高度精炼的特征标识，如消费能力、偏好、年龄、性别等，将所有用户标签综合起来，抽象出本质，如忠诚度、消费度、满意度等，基本就可以勾勒出该用户的商业消费轮廓。

1.3.1　用户画像的产生背景

当互联网步入信息时代后，用户行为数据的极大丰富性给企业及消费者的消费行为带来一系列问题与变革。最大的问题在于电子商务的用户数量相比传统商务，膨胀了成百上千个数量级，单纯依靠人工方式已对其无解，2015 上半年，我国网民已达到 6.68 亿，预计年底能够顺利突破 7 亿，其中使用手机上网人群占整体 88.9%，而手机上网存在着独特性、唯一性和私密性的特点，每个人的手机都是一套独特的生态系统。最大的变革莫过于，消费者的一切行为信息都是可数字化，随着大数据工程技术的日益精湛，带宽、计算资源、存储资源也变得极大丰富起来。这使得企业有能力把专注点回归到问题的本质，即利用信息化管理方式为每位用户建立一个档案，根据用户的生活习惯、消费行为和社会属性等信息，抽象出的一个标签化的用户模型用以精准刻画用户，基于此进而充分挖掘用户潜在的商业价值，随着用户使用时间越长，模型就越能积累多的数据，也越能精确把握用户的消费习性，反过来越能促使用户的消费行为，形成一个良性循环，至此用户画像的概念也就深入企业和用户之心。

1.3.2　用户画像的应用

用户画像的本质就是了解企业的用户，然后完善产品运营提升用户体验，提升盈利，用户画像可以为包括推荐系统、运营推广、策略制定等提供数据支持。除此之外，用户画像可以帮助企业寻找潜在目标用户，在与用户的交互上了解其偏好，促成购买，实现精准运营和营销，用户画像改变了以往闭门造车式的商业交易模式，通过事先调研用户需求反馈，设计制造出更适合用户的产品。具体来讲，用户画像的应用包括：

- 完善及扩充用户信息：用户画像代表了用户的信息全貌，因此寻找足够多的数据是用户画像建模的前提条件。我国在各方面都是很大的长尾市场，互联网很大程度上弥补了信息的不对称，移动互联网能把信息精准送达到任意一个用户面前，尽管如此，根据 2/8 原则还是导致了大多数的用户和商品的数据是空缺着的。同时，在实际中用户的信息也可能提供得不尽完整，如对于没有填写性别信息的用户，用户画像可以通过用户兴趣探索模块，生成用户数据，可见，用户画像不仅消费数据，也可以生成数据。

- 打造健康的生态圈：在掌握用户信息的基础上，电子商务平台就可以对自身的状况进行分析，从相对宏观的角度刻画用户种群的分布，从基础上把握市场的生态环境，挖掘出商品的最大价值，帮助企业提高收入。例如笔者曾经发现，通过与当前热门电影保持同步，通过适时发布作品引导传播、跟进推广周边手机主题，可以很好的带动用户的消费行为，用户的消费与此同时也刺激了第三方设计师紧跟时尚潮流，尽可能第一时间发布引领流行的作品。

- 支撑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。在实际场景中，影响用户对商品的使用黏度的因素很多，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网上商店到退出使用，其间的每一步情况都被记录在案：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，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。

- 市场安全领域的应用：有时候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。

1.4　工程背景

小米科技有限公司作为国内发展较快的互联网企业，活跃用户过亿，移动端用户比例高，有着大量的用户和丰富的用户行为，这些为推荐系统的应用和优化提供了不可或缺的条件，我们基于 MIUI 主题应用商店开发的手机主题推荐系

统，作为用户和主题包之间的桥梁，体现出超强的变现能力。但现有的手机主题推荐系统也面临着一些问题。

(1) 新用户冷启动问题：当一个新用户进入一个站点时，我们对他的兴趣爱好还一无所知，这时如何做出推荐是一个很重要的问题。现有的机制是向用户推荐那写普遍反映比较好的物品，也就是说，推荐完全是基于物品的，这就会使热门的商品越来越热，冷门的商品越来越冷，代价就是加剧了热门商品的马太效应。

(2) 数据稀疏问题：通过观察我们发现只有约 20% 的用户有过多于 5 款/日主题的浏览记录，意味着大多数用户的消费处于待挖掘状态。与此同时，我们发现只有约 20% 的主题包有过多于 10 次/日的浏览次数，意味着大多数主题包的消费处于待挖掘状态，又是一个“蛋和鸡”的问题：要形成好的推荐，首先需要有大量的用户行为支持，这样才能得到足够多的推荐数据，这里问题的关键在于推荐系统如何首先能在数据稀疏的情况下给出优质的服务，打破闭环。

(3) 不断变化的用户喜好：这个问题主要分为俩类：1、用户一直喜欢某种类型的主题包，只是长时间没有机会接触，如一位男性用户喜欢美少女主题包款式，虽然不会主动查找，但如果不经意看到一款制作精美的美女主题包，可能还是会购买，这就是用户的长期兴趣。2、用户之前喜欢某种类型的主题包，之后转为喜欢另外一类主题包，如用户刚开始喜欢清纯系，后来转为温柔系，这时如果向用户推荐温柔系主题包更有可能被其接受，这就是用户的短期兴趣。

(4) 重复推荐的问题：手机主题包属于电子虚拟商品，它的特性是第一次下载需要购买，之后下载则免费，现有的推荐系统会重复推荐用户之前购买过的主题，导致占用有限的推荐位来显示无法变现的信息，并且会给用户一种不专业、不智能的体验。

(5) 其他问题：如隐性喜好 [16]、商品长尾性 [17]，相对来讲这些问题引起的关注度比较小。

我们发现，如果在底层数据仓库层和推荐系统之间加一个用户画像模块，会有效提升推荐系统的各项性能。1、对于新用户冷启动问题、数据稀疏问题，关键是收集足够多的用户基本信息，在没有或者只有少量用户行为的情况下依靠用户画像对用户推荐比较合理的主题。2、对于不断变化的用户喜好，我们通过用户画像存储用户中长期兴趣，通过用户兴趣探索获得用户短期兴趣，并针对手机主题市场的特点，利用线性衰减算法融合用户画像和用户兴趣探索，使得推荐结果能兼顾俩者。3、对于重复推荐的问题，我们在用户画像中维护一个白名单，

用来存储用户曾购买过的所有主题信息，格式为（userId, itemId, buyTime）这样的三元组，避免向用户推荐已购买过的主题。除此之外，我们也通过探索用户小众兴趣提升推荐系统的长尾发掘能力，加强了对小众主题包的推荐力度。主要思路是分析用户所有的行为数据，针对占大多数的冷门主题 (即包含小众标签的主题) 会赋予一个倾斜因子，这样会使得冷门主题更有可能被探索出来。

1.5　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、利用好如此多的数据也带来了很多困惑。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，在这里我们用到的开源工具包括：Redis、Hive、Kafka 等流行的大数据存储、计算工具。

1.6　论文结构

本文的其余正文内容由以下章节组成：

- 第二章是综述了用户画像的推荐系统，对用户画像的研究现状做了介绍，然后介绍了用户画像在推荐系统的应用。
- 第三章讨论了手机主题推荐系统的架构设计与技术实现，简单的介绍了用户画像与用户兴趣探索的作用，和其在推荐系统中的作用。
- 第四章讨论了用户画像模块，包括用户画像数据类型，用户画像的建模，最后给出了实验与分析。
- 第五章介绍了用户兴趣探索模块，包括用户数据的存储和处理，用户兴趣探索模型的俩个关键概念：长尾标签和用户满意度，然后介绍了用户画像和用户兴趣探索的融合的原理；最后给出了实验与分析。

第二章　基于用户画像的推荐系统综述

2.1　引言

自从 1992 年著名的施乐公司的科学家们为了解决困扰已久的信息负载问题，第一次从概念上提出协同过滤的算法模型。1998 年，林登及其同事们成功申请了 item 协同过滤技术的专利，经过多年的工程实践，美国电商亚马逊公司的工程师们骄傲的宣称：在公司所有的销售量，推荐系统占比已经占到整个 Gross Merchandise Volume 的百分之三十以上。不久之后的美国公司 Netflix，因为其创始人与前任公司签署有若干年内不得从事同行工作的限制，于是通过举办推荐算法优化竞赛绕开限制，用以开发出更好的推荐算法。此次竞赛吸引了数以千计的团队参与角逐，期间进行了上百种的算法模型组合、优化的尝试，虽然 Netflix 公司为冠军团队支付了百万美金，但回报是 Netflix 推荐系统的快速发展以及营收的俩位数增长。其中冠军团队凭借 Sigular Value Decomposition 和 Gavin Potter 跨界引入的心理学方法进行的组合算法模型，在诸多优秀团队中脱颖而出。其中，矩阵分解的核心是将一个非常稀疏的用户评分矩阵 R 分解为两个更小的矩阵：只包含 User 特性的矩阵 P 和只包含 Item 特性的矩阵 Q，利用 P 和 Q 相乘的结果 R’ 来拟合原来的评分矩阵 R，使得矩阵 R’ 在 R 相同位置之间的损失函数值尽量的小，通过定义一个 R 和 R’ 之间的距离计算公式 (一般为曼哈顿距离)，如果矩阵 R’ 是正定矩阵，那么把矩阵分解转化成梯度下降求解的局部最优解，就是全局最优解。与此同时，Pandora、LinkedIn、Hulu 等网站在个性化推荐领域都展开你争我抢的竞争势头，使得推荐系统在各个细分行业、垂直领域开始全面开花，都有了不少爆发性进展。但是，对于拥有全品类的综合性购物电商、广告营销，推荐系统的进展还是缓慢，主要原因是因为不同类型的商品，消费者的心态也是不同的，例如大型家电，消费者肯定是先看了又看、选了又选，从价格、定位、功能到噪声比、性价比，大多数都会先做足了调查，才会购买；与此相反，对于日常用品消费者可能眼睛都不眨就购买了，对于这俩种极端的消费情况，推荐系统需要做出截然不同的推荐策略，具体的，单个模型在母婴品类的推荐效果还比较好，但在其他品类就可能很差，很多时候需要根据场景、推荐栏位、品类等不同，设计不同的推荐模型。同时由于用户兴趣随时间会不停的变动 [18–21]，需要一种机制，使得推荐系统能定期对数据进行评估、分析，除此之外不同类型的商品有不同的更新频率，这就对推荐系统提出了更加智能化的挑战。还有，如果定期更新模型，则可能会因为计算资源的限制损害推荐的实时性 [22]，因为模型训练需要一定的 cpu 计算时间，而传统的 Hadoop 的方法实在是无法进行大的更新频率，spark 框架又因为昂贵的内存限制了其应用场景。

传统推荐算法包括基于人口统计学的推荐 [24]、基于商品内容的推荐 [25] 和 user-based/item-based 的协同过滤 [26] 的推荐等都有冷启动问题。基于内容的

推荐对物品冷启动问题免疫，但是无法解决用户冷启动问题 [27]。

由此，笔者在实际工程中，针对传统推荐算法的种种弊端，选择了用户画像。伟大的数学家、计算机学家 Knuth 先生说：如果遇到一个不好搞定的问题，那么就该添加一层中间层，用以屏蔽掉问题。实际上，用户画像作为底层数据仓库和上层推荐系统的缓冲层，起的就是这种作用。

2.2　用户画像的研究现状

2.2.1　用户画像的组成部分

基于内容和用户画像的个性化推荐，有两个实体：内容和用户。需要有一种文本机制联系这两者的东西，我们定义其为标签。内容特征文本化为标签即为内容特征化，用户兴趣文本化标签则称为用户特征化 [28–32]。因此，对于基于用户画像的推荐，主要分为以下几个关键部分：

(1) 标签库

标签是联系用户与用户、用户与商品、商品与商品之间的纽带，也是反应用户兴趣的重要数据源，标签的最终用途在于标记用户行为。标签库则是对标签进行聚合的系统，包括对标签的管理、更新等。在用户画像的过程中有一个很重要的概念叫做颗粒度，就是我们的用户画像应该细化到哪种程度。举一个极端的例子，如果“用户画像”最细的颗粒度应该是细到每一个用户每一具体的生活场景中，但是这基本上是一个不可能完成的任务，同时如果用户画像的颗粒度太大，又会影响推荐精度，一般来说，标签是以层级的形式组织的，如体育为一级维度、篮球为二级维度、NBA 篮球为三级维度等。

(2) 内容特征化

内容特征化即给商品打标签。目前有两种方式：人工打标签和机器自动打标签。在实际工程中，主题推荐系统采用人工打标签方式，具体就是提供一个关键字库，供设计师从中选择适当关键字作为作品的标签。

(3) 用户特征化

用户特征化即为用户打文本标签。通过用户的行为日志和一定的模型算法得到用户的每个标签的权重。用户对内容的行为：点赞、不感兴趣、点击、浏览。对用户的反馈行为如点赞赋予权值 1，默认为 0，不感兴趣为-1；对于用户的浏览行为，则可使用点击、浏览作为权值。对商品发生的行为可以认为对此商品所有标签的行为。用户的兴趣是时间衰减的，即离当前时间越远的兴趣比重越低。时间衰减函数使用 1/[log(t)+1]，t 为事件发生的时间距离当前时间的大小。要考虑到热门商品会干预用户的标签，需要对其标签进行降权。

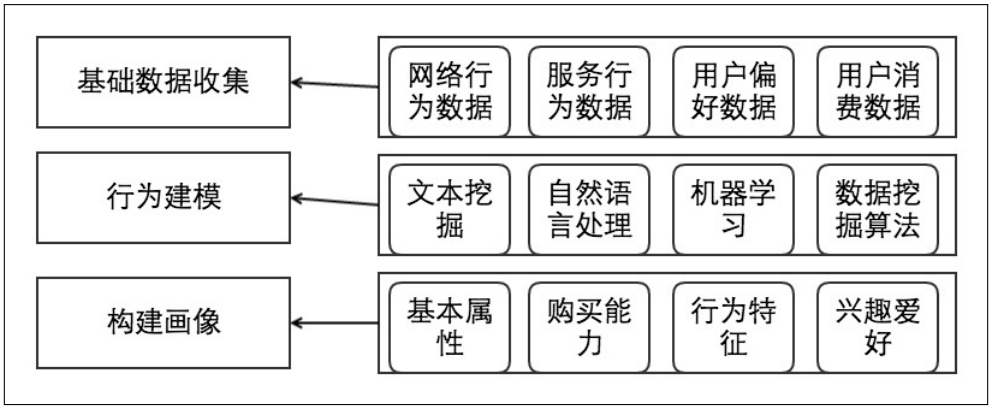

图 **2.1**　用户画像的构建周期示意图

2.2.2　用户画像的构建周期

用户画像，即用户信息标签化，就是企业通过收集与分析消费者社会属性、生活习惯、消费行为等主要信息的数据之后，获得用户的数据标签库。构建周期如图 2.1。

(1) 数据收集

数据收集大致分为四类：1、网络行为数据包括页面浏览量、活跃人数、访问时长、浏览注册转化率、注册活跃转换率等。服务内行为数据：点击浏览路径、网页停留时长、滑屏次数、滑屏频率、滑屏时长。用户内容偏好数据：点击、浏览、收藏内容、评价、评分、评论内容、社交内容、品牌偏好等。用户交易数据（交易类服务）：购买率、折扣率、导流率、流失率等。收集到的数据没必要是百分之百的准确，大体差不多即可。应用中，具体就是在数据清洗阶段过滤一部分不靠谱的异常值，验证、更新数据这块需要在后面的阶段再做判断，比如某用户在性别一栏填的女，但其语言数据显示其为男的概率更大，根据业务再选择丢弃数据还是更新数据。

(2) 用户画像基本成型

该阶段需要利用用户的基本属性，如性别、地域、年龄，得出用户更高层的抽象概念，如消费能力、忠诚度、活跃度、社交爱好等。因为用户画像永远也无法百分百地拟合现实中的一个人，因此，用户画像需要根据变化的基础数据不断修正已有的更高层的抽象概念，尽可能模拟用户的变化趋势。

(3) 数据可视化

最后是数据可视化分析，这部分是最能体现推荐系统的产出，因为人类对数据不如对图画来的敏感，在此步骤中一般是针对群体做进一步的抽象，按照消费习惯、消费能力、消费偏好把用户归类为一类人，比如可以根据用户对价格的敏感度细分出高价值用户、核心用户、高忠诚用户。而决策层所做出的评估也应该是基于某一群体的分布规律。典型的用户画像如图 2.2。

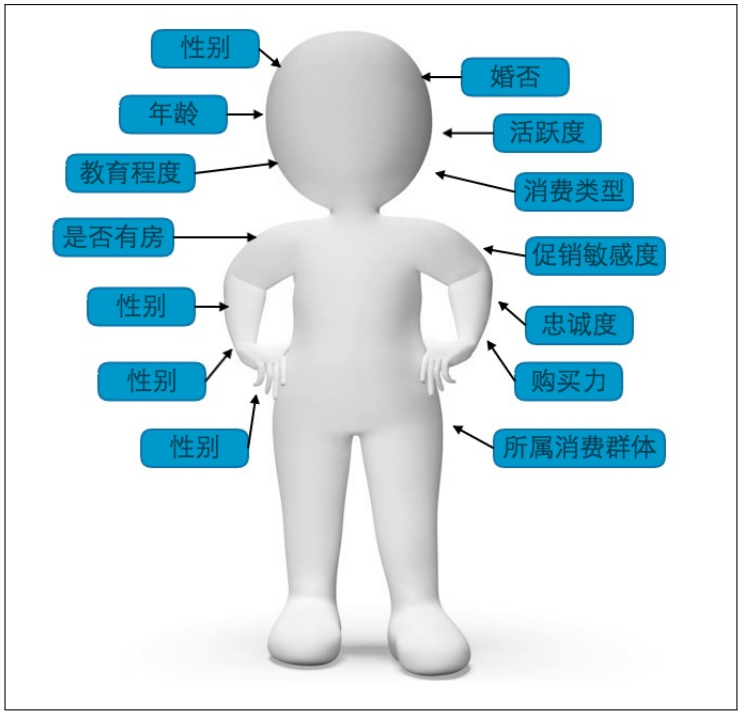

图 **2.2**　用户画像示意图

2.2.3　用户画像的建模

用户画像的建模包括内容标签化和标签权重量化。建模过程：1、内容分析，从原先的物品描述信息中提取有用的信息用一种规范化的标签表示，有时候这种信息源自于作者提供的描述，有时候源自于用户的评价，不管如何，都需要人工做进一步的审核；2、上传、记录用户注册信息，生成用户基本信息，这些信息基本是不会变化的；上传、记录用户行为数据，这些数据是不断变化着的，通常是采用数据挖掘算法从潜在物品集合中取出若干个结果表示用户喜好的模型。例如，一个网页推荐系统，可以通过分析用户过往浏览过的文章，得出用户喜欢浏览类似于范冰冰的花边新闻，如果用户点击了所推荐的文章，则说明分析正确，否则需要根据反馈重新训练模型，从而实现一个反馈-推荐-反馈的闭环；3、推荐系统得出推荐集合后往往需要取 topN，因为推荐系统的本质在精不在多。通过定义一个距离算法，匹配用户标签和商品标签的相关度，相关度一般正则为 0-1 之间，结果是一个二元的离散量：（feature, score）。根据相关度将生成一个用户潜在感兴趣的物品评分列表，然后去掉用户之前看过的商品，取 topN 即可。例如在电影用户画像的建模中，首先分析用户打分比较高的电影的共同特性，包括导演、演员、风格等，这些电影的标签就会成为此用户画像的一部分，根据打分的多少，给定一个合适的权重值。用户-标签用矩阵 A 表示，电影-标签用矩阵 B 表示，A 乘 B 得出矩阵 C，C 代表了用户与电影之间的相关度，固定一个用户，对所有相关度不为零的电影做排序，取 topN 即是推荐结果。用户画像建模的根本在于用户标签的获取和权重的定量分析。

2.2.4　用户画像和推荐系统的评测

首先，用户画像作为一个工具，只用在运用到某一场景才有意义，并能评估出其产出，因此本节主要介绍推荐系统的评测，根据推荐系统的表现好坏才能评估出用户画像的推荐质量。实际工程中，笔者利用 A/B 实验对若干组模型进行定量对比。标准的 A/B 实验是指通过一定的规则把类似的用户群随机分成俩组，采用旧模型的分组叫对照组，采用新模型的分组叫实验组 [33]。通过对用户展示不同的模型，得出用户的使用指标，关键是各种转化率，这样仅仅通过对比俩者的转化率即可得出各个模型的优劣。策略实验的难点在于如何找到合适的实验设计方案。通过时间交错能够在一定程度上减少由时间片带来的误差，这样就有一个难题: 如何选择合适长度的时间片。策略实验往往伴随着携带效应 (carry-over effects)，也就是上一个时间片的策略会对下一个时间片带来影响。笔者和同事们提出一个方案，当选择适当大的时间片的时候，通过 A/A 实验的数据调整 A/B 实验的结果，具体来说，如果 A/A 实验的结果是 0.4%，A/B 实验的结果是 1.2%。那么我们认为 A/A 实验是真实的时间片之间的差异，我们需要用俩者之差的绝对值去调整时间片带来的影响。

2.3　用户画像在推荐系统的应用现状

Amazon 的仓库里堆着数百万图书，Netflix 的服务器中存储有数万部电影，淘宝平台上的小卖家总共拥有 8 亿件物品，除此之外，这三家公司都保留有数以亿计的用户行为数据。互联网电子商务开始积累了海量的用户数据，然后因为数据量过于庞大，有用信息如金矿中的金子一样很难挖掘利用，与此同时，用户发现常常需要面对过多的选择。心理学研究证实过多的选择会使人犹豫不决，导致消极等待，最终可能放弃消费的决定，这个问题严峻到可以造成肉眼可见的用户流失。近代统计学理论的发展加上最近几年的数据科学和数据挖掘工程的进步，为电子商务平台提供更有效的应对方案: 推荐算法。推荐系统在帮助用户解决信息过载问题的同时，提升了企业价值。如今的企业不再局限于传统的推荐功能，通过建立完备的用户画像，推荐系统可以帮助企业更了解用户，在推广、反作弊、精细化运营等领域中发挥重要的作用。

2.3.1　基于用户画像的推荐系统的商业应用

作为全球社交网站中的翘楚，Facebook 在很早的时候就预言到了大数据 + 推荐系统 + 用户画像的无限前景。Facebook 自己的推荐系统就是需要利用分布式计算框架快速的帮助用户找到他们可能感兴趣的人、文章、分析、用户组等。Facebook 是个伟大的公司，一直为开源软件贡献着一份力量，最近在其官网就公布了 Facebook 自己的推荐系统原理、性能及使用情况 [34]。Facebook 的推荐系统需要面对的数据量应该是所有互联网公司中的数一数二，约包含了 1000 亿

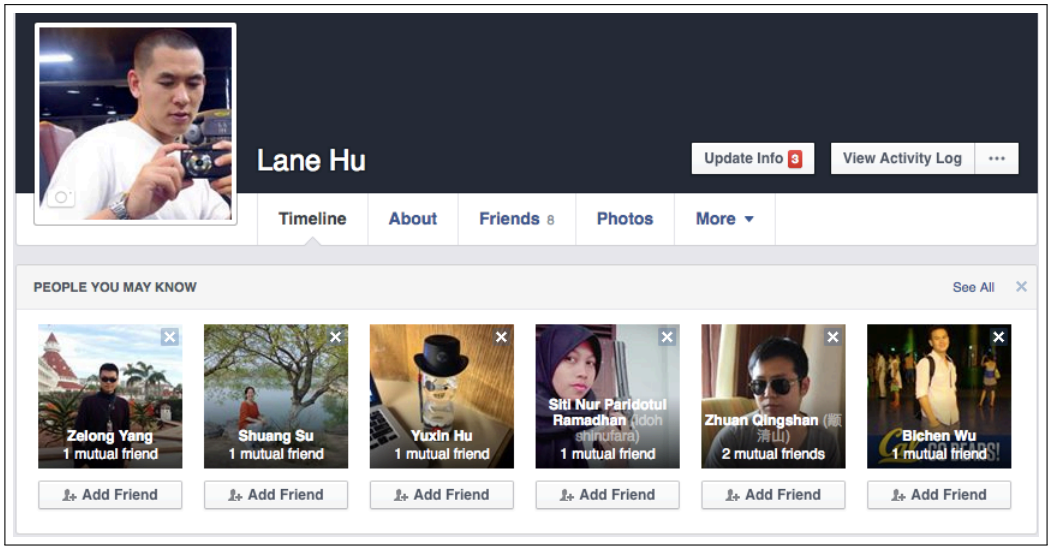

图 **2.3**　Facebook 个性化推荐用户界面

级别的评分数、10 亿级别的用户数以及百万级别的虚拟商品，如何在如此庞大的数据规模下，仍然保持良好性能已经成为世界级的难题，而 Facebook 解决了，通过分析 Facebook 给出了一些实验的结果，表明，Facebook 的系统比传统系统要快 10 倍左右。目前，该方法已经用到 Facebook 的多个应用中，包括用户、用户组的推荐。Facebook 推荐主页如图 2.3。

Facebook 的用户画像进展也十分可观，几乎是与推荐系统同步发展。2011 年 12 月，Facebook 发布了里程碑式的大数据产品——Timeline，通过开发 API 接口，允许用户自行编辑个人的时间轴：在什么时间、什么地点做了什么，遇到了谁，可以说在这条时间线记录这个人的全部生活故事。Timeline 通过帮用户回忆自己的点点滴滴的同时，完成了用户数据捕获、存储，而一旦拥有了这些历史数据，Facebook 就可以做进一步的数据分析、挖掘，这时的 Facebook 就如同和你从小长大的小伙伴，一个懂你的陌生人。可以说用户留下的数据越多，Facebook 就越了解这个人，投放的广告就会更加精准，最终 Facebook 利用庞大的用户数据生态赚足了钱。

2.3.2　推荐系统的主要方法

推荐系统主要有俩种思路：评分预测和 TopN 预测，核心的目标都是找到最适合用户的候选集合 s，从候选集合里挑选目标集合是一个非常复杂的非线性优化问题，通常采用的方案是用局部最优近似非线性最优，通过定义一个的损失函数，选取 TopN[35]。

推荐系统的算法基于统计学、概率论、线性代数、微积分技术，找出用户最有可能喜欢的商品，应该是现代互联网电商的明星应用。目前用的比较广泛的推荐算法还属协同过滤推荐算法，其基本思想是根据与他兴趣相近的用户的选择，得出推荐商品候选集，取 topN 推荐给目标用户，用维度为 m×n 的矩阵表示所有

用户对所有物品的兴趣值，这个值应该是根据用户历史行为数据得出，值越高表示这个用户越喜欢，利用特殊值 0 表示没有接触过。图中行向量表示某个用户对所有商品的喜爱程度，列向量表示某个商品对所有用户的吸引程度，因此单个元素 U_{ij} 表示用户 i 对物品 j 的喜欢程度。协同过滤分为两个阶段：预测阶段和推荐阶段。预测阶段是基于所有原始集商品，预测这个用户有没有可能对其感兴趣，量化为一个数值，只要值不为零即可归为候选集中；推荐是根据预测结果，先去重后去除消费过的商品，然后取去 TopN 推荐给用户。

尽管有这么多的优点，协同过滤算法也存在两大问题：1、数据稀疏性。一个大型的电子商务平台一般有百万级别的物品，用户可能接触到的商品占所有商品的百分之一不到，因此用户之间购买过的物品重叠性非常小，以至于没办法做推荐，一个办法是利用算法添补部分值 [36]。2、扩展性较差，因为一般来讲，电子商务平台中的商品变动很小，用户流入流出、日益增加、变动很大，基于用户的协同过滤算法需要不停的跟新迭代保证跟上用户变动的步伐。遇到这种情况，可以考虑基于商品的协同过滤算法，其基本思想类似于基于用户的协同过滤算法，只是相似性计算对象是商品，而商品一般变动很小可以忽略不计。如果我们知道物品 a 和 b 相似，而一般喜欢 a 的用户也喜欢 b，如果用户 A 喜欢 a，那么我们有很大把握得知 A 也应该喜欢 b，推荐了准没错。而物品之间的相似性比较固定，因此可以一次性计算出物品的相似度，将结果存储到 Redis 中，推荐时查询 Redis 即可。

2.4　本章小结

本章首先介绍了传统推荐算法的存在的弊端，包括冷启动问题；然后介绍了用户画像的组成部分，包括用户的标签库和商品的标签库；之后介绍了用户画像的构建周期，包括数据收集、用户画像成型和数据可视化；然后介绍了用户画像的建模，包括内容标签化和标签权重量化。

第三章　手机主题推荐系统整体设计与实现

3.1　引言

小米主题应用拥有成千上万款主题包，而一个用户整个活跃周期只能接触不到十分之一的主题，所以我们现在面临的一个问题是，如何帮助用户发现新的主题，这些主题同时满足俩个条件：1、不能和用户之前看过的、购买过的主题包重复。2、不能和用户之看买过的、购买过的主题不相关，而这也是我们开发的手机主题推荐系统所要达到的目标之一。除此之外，手机主题推荐系统要达到的目标包括帮助第三方设计师推广其作品。手机主题应用本身既不生产主题包，也不消费主题包，存在的价值就在于提供一个平台，能让用户、设计师和广告商从中受益。每个设计师都希望更多的用户体验、使用他们的主题。得益于个性化推荐系统的投入使用，我们现在可以把更多的主题包直接推送给那些潜在消费者面前。

本章节主要介绍手机主题推荐系统的完整架构。如图 3.1可知推荐引擎主要由推荐模块、用户画像模型、用户兴趣探索模块组成。

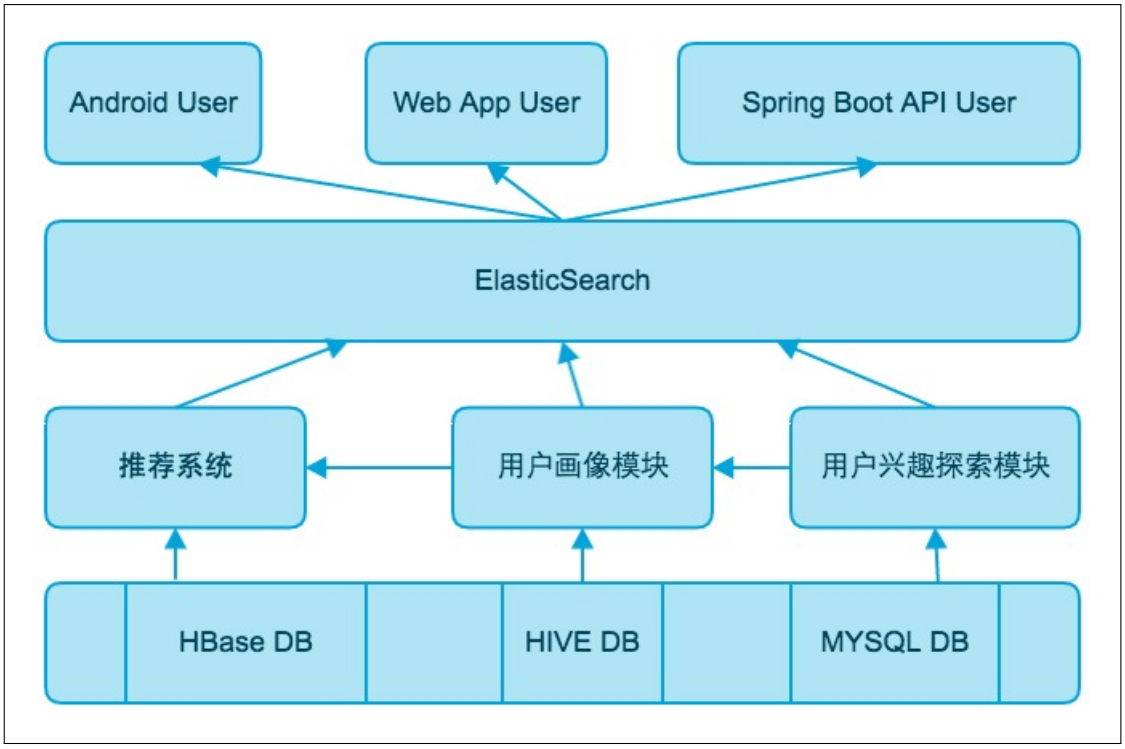

图 **3.1**　推荐系统引擎框架总览图

3.2　手机主题推荐系统设计

推荐系统框架如图 3.1。最顶层显示的是推荐系统对外服务的客户端。由于不同展位的输入输出参数差异较大，因此这一层没有做过多的抽象，每个展位有自己特定的接口 Json 定义，接口层通过调用 ElasticSearch 搜索服务引擎实现秒级别的用户推荐结果列表。推荐系统利用离线方式更新 ElasticSearch 搜索服务器数据。用户画像模块除了作为推荐系统的输入数据外，也可以直接作为 ElasticSearch 的输入。用户兴趣探索模块定期扫描活跃用户和上架主题包，通过分析用户行为日志更新用户画像。从接口层接受到的每次响应请求会被记录成用户行为数据，包括请求的一些必要的上下文信息以及用户及主题包的特征信息。借助 HBase、Hive、Mysql 等数据平台对原始日志进行处理，从而得到需要的各种数据及模型：包括用户的画像信息，用户之间的相似度，商品之间的相似度。在推荐系统的候选集生成这一块，重度使用了 item-based 协同过滤算法，而对于那些行为稀少的用户和新用户，需要根据平台的特点进行做好冷启动策略。对于 Spring Boot API 输入、输出数据格式分别如Listing 3.1和Listing 3.2。

Listing 3.1　Spring Boot API 输入格式

```
{
  "user_id":"123",
  "dims":{"type":"normal", "free_or_charge":"mixed"},
  "white_list":{"id":1141},
  "max_number":"8",
  "start_date":"2016-07-01,
  "end_date":"2016-08-01
}
```

Listing 3.2　Spring Boot API 输出格式

```
{
  "code":0,
  "message":"successfully",
  "data":{
    "total":111,
    "themes":[{
      "id":1141,
      "subscribe":1,
      "business":null,
      "keytype":"market:orderid",
      "tag":null,
      "name":"lovely baby",
      "displayName":"小可爱",
      "description":"家庭，儿童欢乐多",
```

```
        "author":"摆渡车1024",
        "sigModel":2,
        "type":2,
        "dependency":null,
        "createTime":1462447261000,
        "dims":null}
      ]}
  }
```

3.2.1　数据采集和日志格式化

我们的数据采集来源包括移动端埋点和用户请求。目前常见的前端埋点技术有三类：代码触发埋点、可视化触发埋点和延迟埋点，根据手机主题商店的业务特点和用户规模，我们选用可视化触发埋点，当用户在 UI 上点击了某个可埋点的控件时，会自动触发回调函数调用接口发送相应事件的 log 信息，可视化触发埋点不同于代码触发埋点，其理念是把核心代码和配置、资源分开，在 APP 启动的时候通过网络更新配置和资源即可，不必每一个埋点都需要写代码，埋点产生的数据量很少，且只针对特定人群，所以用来做 A/B 测试的数据源。用户关键行为会被上传、存储到服务器。获取数据后根据数据来源和存储方式，将日志格式化为：public 日志、nginx 日志、binlog 日志和 passport 日志，public 日志存放手机端用户请求 log，nginx 日志存放 Web 端用户请求，binlog 日志是将 public 日志同步到 NoSQL DB 的数据，passport 日志存储用户验证信息等数据。

3.2.2　用户画像的收集

当上述日志格式化生成，通过每日定时任务扫描 passport 日志就可以获取新注册用户并为其在 ElasticSearch 创建一个 topic，同时利用移动端埋点功能获取到用户手机 IMEI 号、经纬度等基本信息，利用用户注册手机号或者邮箱账号获取用户的通信录和好友信息，借助好友信息完善此用户的用户画像，除此之外有时可以借助第三方接口获取用户的基本信息。

3.2.3　商品标签的构建

小米手机主题应用商店里的主题包大多数是由第三方设计师创建、当设计师上传成品到官方产品库时会被要求填写作品标签，官方审核员也会更改、删除、添加一些标签，作品上架后用户在浏览、购买时产生的评论文本也会生成一些标签，商品标签的构建也只涉及到标签生成，没有产生权重。

3.2.4　候选集的生成

通过用户与商品的交互行为矩阵，我们最终得到了带有标签权重的候选集，具体算法是利用 Item-based 协同过滤算法生成候选集，定义 N_u 表示用户 u 之前

喜欢的主题集合，则用户 u 对主题 i 的偏好度根据式 3.1可计算得到。

$$p(u,i) = \sum_{j \in N(u)} r(u,j)s(i,j) \tag{3.1}$$

其中，$r_{u,j}$ 表示用户 u 对主题 j 的偏好度，$s_{i,j}$ 表示主题 i 和主题 j 之间的相似度。Item-based 协同过滤算法定义俩个主题之间的相似度由集中在这个俩个主题的用户行为数据计算得出。N_i 为看过主题 i 的用户集合，N_j 为为看过主题 j 的用户集合，因此，主题 i 和主题 j 的相似度计算公式为式 3.2。

$$s(i,j) = \frac{|N(i) \cap N(j)|}{\sqrt{|N(i) \parallel N(j)|}} \tag{3.2}$$

根据式 3.2可知，如果有很多用户同时看了主题 i 和主题 j，那么主题 i 和主题 j 之间的相似度就会很高，不幸的是，这也会导致所有热门主题之间的相似度都很高，导致推荐结果包含热门主题包过多。我们的解决思路是对热门主题包降权，同时控制热门主题所占比例。

3.2.5　排序

排序主要是对候选集的生成的标签权重做排序，但会加入一些倾斜因子，如用户活跃度、主题包的热度、经纬度等因子，最终根据标签权重 + 倾斜因子的排序得到推荐结果。

3.3　用户画像与用户兴趣探索

众所周知用户的需求是动态变化着的，不管是随着季节周期性变动，还是随着年龄发生非逆转变化，都意味着一些标签需要删除掉，一些标签需要加进来。用户画像的数据来源包括：原始的用户行为数据和用户兴趣探索模块，前者只是更新那些显而易见的标签，而后者负责在海量数据中挖掘出那些稍纵即逝的用户行为并准确分析用户的意图，而用户兴趣探索对活跃用户效果相对较好，并且针对小众主题包进行挖掘的效果很棒，可以明显提升推荐结果的多样性。除此之外，我们利用基于时间窗口的遗忘机制解决了新、旧用户兴趣的融合问题，时间窗口机制与自然遗忘规律相似，排前面的标签时效性最好，排后面的标签时效性差，将会被优先淘汰。通过设置时间窗口的大小、时间窗口的滑动速率，可以间接控制新、旧兴趣的比例。

3.4　用户画像与推荐系统

一个推荐系统要给用户提供个性化的、高效的和准确的推荐，则意味着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

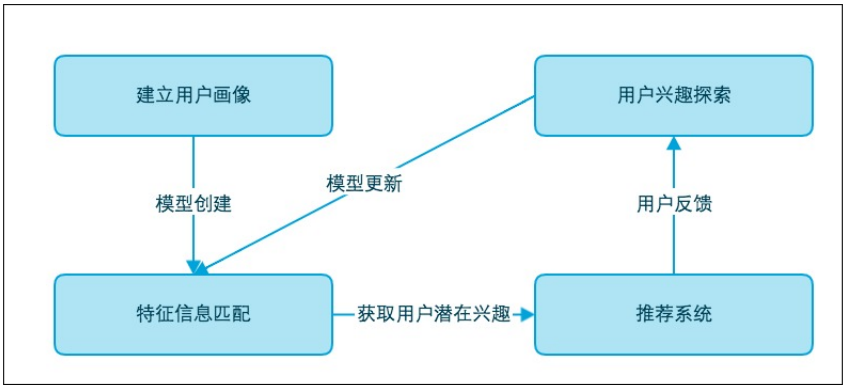

图 **3.2**　用户画像数据流图

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，这就是我们要引进用户画像的根本原因。用户画像模块和兴趣探索模块的关系如图 3.2所示。

一个好的用户画像需要有一个完整的标签体系，包括标签的数量、质量和粒度。其中标签的数量直接影响推荐系统的结果完整度，标签的质量直接影响推荐系统的精度，而标签的粒度会影响推荐系统的用户满意度。完善的标签体系更像一个金字塔，一级是最基本的概念标签，如动漫、运动等，数量被控制在几十个左右，二级标签是上层标签的扩展，如美少女动漫、搞笑动漫、篮球运动、足球运动，三级标签就是指具体化了的标签，如街头篮球、NBA 篮球、CBA 篮球等。对于活跃度高的用户标签倾向于下沉，推荐结果数据多、精度高，活跃度低的用户反之。对于一个新创建的用户画像，刚开始只包含最基本的用户人口信息，随着用户行为数据的累积会逐渐丰富起来。

3.5　本章小结

本章节主要介绍手机主题推荐系统的完整架构；之后详细说明了推荐系统技术，包括数据采集技术、用户画像的收集方式、商品标签的构建，最终是候选集的生成；最后是介绍了用户画像与用户兴趣探索的关系、用户画像与推荐系统的关系

第四章　用户画像模块

4.1　引言

用户画像建模的过程，就是标签量化和标签抽象的过程。小米手机主题用户画像是根据用户社会属性、生活习惯和消费行为等信息而抽象出的一个标签化的用户模型。构建用户画像的核心工作包括：1、给用户贴标签，而标签是通过对用户信息分析而来的高度精炼的特征标识。2、对每个用户标签赋予一定权重以代表该用户对该标签的偏好度。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。小米手机主题用户画像的标签是结构化的，最下层是用户基础信息，包括姓名、年龄、经纬度和职业等，中层是用户的兴趣标签，如正太控、动漫控和运动达人等，最上层是抽象标签，如高、中、低忠诚度用户，高、中、低价值用户等。

4.2　用户画像数据类型

在个性化服务的用户画像建模中，一个完整、成熟的用户画像是包含基础静态数据类型、基础行为数据类型和高维数据类型。

4.2.1　基础静态数据类型

当一个新用户注册时会填写人口基本信息，通过 json 格式从客户端传回服务器，如Listing 4.2。

Listing 4.1　基础静态数据类型

```
{"registerLog": {
  "userId": "001",
  "gender": "male",
  "profession": "student",
  "phone": "null",
  "borthday": "19860820",
  "isWeiboUser": "no",
  "isWeixinUser": "yes",
  "city": "北京市",
  "timestamp": "1453700393",
  "...": "..."
}}
```

有的用户会利用微信、微博提供的第三方免登陆 API，第三方数据可以用来交叉验证用户填写的基础信息数据。用户每次登陆时应用程序还会获得其手机品牌、操作系统等信息。因此，通过解析 server log 得到基础静态数据形式，如表 4.1所示。

图 **4.1**　用户画像标签示例图

表 **4.1**　用户-基础静态数据矩阵表

用户 id	性别	年龄	职业	电话号码	手机运营商	是否为微博用户	...
001	女	23	学生	13948572214	移动	是	...
002	男	30	学生	15811036703	移动	是	...
...	...	...	...	...	...	...	...

4.2.2　基础行为数据类型

用户行为数据是指用户的一些行为，包括购买、试用、浏览和评价等统计量，如Listing 4.2。

Listing 4.2　基础行为数据类型

```
{"actionLog": {
  "userId": "001"
  "actions": [{
     {"itermId": "0822"},
     {"actionType": "jumpIn"},
     {"stayTime": "32000"},
     {"clickNum": "2"},
     {"scrollNum": "5"},
     {"timestamp": "1453701393"},
     {"...": "..."}
  }]
}}
```

基础行为数据是基于用户行为数据得出的统计量，反映了用户的活跃度、消费能力和用户类型，基础行为数据形式如表 4.2。

表 4.2　用户-基础行为数据表

用户 id	购买	试用数	浏览	未支付订单数	活跃时间段	日浏览时长	...
001	2	7	118	0	20:00-22:00	120	...
002	0	3	7	1	13:00-14:00	60	...
...	...	...	...	...	...	...	...

4.2.3　高维数据类型

高维数据即用户的抽象标签，是用户画像模型从基础静态数据和基础行为数据统计、分析、抽象出来，用来衡量用户某一方面的价值，如用户信用是指是否有过作弊行为、退款次数过多等的综合评估，用户价值是指购买次数、单笔消费额和消费频率的综合评估，如表 4.3。

表 4.3　用户-高维数据表

用户 id	信用	价值	忠诚度	活跃度	价格敏感度	奖励敏感度	...
001	高	高	高	高	低	低	...
002	中	中	高	高	高	高	...
...	...	...	...	...	...	...	...

4.3　用户画像建模

用户画像建模的过程就是原始数据进过处理、分析得到可信度高的用户标签信息的过程，对于不同类型的用户数据其建模的侧重功能点也有所区别。

4.3.1　基础静态数据建模

用户基础静态数据的特点是数量不多，但在推荐系统中所占的权重较大，因此对其可信度要求较高，在对基础静态数据建模的时候主要实现俩个功能：根据上下文信息补全为空的标签和根据上下文信息校验已有的标签。

标签补全以用户性别标签为例，新用户注册时如未填写性别信息其值会默认设为 Null，方便用户画像建模时判断。主要思路是通过分析用户上下文信息，包括第三方登入数据、用户语音和头像获得用户真实的性别，如以上方法都未成功获取用户性别，程序会利用线性回归算法挖掘出一个最有可能的性别标签值，代码如Listing 4.3。

Listing 4.3　标签补全算法

```
public String getUserGender(String log) {
    Gson gson = new Gson();
    UserProfile userProfile = gson.fromJson(log,
        UserProfile.class);

    if (userProfile.gender != null) {
```

```
            return userProfile.gender;
        }

        String useId = userProfile.useId;
        //通过第三方应用登陆数据得到用户信息
        UserProfile thirdPartUP =
            gson.fromJson(getThirdPartUserInfo(useId),
            UserProfile.class);
        if (thirdPartUP.gender != null) {
            return thirdPartUP.gender;
        }

        //通过分析用户语音数据得到用户信息
        UserProfile voiceUP =
            gson.fromJson(getUserVoiceUserInfo(useId),
            UserProfile.class);
        if (voiceUP.gender != null) {
            return voiceUP.gender;
        }

        //通过线性回归算法挖掘出用户信息
        UserProfile lrUP =
            gson.fromJson(getLinearRegressionUserInfo(useId),
            UserProfile.class);
        return lrUP.gender;

    }
```

标签校验是指虽然相关信息已经被填写，但程序认为其值具有随意性，需要根据上下文信息加以确认并校验，标签校验由于考虑的因素较多导致计算量大，应用场景较少。用户性别标签校验，代码如Listing 4.4。

Listing 4.4　标签校验算法

```
    public String getRightUserGender(String log) {
    int[] count = {0, 0};
    Gson gson = new Gson();
    UserProfile userProfile = gson.fromJson(log,
        UserProfile.class);

    if (userProfile.gender != null) {
        if (userProfile.gender.equals("male")) {
            count[0]++;
        } else {
            count[1]++;
        }
    }

    String useId = userProfile.useId;
    UserProfile thirdPartUP =
        gson.fromJson(getThirdPartUserInfo(useId),
        UserProfile.class);
    if (thirdPartUP.gender != null) {
        if (thirdPartUP.gender.equals("male")) {
```

```
            count[0]++;
        } else {
            count[1]++;
        }
    }

    UserProfile voiceUP =
        gson.fromJson(getUserVoiceUserInfo(useId),
        UserProfile.class);
    if (voiceUP.gender != null) {
        if (voiceUP.gender.equals("male")) {
            count[0]++;
        } else {
            count[1]++;
        }
    }

    UserProfile lrUP =
        gson.fromJson(getLinearRegressionUserInfo(useId),
        UserProfile.class);
    if (lrUP.gender.equals("male")) {
        count[0]++;
    } else {
        count[1]++;
    }
    if (count[0] >= count[1]) {
        return "male";
    } else {
        return "female";
    }
}
```

4.3.2　基础行为数据建模

基础行为数据建模更新频率较快，计算量较大，因此采用离线方式利用 sql 语句从 hive 表中得出用户在一段时间区间内特定行为的统计数据。需要注意一些用户行为的延迟性，如购买行为，从下单到支付成功可能跨越若干天，因此约定订单量以支付时间为准，有时候遇到网络故障相同订单会被用户提交多次，需要利用 distinct 做去重操作。统计特定用户某段时间的订单量的脚本，如Listing 4.5。

Listing 4.5　基础行为数据建模脚本

```
set hiveconf:ymdwithline=2016-04-06;
set hiveconf:userId=525108009;

select count(distinct a.order_id) score
from theme_dw.dw_v_order_base
where concat_ws('-',year,month,day) between
    date_sub('${hiveconf:ymdwithline}',5) and
    '${hiveconf:ymdwithline}'
and userId='${hiveconf:userId}'
and finish_time like '${hiveconf:ymdwithline}%'
```

4.3.3　高维数据建模

高维数据建模的数据来源包括基础静态数据、基础行为数据，数据类型包括累计量和趋势量，累计量包括用户浏览总数、用户购买总数等，趋势量是指用户最近登录时间、最近购买时间等，利用数据挖掘分类算法得出一个训练模型，需要注意的是用户行为类型、发生时间和发生位置会影响模型的权重计算，即 weight = (行为类型 + 时间上下文 + 空间上下文) × 时间衰减因子。其中，用户行为类型包括浏览、购买、搜索、评论、购买、点击赞和收藏等，我们定义购买权重计为 5，而浏览仅仅为 1。空间上下文是指用户跳转入口方式，我们定义搜索入口权重 3，排行榜入口为 2。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。时间衰减因子根据半衰期公式得出，公式如式 4.1，其中 T 取值为 1，t 为行为发生时间距离当前时间的天数。

$$score = (\frac{1}{2})^{(t/T)} \tag{4.1}$$

以用户活跃度为例，由于日活跃变动过大，月活跃过于滞后，因此按周统计，模型选择线性回归算法，模型输入为基础静态数据、基础行为数据，模型输出为一个 int 型整数，值为 [1，2，3]，分别对应不活跃、较活跃、活跃。代码，代码如Listing 4.6。

Listing 4.6　高维数据建模算法

```
public int getActivityScore(String userId) throws Exception {
    String userBaseInfo = getUserBaseInfo(userId);
    String userActionLog = getUserActionLog(userId);
    Gson gson = new Gson();
    String score = getLinearRegressionActivityScore(
      gson.fromJson(userBaseInfo, UserProfile.class),
      gson.fromJson(userActionLog, UserActions.class));
    double activityScore = Double.parseDouble(score);
    if (activityScore >= 66) {
       return 3;
    } else if (activityScore >= 33) {
       return 2;
    } else {
       return 1;
    }
}
```

4.4　实验与分析

本节的研究目标是如何利用用户画像给新注册用户做出准确的 TopN 推荐并提升用户留存率。严谨的 A/B 测试流程应该先分析 A/A 测试，得出实验本身自带的误差，然后利用这个误差因子修正 A/B 测试结果，最终得到统计结果，但 A/A 测试会严重拖慢节奏，所以本节只重点介绍 A/B 测试。实验目标用户人群

为北京地区，所有从 2015 年 9 月 1 号到 2015 年 9 月 7 号这段时间注册的用户，去除用户注册信息不完整后用户数为 20 万，对照组和测试组 a 和测试组 b 人群比例为 33.3：33.3：33.4，对照组用户人群的推荐结果没有利用用户画像，测试组 a 人群的推荐结果只包含热门主题，测试组 b 人群的推荐结果利用了用户画像，对照组和测试组 b 的推荐候选集为全部主题，测试组 a 的推荐候选集为 Top 20% 热度的主题。实验从 2015 年 9 月 8 号开始到 2015 年 10 月 8 号结束，周期为一个月。

4.4.1 评测指标

本节使用线上 A/B 测试方案，利用用户留存率来评测推荐系统应对冷启动问题的效果。用户留存数是指在某段时间开始使用应用，经过一段时间后仍然继续使用应用的用户，用户留存率是指用户留存数占当时新增用户的比例，计算单位取天，用户留存率研究对象为新注册用户，反映了推荐系统的转换能力，即由初期的不稳定的用户转化为活跃、稳定、忠诚的用户。

4.4.2 对比模型

图 4.2展示了不同模型的实验结果。我们对比了单纯的推荐模型、推荐热门商品的简单推荐模型和融合了用户画像的推荐模型在新注册用户数据集上的用户留存率。图中，横坐标是时间变量，单位为天，纵坐标是用户留存率，每一条曲线代表了一个模型的用户留存率随时间变化的曲线。通过观察曲线可以发现用户留存率随时间流动呈指数分布，头三天就流失了约 90% 的新用户，从第四天用户留存率开始停留在一个比较稳定的阈值，为减少误差头三天的数据不记入统计数据，统计结果显示，融合了用户画像的推荐模型的留存率是 10.3%，比推荐热门商品的简单推荐模型的留存率 8.19% 要高，相对于单纯的推荐模型的留存率 5.76% 同样也高。由此可见用户画像能够很好的解决冷启动问题并得到较高的新注册用户留存率。

4.5 本章小结

本章首先介绍了用户画像数据类型，包括基础静态数据类型、基础行为数据类型；之后介绍了用户画像建模，包括基础静态数据建模、基础行为数据建模和高维数据建模；最后是实验与分析，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，将在下一章节详细介绍。

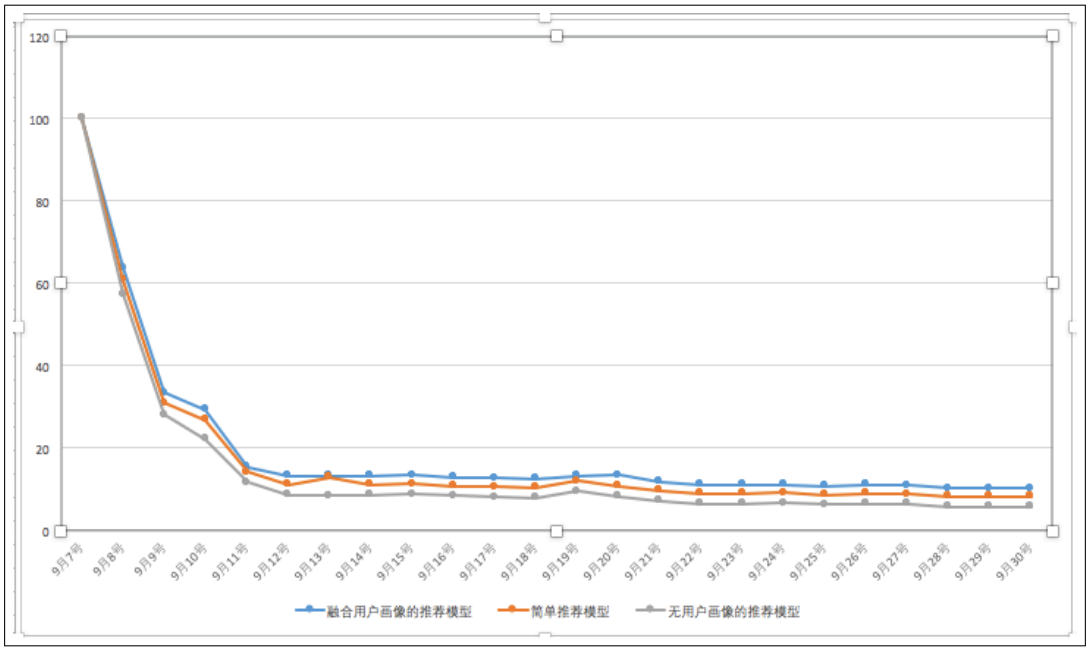

图 **4.2**　新用户留存率实验对比图

第五章　用户兴趣探索

5.1　引言

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数据量化后得出的。但就小米主题市场而言，只有那些热门主题才有足够多的数据可以挖掘，大部分主题只有很少的数据可供分析、挖掘，因此，需要一种方法对用户行为做针对性挖掘，利用少量数据获得出用户真正的偏好度，最终提升商品的销售量，这是问题之一。现有的推荐算法注重用户静态属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，这是问题之二。用户兴趣探索模块可以很好的解决此类问题。

探索用户兴趣的数据来源包括用户行为数据、用户画像和商品特征表。用户画像包括用户基本信息和兴趣标签等，商品特征表包括分类、属性标签等，过程分为几个步骤：首先，利用用户历史行为 (评论，停留时长，评分，点赞，购买等) 量化用户满意度，然后利用用户兴趣特征向量与商品特征矩阵得出相关分数，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。这里涉及到的概念包括用户满意度的量化、用户和商品的关联度、商品属性标签的长尾性，下文会一一给出详细说明。

5.2　用户行为数据的存储和处理

手机主题用户行为数据的特点包括：1、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。2、用户规模增长快，月新注册用户达 10 万数量级。3、每个用户的行为数量较小，即使是活跃用户，每天最多也只产生上百条行为记录。4、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。5、用户行为数据格式不规整，字段丢失率较高。

5.2.1　数据预处理

数据预处理是数据挖掘过程中一个重要步骤，主要工作包括字段去重、无效日志过滤、多表字段的连接等。如统计 2015 年 09 月 06 号 userId 为 001 的投诉数，数据预处理过程如Listing 5.1。

Listing 5.1　数据预处理脚本

```
set hiveconf:ymdwithline=2015-09-06;
set hiveconf:metric=complaint_order_num;
```

```
set hiveconf:user_id=001;

select '${hiveconf:metric}' as metric, count(a.order_id) as
   score
from (
   //去重
   select distinct order_id
   from theme.dw_v_order_base
   //以时间范围date_sub('${hiveconf:ymdwithline}',5) and
      '${hiveconf:ymdwithline}'为条件过滤掉不符合条件的订单
   where concat_ws('-',year,month,day) between
      date_sub('${hiveconf:ymdwithline}',5) and
      '${hiveconf:ymdwithline}'
   //无效订单过滤
   and order_id!=null
   //以用户id为条件过滤掉其他订单
   and user_id=${hiveconf:user_id}
) a
inner join (
   //order_id 字段去重
   select distinct order_id
   from theme.g_comment_complaint
   //type = 3表示用户投诉
   where concat_ws('-',year,month,day) =
      '${hiveconf:ymdwithline}' and type = 3
   //多表字段的连接，如果有一个表有投诉记录，就算一次投诉。
   union
   select distinct order_id
   from theme.dwd_kefu_phone_complaint
   where concat_ws('-',year,month,day) =
      '${hiveconf:ymdwithline}'
) b
on a.order_id = b.order_id
group by metric;
```

5.3　用户兴趣探索模型

用户兴趣探索主要功能模块包括：1，兴趣标签探测，在分析用户行为数据时，如果某些主题标签和用户相关度很低，那么这些标签会作为标签探索候选集。2，长尾标签提取，基于小众标签集，按照某种规则筛选出目标标签。3，用户满意度量化，根据用户所有对某一个主题的行为数据得出这个用户对这个主题的满意度。4，标签权重的更新，不管是不是一次成功的兴趣标签探索，都要对用户画像标签的权重做更新，更新算法利用了线性衰减思想。本章首先介绍一些基本概念，包括实体域、用户行为和用户满意度等。然后详细介绍用户兴趣探索功能模块的实现。

5.3.1 基本概念概述

实体域。如果一个模型是基于分析用户行为得出用户兴趣时，实体就是这个行为针对的对象。不同实体通过标签关联起来，对于手机主题应用市场来说，实体域还包括所有的背景图片，铃声，闹铃等。

用户行为。包括浏览，点击，下载，试用，购买，评论。本文所指的用户行为都是指用户在某手机主题上的行为。

用户满意度。用户满意度是指根据用户作用在主题上的不同行为动作及其属性值，反推得到的用户偏好度。

小众标签集。小众标签集是指用户偏好频率低的主题标签的集合，需要补充是，长尾标签和小众标签大多数是一致的，只是长尾标签针对商品而言，小众标签针对用户而言。

5.3.2 兴趣标签探测功能模块

首先候选标签是与用户相关度低的标签，如用户 001 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，通过计算发现汽车标签对于用户 001 是从未遇到过的标签，即相关度为 0，于是汽车标签将会是潜在的探索标签。事实上用户兴趣探索过程可以在很短的时间内完成，基于 Hive + HDFS 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。

5.3.3 长尾标签抽取功能模块

首先需要介绍标签集中度 (tagFocus) 和标签热度 (tagPopular)，标签集中度针对商品而言，标签热度针对用户而言。

标签集中度。如果某一类主题包集合中包含某兴趣标签的个数为 tagInThemeNum，而其它类包含的总数为 tagInOtherNum，当 tagInThemeNum 大的时候，就说明其集中度高。实际上，如果一个标签在同一类主题集合中频繁出现，则说明该标签能够很好代表这类主题集合的特征，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，标签集中度公式如式 5.1，我们很容易发现，如果一个标签只出现若干主题包，我们通过它就容易定位搜索目标，因此其权重也应该大。反之如果一个词在大量主题包中出现，其权重取较小为好。

$$\mathrm{tagFocus} = \frac{|\mathrm{tagInThemeNum}|}{|\mathrm{tagInThemeNum} + \mathrm{tagInOtherNum}|} \tag{5.1}$$

标签热度。标签热度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。标签热度不是越大越好，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。

长尾标签定义为集中度和热度之比大于一个给定阈值的标签，且本身为小众标签。代码如Listing 5.2。

Listing 5.2　长尾标签抽取算法

```
public Set<String> getLongTailTags(String userId, String
    itemId) throws Exception {
    Set<String> out = new HashSet<>();
    //获取所有小众标签
    HashSet<String> longTailTags = getColdTags();
    //获取所有当前用户画像没有的标签
    Set<String> rawTags = tagExplore(userId, itemId);
    for (String tag : rawTags) {
        if (!longTailTags.contains(tag)) {
            continue;
        }

        //获取标签的集中度
        long tagFocusScore = getTagFocusScore(tag);
        //获取标签的热度
        long tagPopularScore = getTagPopularScore(tag);
        if (tagFocusScore / tagPopularScore <= threshold) {
            continue;
        } else {
            out.add(tag);
        }
    }

    return out;
}
```

5.3.4　用户满意度量化功能模块

表 5.1列举的用户行为包含了部分关键的行为类型，通过将不同行为反映为用户喜好的不同并进行加权累积，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，通过数据挖掘算法滤掉可能的噪音，这样使分析更加精确。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，就需要进行归一化处理使得数据取值在 [0，10] 范围中。

表 5.1　用户行为权重对应表

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	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，页面停留时长可能会很长	1

5.4　用户画像和用户兴趣探索的融合

随着时间的变化，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，通过调节时间窗口大小和更新周期，体现不同的时效性。我们可以把用户画像权重想象成一个自然冷却的过程：任一时刻，用户画像中的标签都有一个当前温度，温度最高的标签权重值最高；如果该用户对某主题发生了一些正向行为，如点赞，该文章包含的标签在用户画像中的温度就会上升，否则温度下降；随着时间流逝，所有标签的温度都逐渐冷却，通过时间窗口向前滑动实现。

这样假设的意义在于我们可以照搬物理学的牛顿冷却定律 (Newton's Law of Cooling)，建立标签权重与时间之间的函数关系：本期分数 = 上期分数 - 冷却系数 * 间隔天数。其中，冷却系数决定了标签融合的更新率，如果想放慢更新率，

冷却系数就取一个较小的值，否则就取一个较大的值。

标签权重的线性衰减算法结合了手机主题用户长期兴趣和短期兴趣，根据时间因素权重自动进行衰减，能准确反映用户兴趣的变化趋势。该模型是指用户对兴趣标签的权重仅代表评价当时的兴趣度，随着时间的推移，用户对该标签的权重将规律性地自动衰减，当权重衰减到 0 时，标签将被淘汰。

5.5　实验与分析

5.5.1　数据集准备

实验中我们利用 2015 年 9 月到 2015 年 10 月的用户行为数据和所有关联的手机主题包。这个数据集包含了 110739 个用户在这段时间对主题包的标签行为，数据集中包含了 8936 个主题包。该数据集每行是一条记录，每条记录包含的信息有：用户 ID，主题 ID，行为类型，行为值，日期，每一条记录代表了某个用户在某个时间点对某个主题包进行了某种行为。保证数据集具有一定的稠密程度，我们去除了用户行为记录少于 10 条的所有用户，最终用户集包含 10646 个用户，2033600 条用户行为记录。

5.5.2　评测指标

使用线上 A/B 测试方案，利用点击购买转化率来评测推荐系统应对马太效应的效果。根据统计我们知道 20% 的热门商品在占了 80% 的曝光机会的同时却只占 50% 的销售量，因为虽然热门商品销量很好但其整体数量偏少，很难满足大多数消费者的需求。相反，占据 80% 的小众商品虽然曝光率低，但凭借其庞大数量和多样性，能满足多数消费者的需求。因此如果适度对小众商品增加曝光机就会大幅提升商品的销售量，这里我们选用的实验指标为商品的点击购买转换率。

5.5.3　对比模型

无兴趣探索模块的推荐模型，在实验中作为基准模型。对照模型包括融合了兴趣探索模块的推荐模型和推荐热门商品的简单推荐模型。

5.5.4　实验结果

我们对比了无兴趣探索模块的推荐模型、推荐热门商品的简单推荐模型和融合了兴趣探索模块的推荐模型在实验期间的有过至少一次销售记录的商品数 itemCount。图 5.1展示了不同模型的实验结果。图中，横坐标是时间变量，单位为天，纵坐标是 itemCount，每一条曲线代表了一个模型的 itemCount 随时间变化的曲线。通过观察曲线可知，融合了兴趣探索模块的推荐模型的 itemCount 月平均数是 3136，推荐热门商品的简单推荐模型的 itemCount 月平均数是 1935，无

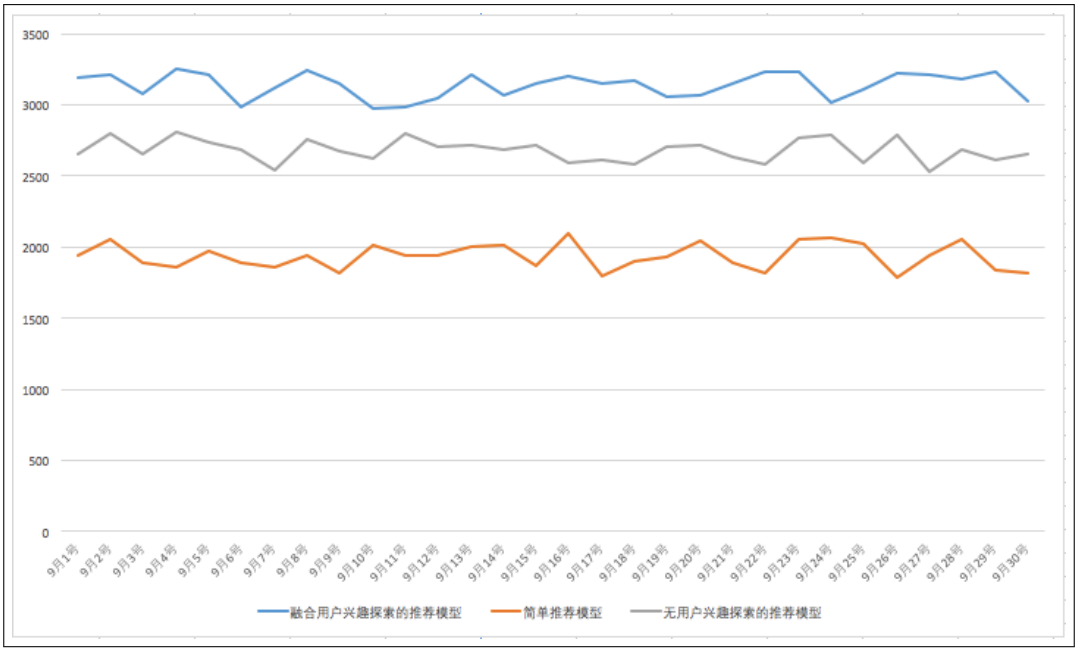

图 **5.1**　推荐多样性实验对比图

兴趣探索模块的推荐模型的 itemCount 月平均数是 2679。实验说明融合了用户兴趣探索的推荐模型相对其他模型有更好的多样性。

我们对比了无兴趣探索模块的推荐模型、推荐热门商品的简单推荐模型和融合了兴趣探索模块的推荐模型在实验期间的点击购买转化率。图 5.2展示了不同模型的实验结果。图中，横坐标是时间变量，单位为天，纵坐标是点击购买转化率，每一条曲线代表了一个模型的点击购买转化率随时间变化的曲线。实验结果显示，融合了兴趣探索模块的推荐模型相对其他模型有更高的点击购买转化率。融合了兴趣探索模块的推荐模型的平均点击购买转化率是 32.74%，比推荐热门商品的简单推荐模型的平均点击购买转化率 9.63% 要高，相对于无兴趣探索模块的推荐模型的平均点击购买转化率 17.54% 也高了不少。由此可见用户兴趣探索能够很好的提升点击购买转化率。

5.6　本章小结

本章首先介绍了用户行为数据特点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模块的组成内容，包括兴趣标签探测功能模块、长尾标签抽取功能模块和用户满意度量化功能模块，之后介绍了用户画像和用户兴趣探索的融合，最后给出了用户兴趣探索实验结果，即用户兴趣探索模块可以为推荐系统带来更好的多样性和更高的购买转化率。

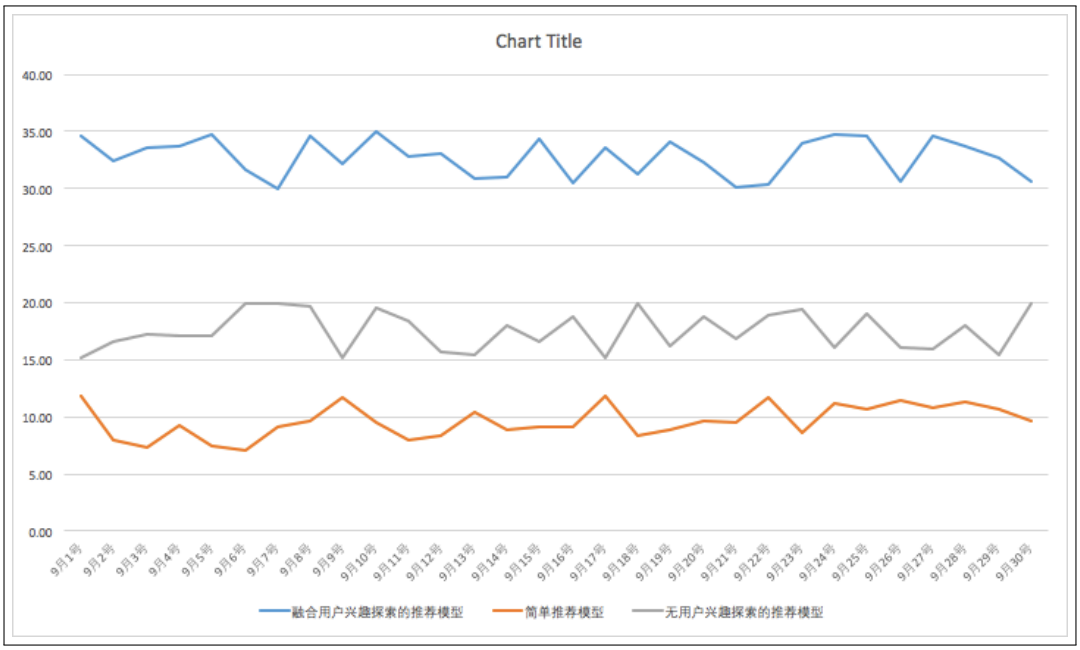

图 **5.2**　转化率实验对比图

第六章　结束语

如果说过去的五年是推荐系统大行其道的时间，那么个性化将成为推荐技术未来五年中最重要的革新之一。一个好的推荐系统需要满足的目标有：能实时提供个性推荐服务，推荐结果满足新颖性、惊喜性和长尾性，而且推荐结果必须足够及时，这样才能在用户浏览之后、购买之前就获得推荐服务，笔者通过引入用户画像模块和用户兴趣探索模块来达到以上目标。

本文介绍的推荐系统由三部分模块组成：用户画像模块，用户兴趣探索模块、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。用户兴趣探索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、计算和排序。

推荐系统的主要方法是通过分析用户的过去预测未来，因此基于用户画像模型的研究是一个很好的突破口，因为用户画像天然支持用户历史信息的存储、查询，同时对用户兴趣探索的研究，无论是从促进用户画像模型的角度出发，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：

- 设计了用户画像模型：通过对基础静态数据、基础行为数据类型和高维数据类型进行建模，得出较为完整、准确的兴趣标签，解决新用户的冷启动问题，提升了推荐系统的精度。

- 设计了用户兴趣探索模型：通过量化用户满意度和计算用户和商品的相关度，实现了用户小众兴趣的探索功能，提升了推荐系统的动态推荐效果。

- 利用线性衰减算法成功融合用户长期兴趣和短期兴趣：本文在研究用户画像建模和用户兴趣探索的基础上，结合电子商务用户兴趣偏好变化频繁的特点，提出了基于线性衰减的用户兴趣融合模型。标签权重的线性衰减算法结合了手机主题用户长期兴趣和短期兴趣，能准确反映用户兴趣的变化趋势。

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。本人认为有待解决的问题有:

- 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：本文的所有工作的评估集中在点击购买转换率上。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，从而给用户推荐热门商品，这并不是一个好的推荐。因为热门商品本身的转化率就很高，这里涉及到了推荐系统的长尾性，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，是笔者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、多样性和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提出宝贵的批评意见和建议。

参考文献

[1] Gediminas Adomavicius, Alexander Tuzhilin. 1999. *User Profiling in Personalization Applications through Rule Discovery and Validation[J]*. ACM, 377-381.

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

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

[4] Han Jiawei, Kamber, Micheline. 2001. *Data mining: concepts and techniques[C]*. Morgan Kaufmann, 5.

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

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

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

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

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

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

[11] Speier Cheri, Valacich Joseph, Vessey Iris. 1999. *The Influence of Task Interruption on Individual Decision Making: An Information Overload Perspective[J]*. Decision Sciences.

[12] Liu Yu, Li Weijia, Yao Yuan, et al. 2011. *An Infrastructure for Personalized Service System Based on Web2.0 and Data Mining[J]*. International Conference on Intelligent Computing and Information Science.

[13] Shumeet Baluja, Rohan Seth, D Sivakumar, et al. 2008. *Video suggestion and discovery for youtube: taking random walks through the view graph[J]*. In Proceeding of the 17th international conference on World Wide Web, pages 895–904.

[14] Robert M, Bell, Yehuda Koren. 2007. *Lessons from the netflix prize challenge[J]*. SIGKDD Explor. Newsl, 9:75–79.

[15] Sia KC, Zhu SChi, Hino Tseng. 2006. *Capturing User Interests by Both Exploitation and Exploration[C]*. Technical report, NEC Labs America.

[16] Thomas Hofmann and Jan Puzicha. 1999. *Latent class models for collaborative filtering[J]*. In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, pages 688–693.

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

[18] Murray CG, Richardson M, Bilenko M, et al. 2008. *Talking the talk vs. walking the walk: salience of information needs in querying vs browsing[J]*. Proc. ACM SIGIR, 705-706.

[19] Eugene Agichtein, Eric Brill, Susan Dumais. 2006. *Improving web search ranking by incorporating user behavior information[J]*. Proc. SIGIR, 19-26.

[20] Broder A. 2002. *A taxonomy of Web search[J]*. ACM SIGIR Forum, 3-10.

[21] Budzik J, Hammond K. 1999. *Watson: anticipating and contextualizing information needs[J]*. Proc ASIS, 727-740.

[22] Yehuda Koren. 2009. *Collaborative filtering with temporal dynamics[J]*. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 447–456.

[23] Robin Burke. November 2002. *Hybrid recommender systems: Survey and experiments[J]*. User Modeling and User-Adapted Interaction, 12:331–370.

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

[25] K Yoshii. 2006. *Hybrid Collaborative and Content-Based Music Recommendation Using Probabilistic Model with Latent User Preferences [C]*. In: Proceedings of the International Conference on Music Information Retrieval.

[26] Jonathan L, Herlocker, Joseph A, et al. 2004. *Evaluating collaborative filtering recommender systems[C]*. ACM Trans, 22:5–53.
[27] Andrew I Schein, Alexandrin Popescul, Lyle H Ungar, et al. 2002. *Methods and Metrics for Cold-Start Recommendations[C]*. New York City, New York: ACM. 253–260.
[28] Siibak, Andra. 2007. *Casanovas of the Virtual World. How Boys Present Themselves on Dating Websites[J]*. Young People at the Crossroads: 5th International Conference on Youth Research. pages 83–91.
[29] Nabeth, Thierry. 2006. *Understanding the Identity Concept in the Context of Digital Social Environments[J]*. FIDIS Deliverables. 2. FIDIS. pp. 74–91.
[30] Marcus Bernd, Machilek Franz, Schütz Astrid. 2006. *Personality in cyberspace: Personal web sites as media for personality expressions and impressions[J]*. Journal of Personality and Social Psychology. 90 (6): 1014–1031.
[31] Adams, Suellen. 2005. *Information Behavior and the Formation and Maintenance of Peer Cultures in Massive Multiplayer Online Roleplaying Games: a Case Study of City of Heroes[J]*. DiGRA: Changing Views - Worlds in Play.
[32] Suler, John. 2004. *The Online Disinhibition Effect[J]*. CyberPsychology & Behavior. 7 (3): 321–326.
[33] Kohavi Ron, Longbotham Roger. 2015. *Online Controlled Experiments and A/B Tests[C]*. In Sammut.
[34] Maja Kabiljo, Aleksandar Ilic. 2015. *Recommending items to more than a billion people[DB/OL]*. https://code.facebook.com/posts/861999383875667/recommending-items-to-more-than-a-billion-people.
[35] Gediminas Adomavicius, Alexander Tuzhilin. 2005. *Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions[J]*. IEEE transactions on knowledge and data engineering.
[36] Daniel Lemire, Anna Maclachlan. 2005 *Slope One Predictors for Online Rating-Based Collaborative Filtering[J]*. In SIAM Data Mining (SDM'05), Newport Beach, California.

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和袁森博士，让我成为大数据挖掘工程师的梦想又更近了一步。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

2018 年 9 月 12 日