分类号_____ 密 级_____
U D C _____ 学校代码_____10689_____

雲南財經大學

YUNNAN UNIVERSITY OF FINANCE AND ECONOMICS

硕士学位论文

基于用户画像与协同过滤的混合推荐系统研究

Hybrid Recommender System Based On User Profile and Collaborative Filtering

姓 名：胡兆山
导 师（职称）：赵昆（教授）
申 请 学 位 类别：管理学硕士
专 业：管理科学与工程
研 究 方 向：推荐系统
学院（中心、所）：信息学院

论文完成时间：2019 年 6 月 6 日

学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：胡兆山　日期：2019年 6月 6日

学位论文版权使用授权书

本人完全了解云南财经大学有关收集、保存、使用学位论文的规定，即：按照有关要求提交学位论文的印刷本和电子版本；学校有权保存并向国家有关部门或机构送交论文和论文电子版，允许学位论文被查阅或借阅；学校可以公布学位论文的全部或部分内容，可以采用影印、缩印或其它复制手段保存、汇编、发表学位论文；授权学校将学位论文的全文或部分内容编入、提供有关数据库进行检索。

（保密的学位论文在解密后遵循此规定）

论文作者签名：胡兆山　导师签名：

日期：2019年 6月 6日　日期：2019年 6月 6日

摘要

随着互联网的迅猛发展，信息呈指数级增长。面对如此庞杂的信息，帮助用户快速、有效地获取需要的信息成为一项具有挑战性的工作，这也是当前学术界研究的热点问题。学术界和业界针对信息过载问题开展了大量的研究和实践工作，提出了多种个性化解决方案，希望为用户提供符合其需求的信息。

而推荐系统就是解决此问题的有效方法，它是一种个性化信息服务系统，通过推荐算法实现有针对性的个性化推荐。而协同过滤是在推荐系统中应用最成功和最广泛的一种推荐算法。协同过滤通过用户评价过项目的历史评分数据来预测未知项目的用户评分，但随着互联网不断发展与普及，像淘宝、抖音、微信这样的大型平台用户数都已过亿，而且其中的项目资源也增长至千万级。但大部分用户仅仅对个别项目进行过评价，而且用户评价过的项目也不尽相同，因此用来进行预测评分的用户-项目评分矩阵变得极端稀疏。很难根据这些数据找到偏好真正相近的用户，传统基于协同过滤方法所生成的推荐质量越来越差，稀疏性问题已成为影响推荐效果的关键问题，因此需要新的方法来解决这个问题。

而用户画像的产生原因是为了准确、高效地分析用户的偏好信息，将其与协同过滤相结合可能会改善这一问题。并且有调查显示 80%的用户愿意向平台提供自己的姓名、年龄、性别等基本信息。因此笔者就对如何使用用户基本信息构建用户画像、如何将用户画像融入到协同过滤进行了研究。提出了一种构建用户画像的用户信息度量模型，一种基于用户画像与协同过滤的混合推荐模型 UPCF。

本文在实验阶段考察了相似度量模型（PCC、COS、ADCOS）、评分预测算法（DFM、WS）、不同用户特征对混合推荐模型 UPCF 的影响，并在其中选出了使 UPCF 模型表现最优的组合。然后将此 UPCF 模型分别与 UBCF、SM 模型在 MAE、Precision、Recall、F1 四种评价指标下进行了比较，实验结果表明本文提出的方法在这四种评价指标上，确实要优于传统的 UBCF、改进的 SM 方法，证明了本文提出的方法提高了协同过滤推荐算法的预测准确性，缓解数据稀疏性问题带来的影响。

关键词：推荐系统；协同过滤；用户画像；混合推荐；数据稀疏问题

Abstract

With the rapid development of the Internet, information has grown exponentially. Facing the ocean of information, how to quickly and effectively help users to obtain the information they really need becomes a challenging task and it is also a hot issue in academic research. Academia and the industry have carried out a lot of research and practice on the problem of information overload, and have proposed a variety of solutions for information personalized, hoping to provide users with information that meets their needs.

The recommender system is one of the most effective ways to solve this problem. Recommender system which implements personalized recommendation through recommendation algorithms is a personalized information service system. And collaborative filtering is one of the most successful and widely used techniques in the recommender system. Collaborative filtering predicts user’s ratings of unknown projects by user’s data of the historical ratings of the projects. However, with the continuous development of the Internet, the number of users and projects which on large platforms like TaoBao has reached nearly 100 million levels and is still increasing. The user's rating of the project is only a small part of the total number of items, resulting in extremely sparse of user ratings matrix which is used to predict unknown ratings. And it is difficult to find users with similar preferences for target user by using sparse ratings matrix, then the quality of recommendations produced by ratings matrix is getting worse. The sparsity of the rating matrix has become a key issue affecting recommendations based on collaborative recommendations. Therefor, we needed a new method to solve this problem.

User profile can accurately and efficiently analyze the user's preference information, so combining it with collaborative filtering may improve the problem of sparsity. And some surveys show that 80% of users are willing to provide their basic information such as name, age and gender for platform. Therefor, author has researched how to

construct user profile using user basic information and how to integrate user profile into collaborative filtering. And author proposed a user information measurement model for constructing user portraits, a hybrid recommendation model UPCF based on user profile and collaborative filtering.

In the experimental stage, the effects of similarity measurement models (PCC, COS, ADCOS), scoring prediction algorithm (DFM, WS), and user characteristics on the hybrid recommendation model UPCF are investigated, and the optimal combination of UPCF model is selected. Then the UPCF model is compared with UBCF, IBCF and SM models under the four evaluation indexes of MAE, Precision, Recall and F1. The experimental results show that the proposed method is superior to the traditional methods UBCF, IBCF and SM. The result proves that the proposed method improves the prediction accuracy of the collaborative filtering recommendation algorithm and mitigates the impact of data sparsity.

Keywords: recommender system; collaborative filtering; user profile; hybrid recommendation; sparsity of the rating matrix

目录

第一章 绪论

第一节 研究背景

目前，不管是企业的运转还是个人的生活，都受到互联网深刻的影响。随着 Web 2.0 使用的普及，信息呈指数级增长，重要的信息和不重要的信息之间的界限变得很容易消失。人们难以在信息的海洋中，找到自己喜欢、或有价值的信息，出现了信息过载问题。因此帮助用户快速有效的获取真正需要的信息成为一项富有挑战性的工作，也是当前学术界研究的热点问题。

而推荐系统就是解决此问题有效方法中的一种，它一种个性化信息服务系统，可借助用户建模技术对用户的长期需求进行描述。并根据用户模型，通过推荐算法实现有针对性的个性化推荐。推荐系统在电子商务、音乐、电影、新闻等领域得到了广泛的应用，并成为公认的最有前途的个性化技术发展方向。而协同过滤是在推荐系统中应用最成功和最广泛的一种技术，它通过用户评价过项目的历史评分数据来预测未知项目的用户评分，数据获取简单。协同过滤以偏好相近的用户为参考，使用偏好相近用户的历史评分数据产生推荐。与人们在决策之前习惯去询问、参考他人意见的心态相符合，推荐效果良好。

但随着互联网不断发展，像淘宝、亚马逊这样大型平台的用户数过亿和项目数也近千万。但大部分用户仅仅对个别项目进行过评价，而且用户评价过的项目也不尽相同，造成用户项目评分矩阵极端稀疏。现有方法很难通过如此稀疏的数据找到与目标用户偏好真正相近的用户，通过评分数据预测产生的推荐质量越来越差。因此需要新的方法来解决这个问题，而用户画像的产生原因是为了准确、高效地分析用户的偏好信息，将其与协同过滤相结合可能会改善这一问题。用户画像即：用户信息标签化。基于协同过滤的推荐系统融入用户画像后，就能够在评分数据的基础上使用关于用户偏好的附加信息来进行推荐。

第二节 研究意义

首先，推荐系统能帮助用户快速发现符合其偏好的高质量信息，来提升用

户体验，增加用户使用时间。并且能够减少用户因浏览到重复或者不符合其偏好的信息带来的不利影响。而协同过滤在推荐系统中是应用最广泛、成功的一种方法。在保障推荐系统效率的前提下，对基于协同过滤的推荐系统的改进具有巨大的价值和意义。本文的研究，以将用户画像引入基于协同过滤的推荐系统，提升推荐系统的准确率为目标。对基于协同过滤的推荐系统的改进具有直接的积极意义。

其次，用户画像能够提供显示用户偏好的信息，帮助推荐系统更加精准的找到偏好相近的用户群体以及用户需求等更为广泛的反馈信息。本文对推荐系统领域用户画像构建进行研究，将其融合到基于协同过滤的推荐系统中。研究思路较为新颖独特，相关研究成果将丰富推荐系统与用户画像领域现有的理论和方法。

第三节　国内外研究现状

一、推荐系统

20 世纪 90 年代提出了推荐系统的概念，至今已近 30 年。期间推荐系统的研究和应用得到了飞速的发展。推荐系统是一种特殊的信息过滤系统，通过分析用户偏好信息，在项目集中找到可能会符合其偏好的项目，然后主动向用户提供推荐[1]。在推荐系统中，“项目”被定义为系统为用户推荐的物品、信息等用户所需要的资源。像在淘宝网中项目为商品，网易云音乐中项目为歌曲，今日头条中项目为新闻。推荐系统的设计目标是在用户缺乏相关领域经验或者面对海量信息而不知所措时，为用户提供一种智能的信息过滤的方法。

在众多推荐技术中，协同过滤算法和基于内容的过滤算法是被研究最多的两种方法。基于内容的过滤算法（Content-Based Filtering，CBF）起源于信息检索技术[2][3],是最早使用在推荐系统中的一种算法。 CBF 就是对项目信息进行相应的处理，形成表示项目内容的特征描述。并且，同用户进行信息交流的时候，会自主的将用户访问过的所有历史记录下来，而且还会在用户所访问的信息的基础上对其进行用户建模（User Modeling），这样就能够对用户的相关兴趣进行特征描述（User Profile）。在以上操作的基础上，就可以将兴趣描述同用户还

没有访问过的信息进行度量，这样就能够从中选择用户还没有访问但又与该描述相近的项目从而推荐给用户。

其特点是只需要对系统中的该用户进行关注即可，不需要对别的用户进行关注，分析自己的相关信息后，要对其进行总结，主要对用户访问的信息或服务所具有的共性进行总结，最后将与所得结果有相同特性的项目推荐给用户。由于现代信息技术的蓬勃发展，出现了海量的数据，如：语音数据、视频数据等，由于这些数据的数据量大，数据维数高，使得人们难以对其特征进行提取，所以基于内容的协同过滤算法将面临这严峻的挑战。由于该算法只对用户自己的相关信息进行分析，因此要向用户推荐相关信息时，对用户潜在兴趣的发现能力有所不足

1992 年 Goldberg 提出“协同过滤”（Collaborative Filtering，CF）的概念，并在后来被广泛的研究和应用[1]。协同过滤假设，如果两个用户 A 和 B 在一些项目上具有相似的行为习惯(例如购买、阅读、观影等)，那么他们在其它项目上也具有相似的偏好，协同过滤因此也被称为社会过滤或协作过滤[2][3]。明尼苏达大学的 GroupLens 研究团队在 1994 年提出了基于协同过滤的开源框架，GroupLens[4]。并在 1997 年将其在新闻组服务中进行了实现[5]。

GroupLens 系统的出现对推荐系统来说具有划时代的意义，该系统是自推荐系统的基础，现在的许多系统都是在 GroupLens 的框架的基础上进行改进。当该系统被构建出来后，为了对该系统的性能进行更近一步的完善，向人们提供了 MovieLens 推荐系统[6]，MovieLens 系统根据观看者的评分向用户推荐电影，据此出现了推荐算法中普遍使用的 MovieLens 数据集，本文实验部分也将使用该数据集进行。

协同过滤算法和基于内容的过滤算法，这两种算法有着许多的不同之处，主要的不同之处在于推荐的策略。协同过滤算法是一种不需要了解用户偏好，仅使用用户对商品的历史评分数据来预测用户对未知商品的评分，来产生推荐的技术。协同过滤算法简单、有效，在很多领域的推荐系统中得到了大量实际应用。虽然 CF 算法可以克服 CBF 算法中存在的诸多缺点，但随着数据规模的不断扩大，传统的 CF 算法逐渐暴露出数据稀疏性的问题[7]，严重制约了该技术的应用。因此，许多研究都围绕如何解决数据稀疏性这一问题展开，相应产生

了许多推荐技术。

Sarwar 等人通过奇异值分解（Singular Value Decomposition，SVD）方法来减少用户-项目评分矩阵的维度[8]。Koren 等人在传统的 FM 模型里加入了隐式评分信息，提出了考虑领域影响的 SVD++算法[9]。这两种方法都是通过将高维的评分矩阵映射到低维空间，得到相对稠密的评分矩阵来解决数据稀疏问题，但这样会造成推荐精度有所降低。Karypis 等人提出了基于项目的协同过滤算法（Item-based CF，IBCF）[10]，像在电商领域主要是用户不断增长，而项目数基于趋于稳定，因此项目的相似性更加稳定。以上方法都是在原有评分数据的基础上，通过矩阵分解、聚类等机器学习的方法来缓解数据稀疏问题。

结合其它有用信息是另外一种缓解数据稀疏性问题的重要手段，这种方法的思想是在其它方法的基础之上引入额外的信息源，使得发现的邻居用户能更为准确，从而缓解数据稀疏问题。Balabanovic 等人根据基于内容和协同过滤优势互补的特点，提出了两者混合的方法[11]。Melville 等人提出了一种名为 content-boosted 的协同过滤方法，该方法引入了额外的文本信息来为用户提供推荐[12]。Ziegler 等人提出把产品的散装分类信息融合到协同过滤算法中来解决数据稀疏性问题[13]。Ba Q 等人首先通过用户统计信息进行用户的聚类，然后与矩阵分解后合成的新评分矩阵，来共同进行最近邻计算与推荐[14]。He 等人将用户的社交信息融入到推荐中，提出了 SNRS 推荐系统[15]。Shambour 等人引入了评分信任度的思想，直接通过用户信任度与项目信任度来进行评分预测，摒弃了传统的相似度计算[16]。吴一帆等人提出了结合用户背景信息的推荐算法，该算法首先将用户背景信息进行量化，然后通过量化后的用户背景信息计算用户之间的相似度, 来预测评分矩阵中空闲处的评分并填充到其中，然后再通过传统的协同过滤算法进行推荐[17]。黄裕洋等人提出一种同时使用用户相似度与项目相似度进行评分预测的方法，加权因子根据数据稀疏度自动调节[18]。孙金刚等人使用项目属性来计算项目相似度，并与传统方法得到的项目相似性通过加权因子结合，提出一种新的相似度量方法[19]。到杨阳等人等人提出一种基于矩阵分解与用户近邻模型的推荐算法[20]。同时使用用户-项目评分矩阵与用户基本信息，并通过随机梯度下降来进行训练求出用户相似度。

二、用户画像

用户画像（User Profiling，UP），也叫做用户建模（User Modeling），一般指通过定义用户属性，给用户一个简短、有效的描述。在大数据分析与深度学习概念出现之前，用户画像已经成为了商业智能、信息系统领域的重要研究方向[21]。20 世纪 90 年代以来，通过自动化技术，隐式的获取用户的反馈数据，以此进行用户画像的推断，成为主流方法途径。传统的用户兴趣、个性、行为习惯等画像信息的理解在推荐系统[22][23]等传统信息检索、数据挖掘任务中已经存在了很久。

用户画像技术最早出现在 90 年代后期，通常意义上按照用户属性、档案（Profile）的表示策略将用户画像技术分为以下四大类，其中有一类为基于用户兴趣 / 偏好的画像方法。用户的兴趣和偏好始终是用户档案信息的重点，在基于内容的推荐系统，除了良好的表示项目外，还需要准确的理解用户的兴趣档案，这样才能对症下药。Carmagnola 提出通过用户产生的标签入手来发现用户的兴趣档案[24]。Sugiyama 等人提出了通过用户浏览行为、结果评分等信息，构建用户的偏好档案的方法[25]。在用户画像与推荐系统的研究中，刘广东设计并实现基于用户画像的商品推送系统[26]。赵荣霞以 WordPress 为研究对象提出了基于用户画像的 WordPress 博文推荐理论[27]。王智囊将医学画像的研究应用于推荐算法中，提出了基于 SVD 的协同过滤与融合画像 Tag 标签特征的推荐算法[28]。

第四节　研究内容、研究方法及创新点

一、研究内容

本文主要研究内容数据整理、用户信息度量、用户画像构建、用户画像与协同过滤融合个四方面的内容。

（一）数据整理

本文研究用 movielens-1m 数据集，数据集包含 6040 名用户对 3952 部电影 1000000 条评分。同时还包含用户的统计信息（性别、年龄、职业）和电影信息（电影名、类型）。但这些信息的结构不符合计算要求，因此需对数据进行整理。

（二）用户信息量化

Bobadilla 等人在文献中提出一种度量奇异值权重的方法[29]，通过这种方法能够过根据用户的评分矩阵度量出数据中隐含的奇异值权重信息。本文用户画像的构建需要这种权重信息，所以根据 Bobadilla 提出的度量奇异值权重的方法，提出一种新的方法来度量用户信息的权重，生成用于构建用户画像的标签（标签为度量出的用户信息的权重）。

（三）用户画像的构建

充分利用用户的统计信息与用户的行为信息，构造出关于偏好的用户画像。对年龄分段、对工作分类，根据实验确定出恰当的年龄分段数与工作种类数。对用户信息都进行了量化后，将得到的标签进行统一，构建出用户画像。

（四）用户画像与协同过滤的融合

用户画像构建完成后，得到了量化的用户信息，就可以方便的将这些附加信息运用到基于协同过滤的推荐中去。本文研究重点就是构建用户画像后，如何与协同过滤相结合，形成一种推荐效果良好的混合推荐系统。

二、研究方法

本文研究主要采用理论分析与建模法、实验法这两种方法。

（一）理论分析与建模方法

利用理论分析方法分析协同过滤、用户画像的构成要素，在用户画像模型、SM (similarity measure based on singularity) 模型与推荐系统模型的基础上，分别建立构建用户画像、基于协同过滤与用户画像混合的研究模型。

（二）实验法

为了增加得出的评价指标结果的可信性，每次实验采用更加严格的五折交叉验证法来进行。把数据分成 5 份，每次拿出 1 份作为验证集，剩下 4 份作为训练集，重复 5 次。最后平均 5 次的结果，作为评价指标的结果。

为了更加客观、直接的看本文所提算法的性能优劣，采用对比试验的方法。通过控制特定变量，将本文提出的算法与传统的基于内存的协同过滤推荐算法进行比较。

三、创新之处

一是将 Bobadilla 提出的根据用户评分矩阵分析得出奇异值信息的方法，通过一些改进引用到了用户信息量化的过程当中。将用户统计信息转化为更易进行计算的数值来表示。

二是对用户画像的构建进行研究，提出了一种通过用户统计信息与用户行为信息来构造关于兴趣、偏好的用户画像方法。这些信息易获取、可扩展性较强，并且成本也较小。

三是在传统的协同过滤推荐系统中引入用户画像。希望能通过用户画像来缓解数据稀疏性的问题，提高推荐质量与个性化程度。

第五节　论文组织结构

第一章，绪论。第一章说明了本文的研究背景及意义，归纳了国内外对协同过滤、用户画像、混合推荐的研究现状，最后介绍了本文的主要研究内容、研究方法与创新之处。

第二章，相关技术理论综述。本章是论文构思写作的理论基础，其中包括三方面的理论知识，一为传统的基于内存的协同过滤算法，二为基于奇异性的相似度模型，三为混合推荐技术。

第三章，基于用户画像与协同过滤的混合推荐算法。第一节阐述了引入用户画像的原因，并介绍了 MovieLens-1M 数据集。之后介绍了用户信息度量模型的提出以及用户画像的构建。第二节阐述在特征组合这种混合思想下，基于用户画像与协同过滤混合推荐算法的提出。

第四章，实验与结果分析。首先介绍了如何将数据进行预处理，使其符合本文提出算法的数据结构要求。明确了评价的检验指标，比较了不同指标下传统推荐算法和混合改进后的算法，最后针对实验结果进行了分析得出了实验结论。

第五章，工作总结与展望。总结了论文的研究成果，以及研究中的不足之处。并根据不足之处对基于用户画像与协同过滤的混合推荐算法的后续改进和未来的研究进行了展望。

第二章 相关技术与理论综述

第一节 基于内存的协同过滤算法

经过 20 多年的发展，在协同过滤的基础上很多新算法被提出，来提高推荐的准确性。Breese 等人归纳了协同过滤技术的发展，将其分成了基于内存（Memory-based）的协同过滤和基于模型（Model-based）的协同过滤两类。而本文主要研究基于内存的协同过滤，在其基础之上借鉴基于内容推荐的思路，用户、产品内容信息融入到基于内存的协同过滤之中，来缓解数据稀疏性问题。Memory-based 协同过滤根据计算的相似度主体的不同，又分成基于用户的协同过滤算法、基于项目的协同过滤算法及两者混合的协同过滤算法。

假设在一系统中，用户 u 对项目 i 的评分为$r_{u,i}$，则用户-项目评分矩阵可表示为表 2.1 所示。

表 2.1 用户-项目评分矩阵

用户 u \ 项目 i	1	…	j	…	m
1	$r_{1,1}$	…	$r_{1,j}$	…	$r_{1,m}$
…	…	…	…	…	…
i	$r_{i,1}$	…	$r_{i,j}$	…	$r_{i,m}$
…	…	…	…	…	…
n	$r_{n,1}$	…	$r_{n,j}$	…	$r_{n,m}$

基于内存的协同过滤使用上述已知的评分数据产生推荐，过程分为以下三步：

（1）计算目标用户与其他用户之间的相似性；

（2）根据相似性，选择要使用的用户；

（3）使用所选用户已给出的评分，以及与目标用户的相似性进行预测。

一、基于用户的协同过滤算法

基于用户的协同过滤（User-Based Collaborative Filtering，UBCF）假设如果两个用户对共同评分项有相似的历史评分，那么他们对剩下的项目就有相似的偏好。UBCF 是基于用户最近邻（K-Nearest Neighbor，KNN）实现的算法，关键过程在于查找目标用户的邻居用户，以及根据邻居用户对项目的历史评分数据对目标用户未访问的项目进行评分预测。1994 年 Resnick 提出使用皮尔逊相关系数（Pearson Correlation Coefficient，PCC）计算用户之间的相似度，如公式（2.1）所示：

$$s(u,u')=\frac{\sum_{i\in I_u\cap I_{u'}}(r_{u,i}-\bar{r}_u)(r_{u',i}-\bar{r}_{u'})}{\sqrt{\sum_{i\in I_u\cap I_{u'}}(r_{u,i}-\bar{r}_u)^2}\sqrt{\sum_{i\in I_u\cap I_{u'}}(r_{u',i}-\bar{r}_{u'})^2}} \tag{2.1}$$

其中 $\bar{r}_u$ 为用户 u 所有评分的平均值，I_u 代表用户 u 评级过的所有项目的集合，i 是用户 u 与用户 u’的共同评分项。除此之外 UBCF 常用的相似度计算方法还有：

（1）余弦相似度（Cosine Similarity，COS)

$$s(u,u')=\frac{\vec{u}\bullet\vec{u}'}{\|\vec{u}\|\times\|\vec{u}'\|}=\frac{\sum_{i\in I_u\cap I_{u'}}(r_{u,i}\times r_{u',i})}{\sqrt{\sum_{i\in I_u\cap I_{u'}}r_{u,i}^{\ 2}}\sqrt{\sum_{i\in I_u\cap I_{u'}}r_{u',i}^{\ 2}}} \tag{2.2}$$

（2）Jaccard 相似度（Jaccard Coefficient）

$$s(u,u')=\frac{|I_u\cap I_{u'}|}{|I_u\cup I_{u'}|} \tag{2.3}$$

上述相似性计算公式中，Jaccard 较为特殊，只能用于二元评分，通过两用户相关项目的交集与并集之比来衡量用户之间的相似度而。而 PCC 和 COS 可用于离散评分和连续评分两种情况。

第二步，根据用户相似度，挑选出目标用户的邻居。邻居选择方法主要有两种，Shardanand 等人提出设置阈值，选择相似度大于阈值的项作为最近邻[2]，Resnick 等人提出选择相似度排名前 K 的项作为最近邻[4]。Herlocker 等人提出后者要优于前者[6]，因此本文选择第二种方法。

第三步，再确定目标用户的相近邻居后，根据目标用户与邻居用户的相似度以及邻居用户对目标项目的评分值实现评分预测。针对目标用户，UBCF 评分预测是根据改进的权重聚合（deviation-from-mean，DFM）方法来实现的，如公

式（2.4）所示，用户 u 的未知项目 i 的做出评分预测。其中 S_u 代表所选出的 K 名用户的集合。

$$p_{u,i} = \bar{r}_u + \frac{\sum_{u' \in S_u} (r_{u',i} - \bar{r}_u) \times s(u,u')}{\sum_{u' \in S_u} s(u,u')} \tag{2.4}$$

二、基于项目的协同过滤算法

传统的基于用户的协同过滤系统，运算量会随着用户数的增长而急剧增加。导致用户之间相似度的稳定性较差。针对 UBCF 中存在的问题，Sarwar 等人提出了基于项目的协同过滤[30]（Item-Based Collaborative Filtering，IBCF）。两者相似，不同之处在于基于项目的协同过滤在第一步计算的是项目与项目之间的相似度。在某些领域（如电商领域）用户数远大于项目数并且项目的变动远小于用户的变动，基于项目的协同过滤表现更优。Sarwar 等人分别使用了 PCC、COS、ADCOS（adjust cosine）来计算项目的相似度，其中 ADCOS 效果最好，如公式（2.5）所示：

$$s(i,i') = \frac{\sum_{u \in U_i \cap U_{i'}} (r_{u,i} - \bar{r}_u)(r_{u,i'} - \bar{r}_u)}{\sqrt{\sum_{u \in U_i \cap U_{i'}} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{u \in U_i \cap U_{i'}} (r_{u,i'} - \bar{r}_u)^2}} \tag{2.5}$$

这里 U_i 代表对项目 i 进行过评分的用户集合，$U_{i'}$ 代表过项目 i'进行过评分的用户集合，u 为对项目 i 与 i' 都进行过评分的用户。

在预测用户 u 对某一项目 i 的评分时，在用户 u 评价过的项目中选择 20 个与要预测的项目 i 最相似的项目。在第三步使用权重聚合（weight sum，WS）的进行评分预测，如公式（2.6）所示：

$$p_{u,i} = \frac{\sum_{i' \in I_u} r_{u,i'} \times s(i,i')}{\sum_{i' \in I_u} s(i,i')} \tag{2.6}$$

本质上，UBCF 与 IBCF 面对的推荐问题是相同的，都是通过用户-项目评分矩阵作为推荐的主要依据。不同之处在于 UBCF 从用户角度出发，而 IBCF 则在项目的角度解决该问题。IBCF 服从“用户未来偏好项目将与以往偏好项目保持一致”的假设，将项目作为主体，使用用户-项目评分矩阵的转置矩阵来度量项目之间的相似性。

三、基于奇异性的相似度量模型

从上述关于 UBCF 与 IBCF 的介绍中可以看出，相似度的计算是基于内存的协同过滤算法的核心组成部分，也是实现准确推荐的关键。为此，有多种计算方法。为改进传统相似度计算模型，许多学者提出了改进方法。

2012 年 Bobadilla 等人提出了一种改进的相似度量模型[29]（similarity measure based on singularity，SM）。SM 模型的特点是，在相似度的计算中对用户评分的相关性进行区分。它把高评分的项目定义为正相关的项目，即用户喜欢的项目。它把低评分的项目定义为负相关的项目，即用户不喜欢的项目。如果两用户与某一项目为正相关（即两用户对这一项目评分为高分），而绝大多数用户与此项目为负相关（即绝大多数用户对这一项目评分为低分），那这一项目对计算两用户的相似度更有价值，如表 2.2 中 user1 与 user6 对 item1 和 item2 的评分。

表 2.2 用户-项目评分矩阵

	Item1	Item2	Item3	Item4	Item5	Item6
User1	4	2	5	5	1	2
User2	2	5	4	4	4	1
User3	1	4	5	4	5	2
User4	3	4	5	5	4	2
User5	2	4	5	5	4	5
User6	5	1	4	2	5	4
User7	1	5	4	4	4	5
User8	2	5	5	4	5	5

基于奇异性的相似度量模型在计算相似度之前，要量化出每一个项目的奇异值。假设对项目的评分是 1 到 5 的整数，4、5 为正相关的评分值，1、2、3 为负相关的评分值。用 P_i 表示对项目 i 评分为正相关的用户集合，用 N_i 表示对项目 i 评分为负相关的用户集合。根据表 2 可得：

P_1={1,6},P_2={2,3,4,5,7,8},P_3={1,2,3,4,5,6,7,8},P_4={1,2,3,4,5,7,8},P_5={2,3,4,5,6,7,8},P_6={5,6,7,8};

N_1={2,3,4,5,7,8},N_2={1,6},N_3=Ø,N_4={6},N_5={1},N_6={1,2,3,4}

S_p^i代表项目 i 的正相关奇异值，计算如公式（2.7）所示。其中$card(P_i)$是对项目 i 评分为正相关的用户数量，$card(U_i)$是对项目 i 评分的用户数量。S_N^i代表项目 i 的负相关奇异值，计算如公式（2.8）所示。

$$S_p^i = 1 - \frac{card(P_i)}{card(U_i)} \tag{2.7}$$

$$S_N^i = 1 - \frac{card(N_i)}{card(U_i)} \tag{2.8}$$

对某一个项目的正相关评分越多，则 S_p^i的值就越小，反之亦然。对某一个项目的负相关评分越多，则S_N^i的值就越大，反之亦然。根据表 2 数据可得：

$S_P^1 = 1 - \frac{2}{8} = 0.75$ ， $S_P^2 = 1 - \frac{6}{8} = 0.25$ ， $S_P^3 = 1 - \frac{8}{8} = 0$ ， $S_P^4 = 1 - \frac{7}{8} = 0.125$ ，
$S_P^5 = 1 - \frac{7}{8} = 0.125$， $S_P^6 = 1 - \frac{4}{8} = 0.5$

$S_N^1 = 1 - \frac{6}{8} = 0.25$ ， $S_N^2 = 1 - \frac{2}{8} = 0.75$ ， $S_N^3 = 1 - \frac{0}{8} = 1$ ， $S_N^4 = 1 - \frac{1}{8} = 0.875$ ，
$S_N^5 = 1 - \frac{1}{8} = 0.875$， $S_N^6 = 1 - \frac{4}{8} = 0.5$

每两个用户的共同评分可以分为三类，A 类是两人评分都为正相关的共同评分项集合，B 类是两人评分都为负相关的共同评分项集合，C 类是一人评分为正相关、一人评分为负相关的共同评分项集合。如果两人的共同评分项 i 属于 A 类，则两人的关于项目 i 的奇异值为$S_P^i \times S_P^i$；如果 i 属于 B 类，则奇异值为$S_N^i \times S_N^i$；如果 i 属于 C 类，则奇异值为$S_P^i \times S_N^i$。

以 user1 与 user6 为例，两用户的共同评分项集合为{1,2,3,4,5,6}。A 类集合是{1,3}，B 类集合是{2}，C 类集合是{4,5,6}。则两用户关于每个项目的奇异值为：item1：$S_P^1 \times S_P^1 = 0.75 \times 0.75 = 0.562$；item2：$S_N^2 \times S_N^2 = 0.75 \times 0.75 = 0.562$；

item3：$S_P^3 \times S_P^3 = 0 \times 0 = 0$；item4：$S_P^4 \times S_N^4 = 0.125 \times 0.875 = 0.109$；

item5：$S_N^5 \times S_P^5 = 0.875 \times 0.125 = 0.109$；item6：$S_N^6 \times S_P^6 = 0.5 \times 0.5 = 0.25$。

之后将的得出的奇异值融合到相似度的计算中。这里使用 MSD（Mean Squared Differences）计算相似度，如公式（2.9）所示：

$$MSD(u,u')=\frac{1}{card(I_u\cap I_{u'})}\sum_{i\in I_u\cap I_{u'}}[1-(r_{u,i}-r_{u',i})^2] \quad (2.9)$$

对评分做标准化处理，使 $r_{u,i}\in[0,1]$，则 $MSD\in[0,1]$。数值越大则代表相似程度越高。最终得到基于奇异性的相似度量方法，如公式（2.10）所示：

$$SM(u,u')=\frac{1}{3}[\frac{\sum_{i\in A}[1-(r_{u,i}-r_{u',i})^2](s_P^i)^2}{card(A)}+\frac{\sum_{i\in B}[1-(r_{u,i}-r_{u',i})^2](s_N^i)^2}{card(B)}+\frac{\sum_{i\in C}[1-(r_{u,i}-r_{u',i})^2](s_P^i\times s_N^i)}{card(C)}] \quad (2.10)$$

第二、三步与 UBCF 相同。

四、混合推荐相关理论介绍

CF 与 CBF 是两种截然不同的的推荐方法，并在不同方面具有自己的优势。但 CBF 和 CF 也受到各自推荐策略的限制，在某些场景下，出现质量低下的推荐，甚至可能会无法实现推荐。如 CF 对一个新出现的项目难以实现推荐，对新用户也有冷启动的问题。而 CBF 所处理的项目必须是能够解析的文本信息，并且在联想性推荐上也无法实现。

对比分析两类推荐算法存在的问题可以发现，其优缺点往往是互补的。CBF 依赖于录入的用户、项目信息，而 CF 实现推荐却不需要用户、项目的内容；CF 无法针对新项目实现推荐，而 CBF 通过录入的用户、项目信息则可以对新项目实现推荐；因此，可以将两种推荐方法在不同形式下进行组合应用，以解决两种推荐存在的问题，从而实现优势互补，提高推荐质量和准确性。本文研究根据文献[28]将 CBF 与 CF 的混合方法划分为以下几种：

(1) 权值组合(Weighted Combination，WC)

(2) 动态切换(Dynamic Switching，DS）

(3) 混合（Mix）

(4) 特征组合（Feature Combination，FC）

(5) 级联（Cascade）

(6) 特征放大（Feature Augmentation，FA）

(7) 元层次组合（Meta-Level Combination，MLC）

本文是将用户画像与 CF 进行混合，来解决数据稀疏问题。上文对 CBF 的介绍中提到，CBF 是根据用户已访问项目的内容信息，对用户进行建模从而生成关于用户偏好的画像，找到符合其偏好的项目。用户画像与 CF 的融合本质上

与 CBF、CF 融合相同，所以这些混合算法的方法对本文依旧适用。

特征组合（Feature Combination，FC）是本文使用的一种混合方法。像（1）（2）（3）（5）这几种都是程度很轻的混合，仅是将两者的推荐结果进行一定的处理。而特征组合不是对 CB 与 CF 推荐结果进行简单的叠加，而是在设计算法时就进行组合，这种混合方法主要体现数据处理阶段。比如 CF 推荐仅使用用户-项目评分数据，没有考虑用户与项目的特征属性，而特征组合就是类似在 CF 的计算相似度的时候不仅仅计算评分矩阵的相似度，而且还要根据特征属性计算用户或项目之间的相似度等等。该混合方法提高了用户兴趣偏好在推荐系统中的作用，不再是邻居爱好什么自己就喜欢什么，在推荐项目中考虑了用户的需求偏好。

第二节 用户画像简介

用户画像概念最早由交互设计之父 Alan Cooper 提出，用户画像是现实用户的抽象表示，是一种建立在真实数据上的用户标签模型。

用户画像在其发展的早期阶段较为简单，并且获取用户信息的成本过高。随着信息技术的发展，用于构建用户画像模型的数据急剧增加。大数据、数据挖掘等技术使得用户画像的架构发生了本质上的飞跃，从简单的、非结构化的、标签相关性低的信息档案质变成复杂高效、系统化的、相对成熟的用户模型。在大数据信息时代用户画像对企业发展、转型具有至关重要的影响。进行海量数据处理时，标签为我们提供了一种把难以处理的信息进行量化的方法。用户画像每个标签的量化，要以目标为导向。

用户构建用户画像的信息主要有三类：

（1）基本信息：为用户注册时的原始信息数据，如人口统计学信息、不可变更的静态信息等。

（2）行为信息：为不断堆积的用户历史行为记录，如浏览、点击、评论等数据。

（3）模型标签：通过模型（如机器学习、深度学习等）学习出的稠密向量。分为可直观理解与不可直观理解两种。

第三章 基于用户画像与协同过滤的混合推荐算法

根据上文对协同过滤介绍可知，数据稀疏性是影响协同过滤推荐发展和应用的关键问题。本章针对此问题对用户相似度计算的影响，提出一种能够生成用户画像的用户信息度量模型，并将用户画像融入到协同过滤的推荐算法中。本章主要介绍了如何通过用户信息度量模型使用用户-项目评分信息、用户统计信息构建用户画像，以及如何将用户画像与传统相似度量模型融合。

第一节 问题的提出

在推荐系统中，大部分用户仅仅对个别项目进行过评价，而且用户评价过的项目也不尽相同，使得用户-项目评分矩阵极端稀疏。如 Movie Lens-1m 数据集，有 6040 位用户对 3952 部电影的 1000000 条评分，稀疏度计算如（3.1）所示：

$$(1-\frac{1000000}{6040\times 3952})\times 100\% = 95.81\% \qquad (3.1)$$

这说明用户-项目评分矩阵中，有 95.81%的部分是没有数据的。而且 MovieLens 数据集还是挑选了评分数在 20 以上的用户，现实中推荐系统的稀疏度能达到 99%以上，这种极度稀疏的数据将给协同过滤推荐带来很大的影响。

清华大学曾春（2002）在其研究中提到，调查显示 80%的用户愿意向平台提供自己的姓名、年龄、性别、教育背景等。但大多数用户不愿意提供像个人收入、感情状况、家庭成员等隐私的信息。并且有 28%的用户愿意平台向其他平台共享自己的信息[31]。

所以本文对用户最易获取的基本信息进行用户画像，找出用户对项目的偏好信息，使用协同过滤算法进行推荐，来提高推荐的准确性。

第二节 用户信息度量模型

一、用户信息标签化

假设获得了一位用户的基本信息，张三（男，27，工程师），虽然这也是对用户张三的画像，但这些内容反应不出他的偏好信息，也无法用于计算。要想利用用户基本信息，并用于推荐系统当中，首先要有适合的用户画像技术。

上一章中介绍了 Bobadilla 提出的基于奇异性的相似度量模型，这种模型通过分析用户-项目评分矩阵的结构，度量出一种称为奇异值的权重信息。本文用户画像的构建需要这种权重信息，所以根据 Bobadilla 提出的度量奇异值权重的方法，提出一种新的方法来度量用户信息的权重，生成用于构建用户画像的特征（特征为度量出的用户信息的权重）。

下面对这种度量用户信息的方法进行介绍。假设评分矩阵如下表所示：

表 3.1 用户-项目评分矩阵

	Item 1	Item 2	Item 3	Item 4
User 1	4		5	5
User 2	2		4	
User 3	1	4		4
User 4	3	4		5
User 5	2	4		

以用户的性别信息为例。假设用户性别已知，将用户的性别信息与评分矩阵相结合，得到用户-项目性别矩阵，如表 3.2 所示：

表 3.2 用户-项目性别矩阵

	Item 1	Item 2	Item 3	Item 4
User 1	M		M	M
User 2	M		M	
User 3	F	F		F
User 4	F	F		F
User 5	F	F		

表 3.2 中 M 代表男性，F 代表女性。假设对一项目产生行为的某一性别的用户越多，则此项目更受这一性别用户的喜爱，反之亦然。

将 M_i 定义为对项目 i 评分的男性用户集合，根据表 3.2 可得：

M_1={1,2}, M_2= Ø, M_3={1,2}, M_4={1}；

将 F_i 定义为对项目 i 评分的女性用户集合，根据表 3.2 可得：

F_1={3,4,5}, F_2={3,4,5}, F_3=Ø,F_4={3,4}；

将 I_M^i 定义为项目 i 的男性化指数（受男性喜爱的程度），计算如公式（3.1）所示。其中 $card(M_i)$ 是对项目 i 评分的男性用户数量，$card(U_i)$ 是对项目 i 评分的总用户数量。

$$I_M^i = \frac{card(M_i)}{card(U_i)} \tag{3.1}$$

将 I_F^i 定义为项目 i 的女性化指数（受女性喜爱的程度），计算如公式（3.2）所示。其中 $card(F_i)$ 是对项目 i 评分的女性用户数量，$card(U_i)$ 是对项目 i 评分的总用户数量。

$$I_F^i = \frac{card(F_i)}{card(U_i)} \tag{3.2}$$

I_M^i、I_F^i 可总称为性别指数。评价某一个项目的男性用户数越多，则 I_M^i 的值就越大，反之亦然。评价某一个项目的女性用户数越多，则 I_F^i 的值就越大，反之亦然。根据表 3.2 数据可得：

$I_M^1 = \frac{2}{5} = 0.4$， $I_M^2 = \frac{0}{3} = 0$， $I_M^3 = \frac{2}{2} = 1$， $I_M^4 = \frac{1}{3} = 0.333$；

$I_F^1 = \frac{3}{5} = 0.6$， $I_F^2 = \frac{3}{3} = 1$， $I_F^3 = \frac{0}{2} = 0$， $I_F^4 = \frac{2}{3} = 0.667$；

最后根据用户产生过行为的项目集，计算它们性别信息的权重，生成用户 j 画像的性别特征 L_M^j 与 L_F^j，L_M^j 特征衡量用户 j 对男性化电影的偏好程度，计算公式如下所示：

$$L_M^j = \frac{\sum_{i \in ItemSet_j} (r_{j,i} \times I_M^i)}{card\ (ItemSet_j)} \tag{3.3}$$

其中 $ItemSet_j$ 为用户 j 评价过的项目集合。用文字来描述 L_M^j 的计算，就是是对用户 j 对其评价过每一项目 i 的评分与项目 i 的男性化指数的乘积求和，然后取平均值。

L_F^j 特征衡量用户 j 对女性化电影的偏好程度，L_F^j 计算公式如下所示：

$$L_F^j = \frac{\sum_{i \in ItemSet_j} (r_{j,i} \times I_F^i)}{card\ (ItemSet_j)} \tag{3.4}$$

用文字来描述 L_F^j 的计算，就是是对用户 j 对其评价过每一项目 i 的评分与项目 i 的女性化指数的乘积求和，然后取平均值。

这样用户 j 就得到了量化的两个通过性别来衡量用户偏好的特征 L_M^j 与 L_F^j。如 user5 产生行为的项目集为{1，2}，$L_M^5 = \frac{(2 \times 0.4 + 4 \times 0)}{2} = 0.4$，$L_F^5 = \frac{(2 \times 0.6 + 4 \times 1)}{2} = 3.6$。

上述公式都以性别信息进行节介绍，公式 3.1、3.2 性别指数的计算，推广到用户信息如下所示：

$$I_l^i = \frac{card(l_i)}{card(U_i)} \tag{3.5}$$

公式 3.3、3.4 为性别特征的计算，推广到用户信息如下所示：

$$L_l^j = \frac{\sum_{i \in ItemSet_j} (r_{j,i} \times I_l^i)}{card\ (ItemSet_j)} \tag{3.6}$$

二、构建用户画像

上文通过提出的用户信息度量模型，以用户性别信息为例，量化出项目的两种性别指数，并根据性别指数度量了用户的性别特征，反应出了用户在性别特征下的偏好。

但年龄、职业与性别信息不同，两者的划分维度多或者说分类太多。年龄是连续性整数，职位种类又各种各样，这样通过用户信息度量模型得出的年龄、职业标签与性别标签就会不在同一数量级，就无法在一起进行运算。年龄、职业特征因数值太小，在之后相似度的计算中仅占有很小的权重，这样就失去了计算的意义。所以本文将年龄划分为青少年、成年、壮年、老年四类，职业划分为技能、研究、经营、事务、其它五类，具体信息及对应标号见表 3.3。

这样就可以根据用户信息度量模型将用户的性别、年龄、职业生成特征以及相对应的权重。为了使表述更加清晰，将所有指数、特征汇总在表 3.4 中。

表 3.3 标签对照表

Gender		Age		Occupation	
F	女性	Tee	青少年	Tec	技能型
M	男性	Adu	成年	Rec	研究型
		Mat	壮年	Ope	经营型
		Old	老年	Obj	事务型
				Else	其它

表 3.4 指数及标签汇总表

符号	符号所代表含义描述（以项目 i 为例）	符号	符号所代表含义描述（以用户 j 为例）
I_M^i	男性化指数	L_M^j	男性化特征
I_F^i	女性化指数	L_F^j	女性化特征
I_{Tee}^i	青少年化指数	L_{Tee}^j	青少年化特征
I_{Adu}^i	成年化指数	L_{Adu}^j	成年化特征
I_{Mat}^i	壮年化指数	L_{Mat}^j	壮年化特征
I_{Old}^i	老年化指数	L_{Old}^j	老年化特征
I_{Tec}^i	技能型指数	L_{Tec}^j	技能型特征
$I_{\mathrm{Re}c}^i$	研究型指数	$L_{\mathrm{Re}c}^j$	研究型特征
I_{Ope}^i	经营型指数	L_{Ope}^j	经营型特征
I_{Obj}^i	事务型指数	L_{Obj}^j	事务型特征
I_{Else}^i	其它型指数	L_{Else}^j	其它型特征

第三节 混合推荐算法设计

上一节中详细介绍了用户信息的度量，其目的本质上就是将稀疏的用户-项目评分矩阵转换成满秩的用户-特征矩阵。这样就可以将其融入传统的基于用户的协同过滤算法当中。

之所以认为将上述得到的用户画像与协同过滤混合会改善推荐系统，是基

于以下假设提出的：

1、假设看某部电影的多为具有某一特征的用户，那这部电影被具有这一特征的用户偏好。

2、假设某位用户看的电影，多为具有某特征的用户偏好的电影。那这位用户会偏好具有此特征用户偏好程度大的电影。

如 I_M^i 就是量化后的电影被男性偏好的程度，用户画像的 L_M^j 特征标签就是量化后的用户对男性化电影的偏好。

通过对传统基于用户的协同过滤分析可知，算法核心有三个部分。分别是为用户-项目评分矩阵的构建、相似度的计算、推荐结果的生成。本文进行混合推荐算法实验的出发点，就是通过改进这三个方面来提高推荐的质量。

特征组合是在 CBF 与 CF 中互相使用对方的特性，以提高特定的推荐能力。对本文来说，就是通过用户信息度量模型，使用用户-项目评分矩阵、用户基本信息来进行用户画像，得到用户-特征矩阵，将其用于基于用户的协同过滤算法中。这是在用户-项目评分矩阵的构建阶段进行了改进。

进行第一步相似度的计算，这里分别使用公式（2.1）皮尔逊相关系数、（2.2）余弦相似度、（2.5）改进的余弦相似度（PCC、COS、ADCOS）进行比较找出最优的一种计算相似度的方法。因为是使用用户画像（user profile）生成的用户-特征矩阵进行相似度计算，所以将改进后的相似度量模型起名为 UPsim 相似度。其中 Label 为特征标签集合。

通过 PCC 相似度算法，使用用户-特征举证来计算用户之间相似度的公式如 3.7 所示，其中 $\overline{L}^u$ 是 u 特征标签的平均值。

$$\text{PCC - UPsim}\left(u,u'\right)=\frac{\sum_{l\in Label}(L_l^u-\overline{L}^u)(L_l^{u'}-\overline{L}^{u'})}{\sqrt{\sum_{l\in Label}(L_l^u-\overline{L}^u)^2}\sqrt{\sum_{l\in Label}(L_l^{u'}-\overline{L}^{u'})^2}} \tag{3.7}$$

使用 COS 进行计算的公式 3.8 所示：

$$\text{COS - UPsim}\left(u,u'\right)=\frac{\sum_{l\in Label}(L_l^u\times L_l^{u'})}{\sqrt{\sum_{l\in Label}(L_l^u)^2}\sqrt{\sum_{l\in Label}(L_l^{u'})^2}} \tag{3.8}$$

使用 ADCOS 进行计算的公式 3.9 所示，其中 $\overline{L}_l$ 是 l 特征标签的平均值：

$$\text{ADCOS - UPsim}\left(u,u'\right)=\frac{\sum_{l\in Label}(L_l^u-\overline{L}_l)(L_l^{u'}-\overline{L}_l)}{\sqrt{\sum_{l\in Label}(L_l^u-\overline{L}_l)^2}\sqrt{\sum_{l\in Label}(L_l^{u'}-\overline{L}_l)^2}} \tag{3.9}$$

根据相似度量模型得到用户之间的相似度后，还要通过聚合方法进行评分预测。本文在第二章基于用户的协同过滤中介绍了改进的聚合方法，DFM；基于项目的协同过滤方法中介绍了权重聚合方法，WS。虽然有如此多的相似度计算方法、聚合方法，但它们两者都是在协同过滤推荐系统的框架中提出或改进的。这两部分中的方法可以直接或稍加改变就能组合使用，因此 DFM、WS 也都可以与本文提出的相似度量模型结合使用。

第四章 推荐算法实验设计与分析

本章首先介绍了如何根据用户信息度量模型生成用户画像，以及使用的评价指标，之后说明了实验流程。然后根据实验流程的安排，对实验目的、步骤、结果进行了介绍，以及对实验结果进行了分析、总结。

第一节 用户画像的生成

一、数据集介绍

推荐系统经过多年的发展，在很多领域得到了推广和应用，并在运行中积累了大量的真实数据。MovieLens-1m 数据集是由 GroupLens 研究小组从 MovieLens 电影评分网站收集的评分数据构成。此真实数据既有一定的数据规模，还在一定程度反应了电影领域中真实数据的分布特征，所以具有较强的代表性。在推荐算法实验中，MovieLens-1m 数据集被大量采用，尤其在 CF 推荐算法的评价中使用更为广泛。鉴于其的权威性，本文在实验中也采用了该数据集。MovieLens 数据集有三个文件，为 users（用户信息）、movies（电影信息）、ratings（用户对电影评分信息）。其中用户信息以如下形式 UserID::Gender::Age::Occupation 存储，分别为用户 ID、性别、年龄、职业，具体信息如表 4.1 所示。

表 4.1 用户信息表

UserID	Gender	Age	Occupation
1	F	1	10
2	M	56	16
3	M	25	15
4	M	45	7
5	M	25	20

其中各标号含义如表 4.2 所示，其中性别 2 类，年龄 7 类，职业 21 类。

表 4.2 标号对照表

Gender		Occupation		11	律师
F	女性	0	不明确	12	程序员
M	男性	1	老师	13	退休人员
		2	艺术家	14	销售
Age		3	行政人员	15	科学家
1	18 岁以下	4	大学生	16	个体经营
18	18-24 岁	5	服务人员	17	工程师
25	25-34 岁	6	医生	18	技工
35	35-44 岁	7	管理者	19	失业者
45	45-49 岁	8	农民	20	作家
50	50-55 岁	9	家庭主妇		
56	56 岁以上	10	青少年		

ratings 文件中包含 3952 位用户对 6040 部电影的 1,000,209 条评分信息，评分区间为[1,5]的整数值，评分的大小反应了用户对电影的兴趣度。具体如表 4.3 所示。

表 4.3 用户对电影评分表

UserID	ItemID	Rating	Timestamp
1	1193	5	978300760
1	661	3	978302109
1	914	3	978301968
1	3408	4	978300275
1	2355	5	978823291

二、生成用户画像

上文提出的用户信息度量模型，在年龄、职业的划分上与 MovieLens-1m 不同。MovieLens-1m 将年龄、职业划分过细，这样通过用户信息度量模型得出的年龄、职业标签与性别标签就会不在同一数量级。年龄、职业特征因数值太小，在之后相似度的计算中仅占有很小的权重，这样就失去了计算的意义。

所以将 MovieLens-1m 中关于职业、年龄的划分归纳为用户信息度量模型中所提到的分类形式。将年龄划分为青少年、成年、壮年、老年四类，职业划分为技能、研究、经营、事务、其它五类，具体信息及对应标号见表 4.4。

表 4.4 标签对照表

Gender		Occupation	
F	女性	Tec	技能型（医生、律师、程序员、工程师）
M	男性	Rec	研究型（老师、艺术家、大学生、科学家、作家）
		Ope	经营型（行政人员、管理者、个体经营者）
Age		Obj	事务型（服务人员、农民、销售、技工）
Tee	青少年（24 岁以下）	Else	其它（不明确、家庭主妇、青少年、退休人员、失业
Ad	成年（25-44 岁）		
Mat	壮年（45-55 岁）		
Old	老年（56 岁以上）		

根据新的年龄、职业划分，生成新的新用户信息表，如表 4.5 所示：

表 4.5 新用户信息

UserID	Gender	Age	Occupation
1	F	Tee	Else
2	M	Old	Ope
3	M	Adu	Rec
4	M	Mat	Ope
5	M	Adu	Rec

之后将 MovieLens-1m 中的 ratings 分成训练集与测试集两部分，训练集占总数据集的 80%，测试集占总数据集的 20%，然后通过训练集的用户-项目评分

生成用户-项目评分矩阵，然后结合信息用户信息表，根据公式 3.5 生成表 4.6 所示的项目-特征矩阵。

表 4.6 项目-特征矩阵

电影 ID	特征	权重值
2031	Obj	0.037037037037
2031	Adu	0.62962962963
2031	Old	0
2031	Mat	0.185185185185
2031	F	0.444444444444

数据预处理完成，然后使用项目-特征矩阵和用户-项目评分矩阵，根据公式 3.6 生成如表 4.7 所示的用户-特征矩阵，即用户画像。

表 4.7 用户-特征矩阵

UserID	特征标签	权重值
1	L_M^j	2.8841841143
1	L_F^j	1.2824825523
1	L_{Tee}^j	1.0200179161
1	L_{Adu}^j	2.3954736392
1	L_{Mat}^j	0.5866681249

表 4.6 与表 4.7 仅列出部分特征标签与权重值。

第二节 推荐算法评价指标

推荐算法评价是推荐系统研究的重要组成部分，也是比较不同推荐算法优劣的主要依据，本文在预测准确率与分类准确率两个方面对算法进行价。

一、预测准确率

预测准确率是评价推荐系统优劣的一个重要指标，它反应了预测分值与用户实际打分的相似程度[32]。本文采用平均绝对误差(Mean Absolute Error, MAE)

来评价算法的预测准确度，MAE 通过统计真实评分与预测评分之间绝对值的均值来实现准确性度量。MAE 值越小预测分值与用户实际打分的相似程度越大，反之，MAE 值越大预测分值与用户实际打分的相似程度越小。对于 n 个真实评分 $R=\{r_1,r_2.....,r_n\}$，推荐算法生成的预测评分为 $P=\{p_1,p_2.....,p_n\}$、，则 MAE 可表示为：

$$MAE=\frac{\sum_{i=1}^{n}|p_i-r_i|}{n} \tag{4.1}$$

基于 MAE 应用的广泛性，本文将采用 MAE 作为准确性度量。

二、分类准确率

除了预测评分外，还有任务需求推荐系统给出二元推荐，即只为用户推荐可能符合其偏好的项目。这种二元推荐因类似于机器学习领域的分类问题，也被称为分类准确性，常用的分类准确性度量标准主要有准确率（Precision）和召回率（Recall）。Billsus 等人(1998)，Basu 等人(1998)，还有 Sarwar 等人(2000a, 2000b)都曾用这两个指标来对推荐算法进行评价。

表 4.8 项目集分类表

	推荐	不推荐	总数
相关	N_{rs}	N_{rn}	N_r
无关	N_{is}	N_{in}	N_i
总数	N_s	N_n	N

而对于多元的推荐，Herlocker 等人(2004)提出可以根据表 4.8 来进行计算准确率与召回率。首先将测试集中的项目分成相关、不相关两类。若数据集中的评分范围是多元评分，需转换成二元评分。将高分划分为相关，低分划分为不相关。然后将评分预测后的项目集划分为推荐、不推荐两类。

准确率被定义为被推荐的相关项目数量与被推荐项目数量的比率，如公式 4.2 所示：

$$P=\frac{N_{rs}}{N_s}=\frac{N_{rs}}{N_{rs}+N_{is}} \tag{4.2}$$

准确率表示的是一个被推荐项目是正确的可能性。召回率被定为被推荐的相关项目数量与相关项目数量的比率，如公式 4.3 所示：

$$R=\frac{N_{rs}}{N_r}=\frac{N_{rs}}{N_{rs}+N_{rn}} \tag{4.3}$$

虽然准确率（precision）与召回率（recall）对推荐系统的分类能力有十分直观的度量，但两者之间存在一定的负相关，不能单独使用其中一种去评价不同算法。F1 评分是对准确率与召回率的一种综合度量，是对二者的协平均。公式下所示：

$$F_1=\frac{2\times P\times R}{P+R}=\frac{2\times N_{rs}}{N_r+N_s} \tag{4.4}$$

第三节　实验设计

一、实验流程介绍

首先将 MovieLens-1m 中的 ratings 分成训练集与测试集两部分，训练集占总数据集的 80%，测试集占总数据集的 20%。

根据用户基本信息与训练集构成的用户-项目评分矩阵生成用户画像后，就可以通过协同过滤的方法进行推荐。本文所提方法的本质就是在计算用户相似度时，使用用户画像所构成的用户-特征矩阵来进行计算。而传统的协同过滤使用用户-项目评分矩阵来计算用户之间的相似度。

为了验证所提算法的有效性，分别进行以下实验：

（1）传统协同过滤方法与本文所提的用户画像与协同过滤混合方法分别使用公式 2.1、2.2、2.5（PCC、COS、ADCOS）进行相似度计算，然后根据加权聚合公式 2.4、2.6（DFM、WS）分别进行评分预测。在不同的邻居数下比较各组合的 MAE 值优劣，挑选出表现最优的相似度计算、评分预测方法组合。

（2）在特定邻居值下，并选用上一实验中选出的最优相似度计算方法、评分预测方法，试验特征组合对推荐结果的影响。如仅使用性别特征，或仅使用

性别、年龄特征。

（3）挑选出最优的相似度计算方法、评分预测方法、特征组合，形成 UPCF 模型。在特定邻居值下，比较 UPCF 与第二章介绍的 UBCF、SM 模型的 MAE 值。之后选定一邻居值，比较四种模型的 Precision、Recall、F1 值。

（4）在训练集中挑选出 m 名用户，假设为没有任何评分记录的新用户，但知晓其性别、年龄、职业。通过 UPCF 进行评分预测，并计算 MAE、Precision、Recall、F1 四种评价指标的值。

二、实验环境

本文的实验环境如下：

硬件：Lenovo 笔记本，处理器 Intel Core i5-4570@2.20GHz。

操作系统：Windows 10 操作系统。

软件：Python 3.5.2，IDE 为 Pycharm。

第四节 实验结果与分析

前三节分别介绍了本文使用的实验数据集、评价指标以及实验安排，对实验中的某些参数、变量的设置进行了说明。为了验证本文所提出的混合推荐算法的有效性，本节根据第三节介绍的实验流程进行实验，并将实验结果进行展示、分析。

算法的实现可形式化描述为：

INPUT：

Train_User_Item_Rating_Matrix (80%)

Test_User_Item_Rating_Matrix (20%)

User_Profile_Matrix

All_Label_Set

OUTPUT：

Prediction_User_Item_Rating_Matrix

```
STEP：
01:  for each user in Train_User_Profile_Matrix:
02:      for each label in All_Label_Set:
03:          use the formula of similarity (formula 3.7 or 3.8 or 3.9)
04:          get the User_Similarity_Matrix
05:  for each user in Test_User_Item_Rating_Matrix:
06:      for each movie in Test_User_Item_Rating_Matrix[user]:
07:          find the nearest neighbor depend on User_Similarity_Matrix
07:              the nearest neighbor who saw the movie
08:              use the formula of prediction (formula 2.4 or formula 2.6)
09:              get the Prediction_User_Item_Rating_Matrix
```

因模型较多，为避免混淆，将各模型代号及含义汇总在表 4.9 中。

表 4.9 各模型代号及含义汇总表

名称	含义
PCC-UPsim	通过本文提出的方法，改进后的 pcc 相似度量模型
pcc_ws	使用 pcc 根据评分矩阵进行相似度计算，ws 进行评分预测
pcc_dfm	使用 pcc 根据评分矩阵进行相似度计算，dfm 进行评分预测
pcc_upsim_ws	使用 PCC-UPsim 根据用户画像进行相似度计算，ws 进行评分预测
pcc_upsim_dfm	使用 PCC-UPsim 根据用户画像进行相似度计算，dfm 进行评分预测
UBCF	基于用户的协同过滤，pcc 相似度计算，dfm 评分预测
IBCF	基于项目的协同过滤，adcos 相似度计算，ws 评分预测
SM	基于奇异性的协同过滤，sm 相似度计算，dfm 评分预测
UPCF	本文提出的基于用户画像与协同过滤的混合推荐算法

像 pcc_ws 这种形式的名称，含义是在某次实验中，选择使用 pcc 进行相似度的计算、使用 ws 进行评分预测的聚合计算。cos、adcos 没有一一列举在表中，若将 pcc 写为 cos，则代表使用 cos 进行相似度计算。带有 upsim 则代表使用本文提出的方法进行相似度的计算。

一、相似度算法对模型的影响

本文所提出的混合方法的本质，就是不再使用用户-项目评分矩阵来计算用户相似度，而是使用用户-特征矩阵（用户画像）计算用户相似度，然后挑选最近邻，对用户未知项目进行评分预测。为了验证方法的有效性，在第一步相似度的计算中，分别在用户-特征矩阵、用户-项目评分矩阵下使用 PCC、COS、ADCOS 这三种相似度计算方法进行计算。在第二步最近邻 K 的选择中，依次取 5 到 200，间隔为 5。在第三步评分预测中，分别使用 DFM、WS 两种聚合方法。

在图 4.1-图 4.6 中，叉号线代表的是本文提出的方法，是根据生成的用户-特征矩阵来进行相似度计算。圆点线代表的传统的基于用户的协同过滤算法，是根据用户-项目评分矩阵来进行相似度的计算。

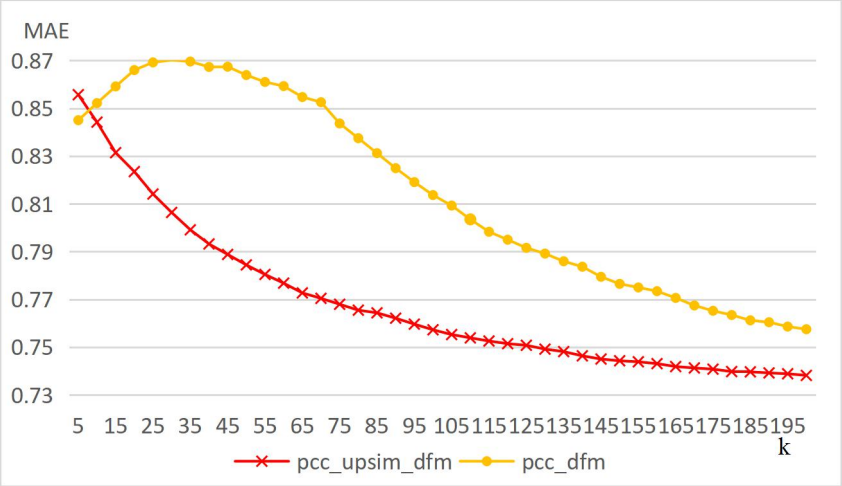

图 4.1 PCC-UPsim 与 PCC 在不同 Neighbor 下，使用 DFM 预测评分得到的 MAE 值

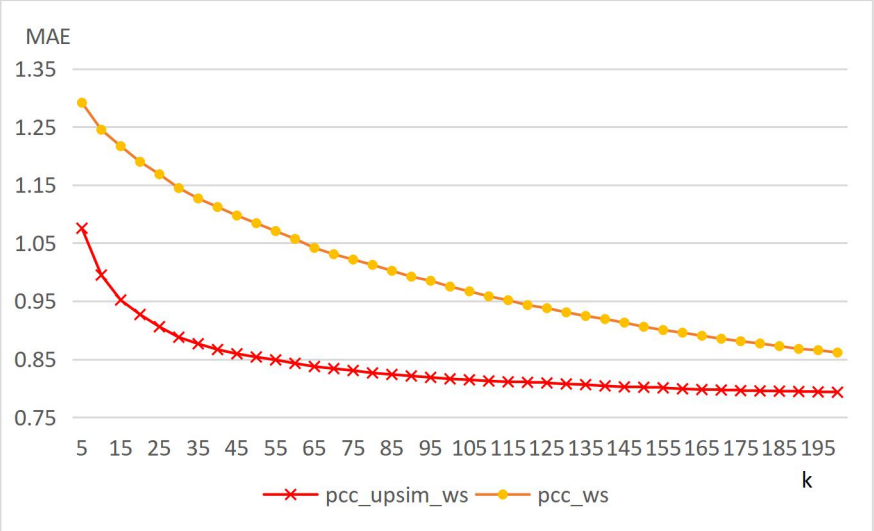

图 4.2 PCC-UPsim 与 PCC 在不同 Neighbor 下，使用 WS 预测评分得到的 MAE 值

通过实验可以看出使用本文提出的相似度量模型的方法，在 WS 方法中除在 K 值小于 15 的情况下，MAE 值都要优于使用传统相似度量模型的方法；在 DFM 方法中，本文提出的相似度量模型方法的 MAE 值都要优于使用传统相似度量模型方法得到的 MAE 值。

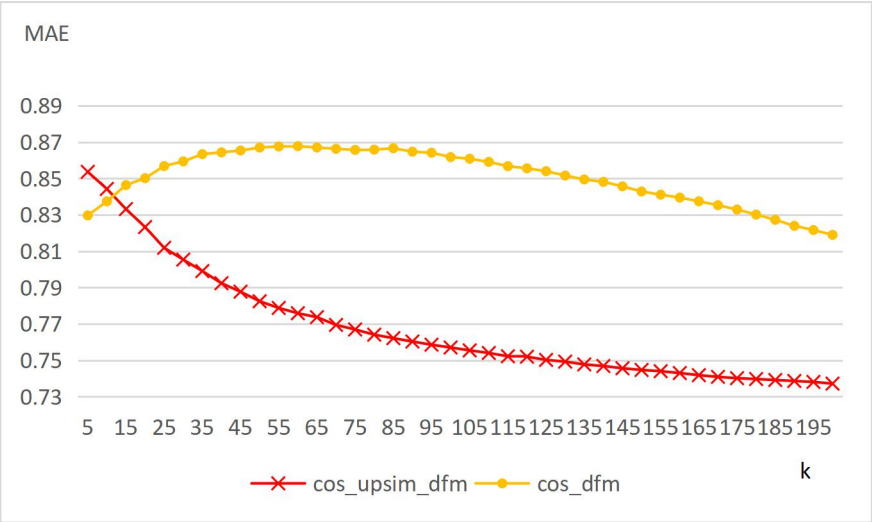

图 4.3 COS-UPsim 与 COS 在不同 Neighbor 下，使用 DFM 预测评分得到的 MAE 值

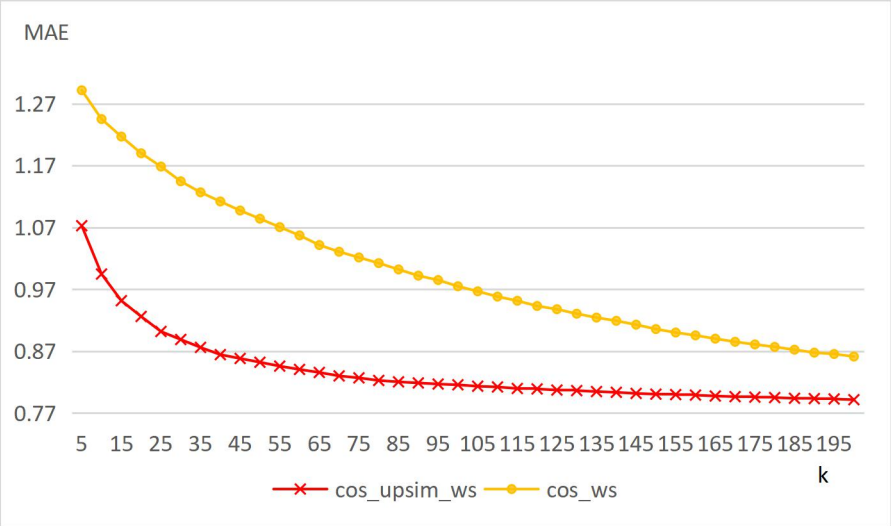

图 4.4 COS-UPsim 与 COS 在不同 Neighbor 下，使用 WS 预测评分得到的 MAE 值

当 k 在 35 到 55 区间，MAE 差值达到最大。随着邻居数的不断增加，传统相似度量算法下的 MAE 值开始减低，并缓慢逼近改进的相似度量算法下的 MAE 值。

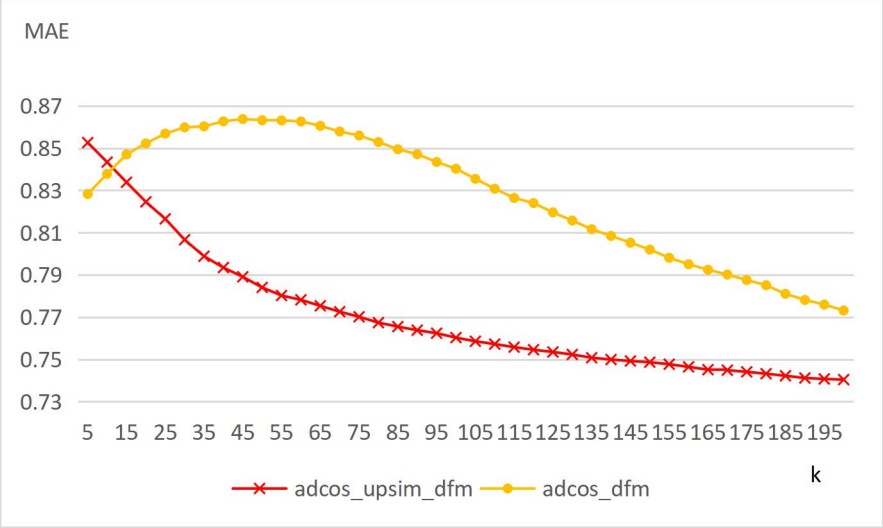

图 4.5 ADCOS-UPsim 与 ADCOS 在不同 Neighbor 下，使用 DFM 预测评分得到的 MAE 值

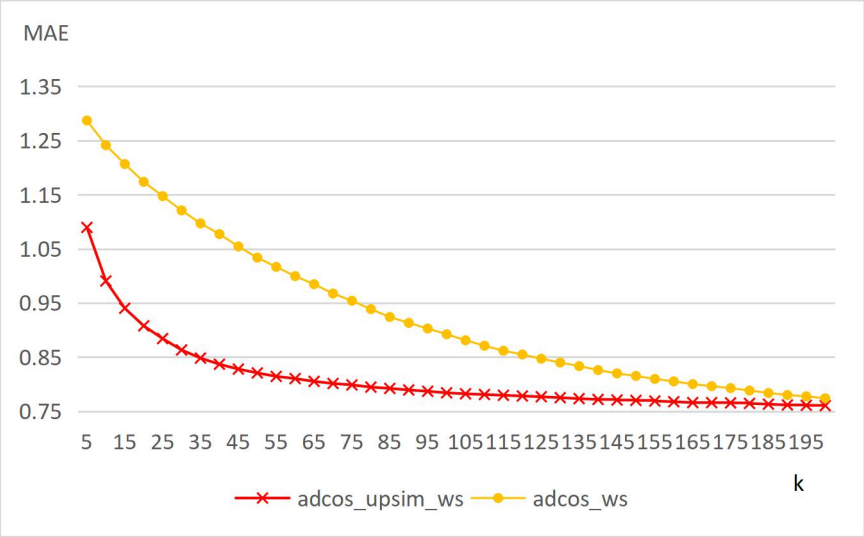

图 4.6 ADCOS-UPsim 与 ADCOS 在不同 Neighbor 下，使用 WS 预测评分得到的 MAE 值

通过这三组实验可以看出本文提出的 UPsim 方法不单单适用于某一种相似度计算模型，PCC、COS、ADCOS 这三种相似度量模型在与 UPsim 结合后，所得到的 MAE 值都优于通过评分矩阵来进行计算的传统的相似度量模型。

为找到最适合本文所提出方法的相似度量模型与评分预测模型组合，取上述三组实验中本文所提方法所能得到的最优 MAE 值，比较结果如图 4.7 所示。通过图 4.7 可以看出 COS-UPsim 与 DFM 的组合能够得到最优的 MAE 值，所以后续实验中，本文所提方法将使用 COS-UPsim 与 DFM。

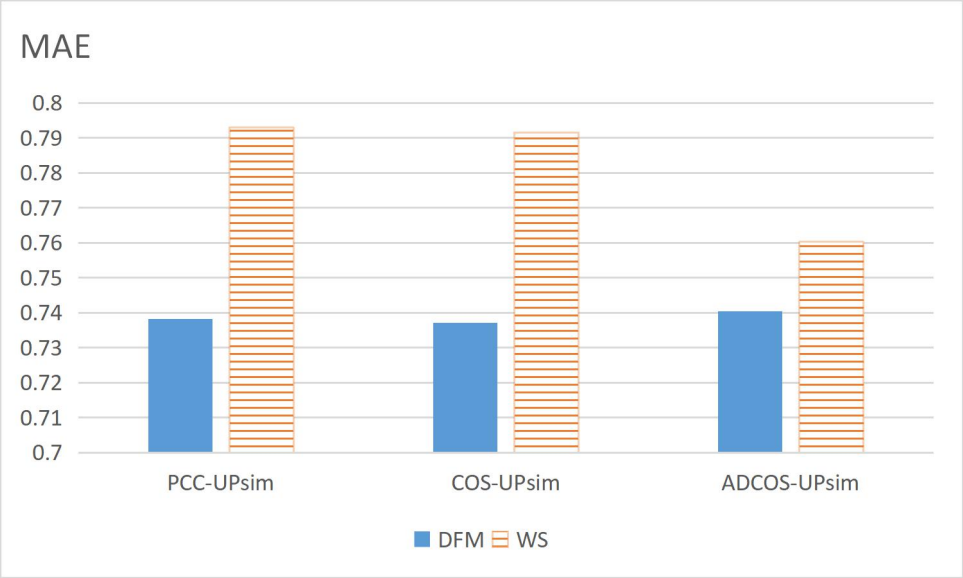

图 4.7 相似度量模型与聚合方法组合下得到的最优 MAE 值

二、用户特征对模型的影响

为了考察用户特征对模型的影响，分别使用不同的用户特征组合来进行相似度的计算。如仅使用用户性别特征来计算用户之间的相似度，或仅使用用户性别、年龄特征来计算用户之间的相似度。

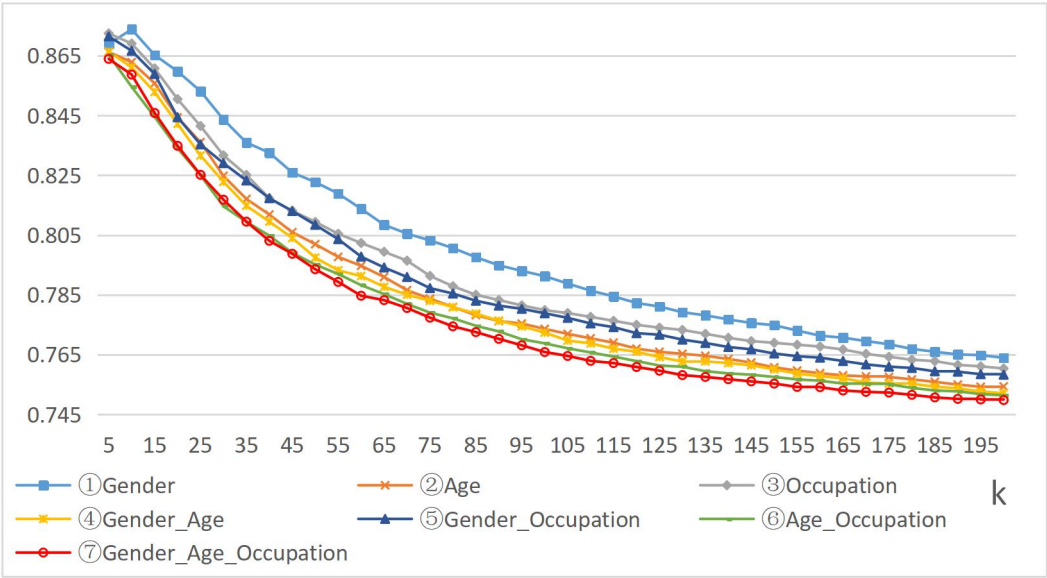

图 4.8 不同用户特征组合下得到的 MAE 值

根据图 4. 8 可以看出，同时使用 Gender、Age、Occupation 的这一组合得到的 MAE 值要优于其它组合。各组合所得到 MAE 值由优到劣顺序为：⑦⑥④②⑤③①。其中很特殊的一个地方就是仅使用 Age 用户特征所得到的 MAE 值，要

优于使用 Gender_Occupation 用户特征组合所得到的 MAE 值。可以看出，在电影领域，用户的年龄信息更能反映出用户的偏好，更具有价值。不过根据结果来看，还是使用所有用户特征信息时得到的结果最优。

三、不同模型对比实验

（一）MAE 值的比较

通过上述实验，得出表现最优的相似度计算（COS-UPsim）、评分预测（DFM）、用户特征（Gender_Age_Occupation）组合，构成 UPCF 模型。

第二章介绍了传统的 UBCF 模型，以及一种改进 SM（Bobadilla，2011）模型。UBCF 使用 PCC 进行用户相似度计算、DFM 进行评分预测。SM 使用一种改进的相似度量模型进行用户相似度计算。此改进的相似度量模型通过分析用户-项目评分矩阵结构，度量出一种被称为奇异值的信息，将其与公式 2.9（一种计算相似度的方法）结合得到，并使用 DFM 进行评分预测。

将 UPCF 与 UBCF 以及 SM 进行比较。K 取值与上述实验相同，结果如图 4.8 所示。

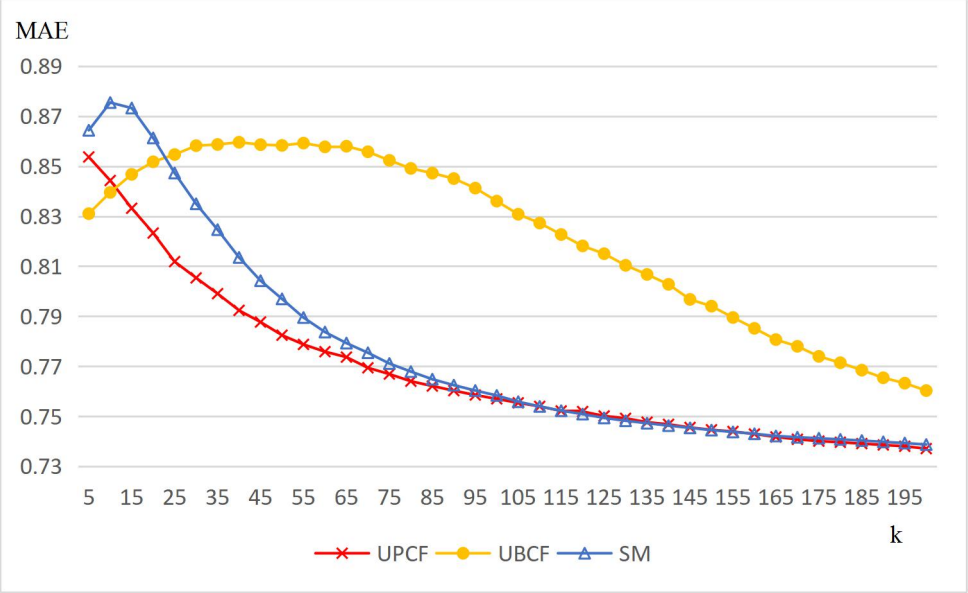

图 4.9 三种模型在不同 Neighbor 下得到的 MAE 值

从图 4.8 中可以看出，本文提出 UPCF 的 MAE 值明显优于 UBCF。与 SM 相比，在 K 小于 95 时本文提出的 UPCF 方法在 MAE 值上表现较优，之后两者几乎没有差别。

（二）Precision、Recall、F1 值的比较

MovieLens-1M 中的评分范围是 1-5 分，需转换成二元评分。这里将 4、5 分划分为相关，1、2、3 分划分为不相关。推荐次序按预测评分由高到低，推荐规定的 Top-N 部电影。

将 UPCF、UBCF 与 SM 的 K 值固定为 MAE 值都趋于稳定的 150，推荐数目 N 为 2 到 20。

准确率具体计算为预测评分排名前 N 的项目与测试集中本来就被评为高分项目的交集，比上预测评分排名前 N 的项目总数。召回率具体计算为预测评分排名前 N 的项目与测试集中本来就被评为高分项目的交集，比上测试集中本来就被评为高分项目总数。

表 4.10 准确率、召回率分值表

	Precision			Recall		
	UPCF	UBCF	SM	UPCF	UBCF	SM
2	0.81303	0.71014	0.78207	0.21214	0.18785	0.20728
10	0.66289	0.60532	0.64248	0.62254	0.59282	0.61578
20	0.52102	0.48527	0.50706	0.79411	0.76766	0.78797

将推荐数为 2、10、20 时的准确率与召回率展示在表 4.10 中。以 UPCF 在 Top-N 为 20 时来看，准确率的具体含义为在推荐的 20 个项目中有 10 个是用户喜欢的正确推荐；召回率的具体含义为在推荐的 20 个项目中包含了 80%用户喜欢的项目。在表 4.4 中可以看出随着推荐数的增多准确率降低，召回率升高并且增加幅度非常之大。当推荐数较少时，UPCF 较 UBCF 的准确率提升较多。

关于三种算法准确率与召回率的比较，更为清晰的展示在图 4.10——准确率与召回率的散点分布图中。其中召回率为横轴，准确率为纵轴，不同点代表不同的推荐数值 N。位于图 4.10 中右上方的线，代表更加优异的准确率与召回率。可以看出 UPCF 在 SM 的右边，SM 在 UBCF 的右边。综合来看 UPCF 优于 SM，SM 优于 UBCF。

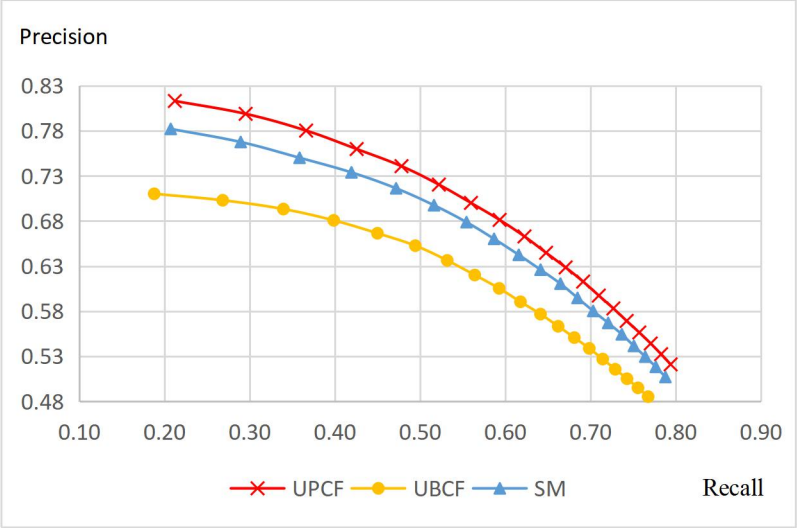

图 4.10 三种模型在不同 N 值下，准确率与召回率的散点分布图

图 4.11 是三种算法的 F1 值，是将准确率与召回率进行综合考量的一种指标，这样就将两个相关联的指标合并为一个，能够更加准确、直观的来评价算法。通过图 4.11 可以看出，本文提出的 UPCF 算法 F1 值最优，明显优于 UBCF 算法，但与 SM 差别较小。

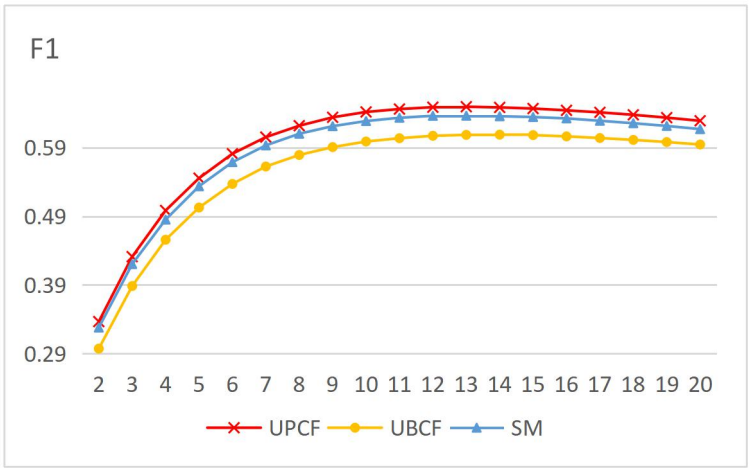

图 4.11 三种模型在不同 N 值下的 F1 值

四、冷启动下的 UPCF

新用户没有任何关于项目的评分记录，所以传统的协同过滤方法无法对新用户进行个性化推荐。本文提出的 UPCF 算法引入了用户统计信息，所以可以根据用户的统计信息来解决新用户的冷启动问题。

假设有一位知道其性别、年龄、职业的新用户，将相对应的特征标签设置

为 1，其余设置为 0。如李四（男，成年，技能型）的用户画像就如表 4.11 所示：

表 4.11 李四的初始用户画像

特征标签	权重	特征标签	权重
L_M^j	1	L_{Tec}^j	1
L_F^j	0	$L_{\mathrm{Re}c}^j$	0
L_{Tee}^j	0	L_{Ope}^j	0
L_{Adu}^j	1	L_{Obj}^j	0
L_{Mat}^j	0	L_{Else}^j	0
L_{Old}^j	0		

这样就可以根据相似度量模型 COS-UPsim 计算李四与其他用户的相似度，然后通过聚合方法 DFM 对项目进行评分预测了。为了验证 UPCF 模型对新用户评分预测的效果，在训练集中随机抽选 100 名作为新用户，按照表 4.11 的形式对这 100 位用户进行用户画像。之后根据本文提出 UPCF 模型对这 100 名用户在测试集中的项目进行评分预测，在 MAE、Precision、Recall 三种评价指标下，与正常情况下的 UPCF、UBCF 进行比较。

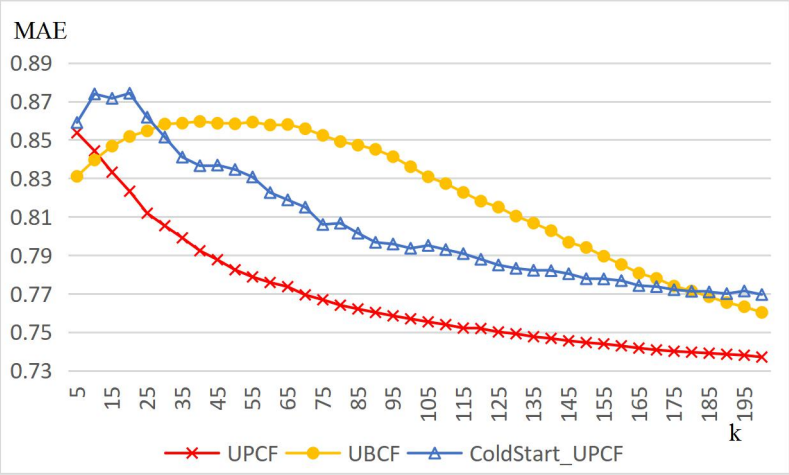

图 4.12 冷启动下 UPCF 与正常情况下 UPCF、UBCF 的 MAE 值

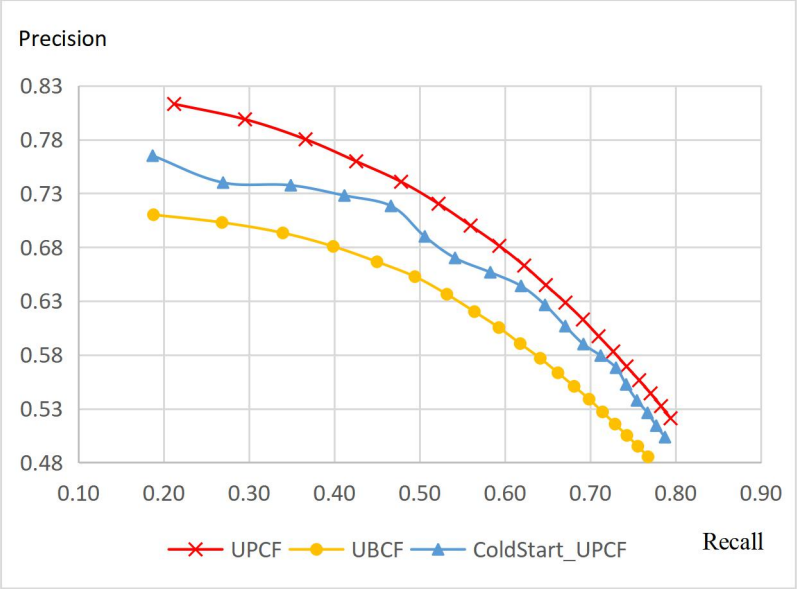

图 4.13 冷启动下 UPCF 与正常情况下 UPCF、UBCF 的 P、R 值

图 4.12 展示了通过 UPCF 模型对 100 名新用户进行评分预测后得到的 MAE 值，与用户-项目评分已知情况下的 UPCF、UBCF 模型得到的 MAE 值。图 4.13 展示了通过 UPCF 模型对 100 名新用户进行评分预测后得到的 Precision、Recall 值，与用户-项目评分已知情况下的 UPCF、UBCF 模型得到的 Precision、Recall 值。从中可以看出，在 MAE、Precision、Recall 评价指标下，冷启动下的 UPCF 得到指标值虽略差于正常情况下的 UPCF 但优于传统的 UBCF，说明 UPCF 能够较好的解决新用户冷启动的问题。

五、总结

通过以上对本文提出的 UPCF 以及 UBCF、SM 三种相似度量模型，在 MAE、Precision、Recall、F1 四种指标的评价下可以看出，本文提出的 UPCF 算法在预测准确率与分类准确率两个方面，较之传统协同过滤推荐系统 UBCF 有了明显的改善与提高。与改进后的 SM 模型相比，在各项指标上 UPCF 也要更加优秀。并且，UPCF 能够较好的解决用户冷启动问题。总体来说，实验证明本文提出的假设正确，基于用户画像与协同过滤的混合推荐系统，确实对提高推荐系统的准确性具有正向作用。

第五章 工作总结与展望

第一节 工作总结

本文首先介绍了推荐系统的背景，以及要解决的问题。并对推荐系统的发展历程以及研究现状进行了综述，其中着重介绍了协同过滤。并在第二章对基于用户的协同过滤、基于项目的协同过滤还有一种基于奇异性的相似度量模型进行了详细介绍。

虽然基于协同过滤的推荐系统实现简单、应用广泛，但随着数据量的不断增大，出现的用户-项目评分矩阵稀疏的问题，给基于协同过滤的推荐系统产生了很大的影响。解决数据稀疏问题，从而改善基于协同过滤推荐系统的的推荐效果，一直是这一领域研究的重点之一。很多学者提出了有效的解决方案，本文按解决方法类别进行了简单介绍。

本文通过改进计算相似度的矩阵以及相似度的计算方法，来缓解此问题对推荐算法的影响，从而提高协同过滤推荐的准确性。同时通过额外使用的用户信息，还能解决新用户冷启动问题。本文的主要工作如下：

1、为了改进计算相似度的矩阵以及相似度的计算，本文在基于用户的协同过滤中引入了用户画像的概念，并根据基于奇异性的相似度量模型提出一种给用户画像的用户信息度量模型。首先将存储用户信息的文件与存储用户-项目评分的文件进行预处理，生成项目-特征矩阵，将用户-项目评分矩阵、项目-特征矩阵根据用户信息度量模型转换为用户-特征矩阵（即用户画像），并以此来计算用户之间的相似度。

2、使用传统的相似度量算法 PCC、COS、ADCOS，分别根据用户-特征矩阵、用户-项目评分矩阵来计算用户之间的相似度，并以各评价指标值来进行比较。实验发现，通过用户-特征矩阵进行计算，得到的预测评分更加准确，说明这种将用户画像的融入，确实改善了协同过滤的数据稀疏问题。并且通过实验证明，本文所提方法能够较好的解决新用户冷启动的问题。

第二节 后续展望

本文对用户画像与协同过滤的混合推荐进行了研究，提出了一种度量用户画像的方法，以及多种尝试方案。所作尝试都是为了改善协同过滤推荐种的数据稀疏问题，实验证明所提方法确实有一定的效果，但是也存在很多问题，以下为后续的研究方向：

1、不同的应用环境可能会对算法的准确性带来一定的影响，但本文仅通过MovieLens-1m 数据集，在电影这一领域进行了实验探讨。所以未来需要在不同的应用环境中来实验本文提出的基于用户画像与协同过滤的混合算法，从中发现算法的不足，以及如何来进一步扩大算法的适用范围。

2、本文提出的用户信息度量模型在进行用户画像时，在年龄、职业的分类上没有经过科学的验证。

3、在评价推荐算法的实验中，仅在不同邻居值比较了 MAE、precision、recall，没有在不同比例、稀疏度下的训练集、测试集进行对比试验。评价指标也相对单一，没有使用覆盖率、惊喜度等评价指标，后续应进行完善，更加全面的说明所提算法。

4、进行用户画像时，得到的项目-特征矩阵也包含很多有价值的信息。但本文在得出项目-特征矩阵后仅用来生成用户-特征矩阵。项目-特征矩阵在计算项目之间相似度、项目与用户之间相似度、Top-N 推荐中都有进行实验尝试的价值。

参考文献

[1] Y Shi, M Larson, A Hanjalic. Collaborative Filtering beyond the User-Item Matrix: A Survey of the State of the Art and Future Challenges[C]. ACM Computing Surveys (CSUR), 2014, Volume 47 Issue 1, Article No.3..

[2] L Yao, Z Xu, B Lev. Synergies Between Association Rules and Collaborative Filtering in Recommender System: An Application to Auto Industry[M]. Data Science and Digital Business, 2019: 65–80.

[3] J. Bobadilla, F Ortega, A Hernando. Recommender system survey[C]. Know-Based Syst, 2013: 109-132.

[4] FM Harper, JA Konstan. The Movielens Datasets: History and Context[C]. In: ACM Transactions on Interactive Intelligent Systems, 2015.

[5] J Lu, D Wu, M Mao, W Wang, G Zhang. Recommender system application developments: A survey[C]. Decis Support Syst, vol 74, 2015:12-32.

[6] M Diaz, C Martin, B Rubio. State-of-the-art, challenges, and open issues in the integration of things and cloud computing[C]. Netw Comput Appl, 2015:99-117..

[7] G. Adomavicius, A. Tuzhilin. Toward the Next Generation of Recommender System: A Survey of the State-of-the-Art and Possible Extensions[C]. IEEE Trans. On Knowl. And Data Eng., vol. 17, iss. 6, pp. 734-749, 2005.

[8] X Luo, MC Zhou, S Li, Z You, Y Xia. A non-negative latent factor model for large-scale spares matrices in recommender systems via alternating direction method[C]. IEEE Trans. Neural Netw, vol 27, No.3, 2016:579-592.

[9] Koren Y. Factorization meets the neighborhood：a multifaceted collaborative filtering model[C]. Proceedings of International Conference on Knowledge Discovery and Data Mining，2008：426-434.

[10] Y Wu, C DuBois, AX Zheng, M Ester. Collaborative denoising auto-encoders for top-n recommender systems[C]. In WSDM, 2016:153-162.

[11] F Ricci, L Rokach, B Shapira. Recommender systems: introduction and challenges. Springer, 2015: 1-34.

[12] Melville P' Mooney R J， Nagarajan R． Content·boosted collaborative filtering for improved recommendations[C] ． Proceedings ofthe National Conference on Aaificial Intelligence．2002：187-192．

[13] Ziegler CN，Lausen G，Schmidt·Thieme L．Taxonomy-driven computation ofproduct recommendations[C] ． Pro ． ceedings of the thirteenth ACM international conference on Information and knowledge management．ACM，2004．406__4l5．

[14] Ba Q，Li X，Bai Z.Clustering collaborative filtering recommendation system based on SVD algorithm[C]. Proceedings of IEEE International Conference on Software Engineering & Service Science，2013：963-967.

[15] He J，Chu W，A Social Network- based Recommender System（SNSR）[M].[S.l.]：Springer US，2010：47-74.

[16] Shambour Q， Lu J.An effective recommender system by unifying user and item trust information for B2B applications[J].Journal of Computer and System Sciences，2015，81（7）：1110-1126.

[17] 吴一帆，王浩然．结合用户背景信息的协同过滤推荐算法[J]．计算机应用，2008，28 (11) : 2973－2974．

[18] 黄裕洋，金远平．一种综合用户和项目因素的协同过滤推荐算法[J].东南大学学报：自然科学版，2010，40 (5): 917-921.

[19] 孙金刚，艾丽蓉．基于项目属性和云填充的协同过滤推荐算法[J].计算机应用，2012，32 (3)：658-660.

[20] 杨 阳，向 阳，熊 磊. 基于矩阵分解与用户近邻模型的协同过滤推荐算法 [J]. 计算机应用，2012，32 (2): 395 － 398.

[21] Fawcett T, Provost F J. Combining data mining and machine learning for effective user profiling. [C]. KDD. S.1., 1996.

[22] Middleton S E, Shadbolt N R, De Roure D C. Ontological user profiling in recommender systems[J]. ACM Transactions on Information Systems (TOTS). 2004, 22 (1):54-88.

[23] Nunes M A S N, Cerri S A, Blanc N. Improving recommendations by using personality traits in user profiles[C]. International Conferences on Knowledge Management and New Media

Technology. [S.1.].2008: 92-100.

[24] Carmagnola F, Cena F, Gena C. User modeling in the social web[C]. Springer, International Conference on Knowledge-Based and Intelligent Information and Engineering Systerns. [S.1.]: Springer, 2007: 745-752.

[25] Sugiyama K, Hatano K, Yoshikawa M. Adaptive web search based on user profile constructed without any effort from users[C]. ACM, WWW. [S.1.]: ACM, 2004: 675 一 84.

[26] 刘广东. 基于用户画像的商品推送系统设计与实现[D]. 西安电子科技大学，硕士研究生学位论文, 2017.

[27] 赵荣霞. 基于用户画像的 WordPress 博文推荐研究[D]. 北京交通大学，硕士研究生学位论文，2018.

[28] 王智囊. 基于用户画像的医疗信息精准推荐的研究[D]. 电子科技大学，硕士研究生学位论文，2016.

[29] Jesús Bobadilla, Fernando Ortega, Antonio Hernando. A collaborative filtering similarity measure based on singularities[J]. Information Processing and Management, 2012, 48: 204–217.

[30] B Sarwar, G Karypis, J Konstan, et al. Item-based collaborative filtering recommendation algorithms[C].Proceedings of the 10th International Conference on World Wide Web, New York: ACM Press, 2001:285-295.

[31] 曾春，邢春晓，周立柱. 个性化服务技术综述[J]. 软件学报，2002, 13(10) : 1952 －1961.

[32] J C Herlocker, J A Konstan, L G Terveen, et al. Evaluating collaborative filtering recommender systems[J]. ACM Transactions on Information Systems, 2004, 22(1): 5-53.

[33] 王自强，冯博琴．个性化推荐系统中遗漏值处理方法的研究[J]．西安交通大学学报， 2004， 38(8)：808-810.

[34] Goldberg K, Roeder T, Gupta D, et al. Eigentaste: A Constant Time Collaborative Filtering Algorithm [J]. Information Retrieval, 2001, 4(2): 133-151.

[35] Soboroff I, Nicholas C.Combining content and collaboration in text filtering.In: Proceedings of the International Joint Conferences on Artificial Intelligence Workshop:Machine Learning for Information filtering, Stockholm,1999,86-91.

[36] 李聪. 电子商务推荐系统中协同过滤瓶颈问题研究[D]. 合肥工业大学，博士学位论文，2009.

[37] Pazzani M J, Billsus D. Content-based recommendation systems[M]. [S .1.]: Springer, 2007:325-341.

[38] Li X, Guo L, Zhao Y E. Tag-based social interest discovery[C]//ACM, WWW. [S.1.]: ACM,2008: 675-684.

[39] Billsus D, Pazzani M J. User modeling for adaptive news access[J]. User Modeling and User-adapted Interaction. 2000. 10 (2-3): 147-180.

[40] 郭光明. 基于社交大数据的用户信用画像方法研究[D]. 中国科技大学，博士研究生学位论文，2017.

[41] 鲁默. 基于用户画像的推荐系统的设计与实现[D]. 东北大学，硕士研究生学位论文，2016.

[42] 付小飞. 基于用户画像的移动广告推荐技术的实现与应用[D]. 电子科技大学，硕士研究生学位论文，2017.

[43] 王宪朋. 基于视频大数据的用户画像构建［J］. 电视技术，2017，41 (6) : 20-23.

[44] 项亮.推荐系统实践[M].人民邮电出版社,2012.

[45] Limitations of current techniques and proposals for scalable, high-performance recommender systems[J]. ACM Transactions on Information Systems, 2011, 5(1):33-65.

[46] Mahdi Jalili. A Survey of Collaborative Filtering Recommender Algorithms and Their Evaluation Metrics[J]. International Journal of System Modeling and Simulation, 2017, 2(2).

[47] 王贾予洋. 推荐系统中基于内存的协同过滤算法研究[D].西安电子科技大学, 硕士研究生学位论文, 2015.

[48] 刘青文. 基于协同过滤推荐算法的研究[D]. 中国科技大学, 博士研究生学位论文, 2013.

[49] F Cacheda, V Carneiro, D Fernandez, et al. Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems[J]. ACM Transactions on Information Systems, 2011, 5(1):33-65.

[50] 任磊. 推荐系统关键技术研究[D]. 华东师范大学，博士研究生学位论文，2012.

[51] Chua F C T, Lauw H W, Lim E P. Generative Models for Item Adoptions Using Social

Correlation[J]. IEEE Transactions on Knowledge & Data Engineering, 2013, 25(9):2036-2048.

[52] Ma H. An experimental study on implicit social recommendation[J]. 2013:73-82.

[53] 李募. 基于用户偏好集产品标签的推荐系统的设计与实现[D]. 北京邮电大学，硕士研究生学位论文，2017.

[54] Yu L, Pan R, Li Z. Adaptive social similarities for recommender systems[C]// ACM Conference on Recommender Systems. ACM, 2011:257-260.

[55] Shani G, Gunawardana A. Tutorial on application-oriented evaluation of recommendation systems[M]. IOS Press, 2013.

[56] 宋瑞平. 混合推荐算法研究[D]. 兰州大学，硕士研究生学位论文，2014.

[57] 刘宇轩. 混合协同过滤算法研究[D]. 北京邮电大学， 硕士研究生学位论文，2013.

致谢

光阴如水，岁月如莲，日子似从指尖流过的细沙，在不经意间悄然滑落。研究生的三年时光也已接近尾声，本科毕业时的场景还历历在目。但与本科不同的是我已告别了懵懂，成立了自己的家庭，也准备好了进入社会。这段研究生的学习生涯，有喜悦、有悲伤、有骄傲、有无奈，但一想到即将告别，更多的是不舍。在此离别之际，我借此机会谨向给予我关心、帮助、指导和支持各位老师、同学和亲友表示衷心的感谢！

首先，我要由衷地感谢我的导师赵昆教授。在入学之初，在赵老师的带领下就确定了自己的研究方向以及自己未来的工作定位。虽学生愚钝，三年时间并没有结出什么成果，但在这三年里，无论是在学习、实习，还是在生活中，赵老师都依然给予了我极大的关心，他总是能抽出宝贵的时间来与我畅谈。赵老师教授我的不仅是知识，还有严谨得学术态度，督导我学会做事、做人。在此，特别感谢赵老师三年来对我的谆谆教诲和耐心指导，尤其是在毕业论文的撰写期间，从选题到初稿，从初稿到修改，最后到定稿，都给了我莫大的指导和帮助，在此谨向赵老师表达我最诚挚的感谢！同时还要感谢我得师娘洪老师，让我在遥远得异地也能感受到了家的温暖。

然后，感谢实验室的陈韬伟老师，带我跨进了跨境电子商务的大门，无私的教授我，开阔了我的视野。感谢我的室友姜浩、卢宁、王熙成，谢谢你们一直以来亲如兄弟姐妹般的互相照顾；感谢林老师，帮我们处理生活、学校中大大小小的琐事，让我们在学校中健康生活、安心读书。同时感谢云南财经大学信息学院所有的老师，包括院领导、任课老师、行政老师等，正因为有你们，我得到了许多专业上的知识升华，同时生活上的许多问题也得到帮助解决，让我不负三年读研时光。此外，还要感谢一直关心和支持我的朋友们，学院 2016 级所有同学、学长学姐、学弟学妹以及研会的小伙伴们，是财大把我们相聚在一起，我们共同学习、互动、活动、分享，不仅在学术上一起努力，不觉乏味，也丰富了业余生活，增强了体质。

最后特别感谢的是我的亲人，特别是我的父母，一直支持我直到毕业，他

们的付出坚定了我的决心是我最有力的后盾，使我能够全身心的投入到学中去；除此之外还要感谢我的妻子，有缘千里来相会，是财大让我们相遇，相互扶持、相互包容、相互理解成长，使我更加懂得“爱”与“责任”是什么。

相聚是短暂的，离别是伤感的，不过我坚信，毕业不仅仅是一段路程的尾声，更是另一段路程的开始。新的路程必定是满是荆棘，肩上的责任必定更重一份，但这也意味着成长，是去向成功的必经之路，我亦会将读研期间所学灵活运用，成为更好的自己！

在读期间完成的研究成果

发表的论文：

[1] 胡兆山. 基于内存的协同过滤算法比较[J]. 信息周刊, 2018(16),73.