## 目录

摘要

Abstract

1. 引言

1.1研究背景及意义

1.2国内外研究现状

1.2.1国外研究现状

1.2.2 国内研究现状

1.3 论文的主要研究内容及工作

1. 推荐系统技术介绍

2.1 推荐系统概述

2.2 协同过滤算法

2.2.1 基于近邻的

2.2.2 基于模型的

2.3 相速度度量方式介绍

2.3.1 皮尔逊系数

2.3.2 余弦

2.3.3 海明距离

1. 两阶段联合哈希的协同过滤算法

ITQ

基于用户视角的两阶段联合哈希算法

实验设计与分析

实验数据集

实验环境

实验过程

评价指标

实验结果

本章小结

1. 总结与展望

总结

展望

参考文献

在校研究成果

致谢

## 摘要

在互联网时代，互联网技术以及电子商务的高速发展生成了海量信息，然而，如果这些海量信息不能个性化的为用户服务，就会使用户迷失在信息的汪洋大海中。传统的协同过滤算法依赖于用户或项目的相似度计算，随着系统中的数据量的极速增加，传统的线性的相似度计算方法表现出了局限性。在此提出一种两阶段联合哈希的协同过滤算法，通过在保留了用户对项目的偏好的情况下，将用户和项目映射到低维空间，使得推荐的过程转化为在低维空间搜索与目标用户具有较小海明距离的项目，省去了用户或项目之间的相似度计算，实现了高效的推荐性能。

本文的主要工作有：

首先，对评分数据从用户或项目视角应用主成分分析技术，得到该视角的低维特征表示，对其应用迭代量化技术进而生成该视角的二值码，从而实现了对评分数据全局特征的提取，也为进一步生成另一视角的二值码奠定了基础。

然后，基于上一过程中得到的其中一个视角的二值码，用已有的评分信息约束用户与项目在海明空间的距离，生成另一视角的二值码，进而实现了对评分数据局部特征的提取，也为利用得到的用户和项目的二值码进行高效的偏好预测做好了准备。

通过以上两个过程的工作，有效的减少了推荐过程中的计算消耗和存储消耗。且能提高推荐质量，通过在数据集上的仿真实验结果表明，提出的算法能显著提高推荐质量。

关键词：两阶段联合哈希算法；协同过滤；主成分分析；迭代量化方法；海明距离

## Abstract

## 第一章 引言

### 1.1 研究背景及意义

当今的信息时代，信息技术的迅速发展导致网上信息量呈爆炸式增长，信息量的爆炸式增长也使得信息的质量良莠不齐，低质量甚至无价值的东西使得用户在面对大量信息时难以获得真正需要的信息，从而降低了对信息的使用效率，导致了信息过载的问题。个性化推荐系统（Personalized Recommendation System）应运而生，作为解决信息过载问题强有力的方法之一，推荐系统根据用户的信息需求、兴趣等产生个性化的推荐，为用户提供更高质量的信息服务，与此同时，也为用户节省了大量的时间和精力。

个性化推荐系统能够利用系统中用户的资料、行为和历史记录等信息以及项目的标签，内容等信息，通过计算相似度的方法对用户的偏好进行预测，来达到将特定的商品或信息推送给目标用户的目的。个性化推荐系统可以主动的为用户提供感兴趣的商品或信息，而不需要用户的主动操作，系统在后台就能够对用户的偏好进行计算预测，这进一步提高了用户或许感兴趣信息的效率。性化推荐技术中的一种主流方法是协同过滤技术，它主要利用系统中的其他用户的信息来预测当前用户对系统中项目的可能评分，进而选择出可能符合当前用户偏好的项目。传统的协同过滤推荐技术通常将用户对项目的评分矩阵作为用户或项目的表示，然后通过对评分矩阵进行处理，预测用户对项目的评分，进而将预测的结果推荐给用户。

此外，随着大数据时代的来临，由于数据量的体积在日益极速膨胀，系统中的用户和项目的数据呈现高维特征，使得传统的依赖于计算用户或项目的相似度来获取近邻的算法计算和存储开销太高，也大大降低了推荐结果的计算速度，这给传统的协同过滤算法带来了巨大的挑战。研究者们在面对大数据的规模制约着传统算法的有效性问题时，提出了两种思路：一种是对现有算法进行改造，设计可以扩展从而能够应对当前大规模数据环境的算法，算法伸缩的技术有分布式学习和随机优化等方法；另一种是对数据进行降维或消减，使其达到传统算法能处理的规模；降维或消减的方法主要包括随机投影和哈希学习等技术。随机投影通过随机的方式产生投影方向，达到降低数据维度的目地。由于其简单快捷，适用于大数据环境，也在解决分类、最小二乘等问题中得到应用与发展；哈希学习通过机器学习机制将数据映射成二值码的形式，从而得到特征的二进制哈希表示，在学习的过程中，要使得哈希码尽可能地保留原数据空间中的近邻关系。哈希学习有利于减小存储空间，提高学习速度，降低数据维度，因为其能显著减少数据的存储和计算开销，已经成为机器学习领域和大数据学习领域的研究热点，在信息检索、模式识别、计算机视觉等领域得到广泛的研究与应用。

因此，本文提出了两阶段联合哈希的协同过滤算法，将传统的协同过滤算法与哈希学习相结合，实现了高质高效的个性化推荐。在大数据时代背景下，本文提出的方法在应对大规模数据协同过滤的扩展性问题和推荐效率问题时更有价值。

### 1.2 本文的主要工作

本文针对大规模数据背景下的个性化推荐系统，提出了一种两阶段联合哈希的协同过滤推荐算法，利用了高效的基于哈希编码的近似最近邻搜索，实现了高效、高质量的个性化推荐。

本文的主要工作包括

首先，对评分数据从用户或项目视角应用主成分分析技术，得到该视角的低维特征表示，对其应用迭代量化技术进而生成该视角的二值码，从而实现了对评分数据全局特征的提取，也为进一步生成另一视角的二值码奠定了基础。

然后，基于上一过程中得到的其中一个视角的二值码，用已有的评分信息约束用户与项目在海明空间的距离，生成另一视角的二值码，进而实现了对评分数据局部特征的提取，也为利用得到的用户和项目的二值码进行高效的偏好预测做好了准备。

通过以上两个过程的工作，有效的减少了推荐过程中的计算消耗和存储消耗。且能提高推荐质量，通过在数据集上的仿真实验结果表明，提出的算法能显著提高推荐质量。

### 1.3 本文的组织结构

### 1.4 本章小结

## 第二章 个性化推荐系统

当今的信息时代，信息技术的迅速发展导致网上信息量呈爆炸式增长，信息量的爆炸式增长也使得信息的质量良莠不齐，低质量甚至无价值的东西使得用户在面对大量信息时难以获得真正需要的信息，从而降低了对信息的使用效率，导致了信息过载的问题。个性化推荐系统（Personalized Recommendation System）应运而生，作为解决信息过载问题强有力的方法之一，推荐系统根据用户的信息需求、兴趣等产生个性化的推荐，为用户提供更高质量的信息服务，与此同时，也为用户节省了大量的时间和精力。

一般而言，个性化推荐系统就是为系统中的用户推荐感兴趣的项目的工具和技术。这些项目的形式可能包括音乐、电影和书籍等，具体来说，推荐系统需要预测目标用户对项目集中的项目的评分或偏好，为进一步的系统的推荐提供决策，实现对用户的个性化需求的技术支撑。

目前，个性化推荐系统已经在互联网领域的各个方面得到了广泛的应用，较为突出的有电子商务、社交网络和在线影音等。以国内著名电商网站当当网为例，网站在各个层面上实现了个性化推荐，如根据当前商品的推荐（图1）和根据历史浏览记录的推荐（图2）。个性化推荐技术一般可分为三种类型，分别是：基于内容的推荐技术、基于协同过滤的推荐技术和基于混合技术的推荐技术。



图1



图2

#### 2.1基于内容的推荐技术

基于内容的推荐技术是根据项目的内容或者通过项目的内容提取出的项目的特征，来为目标用户推荐跟其过去偏好的项目相似的结果。基于内容的推荐算法一般会根据用户的历史信息建立该用户的用户概貌（user profile），得到用户的偏好的描述，然后根据项目集中项目的特征来计算候选项目与目标用户偏好之间的相似度，进而对得到的相似度计算结果进行排序，为用户推荐相似度最高的项目。所以，这种方法的核心在于为用户和项目提取合适的特征和基于提取的特征上的相似度计算。

基于内容的推荐技术有着简单，有效的优点，在广告过滤，垃圾邮件过滤，色情网页过滤等场景下有着不错的发挥；基于内容的推荐技术也有着极其明显的缺点，如难以区分项目内容的品质以及风格，而且无法为用户提供新的感兴趣的推荐服务，它只能发现和用户已有的兴趣相似的项目。

#### 2.2基于协同过滤的推荐技术

协同过滤(Collaborative Filtering)最早由Goldberg等提出并应用于一个电子文档过滤系统Tapestry[1],之后的几十年来，不断的有专家和学者加入到研究这个课题的队伍中来。其中，美国明尼苏达州立大学的GroupLens所设计实现的Grouplens系统[2]，基于用户对新闻的偏好在过去与将来都是一致的，从而利用用户之间的协作，从大量的Usenet新闻中发现用户们感兴趣的新闻，给用户提供个性化的推荐服务。同样来自明尼苏达州立大学的Movielens系统[3]也对协同过滤系统的兴起起到了推动作用，Movielens系统所使用的数据集被公开后，为后来的众多研究者提供了便利，也使得协同过滤算法获得更为广泛的关注与研究。

2006年，美国Netflix公司举行的一场竞赛，将协同过滤算法的热度推到了高峰，Netflix公开了该网站用户对电影评分的部分数据供参赛者实验使用，最后该项竞赛的一百万美元奖金由BPC团队获得，该团队在矩阵分解方法的基础上引入了隐性反馈[4]，结合了近邻模型作为局部修正，又引入了近邻模型中的相似度矩阵分解思想，由此，矩阵分解方法在实践中得到了很大程度的完善，也推动了协同过滤技术的更深发展。

协同过滤的基本思想是根据具有类似兴趣的用户的偏好来对用户进行预测进而完成推荐，利用一个用户对其它项目的评分以及整个用户集过去的评分来预测这个用户对未评分项目的评分。协同过滤的基本思想是：根据用户群体的历史行为，找到与目标用户兴趣偏好相似的若干其他用户作为目标用户的邻居, 然后根据邻居用户的偏好信息计算出目标用户的兴趣列表。其基本思想就是日常生活中, 人们往往会选择兴趣爱好相似的朋友的推荐。协同过滤就是把这一思想运用到推荐系统中来, 利用其他用户对某一项目的评价和用户之间的相似度来向目标用户进行推荐。

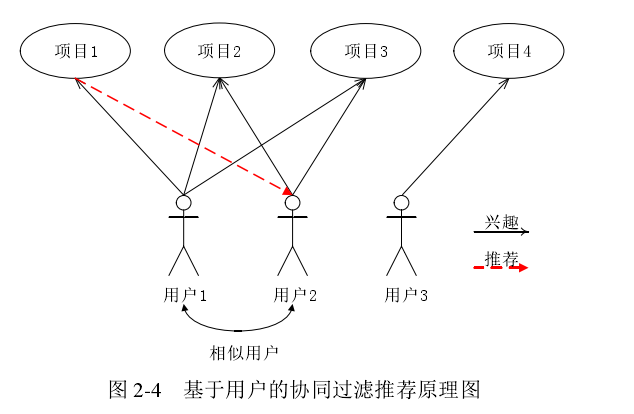
协同过滤是推荐系统应用最为广泛的技术之一，它利用信息之间的关联程度把不相关的信息过滤掉，只保留最有价值的部分信息。基于协同过滤的推荐算法基于以下事实：用户对跟它兴趣相似的其他用户感兴趣的东西往往也会感兴趣。与基于内容的推荐不同的是，协同过滤推荐不用关心项目的具体内容，也不需要给项目属性作标签，所以能够应对复杂的难以表达的场景，而且协同过滤推荐的结果具有新颖性的特点，即常常能给用户推荐新奇的项目，进而提高了用户的体验。

基于协同过滤的个性化推荐技术一般可以分为基于内存的协同过滤(Memory-based CF)、基于模型的协同过滤(Model-based CF)。

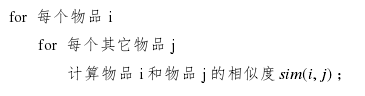
##### 2.2.1 基于内存的协同过滤推荐

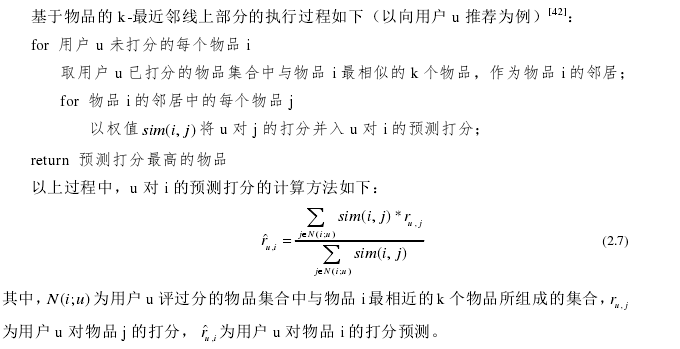
基于内存推荐是协同过滤推荐方法中广为流行的一类方法，其原理是通过直接计算用户或项目的相似度然后利用 K 近邻(KNN)算法来寻找邻居并以此产生推荐，基于内存推荐按照计算相似度的对象的不同可以分为基于用户和基于项目两种。

基于用户的推荐算法的基本思想是根据用户过去对项目的偏好通过计算与目标用户相似度最高的 K 个近邻用户，再计算近邻用户对该项目的评分的加权平均，即为该用户对项目的预测评分，如图 2-4 所示，用户 1 和用户 2 具有相似的历史行为，因此可以认为他们是相似用户，对于用户2来说，用户1即为用户2的近邻，由用户 1对项目1感兴趣可以判断用户2对项目1也感兴趣，因此可以对用户2推荐项目 1。在实际的应用场景下，一般将推荐的过程分为线上和线下两部分，线下主要是通过计算各用户的相似度来得到各用户的近邻，并将邻域信息存储下来；而线上部分就省去了耗时的相似度计算加快了响应速度，具体来说就是只需要查询邻域表来预测目标用户感兴趣的项目即可。

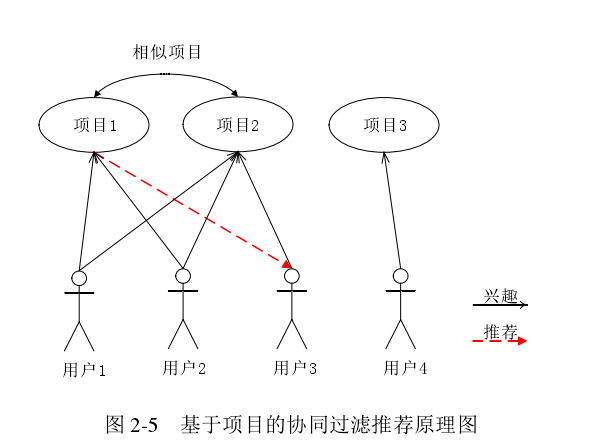


线下部分的执行过程如下：





基于项目的推荐算法与基于用户的推荐算法原理类似，其基本思想是用户的项目的预测评分可以由与该项目相似度最高的 K 个邻居项目获得该用户的评分加权平均计算得到，如图 2-5 所示，对项目 1 感兴趣的用户也都对项目 2 感兴趣，因此项目 1 和项目 2 的相似度较高，它们是相似项目，而用户 3 目前对项目 2 感兴趣。



在实际应用中，基于项目的推荐算法比基于用户的推荐算法效果更好。基于用户的推荐算法运行时间随着用户数量的增长而增长，基于项目的推荐算法的运行时间随着项目数量的增长而增长。对于大多数网站，其用户规模往往比项目规模大很多，用户数量比项目数量更多，所以基于项目的推荐算法往往能够带来显著的性能提升。另一方面，项目之间的相似性相对于用户之间的相似性更为稳定，随着时间的推移，用户的认知很可能发生变化，导致之前计算的用户相似度不准确。另一方面，随着时间的推移，每件项目积累的用户会越来越多，最终项目间的相似性趋于收敛，不再发生剧烈的抖动。而这两种算法都高度依赖相似性的准确度，所以基于项目的推荐算法要优于基于用户的推荐算法。

相似性度量是基于用户的推荐和基于项目的推荐很重要的一项工作，相似性度量用于发现相似用户或项目，它关系着能否准确发现近邻用户或项目，进而影响推荐结果的准确性。相似性度量主要基于对象的描述特征或属性，常利用各对象属性集合或特征向量间的相似系数、 相关系数以及距离等来计算。常用的相似性度量的方法有四种，如下：

1. 皮尔逊相关系数

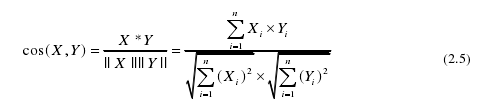
皮尔逊相关系数可以用作度量两个变量之间的相关程度，其取值范围为[-1,1]，1表示这两个变量完全正相关，0表示不相关，-1则表示这两个变量完全负相关。其计算方法如下：



观察上式可以看出，皮尔逊相关系数实际上是两个序列协方差比上方差的乘积，其中，协方差反映了两个序列变化趋势的一致性，除以方差的乘积是用来对协方差进行归一化。

1. 余弦相似性

余弦相似性也把用户看作 n 维空间中的点，并根据这个进行相似性的衡量。假设有两条从原点出发，分别到这两个点的射线。如果两个用户相似，则它们的打分也相似，空间位置很接近，两条射线的方向也会差不多，即两条射线之间的夹角会比较小。反之，如果两个用户不相似，则相应的两个点会相隔较远，从原点到这两点的射线可能指向不同的方向，形成的夹角会比较大。与欧式距离类似，这个夹角同样可以用来度量相似性。在这种情况下，夹角余弦代表相似度值，余弦值越大，表示相似度越高，最大为 1，最小为-1。余弦相似性度量在信息检索、文本挖掘中应用的比较多，用来比较两份文本文档之间的相似程度，文档可以看成由词组成的向量。余弦的计算公式为：

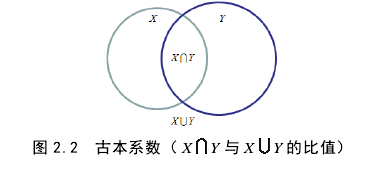


1. Jaccard 相似系数

又称 Tanimoto 系数， 是相似系数的扩展，用来比较二元向量的相似性和分散性的概率。用户间的相似性通过两个向量集合的交集与并集之比来度量， 用户间的相似性 sim( A，B) 为:



基于Jaccard系数的相似性度量不管用户对项目的偏好值是高还是低，只关心用户是否表达过偏好。它用两个用户都打了分的项目的数量除以至少有其中一个用户打过分的项目的数量得到的，也就是说，Jaccard系数实际上是两个打分项目集合交集大小与并集大小的比值，如图 2.5 所示。



(4).基于欧氏距离的相似度

用户之间的欧氏距离是一种 Ld 范式，当 d = 1 时称为曼哈顿距离。可见， 欧氏距离越大，相似度越小。d = 2 时，欧氏距离 dis( A，B) 为



基于内存的协同过滤技术其原理通俗易懂，也易于实现，最重要的是它不关注项目的内容，特征等信息，因此不需要处理项目的内容和特征等信息，这使得基于内存的系统过滤技术通用性的很强，适用于较多的场景。然而，这类方法也存在着一定的缺陷：首先，因为在现实的场景下，得到的用户数据往往是高度稀疏的，这会导致基于内存的协同过滤技术计算用户或项目的相似度时得到的结果不够准确，进而导致了推荐的性能显著下降；另外，在处理历史信息较少的用户或项目，如系统中新增加的用户或项目，协同过滤技术无法对其展开与系统中的其他用户或项目的相似度计算，进而对其进行处理，也就无法将新的项目推荐给用户或者为系统中新增加的用户推荐项目，一定程度上可以说，推荐算法对这类用户和项目是失效的；最后，随着互联网技术的高速发展，系统中的数据规模越来越大，在面对大规模的数据时，基于记忆的协同过滤方法所需要的计算成本非常高，甚至可能无法计算，这意味着算法无法很好的适应大规模数据的系统，即算法的扩展性不强。

##### 2.2.2 基于模型的协同过滤推荐

基于模型的协同过滤推荐是协同过滤推荐的重要推荐算法之一，基于模型推荐事先利用训练数据集离线学习一个预测模型，再将模型应用到在线系统进行推荐。与基于内存的协同过滤推荐算法不同的地方在于，基于模型的协同过滤推荐不是利用启发式规则预测用户对项目的评分，而是采用机器学习和统计的方法对已有数据建立模型，利用建立好的模型来进行评分预测。基于模型的协同过滤推荐性能的好坏与建立的模型有比较大的关系，好的模型能挖掘出更多的用户和项目之间的潜在关系进而获得更准确的推荐结果，但是在现实场景下由于建立模型相对比较耗时，一般都只是在离线完成模型的建立。经典的基于模型的协同过滤推荐方法主要有奇异值分解法、贝叶斯分类法、聚类法、关联规则法等。

**奇异值分解模型**用于协同过滤的主要思想是用户对项目的偏好是基于用户对这些项目的隐含特性的喜好的,而且在项目之间也存在一些共有的特征。用户对某一项目评分高是因为他们对这些潜在的，共有的特征的评分较高,所以通过把系统中用户对项目的评分用线性代数方法分解为一些潜在特征,然后根据用户对这些潜在特征的感兴趣程度来预测用户对他所未评分项目的偏好。

具体来说，用户-项目的评分矩阵R可以用公式(2-6)来表示：

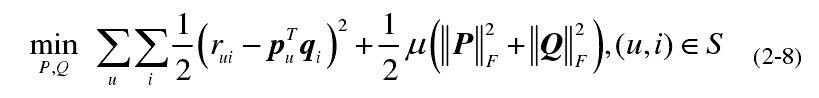
R = PQ’ (2.6)

其中P(N\*K)，Q(M\*K)分别为表征所有用户和所有项目的因子矩阵，K代表了每个用户和每个项目的特征可以用前K个奇异值所对应的K个奇异向量表征，其取值代表了保留了K个维度的特征信息，K的取值很重要，如果太小,就无法得到原始评分矩阵中重要特征,如果太大,就会失去降维的意义,因此需要通过多次试验来确定要保留的维数，一般满足K<<Min(N,M)。式(2-6)实际上是矩阵理论中矩阵奇异值分解公式(2-7)的变形。

R=PΣQ (2.7)

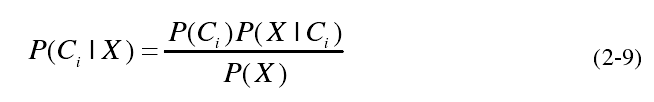
奇异值分解模型中的奇异矩阵P、Q中各个元素作为待估计参数，以用户-项目评分矩阵

中的现有数据来作为训练数据对训练模型，也就是求解优化问题(2-8)，其中μ是权重系数，来缓解模型的过拟合。

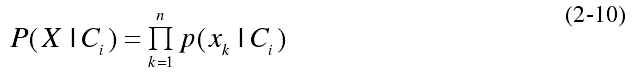


优化问题(2-8)求解后得到P、Q奇异矩阵，然后带入公式(2-6)即可得到预测的用户对项目的评分值。奇异值分解模型将评分预测问题转化为了奇异矩阵的参数估计问题，由于奇异矩阵P、Q中的待估计参数远小于原始用户-项目评分矩阵的元素个数，因此该方法能起到降低矩阵维数的作用，[2]中的实验结果表明奇异值分解方法在某些场合下能改善推荐系统的可扩展性问题,在稀疏的评分矩阵上效果比较好,但是如果原始的评分矩阵稀疏程度较大,其推荐效果甚至不如基于用户的协同过滤推荐。

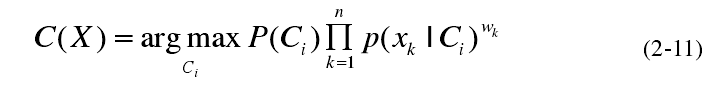
**贝叶斯分类模型**方法的基本思想是把用户对项目的评分分成若干个类，如评分范围属于[1,5]的可以分成5类，每个类代表评分值1、2、3、4、5，然后可以利用目标用户与项目的属性特征来计算该目标用户对该项目的评分属于各个类别的概率，最后取概率最大的评分类中评分值。贝叶斯分类方法的具体数学描述如下：若用户-项目评分矩阵中的每个数据样本有n个属性(用户和项目)，则每个数据样本(评分)可以用一个特征向量表示为，又如果所有的数据样本的分类数为m，分别用表示，对于待分类的数据样本X，根据贝叶斯定理(2-9)可以计算其属于某个类的条件概率：



其中，P（Ci）是样本属于Ci类的先验概率，P（X|Ci）是样本在条件为Ci类情况下属性为X的条件概率，由于P（X）对于所有的类别都是常数，最大化后验概率P（Ci|X）可以转化为最大化P（Ci）P（X|Ci）。为了更方便的计算P（X|Ci）的值，贝叶斯分类器通常都假设各个类之间是相互独立的，也就是说各个属性的取值与其他属性无关。因此有：



先验概率P（x1|Ci）, P（x2|Ci）,…, P（xn|Ci）能够从训练数据集中统计求出。基于此，对于一个类别未知的样本，可以先分别计算该样本属于每一个类别的概率，然后选择其中概率最大的类别作为其类别，即：



**聚类模型**主要有三类：用户聚类、项目聚类和用户-项目联合聚类。用户聚类一般是利用聚类算法将用户集聚类成若干类，使得每一类中是具有较高相似性的用户，在推荐过程中，先在目标用户所在的聚类中心上寻找目标用户的若干个近邻用户，然后根据这些近邻用户对某项目的评分加权即可得到目标用户对该项目的预测评分。同理，项目聚类是对项目集进行聚类，通过在项目子集中寻找某项目的近邻来实现目标用户对该项目的评分预测。用户-项目联合聚类又叫用户-项目协同聚类，考虑到用户和项目的相关性，对用户集和项目集都进行了聚类，将用户聚类和项目聚类两个步骤依次循环迭代直至最终收敛。应用于聚类模型的聚类算法中最常见的是 K-means 算法[44]，K-means是机器学习领域中的一种非监督学习算法。以项目聚类为例，K-means 聚类算法的基本步骤如下：

Step1、设定聚类中心数 K，并从用户-项目评分矩阵中寻找到评分数最多的 K

个项目作为初始聚类中心点，记为

Step2、对于项目集中每个项目 v，计算 v 与各个聚类中心的距离或相似度SIM，得到聚类Ci，将该项目添加到聚类Ci中。

Step 3 、 修 改 聚 类 中 心 ， 计 算 每 个 聚 类 的 均 值 ， 作 为 新 的 聚 类 中 心，如果聚类中心未发生改变或迭代步数达到最大的设定值，则算法终

止，得到 K 个聚类，否则跳转到 Step 2 继续迭代。

基于聚类模型的推荐在聚类中心上搜索目标用户/项目的近邻，相比在全局上做近邻搜索，最近邻的寻找范围大大的缩小了，这大大减少了推荐过程的计算，提升了推荐的速度，在一定程度上有效的缓解了协同过滤推荐系统面临的扩展性问题，然而，在聚类中搜索近邻这种做法依赖于聚类模型的优劣，相比在全局上搜索近邻，聚类模型在改善了扩展性的同时也牺牲了一定的推荐准确性。

**关联规则**挖掘因为其能够发现不同商品在销售过程中的相关性，在零售业有着广泛的应用。作为数据挖掘领域的一个重要分支，关联规则挖掘旨在发现数据集中项集之间的关联。若是项的集合，D是数据库中事务的集合，每个事务T是不同项的集合，且有。规则是形如的蕴含式，其中A,B是一个项集，即,并且。规则在事务集D中成立，具有置信度和支持度，同时满足最小置信度阈值和最小支持度阈值的规则即为强关联规则。

关联规则的挖掘过程可以分为两步：第一步在样本集上生成所有的频繁项集，这些项集的出现频率至少满足大于等于定义的最小支持度。第二步由第一步得到的频繁项集生成强关联规则，规则必须同时满足最小支持度和最小置信度。在以上的两个过程中，第一步要在整个数据库规模上求解，故如果数据库规模很大，则解空间也非常大，导致计算量很大，会影响算法的效率。

关联规则法用于协同过滤系统是[3]首先提出，通过利用Apriori算法挖掘用户的历史历史记录的关联性来进行推荐。Apriori算法首先从聚集树发现匹配用户当前访问操作路径的关联规则,然后再根据推荐度因子的大小确定推荐项,推荐度因子定义为关联规则的置信度乘以距离因子[4]，关联规则模型的生成可以通过离线的方式进行,因此可以保证系统进行有效的，实时的推荐。

#### 2.3基于混合技术的推荐技术

基于内容的推荐与基于协同过滤的推荐都有着各自的优点和缺点，因此，将这两类方法结合打开了新的思路。基于混合技术的推荐方法可以分别使用基于内容的推荐方法和基于协同过滤的推荐方法，然后将两种方法生成的结果进行结合；也可以将一些内容特征作为协同过滤中所使用的信息的一部分，或者反过来把用户和项目的历史数据作为内容特征用于基于内容的推荐技术等等。研究表明，与纯粹的基于内容推荐或协同过滤推荐相比，混合推荐技术可以取得更有效的推荐性能，同时，也有助于应对和处理协同过滤推荐中的冷启动和稀疏性等问题。

#### 2.4 本章小结

## 第三章 哈希学习技术

### 3.1最近邻检索技术

最近邻检索技术（Nearest Neighbor Search, NN）是推荐技术中的一个重要组成部分，尤其是在基于协同过滤的推荐技术中。尽管协同过滤在个性化推荐方面取得很大成功，但其本身也存在着关键问题制约着其进一步发展，随着站点结构、内容复杂度和用户人数的增加，推荐系统需要处理的数据记录呈爆炸式增长，对于这些海量数据的快速计算出推荐结果，如前文所述，协同过滤技术的本质思想是根据相似用户的偏好或与偏好项目相似的物品来为用户进行推荐。因此，如何尽量快速且准确的找到与当前用户或项目最相似的对象就成为了一个无法回避的问题。

### 3.2 经典哈希学习

### 3.3 本章小结

近几年来，哈希学习因为其有利于减小存储空间，提高学习速度，降低数据维度的优势成为了研究者们所青睐的用于解决协同过滤扩展性问题的有效方法。

哈希学习自2007年被引入到机器学习领域，已经迅速发展成机器学习领域和大数据领域的研究热点[9]，在各领域都有优异表现。哈希学习首先由zhou 等在2012年推介到推荐系统领域，在推荐系统中，从数据集中直接训练出用户-项目的二值码是一个NP-难问题[12]。为了从训练集上得到所需的二值码，Zhou等采用了两步学习策略，即先对原空间的样本采用度量学习进行降维，得到低维空间的实数向量的表示，再对得到的实数向量进行量化得到二进制哈希码，在[12]中，作者设计了采用数据集的中位数作为阈值和正交变换的两种量化方法，设计并实现了CFCodeReg和CFCodepair两个分别对应于平方损失和成对损失的度量学习模型，并在MovieLens、EachMovie和Netflix数据集上取得了比谱哈希[14]、矩阵分解模型更好的推荐结果。Zhou等作为将哈希学习引入到协同过滤系统中的先驱，填补了协同过滤系统中哈希学习技术的空白，所采用的两步学习策略给后来的研究者提供了依据与指示。相似的还有zhang等[11]，但是zhang认为在度量学习阶段，内积并不等于相似度，用内积来表示相似度的做法会导致最后的倾向度预测的精度的损失，随后他们提出了正则化后的用户、项目特征来计算余弦相似性，并分别的基于量级和相位来量化得到二进制的哈希码，取得了更高质量的推荐效果。在[10]中，liu等认为在用户与项目这两个维度内部实体之间的联系在生成简约二值码的过程中起着更重要的作用，而不仅仅是用户与项目之间的相互联系，在他们的度量学习阶段，所构造的目标方程包含了每个维度内实体哈希后的损失和整体的维度之间的实体哈希后的损失两个部分，在采用了奇异值分解的技术对目标问题进行优化得到简约的二值码过程中，也得到了对于之后加入的用户或项目进行哈希的哈希函数。以上基于二阶段的哈希方法均取得了精确度的提升，然而，在[13]中，zhang等认为上述的采用二阶段过程的哈希学习方法损失了大量的信息，也导致降低了最后系统的性能。作者设计了实值矩阵分解方法与量化为二值码的矩阵分解方法的对比实验，得到二阶段过程会降低算法的性能的结论。最后作者采用了直接解决离散最优化的问题，通过将两个阶段联合起来，并引入连续值的辅助矩阵，建立辅助矩阵与原矩阵的联系，将约束条件转移到连续值的辅助矩阵上，从而将优化问题分解为包含了辅助矩阵的四个分问题，然后设计了并行离散优化算法和基于小规模的奇异值分解算法解决上述的四个分问题，得到平衡的正则化后的二值码，取得了比其他基于二阶段哈希学习方法更好的推荐效率。



哈希学习作为主要的降维方法，它通过机器学习机制将数据映射成二值码的形式，从而得到特征的二进制哈希表示，在学习的过程中，要使得哈希码尽可能地保留原数据空间中的近邻关系。哈希学习有利于减小存储空间，提高学习速度，降低数据维度，这些优点在推荐系统中可以得到充分的利用，通过借助哈希学习思想对协同过滤系统进行扩展，它可以有效的应对大规模数据对协同过滤系统的制约，且具有运行速度快，占据的存储空间小等特点。当前的基于哈希技术的协同过滤方法[10，11，12]主要是基于二阶段的研究，即第一阶段先对原空间的样本采用度量学习进行降维，得到低维空间的实数向量表示，在第二阶段中，对得到的实数向量量化得到二值码，这种基于二阶段的方法大大增加了信息的损失，因为在第一阶段的降维过程中就造成了信息损失，在第二阶段的量化过程也会造成信息的损失。所以如[13]所提出的离散协同过滤方法，采用在度量学习的过程中量化直至得到收敛的二值码的策略，有效的减少了信息损失，也使得所得到的二值码更高效。然而，在[13]中作者虽然提出了更有效的学习策略，但是在[13]中的度量方程中，只关注了用户与项目之间的关系，并没有考虑到项目的近邻关系所反映的潜在项目特征对所生成二值码的影响。本文在总结了前人工作的基础上，提出了基于项目潜在特征的哈希协同过滤方法，区别于其他采用二阶段学习策略的方法，本文通过引入辅助变量，重构出可以通过正则化算法解决的目标，然后采用离散最优化技术来解决正则化算法的子问题，最后逐位迭代至收敛得到用户/项目的二值码，这种学习策略可以保留更丰富的信息，使推荐过程更加高效，准确。

### 2.3 相速度度量方式介绍

#### 2.3.1 皮尔逊系数

#### 2.3.2 余弦

#### 2.3.3 海明距离

## 第四章 两阶段联合哈希的协同过滤算法

### 3.1 ITQ

### 3.2 基于用户视角的两阶段联合哈希算法

### 3.3实验设计与分析

#### 实验数据集

#### 实验环境

#### 实验过程

#### 评价指标

#### 实验结果

### 本章小结

## 第五章 总结与展望

## 参考文献

1. 电子商务的个性化协同过滤推荐算法研究
2. Application of Dimensionality Reduction in Recommender Systems-A Case Study
3. Mining Navigation History for Recommendation
4. 协同过滤系统的稀疏性与冷启动问题研究

## 在校研究成果

## 致谢