



北京大学

硕士研究生学位论文

题目： 内容感知的个性化
推荐算法研究

姓 名： 李雪鹏

学 号： 1301210747

院 系： 软件与微电子学院

专 业： 软件工程

研究方向： 数据挖掘与商务智能

导师姓名： 李杰教授

二〇一五年七月

版权声明

任何收存和保管本论文各种版本的单位和个人，未经本论文作者同意，不得将本论文转借他人，亦不得随意复制、抄录、拍照或以其他方式传播。否则，引起有碍作者著作权之问题，将可能承担法律责任。

摘要

随着移动互联网、O2O 等概念的兴起,本地生活服务等各项传统服务已经逐步电子商务化,电子商务服务商品化趋势越来越明显,这使得商品本身属性变得越来越复杂,因此个性化推荐系统正面临着一个全新的挑战——如何综合考虑用户对商品的各种属性的偏好程度,给出更加合理化的推荐商品列表。另一方面,伴随着各大互联网公司的迅猛发展以及并行计算、云计算、hadoop 框架、MapReduce 编程模型等一系列技术的发展,现有推荐系统能够收集到更多样的数据,并具有更强的海量数据处理能力,推荐系统变得越来越智能,可以说个性化推荐系统正迎来一个全新的机遇——如何充分利用各类数据(用户偏好数据、商品内容数据、用户评分数据等等),发挥现代计算机集群具有的大规模并行计算能力,进一步提升推荐系统的质量。

为了更好地利用现有推荐系统能够收集到的各类数据,本文提出了一系列旨在充分利用用户偏好和商品内容数据的概率矩阵分解(PMF)扩展模型:

1. 第一种模型 cPMF 旨在更好地利用系统收集到的商品内容数据。该模型在 PMF 模型的基础上,进一步将表征商品的特征向量分为隐式和显式两个部分,其中显式的部分是根据商品内容数据计算得到的。
2. 第二种模型 GcPMF 使用余弦相似度来表达商品显式特征向量与对应用户隐式特征向量的关联,这区别于 cPMF 模型中的内积形式。
3. 第三种模型 CsPMF 旨在增强 cPMF 的灵活性。该模型在 cPMF 的基础上增加了两个系统超参 (w_1, w_2),用于调节显式特征和隐式特征在模型中的比重。
4. 第四种模型 GCsPMF 旨在增强 GcPMF 的灵活性。该模型在 GcPMF 的基础上增加了系统超参 (w_1, w_2),用于调节显式特征和隐式特征在模型中的比重。
5. 第五种模型 BCsPMF 旨在更好地利用系统收集到的商品内容数据以及用户内容数据。与 cPMF 类似,BCsPMF 将用户和商品的特征都划分为隐式和显式两个部分,该模型使用向量内积值来表达商品显式特征向量与对应用户隐式特征向量的关联。

本文在 MovieLens-1M 数据集上对 PMF、cPMF、GcPMF、BCsPMF 模型进行了对比验证,实验结果表明使用了内容相关数据的 cPMF、GcPMF、BCsPMF 模型效果要明显好于 PMF 模型。

关键词: 个性化, 推荐系统, 概率矩阵分解, 内容感知

Research on the content-aware personalized recommendation algorithms

Li Xuepeng (software engineering)

Directed by Li Jie

ABSTRACT

With the development of mobile Internet、O2O, local living services, and other traditional services have been gradually ecommerce, e-commerce service commoditization trends are becoming evident, which makes the product itself attributes become more complex, personalized recommendation system is facing a new challenge - how to consider the degree of user preferences for commodities of various attributes gives rationalization recommended product lists. On the other hand, with the rapid development of major Internet companies as well as parallel computing, cloud computing, hadoop framework, MapReduce programming model and a series of technology, the existing recommendation system is able to collect more samples of data, and have a stronger capacity processing the massive data, recommended systems become more intelligent. As a result, personalized recommendation systems can say is ushering in a new opportunity - how to make full use of various data (user preference data, commodity content data, user data, etc.), play modern computer clusters with large-scale parallel computing capabilities to further enhance the quality of the recommendation system.

In order to make better use of existing recommendation system able to collect all kinds of data, this paper presents a series of matrix factorization extended model for the purpose to be in full use of user preferences and probability commodity content data:

1. The first model cPMF system is designed to make better use of data collected from the merchandise content. The model is based on PMF model, further characterization of the goods into the feature vector implicit and display two parts, some of which display the contents of the data is based on commodity calculated.
2. The second model GcPMF uses cosine similarity Displaying associate degree to express the feature vector corresponding to the user implicitly eigenvectors, inner product form that is different from cPMF model.
3. The third model is designed to enhance cPMF CsPMF flexibility. The model is based on cPMF on the increase over the two systems parameters (w_1 , w_2), is used

to adjust the display characteristics and features in the model implicit GDP.

4. The fourth model is designed to enhance the flexibility of GcPMF of GCsPMF. The model is based on GcPMF on the increase over the reference system (w_1, w_2), is used to adjust the display characteristics and features implicit in the model of proportion.
5. The fifth model BCsPMF system is designed to make better use of data collected from the merchandise content and user content data. Similarly with the cPMF, BCsPMF model will feature vectors within the product user and commodities as a user preference points for merchandise, the model uses it to express merchandise display feature vector associated with the corresponding user implicit feature vectors.

In this paper, MovieLens-1M data sets on the right PMF, cPMF, GcPMF, BCsPMF model were compared to verify the experimental results showed that: Use the cPMF contextual data, GcPMF, BCsPMF model results to be significantly better than the PMF model.

KEY WORDS: Personalized, Recommendation System, Probabilistic Matrix Decomposition, Content-aware

目录

第一章 绪论	1
1.1 选题背景.....	1
1.2 研究目标与内容.....	2
1.3 论文贡献及创新	3
1.4 论文结构与组织.....	3
第二章 国内外研究现状.....	5
2.1 基于协同的推荐	5
2.2 基于内容的推荐	8
2.3 基于知识的推荐	10
2.4 组合推荐	11
2.5 基于矩阵分解的推荐	12
第三章 内容感知的个性化推荐模型	13
3.1 概率矩阵分解模型 (PMF)	14
3.2 内容感知的概率矩阵分解模型一 (cPMF)	17
3.3 内容感知的概率矩阵分解模型二 (GcPMF)	21
3.4 内容感知的概率矩阵分解模型三 (CsPMF)	25
3.5 内容感知的概率矩阵分解模型四 (GCsPMF)	29
3.6 内容感知的概率矩阵分解模型五 (BCsPMF)	34
第四章 模型实现及验证.....	39
4.1 数据集说明	39
4.2 模型评估指标说明	40
4.3 开发环境说明	40
4.3.1 系统配置	41
4.3.2 libmf 介绍	41
4.4 模型基本设置	41
4.4.1 模型参数设置	41
4.4.2 模型初始化	42
4.5 模型效果对比验证	42

4.5.1 PMF vs cPMF 效果对比.....	42
4.5.2 cPMF vs CsPMF 效果对比	45
4.5.3 PMF vs GcPMF 效果对比.....	46
4.5.4 GcPMF vs GCsPMF 效果对比.....	48
4.5.5 PMF vs BCsPMF 效果对比	49
4.5.6 PMF vs Other 效果对比	51
第五章 结论及展望	53
参考文献	55
致谢	58
北京大学学位论文原创性声明和使用授权说明	59

第一章 绪论

1.1 选题背景

随着互联网的日益成熟，越来越多的信息在互联网上传播，用户再也不用抱怨信息量太少了，恰恰相反，现在是有海量的信息等待着用户去选择，用户不能更不想在逐一查看每一条信息之后再做出正确的抉择。因此互联网不仅需要海量的信息，更重要的是需要有一些有效的工具来帮助用户消除信息过载的问题，因此信息检索或者搜索引擎技术一直在发展，在默默地为互联网的发展提供支撑。

然而，伴随着电子商务的逐渐发展，各种各样的商品开始在互联网上出售，面对琳琅满目的商品，用户迫切地希望电子商务系统能够更智能一些，更主动一些，能够将用户最喜爱的商品直接呈现在眼前，因此个性化推荐系统^[1-2]应运而生。个性化推荐系统不仅要解决信息过载的问题，更重要的一点是它要根据用户的偏好主动地给用户推荐商品或者信息。现在已经有很多比较成熟的推荐系统，像亚马逊的商品推荐系统、谷歌的新闻推荐系统、豆瓣的音乐推荐系统等等，它们都会根据用户主动提供的偏好信息(如豆瓣 FM 的推荐系统允许用户给一首歌加上喜欢或者不喜欢的标记)以及用户的历史行为（亚马逊的推荐系统则会记录用户的历史购买、点击、评论等行为）分析出用户的兴趣偏好，然后再结合商品或者信息自身的固有属性主动地给用户推荐用户可能需要的商品或者信息。

随着移动互联网^[3]、O2O^[4]等概念的兴起，本地生活服务等各项传统服务已经逐步电子商务化，电子商务服务商品化趋势越来越明显，这使得商品本身属性变得越来越复杂，因此个性化推荐系统正面临着一个全新的挑战——如何综合考虑用户对商品的各种属性的偏好程度，给出更加合理化的推荐商品列表。另一方面，伴随着各大互联网公司的迅猛发展以及并行计算、云计算、hadoop 框架、MapReduce 编程模型等一系列技术的发展，现有推荐系统能够收集到更多样的数据，并具有更强的海量数据处理能力，推荐系统变得越来越智能，可以说个性化推荐系统正迎来一个全新的机遇——如何充分利用各类数据（用户偏好数据、商品内容数据、用户评分数据等等），发挥现代计算机集群具有的大规模并行计算能力，进一步提升推荐系统的质量。

现在，每天都有数以百万计的用户在众多电子商务网站上进行交易，为了提升用户体验，电子商务网站都会有一套自己的个性化推荐系统，旨在帮助用户解决信息过载问题。这些推荐系统先是根据一些规则选出一些适合用于推荐的商品作为候选集，再利用用户对商品的各类行为（点击、加购、收藏、购买）数据进行建模，预测用户对商品的偏好分，选择偏好分值 Top N 的商品作为备选商品，最后再结合一些上下文

特征对这些备选商品进行再次排序，最终给出推荐结果。

然而，随着电子商务向各大传统行业的全面进军以及各类服务行业的迅猛发展，很多商品已经不仅仅是实体商品，更多的是实体商品与各种第三方提供的服务的综合体，商品变得越来越复杂。现有的推荐系统往往不能综合考虑用户对商品的各个维度的偏好程度，也正是由于这个原因，它会给用户推荐那些在某些属性上非常符合用户的需求，但在另外一些属性上却令用户大失所望的商品。

另一方面，系统能收集到的用户人口统计学知识及商品的内容知识越来越丰富，因此，这些电子商务公司迫切地希望能够充分的利用这些内容知识，来弥补用户行为数据的不足，以期建立更合理的推荐模型，从而给用户推荐那些各个维度都能很好地满足用户需求的商品。

1.2 研究目标与内容

为了解决传统个性化推荐系统因不能综合考虑用户对商品的多种属性的偏好程度，最终导致推荐效果不理想的这类问题，本文在综合调研了推荐系统的各类推荐算法之后，基于 PMF 模型^[29]，提出了一系列内容感知^[38,39]（也即 content-aware，利用已知的用户基本信息确定用户代价特征向量，类似的，利用商品内容信息确定商品代价特征向量）的推荐算法模型，这些模型是旨在充分利用用户偏好和商品内容数据的概率矩阵分解扩展模型，详细介绍如下：

1. 第一种模型 **cPMF** 旨在更好地利用系统收集到的商品内容数据。该模型在 PMF 模型的基础上，进一步将表征商品的特征向量分为隐式和显式两个部分，其中显式的部分是根据商品内容数据计算得到的。商品显式特征向量作为模型的输入，再结合用户评分数据学习表征用户偏好的代价特征向量以及其余隐式的表征用户和商品的特征向量。与 PMF 类似，**cPMF** 模型将用户和商品的特征向量的内积作为用户对商品的偏好分，系统将给用户推荐偏好分 **TOP-K** 的商品。

2. 第二种模型 **GcPMF** 旨在更精准地表达商品数据内容的含义，为了更合理的利用实际数据中多种不同类型的用户偏好特征，消除不同特征之间的单位差异，该模型使用余弦相似度来表达商品显式特征向量与对应用户隐式特征向量的关联，这区别于 **cPMF** 模型中的内积形式。

3. 第三种模型 **CsPMF** 旨在增强 **cPMF** 的灵活性。该模型在 **cPMF** 的基础上增加了两个系统超参 (w_1, w_2)，用于调节显式特征和隐式特征在模型中的比重。

4. 第四种模型 **GCsPMF** 旨在增强 **GcPMF** 的灵活性。该模型在 **GcPMF** 的基础上增加了系统超参 (w_1, w_2)，用于调节显式特征和隐式特征在模型中的比重。

5. 第五种模型 **BCsPMF** 旨在更好地利用系统收集到的商品内容数据以及用户内

容数据。与 cPMF 类似, BCsPMF 将用户和商品的特征都划分为隐式和显式两个部分, 该模型使用向量内积值来表达商品显式特征向量与对应用户隐式特征向量的关联。

本文提出的内容感知的推荐算法模型将商品和用户的内容知识(如用户的兴趣偏好、商品各个维度的属性特征表现)扩充至 PMF 模型, 从而充分利用各类可以离线计算的关于商品和用户的知识, 一方面, 这可以有效地解决传统的 PMF 模型对于代价相关的知识的忽视, 将推荐结果引导至符合用户对商品相关代价需求的方向; 另一方面, 该模型延续了 PMF 模型的高效性, 其计算复杂度与评分矩阵中非零值的个数呈线性关系, 保证了其可扩展性。针对提出的一系列的内容感知的概率矩阵分解模型, 本文详细地介绍了模型的图模型结构, 并给出了使用梯度下降算法求解模型参数的相关公式的详细推导过程。

最后, 本文在 MovieLens-1M 数据集(由 GroupLens 团队提供的包含用户基本信息、电影基本信息以及用户对电影评分的影评数据集)上对 PMF、cPMF、GcPMF、BCsPMF 模型进行了对比验证, 验证了融入用户基本信息和电影基本信息有助于提升推荐模型的准确率, 此外, 还对 CsPMF、GCsPMF 模型与 cPMF、GcPMF 模型进行了对比验证, 验证了通过调节显式特征和隐式特征在模型中的占比将有望进一步提升模型的效果。

1.3 论文贡献及创新

本文的主要贡献有:

1. 基于 PMF 模型, 提出了一系列内容感知的概率矩阵分解扩展模型, 这些模型将用户和商品的内容知识融合进了 PMF 模型, 充分利用了系统收集到的用户基本信息和商品内容数据, 丰富了模型的表达能力。
2. 对本文提出的模型作了详细地解释, 并给出了利用梯度下降算法求解模型参数过程中需要的推导公式。
3. 在实际数据集下对 PMF、cPMF、GcPMF、BCsPMF 模型进行了对比验证, 并对本文提出的模型中参数对模型的影响进行了进一步的分析。

本文的创新之处在于将可以离线获取到的用户和商品的内容相关知识扩充至 PMF 模型, 既增强了 PMF 模型, 又延续了 PMF 模型的高效性, 保持算法复杂度依旧与用户对商品评分矩阵中非零项的个数呈线性关系。

1.4 论文结构与组织

本文的结构安排如下所示:

第一章是绪论部分, 介绍了本论文的选题背景、A 公司现有个性化推荐系统存在的不足、本论文的主要研究内容以及主要贡献与创新。

第二章是研究现状分析部分，介绍国内外学者对个性化推荐的研究现状，分别介绍了基于协同的推荐、基于内容的推荐、基于知识的推荐、基于矩阵分解的推荐和组合推荐。

第三章是模型结构描述部分，详细介绍内容感知的含义、概率矩阵分解模型(PMF)以及本文提出的一系列内容感知的个性化推荐算法模型的框架和相应的公式推导。

第四章是模型实现及对比的部分，详细介绍了作者利用 MovieLens-1m 数据集对 PMF、cPMF、GcPMF、BCsPMF 模型进行了对比验证的过程，并分析了各个模型参数在模型中作用。

第五章是总结与展望部分，对整篇文章进行了总结，并对算法模型中可能的优化点及有待进一步研究的方向进行了说明。

最后向所有给本课题提供过帮助的人们表达了诚挚的谢意，并列出了本文的参考文献及其他附录。

第二章 国内外研究现状

自从亚马逊成功将协同过滤算法应用于它的网上书店的个性化推荐系统,并宣称个性化推荐系统大幅提升了销售额之后,越来越多的国内外学者对个性化推荐相关技术展开了深入的研究。首先,国际学术界对推荐系统的关注度也越来越高:ACM 成立了推荐系统年会,其次,国际顶级数据挖掘会议(KDD)中关于个性化推荐的文章越来越多,并且伴随着会议还会邀请大型企业贡献海量数据举办每年一度的 KDD CUP 比赛,此外,近年来,随着互联网行业的迅猛发展,越来越多的企业尝试使用个性化推荐技术向用户推荐商品或者信息。为了吸引更多的人参与到个性化推荐技术的实践研究当中,许多大公司纷纷效仿 KDD CUP 举办各类推荐相关的大赛(如天猫的天池大数据竞赛、百度的电影推荐大赛),用丰厚的奖金吸引了大量的学生以及工业界从业人员。在推荐系统中,最为核心的便是推荐算法,对于推荐算法的分类,目前学术界和工业界都还没有统一的标准,这里参照文献[5]提出的方法,将推荐算法分为四大类:基于内容的推荐、基于协同的推荐、基于知识的推荐以及组合推荐,并在 2.1 节、2.2 节、2.3 节、2.4 节对它们的研究现状分别进行介绍。最后,在 2.5 节单独介绍了近年来特别热门并且与本文提出的模型密切相关的矩阵分解类算法。

2.1 基于协同的推荐

基于协同的推荐算法最早由亚马逊的工程师们提出,并成功地帮助亚马逊的网上书店大幅提升销售业绩,是目前工业界使用的最为广泛的一种推荐技术。根据相似的参照物不同,协同过滤推荐又可以分为基于用户的协同过滤和基于商品的协同过滤,其基本的思路都是充分利用群体智慧,以基于用户的协同过滤为例,系统会收集用户对商品的历史操作数据,然后根据用户的历史操作行为找到与其相似的用户(该用户的历史操作行为与之类似),再把与之相似用户购买过的商品推荐给当前用户,这类协同过滤算法的基本假设是用户会喜欢与之具有相似偏好用户购买过的商品。根据实现方式的不同,可以将基于协同的推荐算法分为两类:一类是启发式协同推荐,另一类是基于模型的协同推荐。

早期的协同过滤算法多是采用启发式方法,启发式协同推荐的基本思路是对于一个待推荐商品的用户(记为 a),系统先通过用户之间相似度指标找出与之相似的用户(记为 b),然后将 b 与商品之间的关联程度通过 a 与 b 之间的相似度(记为 $\text{Sim}(a,b)$)传递给 a ,最终计算商品与 a 的关联程度(又叫做用户 a 对商品的偏好分,记为 $\text{Score}(a,i)$)。启发式方法的主要研究重点在于用相似度的评价指标,及用户相似程度对于关联商品

的加权方式。常用的相似度评价指标有两种，一种是贾卡德相似度，另外一种是余弦相似度。基于协同的推荐系统通常会使用用户对商品的操作行为（评分、点击、收藏、浏览时长等）来衡量用户之间的相似度。如果使用贾卡德相似度作为指标，则 a 与 b 的相似度公式为：

$$\text{Sim}(a,b) = \frac{|A \cap B|}{|A \cup B|} \quad (2-1)$$

其中 A 和 B 分别表示用户 a 和用户 b 操作过的商品集合。如果使用余弦相似度指标，则将用户对商品的操作表示成为一个向量,每个维度的值对应用户对于商品的操作行为分数，然后计算两个用户对应向量的余弦距离作为两个用户的相似度。常用的加权方式则有以下几种：

$$\text{Score}(a, \text{item}) = \frac{1}{n} \sum_{b \in U_B} \text{Score}(b, \text{item}) \quad (2-2)$$

$$\text{Score}(a, \text{item}) = k \times \sum_{b \in U_B} \text{Sim}(a, b) \times \text{Score}(b, \text{item}) \quad (2-3)$$

$$\text{Score}(a, \text{item}) = \bar{a} + k \times \sum_{b \in U_B} \text{Sim}(a, b) \times (\text{Score}(b, \text{item}) - \bar{b}) \quad (2-4)$$

其中（2-2）是最简单的一种协同形式，表示用所有的相似用户对 item 的评分的平均值作为当前用户对 item 的评分的估计值，（2-3）则利用用户之间的相似度对进行了加权处理，这是应用最广泛的方法；（2-4）则对不同用户评分存在的差异性进行了平滑处理。

近年来，比较流行的协同过滤算法大都是基于模型的方法。这类方法利用用户对商品的历史行为数据作为训练样本，训练一个评分模型，然后根据该评分模型对特定用户和商品进行打分，用于表示用户对商品的偏好程度。这类基于模型的算法引起了很多学者的关注与研究，国内外学者尝试了各种各样的方法用于建模，包括线性回归模型^[6]、概率相关模型^[7]、信息熵模型^[8]、基于贝叶斯的概率模型^[9]以及统计模型^[10]。最近几年，各种图模型算法也被国内外学者所青睐，包含主题分析模型(LDA)^[11]、潜语义分析(PLSA)^[12]等等。

基于协同的推荐算法具有以下优点：

首先，该类算法不需要考虑待推荐的商品的格式，它可以是任何复杂的非结构化的对象（新闻、图片、音频、视频或者它们的综合体都可以基于协同的方式来推荐），因为该类算法不需要对商品的内容进行建模。

其次，该类算法对用户的正常使用的干扰最少，该类算法只需要收集用户对商品

的各类历史行为（如评分、购买、点击、收藏等等），而这些恰恰是计算机最擅长的，并且是对用户影响最小的，绝大部分数据收集工作都是系统自动完成，不会打扰到用户对系统功能的正常使用。

第三，该类方法的推荐结果可以非常个性化，能有效提升用户惊喜度。传统的基于内容的推荐往往是依据用户的历史操作最频繁的一些商品总结归纳出用户的兴趣偏好，这样做最大的问题就是限制了用户的兴趣偏好，因此系统只能推荐出用户已明确表现出来的兴趣偏好相符的商品，对于那些用户尚未表现出来的兴趣偏好却心有余而力不足。而基于协同的推荐能推荐一些与用户具有类似兴趣偏好的其他用户购买过的商品，真正做到千人千面的个性化推荐，推荐的商品很可能与用户某些尚未在系统中明确表现出来的兴趣偏好相符，给用户带来意向不到的惊喜感。

第四，基于协同的推荐算法还是一个普适性很强的算法，由于它不需要关注商品和用户的本身属性，而是依据用户对商品的行为进行建模的，因此在当前应用下行之有效的协同算法，可以很容易推广到其他任何能收集到类似用户行为的应用，不同应用之间可以共用同一套协同过滤算法框架，仅需要调整模型的参数即可以完成算法的推广。

但事物都有两面性，基于协同的推荐算法在以下几个方面存在不足：

首先，基于协同的推荐算法会受到数据稀疏性问题的困扰，需要用到个性化推荐的系统往往存在信息过载问题，这表明待推荐的商品数量是巨大的，但是单个用户的操作比较是少量的，因此用户对商品的行为矩阵势必是比较稀疏的，存在很多的缺失值，对于单个用户来说，他操作过的商品只不过是系统中商品集合很小的一个子集，其他用户也是类似的情况，这样一来，如何在茫茫人海中找到与他操作行为类似的其他用户无疑是很困难的。

其次，基于协同的推荐算法对于冷启动问题容易束手无策，基于协同的推荐核心的数据来源于用户对商品的历史行为，一个新用户，意味着他没有相应的历史数据，这意味着他与系统中任何一个其他用户的相似度为零，推荐系统将无法做出推荐，对于那些历史数据比较少的用户来说，情况也大体相似，因此纯粹的基于协同的推荐算法对于新用户是不够友好的。

第三，基于协同的推荐算法过于依赖用户对商品的历史行为数据，因此使用不同时段的历史数据训练出来的模型往往会有非常大的差异，这使得在使用基于协同的推荐算法时，还需要对用于训练的历史数据的数据量（或者说时间跨度）进行调优，而这需要更换数据集，一旦数据集规模比较大的时候，这将会耗费很多的计算资源，有时候甚至是不可能完成的优化。

第四，基于协同的推荐算法对于作弊行为是非常敏感的，拿网上商城的推荐系统来说（假定它们采用的是基于商品的协同过滤，这也是亚马逊早期采用的协同过滤算

法), 如果某个恶意用户使用多个账户刻意的再买了 A 之后又去买 B, 这样就能让使用基于商品的协同过滤算法的个性化推荐系统将 A 和 B 关联在一起, 然后会给其他购买了 A 的商品的用户推荐 B。即使 A 和 B 事实上不应该存在关联, 但基于协同过滤的推荐系统还是会掉入某些恶意用户设计的陷阱当中。

2.2 基于内容的推荐

基于内容的推荐算法起源于信息检索, 其中很多技术及思想都来源于信息检索, 这类算法是最早被应用于推荐系统的。早期的推荐系统的主体是基于内容的, 它的主体思想是对用户和商品分别进行建模: 先利用信息检索领域常用的各种特征提取方式对商品进行建模, 提取出关键的特征组成特征向量用于表征商品本身, 再利用机器学习的方法使用用户历史访问过(包含喜欢和不喜欢的)的商品的特征, 学习出用户的偏好特征, 最后再给用户推荐与其偏好特征最相符的一组商品。基于内容的推荐算法可以划分为三个步骤:

1) 商品的特征表示

一个商品通常有着区别于其他商品的属性, 这些属性有的是结构化的, 还有一些是非结构化的, 在常见的推荐场景中, 比如社交网站的公共主页推荐, 每个公共主页即是商品, 公共主页的注册年份、感兴趣的领域、注册人的年龄等等比较确定性的属性属于结构化的属性, 而另外一些诸如注册人在公共主页发布的状态, 图片等等则属于非结构化的属性。对于结构化的属性, 特征提取相对简单, 对于非结构化的属性来说则要困难许多。对于类似网页这类非结构化而且文本是内容主体的文档类商品来说, 可以采用信息检索领域里面广泛使用的向量空间模型^[13](Vector Space Model, 简称为 VSM) 表示, 并使用逆文档频率(TF-IDF)方法确定文档中的词的权重^[14]。对于音频、图片等更为复杂的非结构化对象, 则需要采用更为复杂的特征表示方法。

2) 用户偏好模型的建立

基于内容的推荐系统收集用户对商品的历史行为数据, 进而可以获得用户喜爱和不喜爱的商品子集, 通过对这些商品特征的学习, 我们可以归纳出用户的偏好模型。如果我们将用户偏好模型当作一个二分类问题(用户对商品喜欢还是不喜欢)来看待, 那么理论上讲所有的机器学习中二分类的算法都可以用来建模。这里简单介绍一下基于内容的推荐系统可能用到的分类算法:

第一种是最邻近算法(k-Nearest neighbor, 简称 KNN), 要判断一个用户 u 是否偏好一个商品 i , 这种算法在用户已经评价过的商品集合中找到 k 个与商品 i 最相似的商品作为邻近商品集, 然后根据用户 u 对邻近商品集商品中的商品的偏好估计用户 u 对商品 i 的偏好程度。如果商品是文档类的商品, 并且使用了向量空间模型表示, 那么这里

KNN 用两个商品的特征向量的余弦距离来衡量两个商品的距离。

第二种方法是决策树算法，如果商品的大多数属性值是结构化属性，那么使用决策树算法将会是一个不错的选择。使用决策树最大的优点是其结果很直观，便于系统用户理解。使用决策树算法可以很方便的给出合理的推荐原因，比如这个商品的某个属性与你之前偏好的商品的属性一致，这样的推荐理由是跟商品内容息息相关的，很容易被用户理解，也更容易被用户接受。然而，如果商品的大多数属性都来源于非结构化的内容，比如商品是新闻网页，那么决策树的效果可能不会太好。

第三种方法是朴素贝叶斯算法，该算法是贝叶斯算法的一个特例，它假定在类别确定的时候，商品的各个特征维度之间是相互独立的，而贝叶斯网络则没有这样的限制。虽然它有这么严格的条件独立假设，但是实践中它的效果却惊人的好，非常适合用于文本分类问题。因为朴素贝叶斯算法简单易实现，因此它往往是分类问题中优先被尝试的算法。以新闻分类为例，可以利用用户 u 的历史行为数据中表现出偏好的文档进行训练，再用训练好的模型对新的新闻进行分类。

第四种方法是工业界常用的线性分类算法，很多著名的分类器（如逻辑回归分类器、线性 SVM 分类器）都属于这一类。线性分类器通过在高维空间寻找一个超平面，使得这个平面尽量将用户喜欢的商品和用户不喜欢的商品分隔开。

3) 推荐结果的产生

建立用户偏好模型之后，我们可以很快速的产生推荐结果，最简单的方法便是将偏好模型算出的 TOP-N 商品用于推荐，更进一步我们还可以根据不同用户历史访问商品的频率等习惯特征给不同的用户推荐不同个数的商品，从而更加地个性化，进一步优化推荐结果。

基于内容的推荐算法具有以下优点：

首先，推荐结果直观，能给出基于商品内容的推荐原因，容易被用户接受。基于内容的推荐算法会提取商品的特征，会推荐那些与用户偏好商品集有着相似特征的商品，这样便可以很方便的解释推荐原因，这样的推荐原因与商品本身特征相关，容易被用户理解。

其次，每个用户的推荐结果相对独立，对作弊用户的抗干扰能力比较强。基于内容的推荐算法都是根据用户历史偏好的商品集合的特征训练出用户偏好模型的，因此用户之间的偏好模型互不影响，作弊用户刻意购买某些商品不会对其他合法用户造成干扰。

第三，新上架的商品可以立即被推荐 基于内容的推荐算法会单独对商品和用户进行建模，新来的商品只需要进行相应的特征提取，再利用用户偏好模型进行偏好预测即可以预测用户对商品的偏好程度，因此新的商品可以获得和旧商品同样多的机会被推荐出去，这恰恰是纯粹的协同过滤算法不能解决的问题（由于新商品没有相应的用

户反馈数据，纯粹的协同过滤算法会认为新商品不被任何用户青睐，因此新商品将永远得不到推荐）。

另一方面，基于内容的推荐算法也存在着很多不足之处：

首先，基于内容的推荐算法难以应对复杂的诸如多媒体之类的非结构化商品。对于简单的文档量商品，还可以使用向量空间模型进行表示，但是对于那些异常复杂的多媒体（视频、音乐等），特征提取将会更加困难，基于内容的推荐算法将大大受限。

其次，无法挖掘出用户潜在的兴趣偏好。基于内容的推荐算法的核心思想是根据用户表现出偏好（如用户评分较高、访问频繁等）的商品集建立用户偏好模型，这样恰恰限制了用户的偏好，因此基于内容的推荐算法无法推荐那些用户之前未表现出来的兴趣偏好。拿新闻推荐来举例，如果一个用户经常访问娱乐八卦类新闻，那么系统只可能会给用户推荐相关的娱乐新闻，即使用户很可能对数码科技相关的新闻也感兴趣，系统也无从知晓。

第三，无法为新用户推荐商品。基于内容的算法需要收集用户的历史偏好用于建立用户偏好模型，因此一个全新的用户意味着他没有相应的历史偏好，这让系统无法建立用户偏好模型，因此系统无法给出合适的推荐。

基于内容的推荐算法核心是建立用户偏好模型，但实际应用场景中，用户的兴趣偏好往往会随着时间不断变化，因此也有很多研究人员提出了很多自适应的用户偏好模型，这类模型能随着时间的推移，自动的将新的文档纳入到用户偏好模型中，不断更新用户偏好模型，使之能够较好的表征用户模型。文献[15-17]提出了一种自适应的方法来更新用户偏好的文档集合，这种方法将与用户偏好文档集高度相似的文档推荐给用户，并且用这个文档来更新用户偏好文档的各项参数，以此达到动态更新用户偏好模型的效果。文献[18]则将用户的兴趣划分为短期兴趣和长期兴趣，通过对时间进行加权，赋予反映短期兴趣的关键词更大的权重，以期提高用户偏好模型的准确率。

2.3 基于知识的推荐

与其他推荐算法不同的是，基于知识的推荐^[19]算法本质上是一种推理算法的延伸，它更多的依赖相关领域的专家知识。基于知识的推荐系统为用户提供了一个推理逻辑用于筛选合适的商品，它不需要依赖大量的有关于商品和用户的统计数据（类似用户点击、收藏、评价过商品这类行为数据），它依赖的是应用场景下人类专家的领域知识，根据专家的领域知识建立推理模型，然后利用这个模型给用户推荐合适的商品。文献[19]中利用餐饮领域的专家知识（具体来说哪些菜式受哪些人群欢迎）给用户推荐合适的饭店，类似的，利用电影领域的专家知识，给观众推荐合适的电影。同时，文献[19]的作者还指出，基于知识的推荐跟其他类别的推荐算法有很强的互补性，如果加以

结合可能可以提升推荐系统的效果。

2.4 组合推荐

正如上文介绍的那样，单一的推荐算法有其优势也存在不足，单个模型很难满足现实生活中应用场景对推荐系统的要求，自然而然地人们便尝试将多种单一模型组合到一起，以期能够扬长避短，实现一加一大于二的效果，通过模型之间的互补来提升推荐的效果。组合推荐，顾名思义就是多种推荐算法模型的组合，尽管理论上我们有很多种模型之间相互组合的方式，但实践中人们使用的最多的还是基于内容和基于协同的模型的组合。文献[20]对 PureSVD(基于隐变量的协同过滤模型)、NNCos(非归一化的基于余弦相似度的协同过滤模型)以及 LSA（基于隐语义分析的基于内容推荐模型）这 3 种单模型以及它们的组合模型（LSA+NNCos 和 LSA+PureSVD 以及 LSA+NNCos）的效果进行了试验，结果表明组合模型的推荐效果要优于单一模型。文献[28]通过组合 107 种不同的单模型，综合了各个模型的结果，最终获得了 Netflix prize。

文献[5]中根据组合方式的不同将组合推荐分为后融合、中融合以及前融合这三种。

后融合是先训练出单个模型，并用单模型产出推荐结果，在将两个或者多个单模型的推荐结果进行融合，后融合中融合单模型产出的策略是研究者们研究的重点，不同的融合策略往往有不同的效果。文献[21]使用了一种投票机制来对多个模型的推荐结果进行融合，文献[22]则将多个单模型产出的用户偏好分值进行线性组合后得到最终的用户偏好分值，再利用这个分值给用户推荐得分较高的那些商品。在工业界还有很多新奇的融合方式，值得一提的是，在天猫天池大数据竞赛中有些参赛人员还试图通过对多个单模型得到的商品排名进行非线性组合得到商品的最终排名（具体的组合方式为：对于某一用户 u ，模型 A 认为商品集应该按照某一顺序推荐给他，模型 B 认为商品集应该按照另一种顺序推荐给他，记商品 i 在模型 A 中对于用户 u 的排名为 ra ，在模型 B 中对于用户 u 的排名为 rb ，然后将 $(ra*ra + rb*rb)$ 在所有商品中的排名作为组合模型中该商品对于用户 u 的排名），这种直接对排名进行融合的最大优点就是它不需要归一化，不需要单模型之间产出的分值在同一个范围内。

中融合的组合方式又分为两种，一种是以基于协同过滤的方法为整体框架，将基于内容的方法融合到框架中；另一种是以基于内容的方法为整体框架，将协同过滤的方法与思想融合到框架中。前者为了克服协同过滤推荐算法存在的数据稀疏性问题，将用户的人口统计学特征（如用户的年龄、教育水平、工作情况等等）收集起来，用于评价用户之间相似度的一个维度，这实际上便是将用户当作对象，使用基于内容的思路进行推荐，这样可以更充分地利用系统能够收集到的数据，在一定程度上弥补了协同过滤的短板，因此被工业界的推荐系统所广泛使用。文献[23]中作者采用了多种基

于内容的方法来对用户进行聚类,挖掘用户与用户之间的内在联系,从而弥补纯粹协同过滤算法的不足,以期达到更好的推荐效果。文献[24]中使用基于内容的方法为用户产生新的特征,用于补充用户对商品的评分数据,也属于这一类中融合方式。后一种融合方式利用降维技术将基于内容的推荐系统中待推荐对象的特征进行了简化。例如,文献[25]中采用 LSI 算法 (latent semantic indexing, 即隐性语义索引), 在基于内容的推荐算法框架下, 将用户的特征向量进行了精简处理, 这有效地提升了推荐模型的效率。

前融合是指直接将多种推荐方法统一到一个框架下进行建模, 这种方法近年来备受关注。文献[26]中作者将电影的类型和用户的年龄一起作为特征, 统一用于训练, 最终学得用于推荐的分类器。文献[27]中使用了贝叶斯混合方式的回归模型把用户和对象的所有特征都统一放到一个统计模型中, 并采用了马尔科夫蒙特卡罗方法获的模型中的参数的值, 再利用相应的效用函数来计算商品对于用户的效用值, 效用值越高则越有可能被系统推荐给用户。

2.5 基于矩阵分解的推荐

近年来, 基于矩阵分解的推荐算法得到很多研究者的关注, 这类算法的基本思路是利用评分矩阵学习分别表示用户偏好和商品类型的潜在特征向量, 并用这两个向量的内积作为用户对商品的偏好分值, 最后系统将选择偏好分值较高的那些商品进行推荐。

典型的矩阵分解类的算法模型包括 SVD(奇异值分解模型)^[31]、NMF(非负矩阵分解模型)^[32]、PMF(概率矩阵分解模型)^[29]。其中 SVD 是利用线性代数求解用于表征用户和商品的潜在特征向量, 这些向量的元素值可能有正有负。为了使得这些特征向量有一定的物理含义, NMF 规定潜在特征向量中各个元素是非负值, 这使得模型结果更直观。PMF 则使用概率论从贝叶斯的角度对这些特征向量的含义作出了解释, 并使用多维高斯分布作为这些特征向量的先验分布。

此外, 基于矩阵分解的模型, 很多学者提出了改进模型, 如 Yong Ge 等人^[30]提出了两种代价敏感的 PMF 扩展模型并将其应用于旅行路线推荐, 涂丹丹等人^[33]提出了一种联合的 PMF 模型, 将用户和网页、用户和广告、广告和网页三种相关矩阵数据都充分利用起来, 从而实现了更好的上下文广告推荐系统。

矩阵分解模型的优势在于其算法复杂度与评分矩阵中的非零值呈线性关系, 这使得它比较适合用于大规模数据, 为了让其更好地应用于现实的大数据场景, 很多学者对矩阵分解模型并行处理^[34-36]进行了深入的研究。其中, Y. Zhuang 等人^[36]还将提供了并行矩阵分解模型(libmf)的开源代码, 相关研究人员可以在官网上下载并自由使用。

第三章 内容感知的个性化推荐模型

本章将详细介绍本文提出的内容感知的个性化推荐模型，与传统的个性化推荐模型不同的是，这类模型能够感知用户基本信息和商品内容中所包含的用户偏好和商品类型，因此这类模型计算出的用户对商品的偏好分更能体现出用户购买商品所需要付出的代价（指用户偏好和商品类型的吻合度），因此我们将这类模型统称为内容感知的个性化推荐模型。内容感知的个性化推荐模型综合考虑用户和商品的内容特征，比传统推荐模型考虑的更加全面，举例来说：用户在网上买东西，如果推荐系统已经知道各个物品的物流服务水平，并且了解用户对物流水平的要求，那么推荐系统就可以结合这两种内容数据给出更加合理的推荐。

将在 3.1 节介绍 **PMF** 模型的整理框架，本文提出的内容感知的个性化推荐模型都是基于 **PMF** 的扩展模型，关于潜在特征向量的部分与 **PMF** 保持一致。

在 3.2 节介绍内容感知个性化推荐模型的第一个版本 **cPMF**，该版本在 **PMF** 模型的基础上增加了用于表征代价相关信息的向量，其整体框架与 **PMF** 基本一致。

在 3.3 节介绍内容感知个性化推荐模型的第二个版本 **GcPMF**，该版本将 **cPMF** 中通过向量内积形式关联用户和商品的代价特征的方式修改为通过余弦相似度进行关联，这样做是为了消除各种内容特征不同纬度之间的单位差异。

在 3.4 节介绍内容感知个性化推荐模型的第三个版本 **CsPMF**，该版本在 **cPMF** 的基础上，将代价相关信息向量与潜在因子向量进行了加权处理，将权重作为系统超参，通过调节这两个参数可以很方便地控制系统对代价相关信息的重视程度，这样做的目的是为了进一步提升系统的灵活性。

在 3.5 节介绍内容感知个性化推荐模型的第四个版本 **GCsPMF**，该版本是在 **GcPMF** 的基础上，将代价相关信息向量与潜在因子向量进行加权处理，并将权重作为系统超参，与 **CsPMF** 类似，这样做也是为了提升系统的灵活性。

在 3.6 节介绍内容感知个性化推荐模型的第五个版本 **BCsPMF**(Bothway cost sensitive PMF)，该版本是为了更好的利用用户相关的内容知识（如用户的年龄、教育水平、消费水平等等）而提出的双向内容感知模型，该模型能更好地利用可以离线获取的用户相关的内容知识，在 **CsPMF** 的基础之上，它新增了用户偏好决定的代价相关向量（可以根据用户相关数据离线计算好，在图模型中是显变量）以及与之对应的商品跟用户偏好相关的潜在因子向量，跟 **CsPMF** 类似，通过向量内积值将商品的潜在因子向量和用户偏好向量进行关联，该模型旨在充分利用内容相关的数据，是基于协同过滤的框架，融合基于内容的方法的一种思路。

3.1 概率矩阵分解模型（PMF）

PMF 模型的图模型如图 3.1 所示。假定收集到的数据当中有 N 个用户， M 件商品，用户对商品的评分数据用矩阵 R 表示，矩阵的行向量 R_i 表示一个用户的评分向量，如果用户对某一商品进行了评分，则向量中对应维度的值即为评分，否则对应维度的值为 0；类似的，矩阵的列向量 R_j 表示一个商品的评分向量，如果商品被某一用户评分，则 R_j 中对应维度的值即为评分值，否则该维度值为 0。在图 3.1 中， U_i 表示第 i 个用户的潜在因子向量， V_j 表示第 j 件商品的潜在因子向量，假定 U_i 和 V_j 均有 D 维，每一维都表征某种潜在的特征； R_{ij} 表示第 i 个用户对第 j 个商品的评分值，但第 i 个用户对第 j 个商品进行了评分，则 R_{ij} 为评分值，否则 R_{ij} 的值为零， R_{ij} 是显变量，可直接从训练数据中获得； σ_U 、 σ_V 、 σ 是系统的超参，其中 σ_U 表示 U_i 对应的高斯分布的方差， σ_V 表示 V_j 对应的高斯分布的方差， σ 表示 R_{ij} 对应的高斯分布的方差。

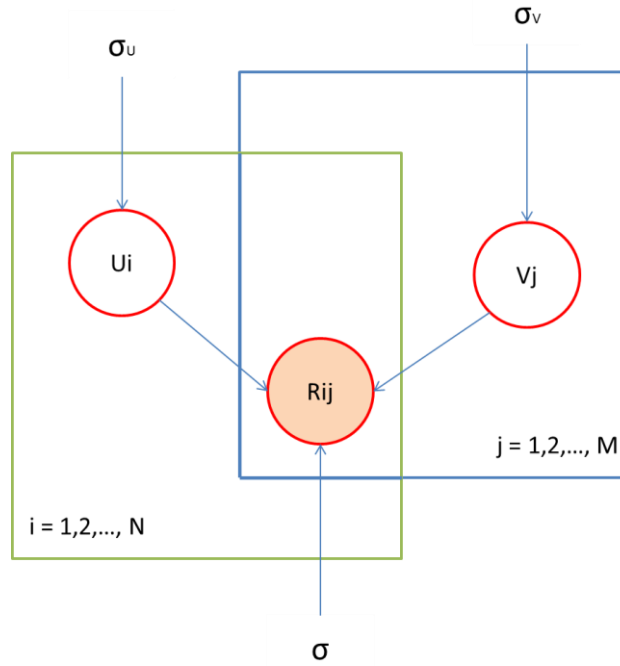


图 3.1 概率矩阵分解模型（PMF）的图模型

用 U 表示所有用户的潜在因子向量的集合， V 表示所有商品的潜在因子向量的集合， R 表示用户对商品的评分矩阵。PMF 模型中为了表达评分具有一定得波动性，用一个高斯分布来表示 R_{ij} 跟 U_i 和 V_j 之间的关联，这样我们可以求出评分矩阵的条件概率为：

$$p(R|U, V, \sigma^2) = \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j), \sigma^2)]^{I_{ij}} \quad (3.1)$$

其中， $\mathcal{N}(x | u, \sigma^2)$ 是期望为 u ，方差为 σ^2 的高斯分布的概率密度函数；此外， I_{ij} 是一

个标记函数，用于标记第 i 个用户是否评论了第 j 件商品，其定义如下：

$$I_{ij} = \begin{cases} 1, & \text{第 } i \text{ 个用户评论了第 } j \text{ 件商品;} \\ 0, & \text{其他情况} \end{cases} \quad (3.2)$$

公式 (3.1) 中的 $h(U_i, V_j)$ 是用户和商品对应的潜在因子向量的函数，可以看做是两者的某种距离。对于用户的潜在因子向量，PMF 使用一个球形的多维高斯分布作为先验分布，因此我们可以得到 U 的条件概率分布为：

$$p(U | \sigma_U^2) = \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \quad (3.3)$$

类似的，对于商品的潜在因子向量，PMF 也使用类似的多维高斯分布，但参数有所不同，进而我们可以得到 V 的条件概率分布为：

$$p(V | \sigma_V^2) = \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \quad (3.4)$$

根据图(3.1)所示的图模型以及贝叶斯定理，我们可以推导出 U, V 这两个变量的后验概率为：

$$\begin{aligned} p(U, V | R, \sigma_U^2, \sigma_V^2, \sigma^2) &\propto p(R | U, V, \sigma^2) \times p(U | \sigma_U^2) \times p(V | \sigma_V^2) \\ &= \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j), \sigma^2)]^{I_{ij}} \times \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \times \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \end{aligned} \quad (3.5)$$

对式(3.5)中后验概率求对数，可以得到后验概率的对数值为：

$$\begin{aligned} \ln p(U, V | R, \sigma_U^2, \sigma_V^2, \sigma^2) &= -\frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j))^2 - \frac{1}{2\sigma_U^2} \sum_{i=1}^N U_i^T U_i - \frac{1}{2\sigma_V^2} \sum_{j=1}^M V_j^T V_j \\ &\quad - \frac{1}{2} \left[\left(\sum_{i=1}^N \sum_{j=1}^M I_{ij} \right) \ln \sigma^2 + ND \ln \sigma_U^2 + MD \ln \sigma_V^2 \right] + Const \end{aligned} \quad (3.6)$$

其中， $Const$ 表示常量。根据对数函数的单调性，我们可以很容易证明最大化 (3.5) 式等价于最大化 (3.6) 式，更进一步，通过观察 (3.6) 式，我们发现最后一项是常数项，倒数第二项是由模型的超参决定的，因此我们可以归纳出模型的损失函数 E ，并将最大化 (3.6) 式的问题进一步转换为最小化损失函数 E 的问题，损失函数 E 的表达

式如下：

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j))^2 + \frac{\lambda_U}{2} \sum_{i=1}^N U_i^T U_i + \frac{\lambda_V}{2} \sum_{j=1}^M V_j^T V_j \quad (3.7)$$

其中， λ_U 的表达式为：

$$\lambda_U = \frac{\sigma^2}{\sigma_U^2} \quad (3.8)$$

λ_V 的表达式为：

$$\lambda_V = \frac{\sigma^2}{\sigma_V^2} \quad (3.9)$$

为了便于后续的求导，这里将 $h(U_i, V_j)$ 定义为 U_i 与 V_j 的内积，用于表征用户和商品的潜在因子之间的距离，其表达式为：

$$h(U_i, V_j) = U_i^T V_j \quad (3.10)$$

接下来，我们便可以利用梯度下降法求解损失函数 E 的小值， E 对 U_i 求偏导得到：

$$\frac{\partial E}{\partial U_i} = \sum_{j=1}^M I_{ij} (h(U_i, V_j) - R_{ij}) h' (U_i, V_j) + \lambda_U U_i = \sum_{j=1}^M I_{ij} (h(U_i, V_j) - R_{ij}) V_j + \lambda_U U_i \quad (3.11)$$

E 对 V_j 求偏导得到：

$$\frac{\partial E}{\partial V_j} = \sum_{i=1}^N I_{ij} (h(U_i, V_j) - R_{ij}) h' (U_i, V_j) + \lambda_V V_j = \sum_{i=1}^N I_{ij} (h(U_i, V_j) - R_{ij}) U_i^T + \lambda_V V_j \quad (3.12)$$

综上所述，PMF 模型将用户对商品的评分看成是一个由用户和商品潜在因子距离为期望，超参 σ^2 为方差的高斯分布产生的，每个评分之间相互独立，并假定用户和商品的潜在因子向量的先验分布是特殊形式的多维高斯分布（各维期望为 0，协方差矩阵为超参与单位矩阵 I 的乘积），再通过贝叶斯定理，可以求出 U 、 V 的后验概率，再将后验概率取对数并利用梯度下降算法求出使之最大化的参数 U 和 V ，再通过（3.10）式去估计那些评分矩阵缺失的值。需要注意的是通过（3.10）式计算得到的估计值可能会超出评分的范围，事实上，在实践中，人们往往会将（3.10）式得到的值再通过一个逻辑斯特函数（logistic function）将范围限制到[0,1]，类似的，对于评分矩阵中的 R_{ij} ，

则会通过最大最小值标准化方法（Max-Min Normalization）进行标准化，使它们的范围也变成[0,1]。

通过观察（3.7）式，我们可以发现 PMF 模型损失函数的计算复杂度为 $O(LD)$ ， L 表示评分矩阵中非零值的个数， D 表示潜在因子向量的维数，类似的，观察（3.11）式和（3.12）式，我们还可以看出损失函数的各个偏导的计算复杂度也为 $O(LD)$ ，因此 PMF 模型的复杂度为 $O(LD)$ 。这表明 PMF 模型的复杂度跟评分矩阵中的非零值个数是线性关系，因此 PMF 模型的扩展性比较强，比较适合并行计算，通过批量更新(batch update)，我们可以进一步提升训练的效率。

3.2 内容感知的概率矩阵分解模型一（cPMF）

cPMF 模型的图模型如图 3.2 所示。假定收集到的数据当中有 N 个用户， M 件商品，用户对商品的评分数据用矩阵 R 表示，矩阵的行向量 R_i 表示一个用户的评分向量，如果用户对某一商品进行了评分，则向量中对应维度的值即为评分，否则对应维度的值为 0；类似的，矩阵的列向量 R_j 表示一个商品的评分向量，如果商品被某一用户评分，则 R_j 中对应维度的值即为评分值，否则该维度值为 0。

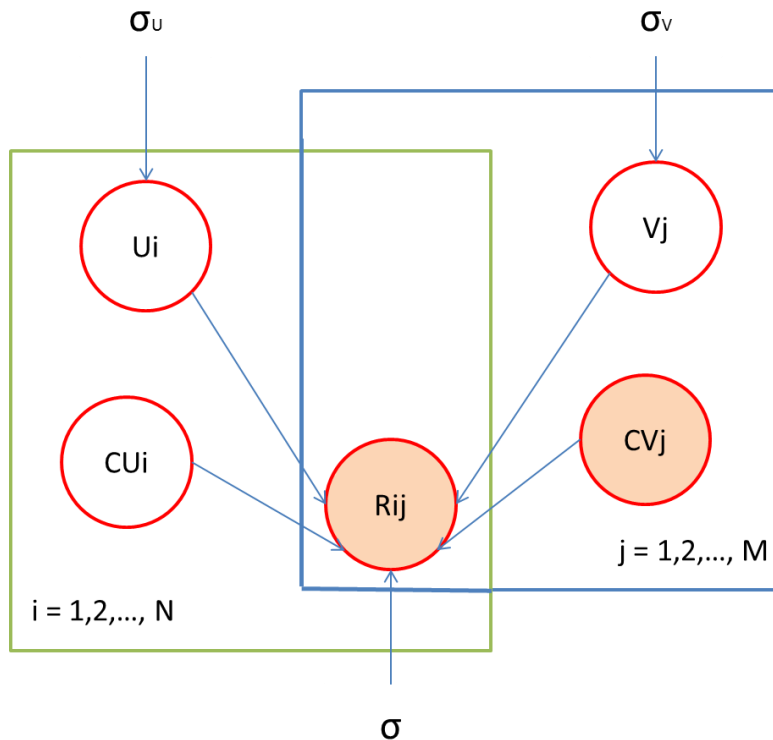


图 3.2 内容感知的概率矩阵分解模型（cPMF）的图模型

在图 3.2 中， U_i 表示第 i 个用户的潜在因子向量， V_j 表示第 j 件商品的潜在因子向

量，假定 U_i 和 V_j 均有 D 维，每一维都表征某种潜在的特征； CU_i 表示第 i 个用户的潜在代价向量， CV_j 表示第 j 件商品的代价相关向量， CU_i 和 CV_j 均有 K 维， CV_j 的每一维对应于一种需要考虑的代价； R_{ij} 表示第 i 个用户对第 j 个商品的评分值，但第 i 个用户对第 j 个商品进行了评分，则 R_{ij} 为评分值，否则 R_{ij} 的值为零。 R_{ij} 是显变量，可直接从训练数据中获得， CV_j 是第 j 件商品的代价相关向量，它也是显变量，可以离线计算好， σ_U 、 σ_V 、 σ 是系统的超参，其中 σ_U 表示 U_i 对应的高斯分布的方差， σ_V 表示 V_j 对应的高斯分布的方差， σ 表示 R_{ij} 对应的高斯分布的方差。

用 U 表示所有用户的潜在因子向量的集合， V 表示所有商品的潜在因子向量的集合， R 表示用户对商品的评分矩阵， CU 表示所有用户的潜在代价相关向量的集合， CV 表示所有商品的代价相关向量的集合。与 PMF 模型类似，cPMF 模型中为了表达评分具有一定波动性，用一个高斯分布来表示 R_{ij} 的产生，对照图 3.2，我们可以求出评分矩阵的条件概率为：

$$p(R|U, V, CU, CV, \sigma^2) = \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, CU_i, CV_j), \sigma^2)]^{I_{ij}} \quad (3.13)$$

其中， $\mathcal{N}(x | u, \sigma^2)$ 是期望为 u ，方差为 σ^2 的高斯分布的概率密度函数；此外， I_{ij} 是一个标记函数，用于标记第 i 个用户是否评论了第 j 件商品，其定义如下：

$$I_{ij} = \begin{cases} 1, & \text{第 } i \text{ 个用户评论了第 } j \text{ 件商品;} \\ 0, & \text{其他情况} \end{cases} \quad (3.14)$$

公式 (3.13) 中的 $h(U_i, V_j, CU_i, CV_j)$ 是用户和商品的潜在因子向量和代价相关向量的函数，用于表征第 i 个用户和第 j 件商品在多个维度上的关联度。对于用户的潜在因子向量，与 PMF 模型类似，cPMF 也使用一个球形的多维高斯分布作为先验分布，因此我们可以得到 U 的条件概率分布为：

$$p(U | \sigma_U^2) = \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \quad (3.15)$$

类似的，对于商品的潜在因子向量，cPMF 也使用类似的多维高斯分布，但参数有所不同，进而我们可以得到 V 的条件概率分布为：

$$p(V | \sigma_V^2) = \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \quad (3.16)$$

根据图(3.2)所示的图模型以及贝叶斯定理，我们可以推导出 U, V, CU 这三个变量的后验概率为：

$$\begin{aligned}
 p(U, V, CU | R, \sigma_U^2, \sigma_V^2, \sigma^2, CV) &\propto p(R | U, V, CU, CV, \sigma^2) \times p(U | \sigma_U^2) \times p(V | \sigma_V^2) \\
 &= \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, CU_i, CV_j), \sigma^2)]^{I_{ij}} \times \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \\
 &\quad \times \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I)
 \end{aligned} \tag{3.17}$$

对式(3.17)中后验概率求对数，可以得到后验概率的对数值为：

$$\begin{aligned}
 \ln p(U, V, CU | R, \sigma_U^2, \sigma_V^2, \sigma^2, CV) &= -\frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, CU_i, CV_j))^2 - \frac{1}{2\sigma_U^2} \sum_{i=1}^N U_i^T U_i \\
 &\quad - \frac{1}{2\sigma_V^2} \sum_{j=1}^M V_j^T V_j - \frac{1}{2} \left[\left(\sum_{i=1}^N \sum_{j=1}^M I_{ij} \right) \ln \sigma^2 + ND \ln \sigma_U^2 + MD \ln \sigma_V^2 \right] + Const
 \end{aligned} \tag{3.18}$$

其中，Const 表示常量。根据对数函数的单调性，我们可以很容易证明最大化(3.17)式等价于最大化(3.18)式，更进一步，通过观察(3.18)式，我们发现最后一项是常数项，倒数第二项是由模型的超参决定的，因此我们可以归纳出模型的损失函数 E，并将最大化(3.18)式的问题进一步转换为最小化损失函数 E 的问题，损失函数 E 的表达式如下：

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, CU_i, CV_j))^2 + \frac{\lambda_U}{2} \sum_{i=1}^N U_i^T U_i + \frac{\lambda_V}{2} \sum_{j=1}^M V_j^T V_j \tag{3.19}$$

其中， λ_U 的表达式为：

$$\lambda_U = \frac{\sigma^2}{\sigma_U^2} \tag{3.20}$$

λ_V 的表达式为：

$$\lambda_V = \frac{\sigma^2}{\sigma_V^2} \tag{3.21}$$

为了便于后续的求导，这里将 $h(U_i, V_j, CU_i, CV_j)$ 定义为 CU_i 与 CV_j 的内积再加上 U_i 与 V_j 的内积，用于表征第 i 个用户和第 j 件商品的关联程度，其表达式为：

$$h(U_i, V_j, CU_i, CV_j) = CU_i^T CV_j + U_i^T V_j \quad (3.22)$$

接下来，我们便可以利用梯度下降法求解损失函数 E 的小值。

E 对 U_i 求偏导得到：

$$\frac{\partial E}{\partial U_i} = \sum_{j=1}^M I_{ij} (h(U_i, V_j, CU_i, CV_j) - R_{ij}) \cdot V_j + \lambda_u U_i \quad (3.23)$$

E 对 V_j 求偏导得到：

$$\frac{\partial E}{\partial V_j} = \sum_{i=1}^N I_{ij} (h(U_i, V_j, CU_i, CV_j) - R_{ij}) \cdot U_i^T + \lambda_v V_j \quad (3.24)$$

E 对 CU_i 求偏导得到：

$$\frac{\partial E}{\partial CU_i} = \sum_{j=1}^M I_{ij} (h(U_i, V_j, CU_i, CV_j) - R_{ij}) \cdot S'(CU_i, CV_j) \quad (3.25)$$

其中 $S'(CU_i, CV_j)$ 是指 $S(CU_i, CV_j)$ 对 CU_i 的偏导， $S(CU_i, CV_j)$ 表示 CU_i 和 CV_j 的相似度，定义如下：

$$S(CU_i, CV_j) = CU_i^T CV_j \quad (3.26)$$

$$S'(CU_i, CV_j) = S(CU_i, CV_j) = CV_j \quad (3.27)$$

综上所述，cPMF 模型将用户对商品的评分看成是一个由用户和商品关联程度（由潜在因子向量和代价向量共同决定）为期望，超参 σ^2 为方差的高斯分布产生的，每个评分之间相互独立，并假定用户和商品的潜在因子向量的先验分布是特殊形式的多维高斯分布（各维期望为 0，协方差矩阵为超参与单位矩阵 I 的乘积），再通过贝叶斯定理，可以求出 U ， V ， CU 的后验概率，再将后验概率取对数并利用梯度下降算法求出使之最大化的参数 U ， V 和 CU ，再通过（3.22）式去估计那些评分矩阵缺失的值。需要注意的是通过（3.22）式计算得到的估计值可能会超出评分的范围，事实上，在实践中，人们往往会将（3.22）式得到的值再通过一个逻辑斯特函数（logistic function）将范围限制到 $[0,1]$ ，类似的，对于评分矩阵中的 R_{ij} ，则会通过最大最小值标准化方法（Max-Min Normalization）进行标准化，使它们的范围也变成 $[0,1]$ 。

通过观察(3.19)式，我们可以发现cPMF模型损失函数的计算复杂度为 $O(L(D + K))$ ， L 表示评分矩阵中非零值的个数， D 和 K 分别表示潜在因子向量的维数与代价相关向量的维数，类似的，观察（3.23）式、（3.24）式以及（3.25）式，我们还可以看出损失

函数的各个偏导的计算复杂度也为 $O(L(D+K))$ ，因此 cPMF 模型的复杂度为 $O(L(D+K))$ 。这表明 cPMF 模型的复杂度跟评分矩阵中的非零值个数是线性关系，因此 cPMF 模型的扩展性比较强，比较适合并行计算，通过批量更新 (batch update)，我们可以进一步提升训练的效率。

3.3 内容感知的概率矩阵分解模型二 (GcPMF)

GcPMF 模型的图模型如图 3.3 所示。假定收集到的数据当中有 N 个用户， M 件商品，用户对商品的评分数据用矩阵 R 表示，矩阵的行向量 R_i 表示一个用户的评分向量，如果用户对某一商品进行了评分，则向量中对应维度的值即为评分，否则对应维度的值为 0；类似的，矩阵的列向量 R_j 表示一个商品的评分向量，如果商品被某一用户评分，则 R_j 中对应维度的值即为评分值，否则该维度值为 0。

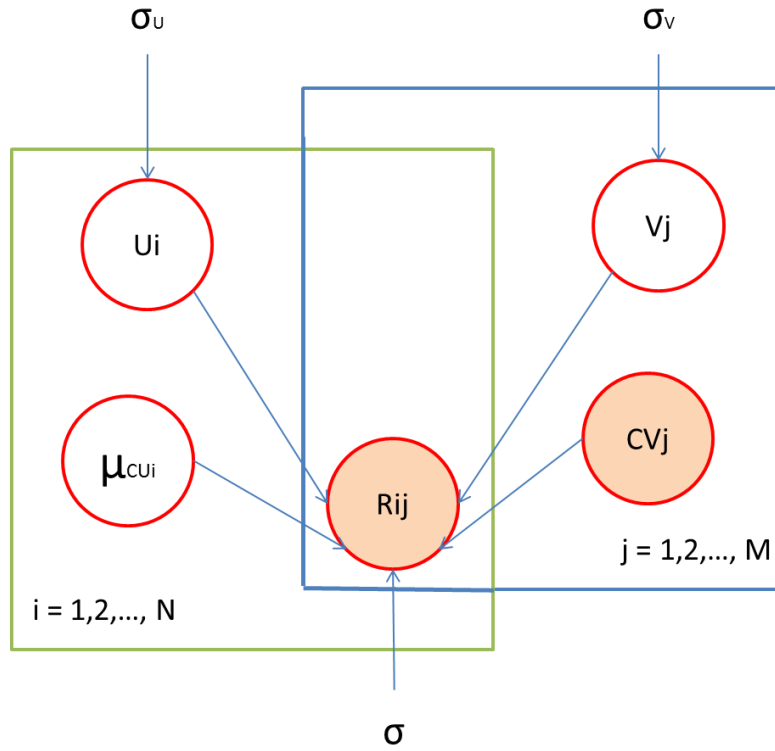


图 3.3 内容感知的概率矩阵分解模型 (GcPMF) 的图模型

在图 3.3 中， U_i 表示第 i 个用户的潜在因子向量， V_j 表示第 j 件商品的潜在因子向量，假定 U_i 和 V_j 均有 D 维，每一维都表征某种潜在的特征； μ_{cui} 表示第 i 个用户的潜在代价向量的期望值， CV_j 表示第 j 件商品的代价相关向量， μ_{cui} 和 CV_j 均有 K 维， CV_j 的每一维对应于一种需要考虑的代价； R_{ij} 表示第 i 个用户对第 j 个商品的评分值，但第 i 个用户对第 j 个商品进行了评分，则 R_{ij} 为评分值，否则 R_{ij} 的值为零。 R_{ij} 是显变

量，可直接从训练数据中获得， CV_j 是第 j 件商品的代价相关向量，它也是显变量，可以离线计算好， σ_U 、 σ_V 、 σ 是系统的超参，其中 σ_U 表示 U_i 对应的高斯分布的方差， σ_V 表示 V_j 对应的高斯分布的方差， σ 表示 R_{ij} 对应的高斯分布的方差。

用 U 表示所有用户的潜在因子向量的集合， V 表示所有商品的潜在因子向量的集合， R 表示用户对商品的评分矩阵， CU 表示所有用户的潜在代价相关向量的期望的集合， CV 表示所有商品的代价相关向量的集合。与 PMF 模型类似，GcPMF 模型中为了表达评分具有一定波动性，用一个高斯分布来表示 R_{ij} 的产生，对照图 3.3，我们可以求出评分矩阵的条件概率为：

$$p(R|U, V, CU, CV, \sigma^2, \sigma_{CU}^2) = \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, \mu_{CV_i}, CV_j), \sigma^2)]^{I_{ij}} \quad (3.28)$$

其中， $\mathcal{N}(x|u, \sigma^2)$ 是期望为 u ，方差为 σ^2 的高斯分布的概率密度函数；此外， I_{ij} 是一个标记函数，用于标记第 i 个用户是否评论了第 j 件商品，其定义如下：

$$I_{ij} = \begin{cases} 1, & \text{第 } i \text{ 个用户评论了第 } j \text{ 件商品;} \\ 0, & \text{其他情况} \end{cases} \quad (3.29)$$

公式 (3.28) 中的 $h(U_i, V_j, \mu_{CV_i}, CV_j)$ 是用户和商品的潜在因子向量和代价相关向量的函数，用于表征第 i 个用户和第 j 件商品在多个维度上的关联度，可用于估计第 i 个用户对第 j 件商品的评分。对于用户的潜在因子向量，与 PMF 模型类似，GcPMF 也使用一个球形的多维高斯分布作为先验分布，因此我们可以得到 U 的条件概率分布为：

$$p(U | \sigma_U^2) = \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \quad (3.30)$$

类似的，对于商品的潜在因子向量，GcPMF 也使用类似的多维高斯分布，但参数有所不同，进而我们可以得到 V 的条件概率分布为：

$$p(V | \sigma_V^2) = \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \quad (3.31)$$

根据图(3.3)所示的图模型以及贝叶斯定理，我们可以推导出 U, V, CU 这三个变量的后验概率为：

$$\begin{aligned}
 p(U, V, CU | R, \sigma_U^2, \sigma_V^2, \sigma^2, \sigma_{CU}^2, CV) &\propto p(R | U, V, CU, CV, \sigma_{CU}^2, \sigma^2) \times p(U | \sigma_U^2) \times p(V | \sigma_V^2) \\
 &= \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, \mu_{CUI}, CV_j), \sigma^2)]^{I_{ij}} \times \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \\
 &\quad \times \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I)
 \end{aligned} \tag{3.32}$$

对式(3.32)中后验概率求对数，可以得到后验概率的对数值为：

$$\begin{aligned}
 \ln p(U, V, CU | R, \sigma_U^2, \sigma_V^2, \sigma^2, \sigma_{CU}^2, CV) &= -\frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, \mu_{CUI}, CV_j))^2 \\
 &\quad - \frac{1}{2\sigma_U^2} \sum_{i=1}^N U_i^T U_i - \frac{1}{2\sigma_V^2} \sum_{j=1}^M V_j^T V_j \\
 &\quad - \frac{1}{2} \left[\left(\sum_{i=1}^N \sum_{j=1}^M I_{ij} \right) \ln \sigma^2 + ND \ln \sigma_U^2 + MD \ln \sigma_V^2 \right] + Const
 \end{aligned} \tag{3.33}$$

其中，Const 表示常量。根据对数函数的单调性，我们可以很容易证明最大化（3.32）式等价于最大化（3.33）式，更进一步，通过观察（3.33）式，我们发现最后一项是常数项，倒数第二项是由模型的超参决定的，因此我们可以归纳出模型的损失函数 E，并将最大化（3.33）式的问题进一步转换为最小化损失函数 E 的问题，损失函数 E 的表达式如下：

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, \mu_{CUI}, CV_j))^2 + \frac{\lambda_U}{2} \sum_{i=1}^N U_i^T U_i + \frac{\lambda_V}{2} \sum_{j=1}^M V_j^T V_j \tag{3.34}$$

其中， λ_U 的表达式为：

$$\lambda_U = \frac{\sigma^2}{\sigma_U^2} \tag{3.35}$$

λ_V 的表达式为：

$$\lambda_v = \frac{\sigma^2}{\sigma_v^2} \quad (3.36)$$

为了表达第 i 个用户与第 j 件商品的关联, 这里将 $h(U_i, V_j, \mu_{cvi}, CV_j)$ 定义为 CV_j 与 μ_{cvi} 的余弦相似度, 再加上 U_i 与 V_j 的余弦相似度, 其表达式为:

$$h(U_i, V_j, \mu_{cvi}, CV_j) = \frac{cvi^T cvj}{|cvi||cvj|} + \frac{ui^T vj}{|ui||vj|} \quad (3.37)$$

接下来, 我们便可以利用梯度下降法求解损失函数 E 的小值。

E 对 U_i 求偏导得到:

$$\frac{\partial E}{\partial ui} = \sum_{j=1}^M I_{ij} (h(U_i, V_j, CV_i, CV_j) - R_{ij}) \cdot S_a' (U_i, V_j) \cdot V_j + \lambda_v U_i \quad (3.38)$$

E 对 V_j 求偏导得到:

$$\frac{\partial E}{\partial vj} = \sum_{i=1}^N I_{ij} (h(U_i, V_j, CV_i, CV_j) - R_{ij}) \cdot S_b' (U_i, V_j) \cdot U_i^T + \lambda_v V_j \quad (3.39)$$

E 对 CU_i 求偏导得到:

$$\frac{\partial E}{\partial cvi} = \sum_{j=1}^M I_{ij} (h(U_i, V_j, CV_i, CV_j) - R_{ij}) U_i^T V_j \cdot S_a' (CV_i, CV_j) \quad (3.40)$$

其中 $S(a, b)$ 表示 a 和 b 的余弦相似度, 定义如下:

$$S(a, b) = \frac{a^T b}{|a||b|} \quad (3.41)$$

$S_a' (a, b)$ 是指 $S(a, b)$ 对 a 的偏导, 定义如下:

$$S_a' (a, b) = \frac{(b \cdot |a|^2 - a^T b \cdot a)}{|a||b||a|^2} \quad (3.42)$$

$S_b' (a, b)$ 是指 $S(a, b)$ 对 b 的偏导, 定义如下:

$$S_b' (a, b) = S_a' (b, a) \quad (3.43)$$

综上所述, GcPMF 模型将用户对商品的评分看成是一个由用户和商品关联程度(由潜在因子向量和代价向量共同决定)为期望, 超参 σ^2 为方差的高斯分布产生的, 每个

评分之间相互独立，并假定用户和商品的潜在因子向量的先验分布是特殊形式的多维高斯分布（各维期望为 0，协方差矩阵为超参与单位矩阵 I 的乘积），还假设用户潜在代价相关向量服从多维高斯分布，用户潜在代价向量与商品代价向量通过余弦相似度进行关联，再通过贝叶斯定理，可以求出 U ， V ， CU 的后验概率，再将后验概率取对数并利用梯度下降算法求出使之最大化的参数 U ， V 和 CU ，再通过 (3.37) 式去估计那些评分矩阵缺失的值。需要注意的是通过 (3.37) 式计算得到的估计值可能会超出评分的范围，事实上，在实践中，人们往往会将 (3.37) 式得到的值再通过一个逻辑斯特函数（logistic function）将范围限制到 $[0,1]$ ，类似的，对于评分矩阵中的 R_{ij} ，则会通过最大最小值标准化方法（Max-Min Normalization）进行标准化，使它们的范围也变成 $[0,1]$ 。

通过观察 (3.34) 式，我们可以发现 GcPMF 模型损失函数的计算复杂度为 $O(L(D + K))$ ， L 表示评分矩阵中非零值的个数， D 和 K 分别表示潜在因子向量的维数与代价相关向量的维数，类似的，观察 (3.38) 式、(3.39) 式以及 (3.40) 式，我们还可以看出损失函数的各个偏导的计算复杂度也为 $O(L(D + K))$ ，因此 GcPMF 模型的复杂度为 $O(L(D + K))$ 。这表明 GcPMF 模型的复杂度跟评分矩阵中的非零值个数是线性关系，因此 GcPMF 模型的扩展性比较强，比较适合并行计算，通过批量更新（batch update），我们可以进一步提升训练的效率。

3.4 内容感知的概率矩阵分解模型三（CsPMF）

CsPMF 模型的图模型如图 3.4 所示。假定收集到的数据当中有 N 个用户， M 件商品，用户对商品的评分数据用矩阵 R 表示，矩阵的行向量 R_i 表示一个用户的评分向量，如果用户对某一商品进行了评分，则向量中对应维度的值即为评分，否则对应维度的值为 0；类似的，矩阵的列向量 R_j 表示一个商品的评分向量，如果商品被某一用户评分，则 R_j 中对应维度的值即为评分值，否则该维度值为 0。

在图 3.4 中， U_i 表示第 i 个用户的潜在因子向量， V_j 表示第 j 件商品的潜在因子向量，假定 U_i 和 V_j 均有 D 维，每一维都表征某种潜在的特征； CU_i 表示第 i 个用户的潜在代价向量， CV_j 表示第 j 件商品的代价相关向量， CU_i 和 CV_j 均有 K 维， CV_j 的每一维对应于一种需要考虑的代价； R_{ij} 表示第 i 个用户对第 j 个商品的评分值，但第 i 个用户对第 j 个商品进行了评分，则 R_{ij} 为评分值，否则 R_{ij} 的值为零。 R_{ij} 是显变量，可直接从训练数据中获得， CV_j 是第 j 件商品的代价相关向量，它也是显变量，可以离线计算好， σ_U 、 σ_V 、 σ 、 w_1 、 w_2 是系统的超参，其中 σ_U 表示 U_i 对应的高斯分布的方差， σ_V 表示 V_j 对应的高斯分布的方差， σ 表示 R_{ij} 对应的高斯分布的方差， w_1 与 w_2 分别表示代价相关因子和其余潜在因子之间的权重，用于调节代价相关因子在模型

中的比重。

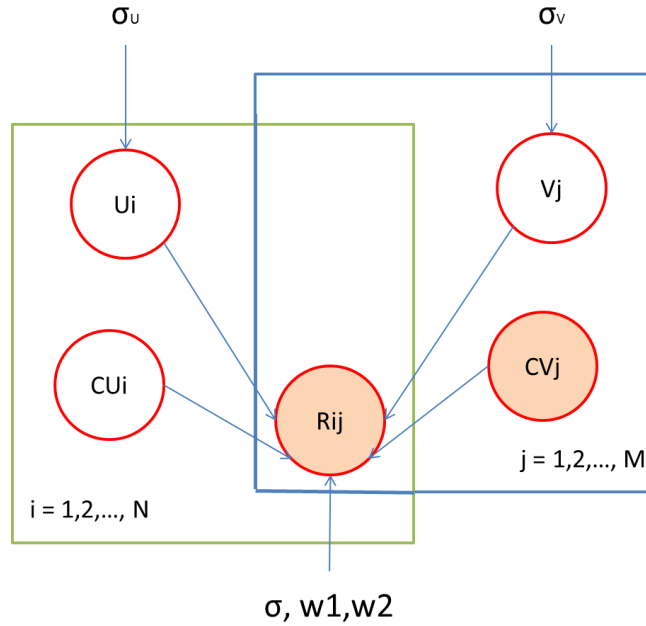


图 3.4 内容感知的概率矩阵分解模型（CsPMF）的图模型

用 U 表示所有用户的潜在因子向量的集合， V 表示所有商品的潜在因子向量的集合， R 表示用户对商品的评分矩阵， CU 表示所有用户的潜在代价相关向量的集合， CV 表示所有商品的代价相关向量的集合。与 PMF 模型类似，CsPMF 模型中为了表达评分具有一定得波动性，用一个高斯分布来表示 R_{ij} 的产生，对照图 3.4，我们可以求出评分矩阵的条件概率为：

$$p(R|U, V, CU, CV, \sigma^2) = \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, CU_i, CV_j), \sigma^2)]^{I_{ij}} \quad (3.44)$$

其中， $\mathcal{N}(x | u, \sigma^2)$ 是期望为 u ，方差为 σ^2 的高斯分布的概率密度函数；此外， I_{ij} 是一个标记函数，用于标记第 i 个用户是否评论了第 j 件商品，其定义如下：

$$I_{ij} = \begin{cases} 1, & \text{第 } i \text{ 个用户评论了第 } j \text{ 件商品;} \\ 0, & \text{其他情况} \end{cases} \quad (3.45)$$

公式 (3.44) 中的 $h(U_i, V_j, CU_i, CV_j)$ 是用户和商品的潜在因子向量和代价相关向量的函数，用于表征第 i 个用户和第 j 件商品在多个维度上的关联度，值得注意的是该函数包含系统超参 $w1$ 和 $w2$ ，用于调节代价相关因子与其余潜在因子之间的比重。对于用户的潜在因子向量，与 PMF 模型类似，CsPMF 也使用一个球形的多维高斯分布作为先验分布，因此我们可以得到 U 的条件概率分布为：

$$p(U | \sigma_U^2) = \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \quad (3.46)$$

类似的，对于商品的潜在因子向量，CsPMF 也使用类似的多维高斯分布，但参数有所不同，进而我们可以得到 V 的条件概率分布为：

$$p(V | \sigma_V^2) = \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \quad (3.47)$$

根据图(3.4)所示的图模型以及贝叶斯定理，我们可以推导出 U, V, CU 这三个变量的后验概率为：

$$\begin{aligned} p(U, V, CU | R, \sigma_U^2, \sigma_V^2, \sigma^2, CV) &\propto p(R | U, V, CU, CV, \sigma^2) \times p(U | \sigma_U^2) \times p(V | \sigma_V^2) \\ &= \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, CU_i, CV_j), \sigma^2)]^{I_{ij}} \times \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \\ &\quad \times \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \end{aligned} \quad (3.48)$$

对式(3.48)中后验概率求对数，可以得到后验概率的对数值为：

$$\begin{aligned} \ln p(U, V, CU | R, \sigma_U^2, \sigma_V^2, \sigma^2, CV) &= -\frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, CU_i, CV_j))^2 - \frac{1}{2\sigma_U^2} \sum_{i=1}^N U_i^T U_i \\ &\quad - \frac{1}{2\sigma_V^2} \sum_{j=1}^M V_j^T V_j - \frac{1}{2} \left[\left(\sum_{i=1}^N \sum_{j=1}^M I_{ij} \right) \ln \sigma^2 + ND \ln \sigma_U^2 + MD \ln \sigma_V^2 \right] + Const \end{aligned} \quad (3.49)$$

其中， $Const$ 表示常量。根据对数函数的单调性，我们可以很容易证明最大化（3.48）式等价于最大化（3.49）式，更进一步，通过观察（3.49）式，我们发现最后一项是常数项，倒数第二项是由模型的超参决定的，因此我们可以归纳出模型的损失函数 E ，并将最大化（3.49）式的问题进一步转换为最小化损失函数 E 的问题，损失函数 E 的表达式如下：

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, CU_i, CV_j))^2 + \frac{\lambda_U}{2} \sum_{i=1}^N U_i^T U_i + \frac{\lambda_V}{2} \sum_{j=1}^M V_j^T V_j \quad (3.50)$$

其中， λ_U 的表达式为：

$$\lambda_U = \frac{\sigma^2}{\sigma_U^2} \quad (3.51)$$

λ_V 的表达式为:

$$\lambda_V = \frac{\sigma^2}{\sigma_V^2} \quad (3.52)$$

为了更好的控制代价相关向量在模型中的比重, 这里将 $h(U_i, V_j, CU_i, CV_j)$ 定义为 CU_i 与 CV_j 的内积与 U_i 和 V_j 的内积的加权和, 用于表征第 i 个用户和第 j 件商品的关联程度, 其表达式为:

$$h(U_i, V_j, CU_i, CV_j) = w1 \cdot CU_i^T CV_j + w2 \cdot U_i^T V_j \quad (3.53)$$

接下来, 我们便可以利用梯度下降法求解损失函数 E 的小值。

E 对 U_i 求偏导得到:

$$\frac{\partial E}{\partial U_i} = \sum_{j=1}^M I_{ij}(h(U_i, V_j, CU_i, CV_j) - R_{ij}) \cdot w2 \cdot S_a'(U_i, V_j) + \lambda_U U_i \quad (3.54)$$

E 对 V_j 求偏导得到:

$$\frac{\partial E}{\partial V_j} = \sum_{i=1}^N I_{ij}(h(U_i, V_j, CU_i, CV_j) - R_{ij}) \cdot w2 \cdot S_b'(U_i, V_j) + \lambda_V V_j \quad (3.55)$$

E 对 CU_i 求偏导得到:

$$\frac{\partial E}{\partial CU_i} = \sum_{j=1}^M I_{ij}(h(U_i, V_j, CU_i, CV_j) - R_{ij}) \cdot w1 \cdot S_a'(CU_i, CV_j) \quad (3.56)$$

其中 $S(a, b)$ 表示 a 和 b 的向量内积值, 定义如下:

$$S(a, b) = a^T b \quad (3.57)$$

$S_a'(a, b)$ 是指 $S(a, b)$ 对 a 的偏导, 定义如下:

$$S_a'(a, b) = b \quad (3.58)$$

$S_b'(a, b)$ 是指 $S(a, b)$ 对 b 的偏导, 且 $S_b'(a, b) = a$ 。

综上所述, CsPMF 模型将用户对商品的评分看成是一个由用户和商品关联程度(由

潜在因子向量和代价向量共同决定)为期望,超参 σ^2 为方差的高斯分布产生的,每个评分之间相互独立,并假定用户和商品的潜在因子向量的先验分布是特殊形式的多维高斯分布(各维期望为0,协方差矩阵为超参与单位矩阵 I 的乘积),用户和商品之间的关联程度用代价相关因子向量余弦相似度和其余潜在因子向量余弦相似度的加权和表征,再通过贝叶斯定理,可以求出 U , V , CU 的后验概率,再将后验概率取对数并利用梯度下降算法求出使之最大化的参数 U , V 和 CU ,再通过(3.53)式去估计那些评分矩阵缺失的值。需要注意的是通过(3.53)式计算得到的估计值可能会超出评分的范围,事实上,在实践中,人们往往会将(3.53)式得到的值再通过一个逻辑斯特函数(logistic function)将范围限制到 $[0,1]$,类似的,对于评分矩阵中的 R_{ij} ,则会通过最大最小值标准化方法(Max-Min Normalization)进行标准化,使它们的范围也变成 $[0,1]$ 。

通过观察(3.50)式,我们可以发现cPMF模型损失函数的计算复杂度为 $O(L(D+K))$, L 表示评分矩阵中非零值的个数, D 和 K 分别表示潜在因子向量的维数与代价相关向量的维数,类似的,观察(3.54)式、(3.55)式以及(3.56)式,我们还可以看出损失函数的各个偏导的计算复杂度也为 $O(L(D+K))$,因此CsPMF模型的复杂度为 $O(L(D+K))$ 。这表明CsPMF模型的复杂度跟评分矩阵中的非零值个数是线性关系,因此CsPMF模型的扩展性比较强,比较适合并行计算,通过批量更新(batch update),我们可以进一步提升训练的效率。

3.5 内容感知的概率矩阵分解模型四(GCsPMF)

GCsPMF模型的图模型如图3.5所示。假定收集到的数据当中有 N 个用户, M 件商品,用户对商品的评分数据用矩阵 R 表示,矩阵的行向量 R_i 表示一个用户的评分向量,如果用户对某一商品进行了评分,则向量中对应维度的值即为评分,否则对应维度的值为0;类似的,矩阵的列向量 R_j 表示一个商品的评分向量,如果商品被某一用户评分,则 R_j 中对应维度的值即为评分值,否则该维度值为0。在图3.5中, U_i 表示第 i 个用户的潜在因子向量, V_j 表示第 j 件商品的潜在因子向量,假定 U_i 和 V_j 均有 D 维,每一维都表征某种潜在的特征; μ_{cui} 表示第 i 个用户的潜在代价向量的期望值, CV_j 表示第 j 件商品的代价相关向量, μ_{cui} 和 CV_j 均有 K 维, CV_j 的每一维对应于一种需要考虑的代价; R_{ij} 表示第 i 个用户对第 j 个商品的评分值,但第 i 个用户对第 j 个商品进行了评分,则 R_{ij} 为评分值,否则 R_{ij} 的值为零。 R_{ij} 是显变量,可直接从训练数据中获得, CV_j 是第 j 件商品的代价相关向量,它也是显变量,可以离线计算好, σ_U 、 σ_V 、 σ 、 w_1 、 w_2 是系统的超参,其中 σ_U 表示 U_i 对应的高斯分布的方差, σ_V 表示 V_j 对应的高斯分布的方差, σ 表示 R_{ij} 对应的高斯分布的方差, w_1 和 w_2 是代价相关因子与其余潜在因子的权重参数,用于调节代价相关因子在模型中的比重。

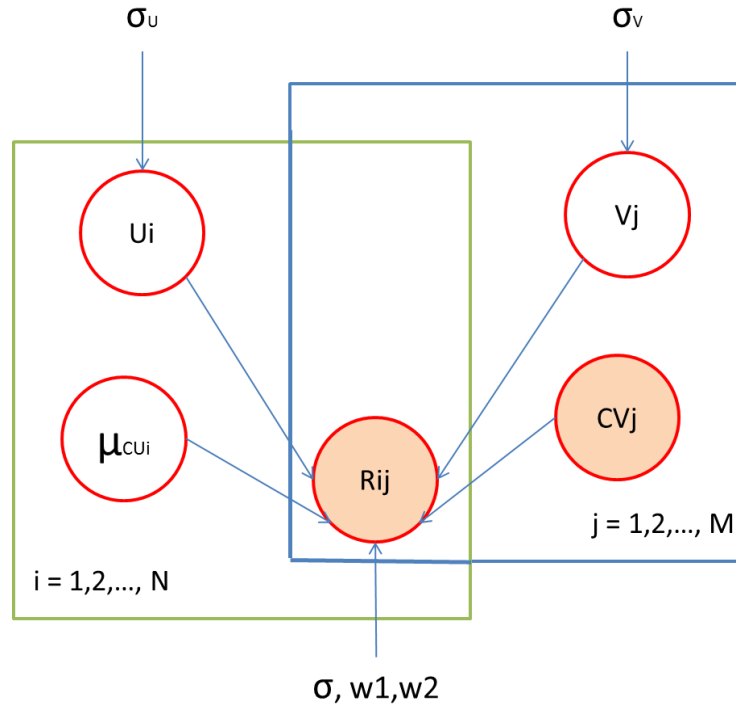


图 3.5 内容感知的概率矩阵分解模型（GCsPMF）的图模型

用 U 表示所有用户的潜在因子向量的集合， V 表示所有商品的潜在因子向量的集合， R 表示用户对商品的评分矩阵， CU 表示所有用户的潜在代价相关向量的期望的集合， CV 表示所有商品的代价相关向量的集合。与 PMF 模型类似，GCsPMF 模型中为了表达评分具有一定得波动性，用一个高斯分布来表示 R_{ij} 的产生，对照图 3.5，我们可以求出评分矩阵的条件概率为：

$$p(R|U, V, CU, CV, \sigma^2, \sigma_{cu}^2) = \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, \mu_{cui}, CV_j), \sigma^2)]^{I_{ij}} \quad (3.59)$$

其中， $\mathcal{N}(x|u, \sigma^2)$ 是期望为 u ，方差为 σ^2 的高斯分布的概率密度函数；此外， I_{ij} 是一个标记函数，用于标记第 i 个用户是否评论了第 j 件商品，其定义如下：

$$I_{ij} = \begin{cases} 1, & \text{第 } i \text{ 个用户评论了第 } j \text{ 件商品;} \\ 0, & \text{其他情况} \end{cases} \quad (3.60)$$

公式 (3.59) 中的 $h(U_i, V_j, \mu_{cui}, CV_j)$ 是用户和商品的潜在因子向量和代价相关向量的函数，用于表征第 i 个用户和第 j 件商品在多个维度上的关联度，可用于估计第 i 个用户对第 j 件商品的评分，与 GcPMF 模型不同的是，GCsPMF 模型在这里增加了 $w1, w2$ 这两个系统超参，用于调节代价相关因子与其余潜在因子影响该评分的比重。对于用

户的潜在因子向量，与 PMF 模型类似，GCsPMF 也使用一个球形的多维高斯分布作为先验分布，因此我们可以得到 U 的条件概率分布为：

$$p(U | \sigma_U^2) = \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \quad (3.61)$$

类似的，对于商品的潜在因子向量，GCsPMF 也使用类似的多维高斯分布，但参数有所不同，进而我们可以得到 V 的条件概率分布为：

$$p(V | \sigma_V^2) = \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \quad (3.62)$$

根据图(3.5)所示的图模型以及贝叶斯定理，我们可以推导出 U, V, CU 这三个变量的后验概率为：

$$\begin{aligned} p(U, V, CU | R, \sigma_U^2, \sigma_V^2, \sigma^2, \sigma_{CU}^2, CV) &\propto p(R | U, V, CU, CV, \sigma_{CU}^2, \sigma^2) \times p(U | \sigma_U^2) \times p(V | \sigma_V^2) \\ &= \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, \mu_{CUI}, CV_j), \sigma^2)]^{I_{ij}} \times \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \\ &\quad \times \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \end{aligned} \quad (3.63)$$

对式(3.63)中后验概率求对数，可以得到后验概率的对数值为：

$$\begin{aligned} \ln p(U, V, CU | R, \sigma_U^2, \sigma_V^2, \sigma^2, \sigma_{CU}^2, CV) &= -\frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, \mu_{CUI}, CV_j))^2 \\ &\quad - \frac{1}{2\sigma_U^2} \sum_{i=1}^N U_i^T U_i - \frac{1}{2\sigma_V^2} \sum_{j=1}^M V_j^T V_j \\ &\quad - \frac{1}{2} \left[\left(\sum_{i=1}^N \sum_{j=1}^M I_{ij} \right) \ln \sigma^2 + ND \ln \sigma_U^2 + MD \ln \sigma_V^2 \right] + Const \end{aligned} \quad (3.64)$$

其中，Const 表示常量。根据对数函数的单调性，我们可以很容易证明最大化 (3.63) 式等价于最大化 (3.64) 式，更进一步，通过观察 (3.64) 式，我们发现最后一项是常数项，倒数第二项是由模型的超参决定的，因此我们可以归纳出模型的损失函数 E ，并将最大化 (3.64) 式的问题进一步转换为最小化损失函数 E 的问题，损失函数 E 的表达式如下：

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, \mu_{CVi}, CV_j))^2 + \frac{\lambda_U}{2} \sum_{i=1}^N U_i^T U_i + \frac{\lambda_V}{2} \sum_{j=1}^M V_j^T V_j \quad (3.65)$$

其中, λ_U 的表达式为:

$$\lambda_U = \frac{\sigma^2}{\sigma_U^2} \quad (3.66)$$

λ_V 的表达式为:

$$\lambda_V = \frac{\sigma^2}{\sigma_V^2} \quad (3.67)$$

为了表达第 i 个用户与第 j 件商品的关联, 这里将 $h(U_i, V_j, \mu_{CVi}, CV_j)$ 定义为 CV_j 和 μ_{CVi} 的余弦相似度与 U_i 和 V_j 的余弦相似度的加权和, 其表达式为:

$$h(U_i, V_j, \mu_{CVi}, CV_j) = w_1 \frac{\mu_{CVi}^T CV_j}{|\mu_{CVi}| |CV_j|} + w_2 \frac{U_i^T V_j}{|U_i| |V_j|} \quad (3.68)$$

接下来, 我们便可以利用梯度下降法求解损失函数 E 的小值。

E 对 U_i 求偏导得到:

$$\frac{\partial E}{\partial U_i} = \sum_{j=1}^M I_{ij} (h(U_i, V_j, \mu_{CVi}, CV_j) - R_{ij}) \cdot w_2 \cdot S_a' (U_i, V_j) \cdot V_j + \lambda_U U_i \quad (3.69)$$

E 对 V_j 求偏导得到:

$$\frac{\partial E}{\partial V_j} = \sum_{i=1}^N I_{ij} (h(U_i, V_j, \mu_{CVi}, CV_j) - R_{ij}) \cdot w_2 \cdot S_b' (U_i, V_j) \cdot U_i^T + \lambda_V V_j \quad (3.70)$$

E 对 μ_{CVi} 求偏导得到:

$$\frac{\partial E}{\partial \mu_{CVi}} = \sum_{j=1}^M I_{ij} (h(U_i, V_j, \mu_{CVi}, CV_j) - R_{ij}) U_i^T V_j \cdot S_a' (\mu_{CVi}, CV_j) \quad (3.71)$$

其中 $S(a, b)$ 表示 a 和 b 的余弦相似度, 定义如下:

$$S(a, b) = \frac{a^T b}{|a| |b|} \quad (3.72)$$

$S_a'(a, b)$ 是指 $S(a, b)$ 对 a 的偏导，定义如下：

$$S_a'(a, b) = \frac{(b \cdot |a|^2 - a^T b \cdot a)}{|a||b||a|^2} \quad (3.73)$$

$S_b'(a, b)$ 是指 $S(a, b)$ 对 b 的偏导，由于余弦相似度具有对称性，可得：

$$S_b'(a, b) = S_a'(b, a) \quad (3.74)$$

$S_b'(a, b)$ 定义如下：

$$S_b'(a, b) = \frac{(a \cdot |b|^2 - a^T b \cdot b)}{|a||b||b|^2} \quad (3.75)$$

综上所述，GCsPMF 模型将用户对商品的评分看成是一个由用户和商品关联程度（由潜在因子向量和代价向量共同决定）为期望，超参 σ^2 为方差的高斯分布产生的，每个评分之间相互独立，并假定用户和商品的潜在因子向量的先验分布是特殊形式的多维高斯分布（各维期望为 0，协方差矩阵为超参与单位矩阵 I 的乘积），还假设用户潜在代价相关向量服从多维高斯分布，用户潜在代价向量与商品代价向量通过余弦相似度进行关联，代价相关因子和其余潜在因子通过 w_1 和 w_2 进行加权，利用贝叶斯定理，可以求出 U , V , CU 的后验概率，再将后验概率取对数并利用梯度下降算法求出使之最大化的参数 U , V 和 CU ，然后通过 (3.68) 式去估计那些评分矩阵缺失的值。需要注意的是通过 (3.68) 式计算得到的估计值可能会超出评分的范围，事实上，在实践中，人们往往会将 (3.68) 式得到的值再通过一个逻辑斯特函数（logistic function）将范围限制到 $[0, 1]$ ，类似的，对于评分矩阵中的 R_{ij} ，则会通过最大最小值标准化方法（Max-Min Normalization）进行标准化，使它们的范围也变成 $[0, 1]$ 。

通过观察 (3.65) 式，我们可以发现 GCsPMF 模型损失函数的计算复杂度为 $O(L(D + K))$ ， L 表示评分矩阵中非零值的个数， D 和 K 分别表示潜在因子向量的维数与代价相关向量的维数，类似的，观察 (3.69) 式、(3.70) 式以及 (3.71) 式，我们还可以看出损失函数的各个偏导的计算复杂度也为 $O(L(D + K))$ ，因此 GcPMF 模型的复杂度为 $O(L(D + K))$ 。这表明 GCsPMF 模型的复杂度跟评分矩阵中的非零值个数是线性关系，因此 GCsPMF 模型的扩展性比较强，比较适合并行计算，通过批量更新（batch update），我们可以进一步提升训练的效率。

3.6 内容感知的概率矩阵分解模型五（BCsPMF）

BCsPMF 模型的图模型如图 3.6 所示。假定收集到的数据当中有 N 个用户， M 件商品，用户对商品的评分数据用矩阵 \mathbf{R} 表示，矩阵的行向量 \mathbf{R}_i 表示一个用户的评分向量，如果用户对某一商品进行了评分，则向量中对应维度的值即为评分，否则对应维度的值为 0；类似的，矩阵的列向量 \mathbf{R}_j 表示一个商品的评分向量，如果商品被某一用户评分，则 \mathbf{R}_j 中对应维度的值即为评分值，否则该维度值为 0。

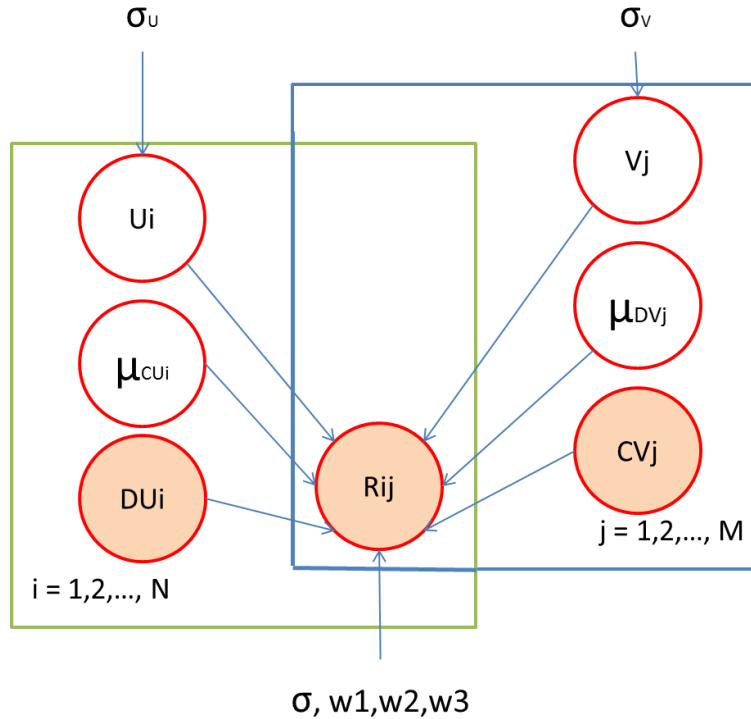


图 3.6 内容感知的概率矩阵分解模型（BCsPMF）的图模型

在图 3.6 中， U_i 表示第 i 个用户的潜在因子向量， V_j 表示第 j 件商品的潜在因子向量，假定 U_i 和 V_j 均有 D 维，每一维都表征某种潜在的特征； μ_{cui} 表示第 i 个用户的潜在代价向量的期望值， CV_j 表示第 j 件商品的代价相关向量， μ_{cui} 和 CV_j 均有 P 维， CV_j 的每一维对应于一种需要考虑的代价； μ_{dvj} 表示第 j 件商品的潜在代价向量的期望值， DU_i 表示第 i 个用户的代价相关向量， μ_{dvj} 和 DU_i 均有 Q 维， DU_i 的每一维对应于一种可以离线计算得到的用户偏好； R_{ij} 表示第 i 个用户对第 j 个商品的评分值，但第 i 个用户对第 j 个商品进行了评分，则 R_{ij} 为评分值，否则 R_{ij} 的值为零。 R_{ij} 是显变量，可直接从训练数据中获得， CV_j 是第 j 件商品的代价相关向量，它也是显变量，可以离线计算好， σ_u 、 σ_v 、 σ 、 $w1$ 、 $w2$ 、 $w3$ 是系统的超参，其中 σ_u 表示 U_i 对应的高斯分布的方差， σ_v 表示 V_j 对应的高斯分布的方差， σ 表示 R_{ij} 对应的高斯分布的方差， $w1$ 、 $w2$ 、 $w3$ 是用户代价因子、商品代价因子与其余潜在因子的权重参数，用于调节代价相

关因子在模型中的比重。

用 U 表示所有用户的潜在因子向量的集合, V 表示所有商品的潜在因子向量的集合, R 表示用户对商品的评分矩阵, CU 表示所有用户的潜在代价相关向量的期望的集合, CV 表示所有商品的代价相关向量的集合, DU 表示所有用户的代价相关向量的集合, DV 表示所有商品的潜在代价相关向量的期望的集合。与 PMF 模型类似, BCsPMF 模型中为了表达评分具有一定得波动性, 用一个高斯分布来表示 R_{ij} 的产生, 对照图 3.6, 我们可以求出评分矩阵的条件概率为:

$$p(R|U,V,CU,CV,DU,DV,\sigma^2,\sigma_{CU}^2,\sigma_{DV}^2) = \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, \mu_{CV_i}, CV_j, \mu_{DV_j}, DU_i), \sigma^2)]^{I_{ij}} \quad (3.76)$$

其中, $\mathcal{N}(x|u, \sigma^2)$ 是期望为 u , 方差为 σ^2 的高斯分布的概率密度函数; 此外, I_{ij} 是一个标记函数, 用于标记第 i 个用户是否评论了第 j 件商品, 其定义如下:

$$I_{ij} = \begin{cases} 1, & \text{第 } i \text{ 个用户评论了第 } j \text{ 件商品;} \\ 0, & \text{其他情况} \end{cases} \quad (3.77)$$

公式 (3.76) 中的 $h(U_i, V_j, \mu_{CV_i}, CV_j, \mu_{DV_j}, DU_i)$ 是用户和商品的潜在因子向量和代价相关向量的函数, 用于表征第 i 个用户和第 j 件商品在多个维度上的关联度, 可用于估计第 i 个用户对第 j 件商品的评分, 与 GCsPMF 模型类似, BCsPMF 模型在这里用 w_1, w_2, w_3 这三个系统超参来调节代价相关因子与其余潜在因子影响该评分的比重。对于用户的潜在因子向量, 与 PMF 模型类似, BCsPMF 也使用一个球形的多维高斯分布作为先验分布, 因此我们可以得到 U 的条件概率分布为:

$$p(U|\sigma_U^2) = \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \quad (3.78)$$

类似的, 对于商品的潜在因子向量, BCsPMF 也使用类似的多维高斯分布, 但参数有所不同, 进而我们可以得到 V 的条件概率分布为:

$$p(V|\sigma_V^2) = \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I) \quad (3.79)$$

根据图(3.6)所示的图模型以及贝叶斯定理, 我们可以推导出 U, V, CU, DV 这三个变量的后验概率为:

$$\begin{aligned}
 & p(U, V, CU, DV | R, \sigma_U^2, \sigma_V^2, \sigma^2, \sigma_{CU}^2, CV, \sigma_{DV}^2, DU) \\
 & \propto p(R | U, V, CU, CV, DU, DV, \sigma^2, \sigma_{CU}^2, \sigma_{DV}^2) \times p(U | \sigma_U^2) \times p(V | \sigma_V^2) \\
 & = \prod_{i=1}^N \prod_{j=1}^M [\mathcal{N}(R_{ij} | h(U_i, V_j, \mu_{CU_i}, CV_j, \mu_{DV_j}, DU_i), \sigma^2)]^{I_{ij}} \times \prod_{i=1}^N \mathcal{N}(U_i | 0, \sigma_U^2 I) \\
 & \quad \times \prod_{j=1}^M \mathcal{N}(V_j | 0, \sigma_V^2 I)
 \end{aligned} \tag{3.80}$$

对式(3.80)中后验概率求对数，可以得到后验概率的对数值为：

$$\begin{aligned}
 & \ln p(U, V, CU, DV | R, \sigma_U^2, \sigma_V^2, \sigma^2, \sigma_{CU}^2, CV, \sigma_{DV}^2, DU) \\
 & = -\frac{1}{2\sigma^2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, \mu_{CU_i}, CV_j, \mu_{DV_j}, DU_i))^2 - \frac{1}{2\sigma_U^2} \sum_{i=1}^N U_i^T U_i \\
 & \quad - \frac{1}{2\sigma_V^2} \sum_{j=1}^M V_j^T V_j - \frac{1}{2} \left[\left(\sum_{i=1}^N \sum_{j=1}^M I_{ij} \right) \ln \sigma^2 + ND \ln \sigma_U^2 + MD \ln \sigma_V^2 \right] + Const
 \end{aligned} \tag{3.81}$$

其中，Const 表示常量。根据对数函数的单调性，我们可以很容易证明最大化（3.80）式等价于最大化（3.81）式，更进一步，通过观察（3.81）式，我们发现最后一项是常数项，倒数第二项是由模型的超参决定的，因此我们可以归纳出模型的损失函数 E，并将最大化（3.81）式的问题进一步转换为最小化损失函数 E 的问题，损失函数 E 的表达式如下：

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M I_{ij} (R_{ij} - h(U_i, V_j, \mu_{CU_i}, CV_j, \mu_{DV_j}, DU_i))^2 + \frac{\lambda_U}{2} \sum_{i=1}^N U_i^T U_i + \frac{\lambda_V}{2} \sum_{j=1}^M V_j^T V_j \tag{3.82}$$

其中， λ_U 的表达式为：

$$\lambda_U = \frac{\sigma^2}{\sigma_U^2} \tag{3.83}$$

λ_V 的表达式为：

$$\lambda_V = \frac{\sigma^2}{\sigma_V^2} \tag{3.84}$$

为了表达第 i 个用户与第 j 件商品的关联，这里将 $h(U_i, V_j, \mu_{cvi}, CV_j, \mu_{dvj}, DU_i)$ 定义为用户代价相关因子、商品代价相关因子以及其余潜在因子的函数的加权和，其表达式为：

$$h(U_i, V_j, \mu_{cvi}, CV_j, \mu_{dvj}, DU_i) = w_1 \cdot \mu_{cvi}^T CV_j + w_2 \cdot \mu_{dvj}^T DU_i + w_3 \cdot U_i^T V_j \quad (3.85)$$

其中， w_1, w_2, w_3 为系统超参，用于控制代价相关因子和其余潜在因子对预测评分影响的比重。接下来，我们便可以利用梯度下降法求解损失函数 E 的小值。

E 对 U_i 求偏导得到：

$$\frac{\partial E}{\partial U_i} = \lambda_U U_i + \sum_{j=1}^M I_{ij} (h(U_i, V_j, \mu_{cvi}, CV_j, \mu_{dvj}, DU_i) - R_{ij}) \cdot w_3 \cdot S_a' (U_i, V_j) \cdot V_j \quad (3.86)$$

E 对 V_j 求偏导得到：

$$\frac{\partial E}{\partial V_j} = \lambda_V V_j + \sum_{i=1}^N I_{ij} (h(U_i, V_j, \mu_{cvi}, CV_j, \mu_{dvj}, DU_i) - R_{ij}) \cdot w_3 \cdot S_b' (U_i, V_j) \cdot U_i^T \quad (3.87)$$

E 对 μ_{cvi} 求偏导得到：

$$\frac{\partial E}{\partial \mu_{cvi}} = \sum_{j=1}^M I_{ij} (h(U_i, V_j, \mu_{cvi}, CV_j, \mu_{dvj}, DU_i) - R_{ij}) \cdot w_1 \cdot S_a' (\mu_{cvi}, CV_j) \cdot CV_j \quad (3.88)$$

E 对 μ_{dvj} 求偏导得到：

$$\frac{\partial E}{\partial \mu_{dvj}} = \sum_{i=1}^N I_{ij} (h(U_i, V_j, \mu_{cvi}, CV_j, \mu_{dvj}, DU_i) - R_{ij}) \cdot w_2 \cdot S_a' (\mu_{dvj}, DU_i) \cdot DU_i \quad (3.89)$$

其中 $S(a, b)$ 表示 a 和 b 的向量内积值，定义如下：

$$S(a, b) = a^T b \quad (3.90)$$

$S_a' (a, b)$ 是指 $S(a, b)$ 对 a 的偏导，定义如下：

$$S_a' (a, b) = b \quad (3.91)$$

$S_b' (a, b)$ 是指 $S(a, b)$ 对 b 的偏导，且 $S_b' (a, b) = a$ 。

综上所述，BCsPMF 模型将用户对商品的评分看成是一个由用户和商品关联程度（由潜在因子向量和代价向量共同决定）为期望，超参 σ^2 为方差的高斯分布产生的，每个评分之间相互独立，并假定用户和商品的潜在因子向量的先验分布是特殊形式的

多维高斯分布（各维期望为 0，协方差矩阵为超参与单位矩阵 I 的乘积），还假设用户的潜在代价相关向量服从多维高斯分布，用户潜在代价向量与商品代价向量通过向量内积进行关联，类似的，商品的潜在代价相关向量也服从多维高斯分布，商品的潜在代价向量与用户的代价向量通过向量内积进行关联，代价相关因子和其余潜在因子通过 w_1 、 w_2 、 w_3 进行加权，利用贝叶斯定理，可以求出 U ， V ， CU ， DV 的后验概率，再将后验概率取对数并利用梯度下降算法求出使之最大化的参数 U ， V ， CU 和 DV ，然后通过 (3.85) 式去估计那些评分矩阵缺失的值。需要注意的是通过 (3.85) 式计算得到的估计值可能会超出评分的范围，事实上，在实践中，人们往往会将 (3.85) 式得到的值再通过一个逻辑斯特函数（logistic function）将范围限制到 $[0,1]$ ，类似的，对于评分矩阵中的 R_{ij} ，则会通过最大最小值标准化方法（Max-Min Normalization）进行标准化，使它们的范围也变成 $[0,1]$ 。

通过观察 (3.82) 式，我们可以发现 BCsPMF 模型损失函数的计算复杂度为 $O(L(D + P + Q))$ ， L 表示评分矩阵中非零值的个数， D 、 P 、 Q 分别表示潜在因子向量的维数、商品代价相关向量的维数以及用户代价相关向量的维数；类似的，观察 (3.86) 式、(3.87) 式、(3.88) 式以及 (3.89) 式，我们还可以看出损失函数的各个偏导的计算复杂度也为 $O(L(D + P + Q))$ ，因此 GcPMF 模型的复杂度为 $O(L(D + P + Q))$ 。这表明 GCsPMF 模型的复杂度跟评分矩阵中的非零值个数是线性关系，因此 BCsPMF 模型的扩展性比较强，比较适合并行计算，通过批量更新（batch update），我们可以进一步提升训练的效率。

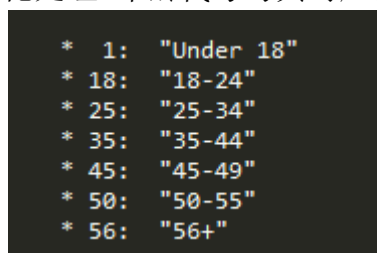
第四章 模型实现及验证

4.1 数据集说明

本文采用的数据集是 movieLens-1M (<http://www.grouplens.org/>), 该数据集由 GroupLens 项目组成员 Shyong Lam 和 Jon Herlocker 在其实验性电影推荐站点 MovieLens 收集并进行了适当的预处理。该数据集中包含三个文件, 分别存储用户基本信息、电影基本信息以及用户评分信息。这三个文件中分别包含了 6040 个用户的基本信息、3883 部电影的基本信息以及 1000209 条用户对电影的评分信息。

用户评分数据包含用户 ID、电影 ID、星级评分(范围为 1-5)以及评分发生的时间戳, 值得一提的是该份数据中每个用户至少有 20 条评分。

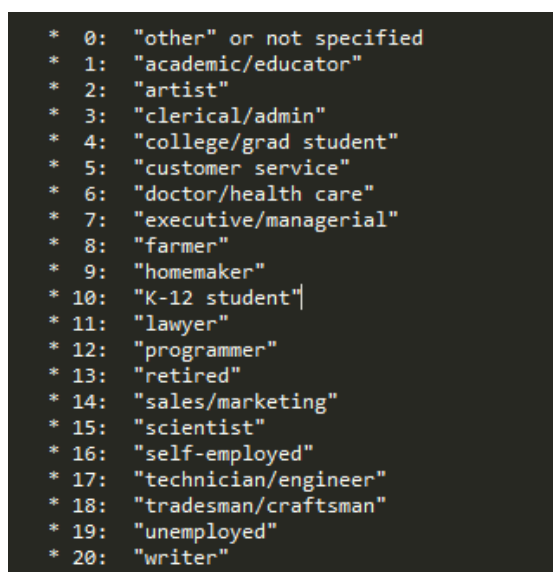
用户基本信息数据包含用户 ID、用户性别、用户年龄、用户的职业编号以及用户的邮编。用户的年龄作了离散化处理。年龄代号与其对应范围的对应关系如图 4.1 所示:



* 1:	"Under 18"
* 18:	"18-24"
* 25:	"25-34"
* 35:	"35-44"
* 45:	"45-49"
* 50:	"50-55"
* 56:	"56+"

图 4.1 年龄离散化处理

另外, 用户的职业编号具体对应关系如图 4.2 所示:



* 0:	"other" or not specified
* 1:	"academic/educator"
* 2:	"artist"
* 3:	"clerical/admin"
* 4:	"college/grad student"
* 5:	"customer service"
* 6:	"doctor/health care"
* 7:	"executive/managerial"
* 8:	"farmer"
* 9:	"homemaker"
* 10:	"K-12 student"
* 11:	"lawyer"
* 12:	"programmer"
* 13:	"retired"
* 14:	"sales/marketing"
* 15:	"scientist"
* 16:	"self-employed"
* 17:	"technician/engineer"
* 18:	"tradesman/craftsman"
* 19:	"unemployed"
* 20:	"writer"

图 4.2 用户职业编号

电影基本信息数据包含电影 ID、电影的标题（含年份）以及电影的类型，该份数据中总共涉及 18 种电影类型，详细类型说明如图 4.3 所示：

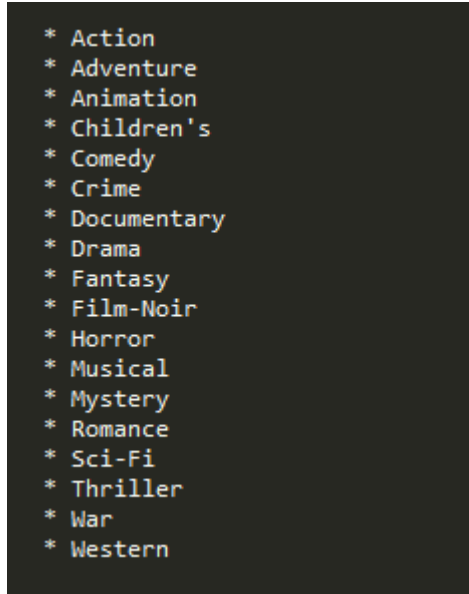


图 4.3 电影类型汇总

本论文使用电影的类型作为内容数据和用户对电影的评分数据训练了 cPMF、CsPMF、GcPMF、GCsPMF 模型；使用了电影的类型、用户的年龄、用户的职业以及用户对电影的评分数据训练了 BCsPMF 模型。

4.2 模型评估指标说明

本文采用均方根误差(RMSE)^[37]作为模型的评估指标，RMSE 全称为 Root Mean Square Error。

RMSE 表示在测试集上模型预测评分与真实评分的差值的平方和的平均值的根值。RMSE 值越小，模型的预测准确度越高，它的数学定义如下：

$$RMSE = \sqrt{\frac{1}{n} \sum_{r_{ij} \in R} (r_{ij} - p_{ij})^2} \quad (4.1)$$

其中， R 为验证集， n 为验证集中评分的总次数， r_{ij} 表示验证集中第 i 个用户对第 j 个物品的评分， p_{ij} 表示模型预测第 i 个用户对第 j 个物品的评分。

4.3 开发环境说明

本文的各个模型是作者基于开源的矩阵分解工具包 libmf，利用 vs2013 进行二次开发的，另外，使用 python 2.7 对 movielens 数据集进行了整理，转换成各个模型所需要

的输入数据格式。

4.3.1 系统配置

本文的模型在作者的笔记本上实现并运行，电脑详细配置情况如图 4.4 所示：



图 4.4 系统环境配置图

4.3.2 libmf 介绍

Libmf 是由台湾国立大学的机器学习项目组开发的，它是一个专门用于推荐系统的矩阵分解开源工具包，目前已经更新到 1.2 版本，其项目主页为 <http://www.csie.ntu.edu.tw/~cjlin/libmf/>。

与其他矩阵分解工具包相比，libmf 具有以下几个优点：

- 首先，libmf 适合在多核的机器上进行并行计算，因此适用于大规模数据。
- 其次，libmf 使用了 SSE 指令集来加速向量操作。
- 第三，经过开发团队实际验证，libmf 能在 20 分钟之内使得具有 17 亿评分的数据集收敛到合理的误差。
- 第四，libmf 提供了交叉验证，能够更方便的进行参数选择
- 第五，libmf 代码清晰简洁，整体架构良好，可扩展性很高

4.4 模型基本设置

4.4.1 模型参数设置

基本的 PMF 模型中主要有正则项系数、学习率、潜在因子个数、迭代次数、是否使用交叉验证（如果使用，采用几折交叉验证）这几个参数，默认值分别为 0.1、0.1、8、20、否。cPMF 和 GcPMF 模型的参数与 PMF 模型一致。

在 CsPMF 模型中，相对于 PMF 模型，增加了 w_1 和 w_2 两个参数，用于控制潜在因子和显式因子在模型中的比重，默认值分别为 1、1。GCsPMF 模型的参数与 CsPMF 模型一致。

在 BCsPMF 模型中，相对于 CsPMF 模型，增加了 w_3 ，与 w_1 、 w_2 一起用于控制

潜在因子和商品显式因子以及用户显式因子在模型中的比重，默认值均为 1。

4.4.2 模型初始化

在 PMF 模型中，需要对表征用户和商品的潜在因子向量进行初始化，该模型假设这些向量产生于多维高斯分布，因此在实现过程中，采用 c++ 的 `random` 库中的 `norm_distribution` 结合随机引擎 `default_random_engine` 产生潜在因子向量的各个值，这里将高斯分布的期望设置为 0，方差设置为 1。

在 cPMF、GcPMF、CsPMF、GCsPMF 模型中，采用与 PMF 模型相同的方法对用户和商品的潜在因子向量进行初始化。对于显式的商品因子向量，直接从输入数据中读取并存储，此外，与之对应的隐式的用户因子向量则依旧从高斯分布采样得到。

在 BCsPMF 模型中，采样与 PMF 模型相同的方法对用户和商品的潜在因子向量进行初始化。此外，与 cPMF 等模型类似，对于显式的商品因子向量和显式的用户因子向量，都是直接从输入数据中读取并存储，对应的隐式因子向量则依然从高斯分布采样得到。

4.5 模型效果对比验证

4.5.1 PMF vs cPMF 效果对比

本文基于 libfm 源码，实现了 PMF（参考文献[36]和论文[29]，是当前效果最佳的 PMF 模型）以及 cPMF 模型，cPMF 模型相对于 PMF 模型增加了对商品显式内容特征数据的存储以及模型学习对应的用户的潜在特征向量的过程。cPMF 模型用商品显式内容特征向量和用户潜在特征向量的内积表示商品和用户在这些显式维度上的关联程度，类似的，用商品其余潜在维度的特征向量和用户其余潜在维度的特征向量的内积表示用户和商品在其余显式维度的关联程度，最后将上述两个内积值相加，作为用户和商品的关联分值，分值越高，表明用户越有可能给商品更高的评分。

本论文采用均方根误差（RMSE）来衡量模型计算得到的关联分值与用户实际给商品的评分之间的差距，本次实验依据单一变量原则，分别比较了 PMF 模型和 cPMF 模型在不同的正则项系数、不同的学习率参数以及不同的隐式因子数目下产生的效果，试验结果表明 cPMF 模型明显优于 PMF 模型。

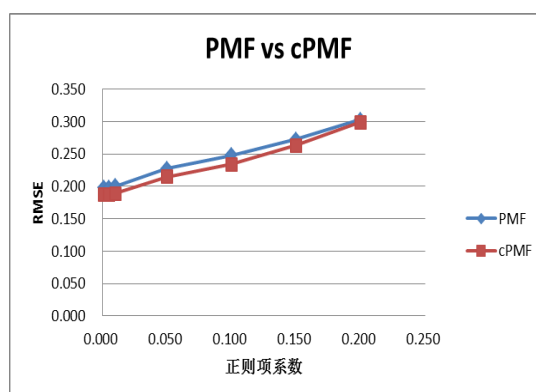


图 4.5 PMF vs cPMF 变化正则项系数

从图 4.5 中可以看出，对于不同的正则项系数，cPMF 均表现地优于 PMF 模型，相对具有更小的均方根误差。另一方面，随着正则项系数的增加，无论是 cPMF 模型，还是 PMF 模型，均方根误差均随着增加，这说明 cPMF 模型和 PMF 模型对于正则项系数均比较敏感。详细的测试数据请参见表 4.1。

表 4.1 PMF vs cPMF 不同正则项系数下的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	PMF 的 RMSE	cPMF 的 RMSE
参数命令	-l	-r	-t	-k		
默认参数值	0.1	0.1	20	8	0.2484	0.2354
	0.1	0.1	40	8	0.2481	0.2340
增大正则项系数	0.15	0.1	40	8	0.2722	0.2629
	0.2	0.1	40	8	0.3029	0.2991
减小正则项系数	0.05	0.1	40	8	0.2271	0.2146
	0.01	0.1	40	8	0.2001	0.1887
	0.005	0.1	40	8	0.1984	0.1869
	0.001	0.1	160	8	0.1978	0.1868

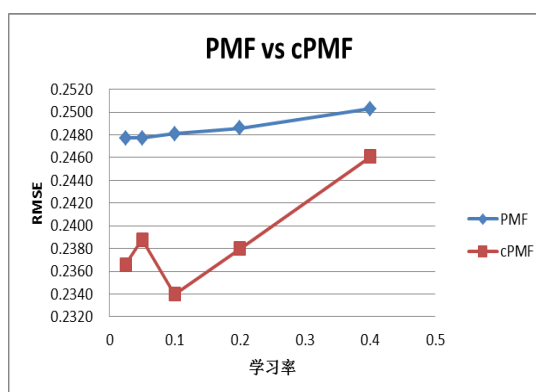


图 4.6 PMF vs cPMF 变化学习率参数

从图 4.6 中可以看出，在不同的学习率参数下，cPMF 模型均优于 PMF 模型。另一

方面，cPMF 模型对于学习率非常敏感，其最优参数在 0.1 附近，增大或者减小学习率，其均方根误差都会增大，而 PMF 模型对于学习率参数，则相对稳定。详细的测试数据请参见表 4.2。

表 4.2 PMF vs cPMF 不同学习率参数下的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	PMF 的 rmse	cPMF 的 rmse
参数命令	-l	-r	-t	-k		
默认参数值	0.1	0.1	20	8	0.2484	0.2354
	0.1	0.1	40	8	0.2481	0.2340
增大学习率	0.1	0.2	40	8	0.2486	0.2380
	0.1	0.4	40	8	0.2503	0.2461
减小学习率， 增加迭代次数	0.1	0.05	160	8	0.2477	0.2388
	0.1	0.025	160	8	0.2477	0.2366

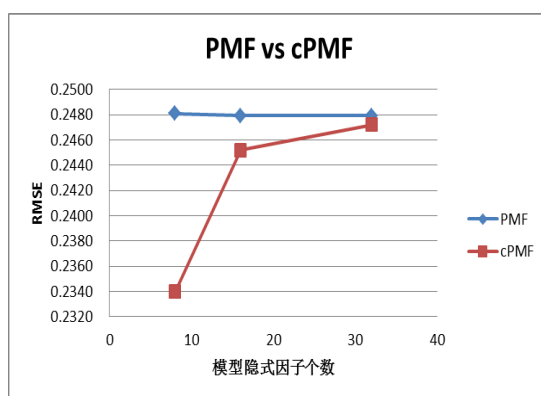


图 4.7 PMF vs cPMF 变化隐式因子个数

从图 4.7 中可以看出，随着模型隐式因子个数的增加，PMF 模型的均方根误差会小幅度减小，但是 cPMF 模型的均方根误差却大幅度的增加，但各种参数下，cPMF 模型的误差均小于 PMF 模型的均方根误差。一方面，cPMF 模型效果优于 PMF 模型，另一方面，cPMF 模型对于模型隐式因子个数比较敏感。详细的测试数据请参见表 4.3。

表 4.3 PMF vs cPMF 不同隐式因子个数下的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	PMF 的 rmse	CsPMF 的 rmse
参数命令	-l	-r	-t	-k		
默认参数值	0.1	0.1	20	8	0.2484	0.2354
	0.1	0.1	40	8	0.2481	0.2340
增加隐式因子个数	0.1	0.1	160	16	0.2479	0.2452
	0.1	0.1	160	32	0.2479	0.2472

4.5.2 cPMF vs CsPMF 效果对比

CsPMF 模型比 cPMF 模型要多两个系统参数—— w_1 和 w_2 ，它们分别表示显式因子和潜在因子对用户和商品关联分的影响权重。本论文采用均方根误差（RMSE）来衡量模型计算得到的关联分值与用户实际给商品的评分之间的差距，本次实验依据单一变量原则，分别比较了 cPMF 模型和 CsPMF 模型在不同权重下产生的效果，试验结果表明通过调整显式因子和潜在因子在系统中的比重，CsPMF 模型可以取得明显优于 cPMF 模型的效果。

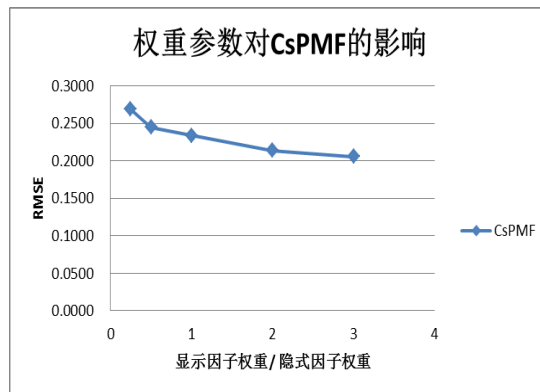


图 4.8 权重参数对 CsPMF 模型的影响

从图 4.8 中可以看出，随着隐式因子权重在模型中的占比增加，CsPMF 模型的均方根误差逐渐减小，这表明引入两个权重参数用于调节显式因子与隐式因子在模型中的比例确实有助于提升模型的效果。详细的测试数据请参见表 4.4。

表 4.4 CsPMF 不同的显式和潜在因子权重的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	显式因子权重	隐式因子权重	显式/隐式	CsPMF 的 RMSE
参数命令	-l	-r	-t	-k	-w1	-w2		
默认参数值	0.1	0.1	20	8	1	1	1	0.2354
降低隐式因子权重	0.1	0.1	40	8	3	1	3	0.2055
	0.1	0.1	40	8	2	1	2	0.2140
	0.1	0.1	40	8	1	1	1	0.2340
增大隐式因子权重	0.1	0.1	40	8	1	2	0.5	0.2448
	0.1	0.1	40	8	1	4	0.25	0.2687

4.5.3 PMF vs GcPMF 效果对比

本文基于 libfm 源码，实现了 GcPMF 模型，GcPMF 模型相对于 PMF 模型增加了对商品显式内容特征数据的存储以及模型学习对应的用户的潜在特征向量的过程。GcPMF 模型用商品显式内容特征向量和用户潜在特征向量的余弦相似度表示商品和用户在这些显式维度上的关联程度，类似的，用商品其余潜在维度的特征向量和用户其余潜在维度的特征向量的余弦相似度表示用户和商品在其余显式维度的关联程度，最后将上述两个内积值相加，作为用户和商品的关联分值，分值越高，表明用户越有可能给商品更高的评分。本论文采用均方根误差 (RMSE) 来衡量模型计算得到的关联分值与用户实际给商品的评分之间的差距，本次实验依据单一变量原则，分别比较了 PMF 模型和 GcPMF 模型在不同的正则项系数、不同的学习率参数以及不同的隐式因子数目下产生的效果，试验结果表明 GcPMF 模型明显比 PMF 模型均方根误差大。直接将显式因子向量的余弦相似度和潜在因子向量的余弦相似度直接相加得到关联分值这种方式不够合理，我们需要引入权重参数，来控制潜在因子和显式因子的权重，这也就是 GCsPMF 模型，我们将在 4.5.4 节介绍 GCsPMF 模型的效果。

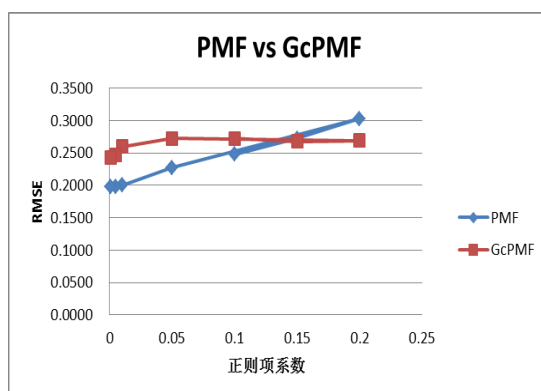


图 4.9 PMF vs GcPMF 变化正则项系数

从图 4.9 中可以看出，GcPMF 模型只有在正则项系数较大的情况下才比 PMF 模型稍微好一些，整体看来 PMF 模型效果要好一些。另一方面，PMF 模型对于正则项系数要相对敏感，而 GcPMF 模型则要鲁棒一些。详细的对比试验数据请参看表 4.5。

表 4.5 PMF vs GcPMF 不同正则项系数下的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	PMF 的 RMSE	GcPMF 的 RMSE
参数命令	-l	-r	-t	-k		
默认参数值	0.1	0.1	20	8	0.2484	0.2741
	0.1	0.1	40	8	0.2481	0.2724
增大正则项系数	0.15	0.1	40	8	0.2722	0.2681
	0.2	0.1	40	8	0.3029	0.2690
减小正则项	0.05	0.1	40	8	0.2271	0.2723

系数	0.01	0.1	40	8	0.2001	0.2598
	0.005	0.1	40	8	0.1984	0.2467
	0.001	0.1	160	8	0.1978	0.2428

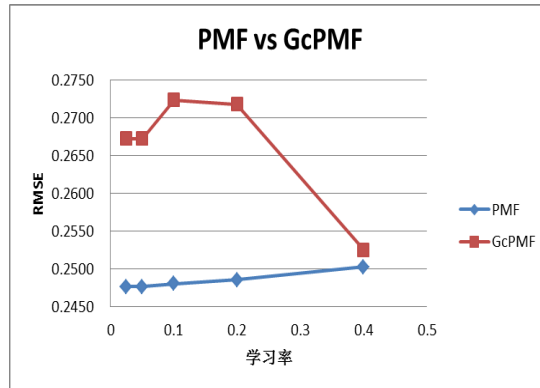


图 4.10 PMF vs GcPMF 变化学习率参数

从图 4.10 中可以看出，在不同的学习率下，GcPMF 模型均比 PMF 模型效果要差，而且 PMF 模型对于学习率参数相对鲁棒，而 GcPMF 模型对于学习率参数非常敏感。详细的对比试验数据请查看表 4.6。

表 4.6 PMF vs GcPMF 不同学习率参数下的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	PMF 的 rmse	cPMF 的 rmse
参数命令	-l	-r	-t	-k		
默认参数值	0.1	0.1	20	8	0.2484	0.2741
	0.1	0.1	40	8	0.2481	0.2724
增大学习率	0.1	0.2	40	8	0.2486	0.2718
	0.1	0.4	40	8	0.2503	0.2526
减小学习率， 增加迭代次数	0.1	0.05	160	8	0.2477	0.2673
	0.1	0.025	160	8	0.2477	0.2673

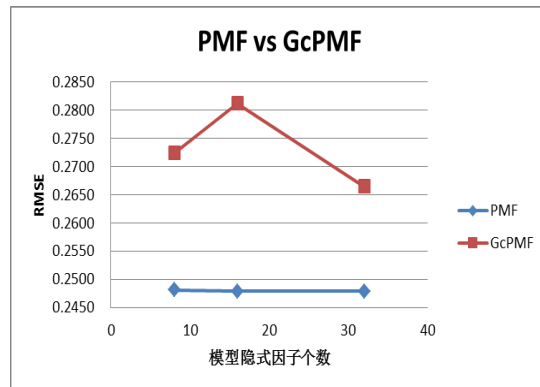


图 4.11 PMF vs GcPMF 变化隐式因子个数

从图 4.11 中可以看出，在不同的模型隐式因子个数参数下，GcPMF 模型均比 PMF 模型效果要差，而且 PMF 模型该参数相对鲁棒，而 GcPMF 模型该参数比较敏感。详细的对比实验数据请参看表 4.7。

表 4.7 PMF vs GcPMF 不同隐式因子个数下的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	PMF 的 rmse	CsPMF 的 rmse
参数命令	-l	-r	-t	-k		
默认参数值	0.1	0.1	20	8	0.2484	0.2741
	0.1	0.1	40	8	0.2481	0.2724
增加隐式因子个数	0.1	0.1	160	16	0.2479	0.2812
	0.1	0.1	160	32	0.2479	0.2665

4.5.4 GcPMF vs GCsPMF 效果对比

GCsPMF 模型比 GcPMF 模型要多两个系统参数—— w_1 和 w_2 ，它们分别表示显式因子和潜在因子对用户和商品关联分的影响权重。本论文采用均方根误差 (RMSE) 来衡量模型计算得到的关联分值与用户实际给商品的评分之间的差距，本次实验依据单一变量原则，分别比较了 GCsPMF 模型和 GcPMF 模型在不同权重下产生的效果，试验结果表明通过调整显式因子和潜在因子在系统中的比重，GCsPMF 模型可以取得明显优于 GcPMF 模型的效果，并且在适合的权重参数下，GCsPMF 模型的均方根误差要明显小于 PMF 模型。

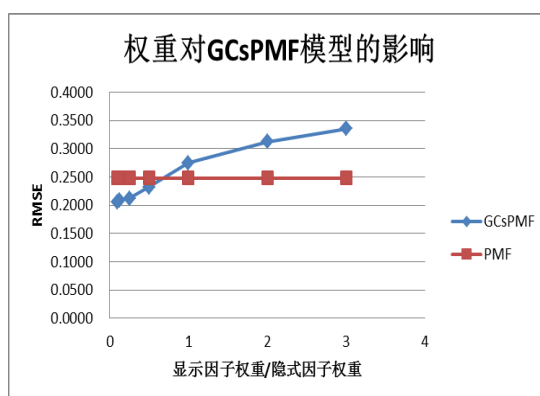


图 4.12 权重参数对 GCsPMF 模型的影响

从图 4.12 中可以看出，随着隐式因子权重在模型中的占比增加，GCsPMF 模型的均方根误差逐渐减小，当显式因子与隐式因子的权重比值小于 0.5 时，GCsPMF 模型的均方根误差要小于 PMF 模型，这表明引入两个权重参数用于调节显式因子与隐式因子在模型中的比例确实有助于提升模型的效果。详细的对比实验数据请参看表 4.8。

表 4.8 GCsPMF 不同的显式和潜在因子权重的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	显式因子权重	隐式因子权重	显式/隐式	GCsPMF 的 RMSE	PMF 的 RMSE
参数命令	-l	-r	-t	-k	-w1	-w2			
默认参数值	0.1	0.1	20	8	1	1	1	0.2752	0.2484
降低隐式因子权重	0.1	0.1	40	8	3	1	3	0.3359	0.2481
	0.1	0.1	40	8	2	1	2	0.3129	0.2481
	0.1	0.1	40	8	1	1	1	0.2752	0.2481
增大隐式因子权重	0.1	0.1	40	8	1	2	0.5	0.2323	0.2481
	0.1	0.1	40	8	1	4	0.25	0.2125	0.2481
	0.1	0.1	40	8	1	8	0.125	0.2089	0.2481
	0.1	0.1	40	8	1	10	0.1	0.2059	0.2481

4.5.5 PMF vs BCsPMF 效果对比

本文基于 libfm 源码，实现了 PMF 和 BCsPMF 模型，BCsPMF 模型相对于 PMF 模型增加了对用户和商品内容数据的处理。一方面，BCsPMF 增加了对商品显式内容特征数据的存储以及模型学习对应的用户的潜在特征向量的过程，另一方面，BCsPMF 增加了对用户显式内容特征数据的存储以及模型学习对应的用户的潜在特征向量的过程。BCsPMF 模型用商品显式内容特征向量和用户潜在特征向量的内积表示商品和用户在这些显式维度上的关联程度。同样的，BCsPMF 模型用用户显式内容特征向量和商品的潜在特征向量的内积表示商品和用户在这些显式维度上的关联程度。类似的，用商品其余潜在维度的特征向量和用户其余潜在维度的特征向量的内积表示用户和商品在其余显式维度的关联程度，最后将上述三个内积值相加，作为用户和商品的关联分值，分值越高，表明用户越有可能给商品更高的评分。

本论文采用均方根误差 (RMSE) 来衡量模型计算得到的关联分值与用户实际给商品的评分之间的差距，本次实验依据单一变量原则，分别比较了 PMF 模型和 BCsPMF 模型在不同的正则项系数、不同的学习率参数以及不同的隐式因子数目下产生的效果，试验结果表明 BCsPMF 模型明显优于 PMF 模型。

从图 4.13 中可以看出，BCsPMF 模型的 RMSE 基本稳定在 0.2 以下，但是 PMF 模型的 RMSE 最差为 0.3，最优仅为 0.2 左右，且随着正则项系数的增加，不断的增加。BCsPMF 模型的误差要明显小于 PMF，此外，PMF 模型对于正则项系数比较敏感，而 BCsPMF 模型则表现的相当鲁棒。详细的对比实验数据请参看表 4.9。

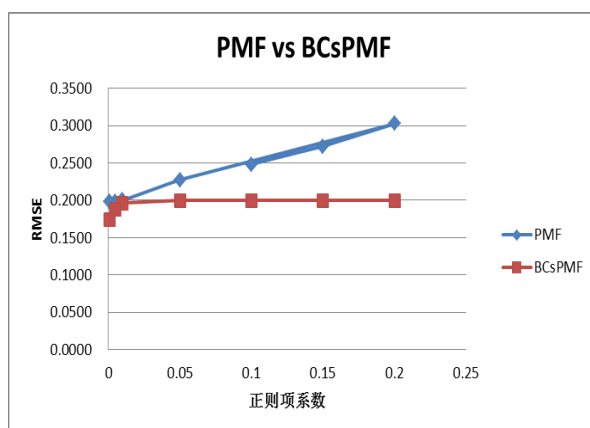


图 4.13 PMF vs BcsPMF 变化正则项系数

表 4.9 PMF vs BCsPMF 不同正则项系数下的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	PMF 的 RMSE	BCsPMF 的 RMSE
参数命令	-l	-r	-t	-k		
默认参数值	0.1	0.1	20	8	0.2484	0.2023
	0.1	0.1	40	8	0.2481	0.1997
增大正则项系数	0.15	0.1	40	8	0.2722	0.1997
	0.2	0.1	40	8	0.3029	0.1998
减小正则项系数	0.05	0.1	40	8	0.2271	0.1997
	0.01	0.1	40	8	0.2001	0.1960
	0.005	0.1	40	8	0.1984	0.1882
	0.001	0.1	160	8	0.1978	0.1745

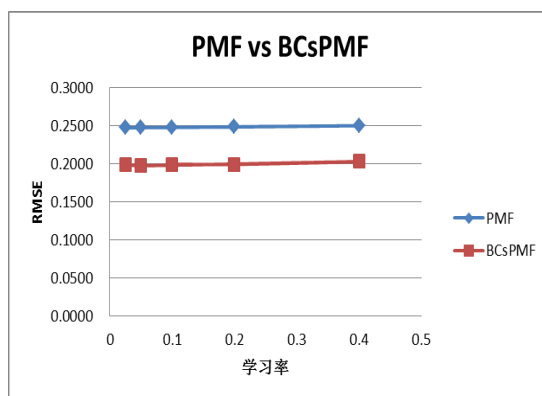


图 4.14 PMF vs BcsPMF 变化学习率参数

从图 4.14 中可以看出，在不同的学习率下，BCsPMF 模型的误差均小于 PMF 模型的误差，另一方面，从图中我们可以看出在特定的正则项系数下，不同的学习率对误差结果影响不大，或者说两个模型对学习率参数都不太敏感。详细的对比实验数据请参看表 4.10。

表 4.10 PMF vs BCsPMF 不同学习率下的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	PMF 的 rmse	BCsPMF 的 rmse
参数命令	-l	-r	-t	-k		
默认参数值	0.1	0.1	20	8	0.2484	0.2023
	0.1	0.1	40	8	0.2481	0.1997
增大学习率	0.1	0.2	40	8	0.2486	0.1995
	0.1	0.4	40	8	0.2503	0.2032
减小学习率，增加迭代次数	0.1	0.05	160	8	0.2477	0.1978
	0.1	0.025	160	8	0.2477	0.1995

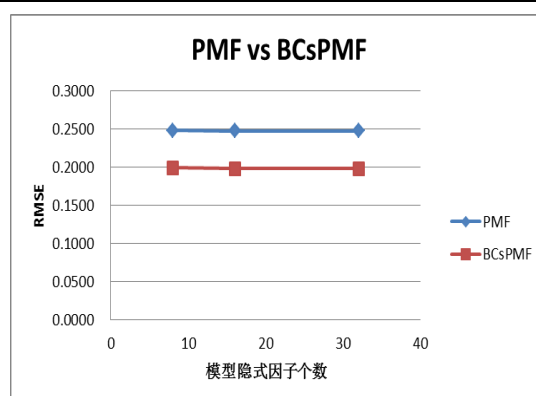


图 4.15 PMF vs BcsPMF 变化模型隐式因子个数

从图 4.15 中可以看出，在不同的隐式因子个数情况下，BCsPMF 模型均优于 PMF 模型，具有更小的均方根误差。另一方面，增大隐式因子的个数，均方根误差会不断减少，但是幅度比较小。详细的对比实验数据请参看表 4.11。

表 4.11 PMF vs BCsPMF 不同隐式因子个数下的对比试验数据

参数说明	正则项系数	学习率	迭代次数	隐式因子个数	PMF 的 rmse	BCsPMF 的 rmse
参数命令	-l	-r	-t	-k		
默认参数值	0.1	0.1	20	8	0.2484	0.2023
	0.1	0.1	40	8	0.2481	0.1997
增加隐式因子个数	0.1	0.1	160	16	0.2479	0.1985
	0.1	0.1	160	32	0.2479	0.1985

4.5.6 PMF vs Other 效果对比

为了更直观地比较 PMF 模型与本文提出的几种模型，下面给出 PMF 模型与 CsPMF 模型 ($w_1=3$ 、 $w_2=1$)、GCsPMF 模型 ($w_1=1$ 、 $w_2=8$) 以及 BCsPMF 模型的整体对比试验结果图。

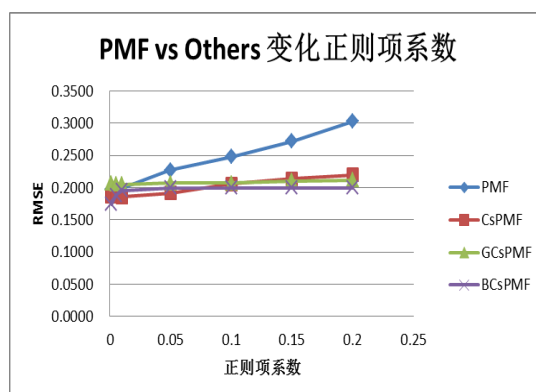


图 4.16 PMF vs Others 变化正则项系数

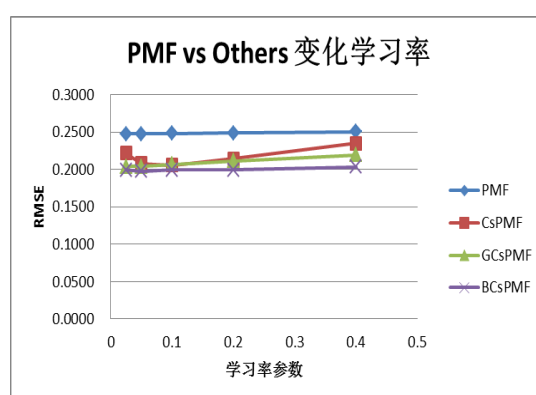


图 4.17 PMF vs Others 变化学习率参数

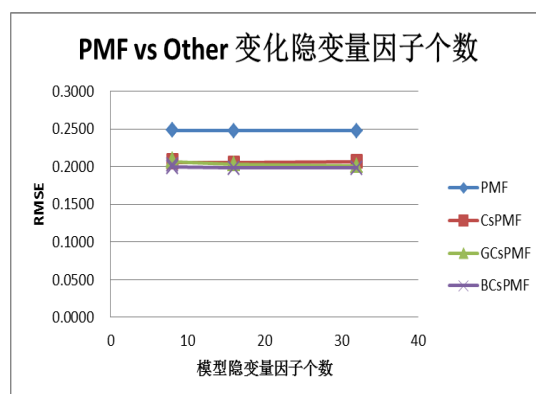


图 4.18 PMF vs Others 变化模型隐式因子个数

第五章 结论及展望

为了更有效地利用系统收集到的关于商品和用户的内容信息，本文提出了一系列概率矩阵分解的扩展模型。这些模型能够将商品和用户的内容信息转换为显式的代价因子，再结合潜在的代价因子，共同决定模型预测结果。在 `movielens-1m` 数据集上对这些模型进行了验证及对比，验证了将内容数据融入矩阵分解模型有助于提升模型的准确率。

在第一章中介绍了将现有推荐模型在处理内容丰富的商品推荐问题上存在的不足，并结合现有大数据相关技术的发展，提出了将商品和用户内容数据融入传统推荐模型的想法。

在第二章中对现有的推荐算法进行了系统的介绍。先是概况性的介绍了基于内容、基于协同、基于知识以及组合式推荐算法的研究现状以及优缺点，再重点介绍跟本文提出的模型密切相关的矩阵分解类算法，并概述了前人对矩阵分解类算法在模型及系统实现上的改进及创新。

在第三章对本文提出的一系列内容感知的算法模型进行了详细的介绍，并给出了相应的图模型以及求解模型参数过程中完整的公式推导过程。各个模型的介绍如下：

第一种模型 **cPMF** 旨在更好地利用系统收集到的商品内容数据。该模型在 **PMF** 模型的基础上，进一步将表征商品的特征向量分为隐式和显式两个部分，其中显式的部分是根据商品内容数据计算得到的。商品显式特征向量作为模型的输入，再结合用户评分数据学习表征用户偏好的代价特征向量以及其余隐式的表征用户和商品的特征向量。与 **PMF** 类似，**cPMF** 模型将用户和商品的特征向量的内积作为用户对商品的偏好分，系统将给用户推荐偏好分 **TOP-K** 的商品。

第二种模型 **GcPMF** 旨在更精准地表达商品数据内容的含义，为了更合理的利用实际数据中多种不同类型的用户偏好特征，消除不同特征之间的单位差异，该模型使用余弦相似度来表达商品显式特征向量与对应用户隐式特征向量的关联，这区别于 **cPMF** 模型中的内积形式。

第三种模型 **CsPMF** 旨在增强 **cPMF** 的灵活性。该模型在 **cPMF** 的基础上增加了两个系统超参 (w_1, w_2)，用于调节显式特征和隐式特征在模型中的比重。

第四种模型 **GCsPMF** 旨在增强 **GcPMF** 的灵活性。该模型在 **GcPMF** 的基础上增加了系统超参 (w_1, w_2)，用于调节显式特征和隐式特征在模型中的比重。

第五种模型 **BCsPMF** 旨在更好地利用系统收集到的商品内容数据以及用户内容数据。与 **cPMF** 类似，**BCsPMF** 将用户和商品的特征都划分为隐式和显式两个部分，该模型使用向量内积值来表达商品显式特征向量与对应用户隐式特征向量的关联。

在第四章中介绍了本文提出的模型在 **movieLens-1m** 数据集上的效果，并对模型中各个参数对模型的影响进行了试验。首先，通过对比 **PMF** 及 **CsPMF** 等模型，我们可以明显的发现将内容数据融入模型的 **CsPMF** 等模型要优于 **PMF** 模型。其次，通过调整模型参数并对比，发现传统的 **PMF** 模型对正则项系数这个参数非常敏感，而本文提出的 **BCsPMF** 模型相对来说更具鲁棒性，此外不同的学习率会对模型的收敛速度以及最终效果有一定影响，而不同的潜在因子数量对模型的影响效果较小。

总的来说，本文工作解决了传统的推荐算法中存在的某些问题，将商品和用户的内容数据融入矩阵分解模型，更好的利用了现代推荐系统能够收集到的数据，丰富了矩阵分解模型的表达能力。

本论文采用了内积形式和余弦相似度的方法将用户和商品的特征向量进行关联，将来希望采用其他类型的关联方式（对连续型特征采用多维高斯分布、离散型特征多维伯努利分布），以期获得更好的效果。此外，将来我们还希望能在更多的数据集上验证本论文提出的模型，并且与其他非矩阵分解类的推荐算法进行对比实验。

参考文献

- [1] 王国霞, 刘贺平. 个性化推荐系统综述[J]. Computer Engineering and Applications, 2012, 48(7).
- [2] Kim J, Lee E. Intelligent information recommend system on the Internet[C]//Parallel Processing, 1999. Proceedings. 1999 International Workshops on. IEEE, 1999: 376-380.
- [3] 罗军舟, 吴文甲, 杨明. 移动互联网: 终端, 网络与服务[J]. 计算机学报, 2011, 34(11): 2029-2051.
- [4] 吴芝新. 简析 O2O 电子商务模式[J]. 重庆科技学院学报: 社会科学版, 2012 (13): 73-74.
- [5] 许海玲, 吴潇, 李晓东, 等. 互联网推荐系统比较研究 [J]. 软件学报, 2009, 20(2): 350-362
- [6] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms[C]//Proceedings of the 10th international conference on World Wide Web. ACM, 2001: 285-295.
- [7] Getoor L, Sahami M. Using probabilistic relational models for collaborative filtering[C]//Workshop on Web Usage Analysis and User Profiling (WEBKDD'99). 1999.
- [8] Pavlov D Y, Pennock D M. A maximum entropy approach to collaborative filtering in dynamic, sparse, high-dimensional domains[C]//Advances in neural information processing systems. 2002: 1441-1448.
- [9] Chien Y H, George E I. A bayesian model for collaborative filtering[C]//Proceedings of the 7th International Workshop on Artificial Intelligence and Statistics. San Francisco: Morgan Kaufman Publishers, [http://uncertainty99. microsoft. com/proceedings. htm], 1999.
- [10] Ungar LH, Foster DP. Clustering methods for collaborative filtering. In: Proc, of the Workshop on Recommendation Systems. Menlo Park: AAAI Press. 1998. 112—125
- [11] Marlin B M. Modeling user rating profiles for collaborative filtering[C]//Advances in neural information processing systems. 2003. 627-634
- [12] Hofmann T. Collaborative filtering via gaussian probabilistic latent semantic analysis[C]//Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM, 2003: 259-266.
- [13] Salton G, Wong A, Yang C S. A vector space model for automatic indexing[J]. Communications of the ACM, 1975, 18(11): 613-620.
- [14] Ramos J. Using tf-idf to determine word relevance in document queries[C]//Proceedings of the first instructional conference on machine learning. 2003.
- [15] Somlo G, Howe A. Adaptive Lightweight Text Filtering [C] / / Process of the 4th International Symposium on Intelligent Data Analysis, 2001: 319- 329
- [16] Zhang Yi, Callan J, Minka T . Novelty and Redundancy Detection in Adaptive Filtering [C] / / Process of 25th Annual International ACM SIGIR Conference, 2002: 81- 88
- [17] Roberrson S, Walker S. Threshold Setting and Performance Optimization in Adaptive Filtering[J] . Information Retrieval, 2002, 5: 239- 256

- [18] Chang Ye-in, Shen Jun- hong , Chen T L. A Data Mining- based Method for the Incremental Update for Supporting Personalized Information Filtering[J] . Journal of Information Science and Engineering , 2008, 24(1) : 129- 142.
- [19] Trewin S. Knowledge-based recommender systems[J]. Encyclopedia of Library and Information Science: Volume 69-Supplement 32, 2000: 180.
- [20] Cremonesi P, Turrin R, Airoidi F. Hybrid algorithms for recommending new items[C]//Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems. ACM, 2011: 33-40.
- [21] Pazzani M. A framework for collaborative, content-based, and demographic filtering. Artificial Intelligence Review, 1999,13(5-6):393-408
- [22] Claypool M,Gokhale A,Miranda T,Murnikov P,Netes D,Sartin M. Combining content—based and collaborative filters in an online newspaper. In: Proc. of the ACM SIGIR'99 Workshop Recommender Systems: Algorithms and Evaluation. New York: ACM Press,1999
- [23] Good N, Schafer JB, Konstan JA, Borchers A, Sarwar B, Herlocker JL, Riedl J. Combining collaborative filtering with personal agents for better recommendations. In:Proc. of the 16th National Conf. on Artificial Intelligence. Menlo Park: AAAI Press, 1999.439-446
- [24] Melville P,Mooney RJ,Nagarajan R . Content-Boosted collaborative filtering for improved recommendations. In: Proc. of the 18th National Conf. on Artificial Intelligence. Menlo Park: American Association for Artificial Intelligence,2002. 187—192
- [25] Soborof I,Nicholas C. Combining content and collaboration in text filtering. In: Proc. of the Int'l Joint Conf. on Artificial Intelligence Workshop : Machine Learning for Information Filtering. Stockholm,1999. 86—91.
- [26] Basu C, Hirsh H, Cohen W. Recommendation as classification: Using social and content-based information in recommendation . In : Proc. of the AAAI'98 . Menlo Park : AAAI Press. 1998. 714—720.
- [27] Ansari A, Essegiaier S, Kohli R. Internet recommendations systems. Journal of Marketing Research, 2000, 37(3): 363—375.
- [28] R. Bell, Y. Koren, and C. Volinsky. The BellKor solution to the Netflix Prize.
- [29] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In Neural Information Processing Systems 21 (NIPS 2008), 2008.
- [30] Ge Y, Liu Q, Xiong H, et al. Cost-aware travel tour recommendation[C]//Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2011: 983-991.
- [31] Golub G, Kahan K. Calculating the singular values and pseudo-inverse of a matrix. Journal of the Society for Industrial and Applied Mathematics, 1965,2(2):205 224. [doi: 10.1137/0702016]
- [32] Lee DD, Seung HS. Algorithms for non-negative matrix factorization. In: Advances in Neural Information Processing Systems. 2000. 556 562.
- [33] 涂丹丹, 舒承椿, 余海燕. 基于联合概率矩阵分解的上下文广告推荐算法*[J]. 软件学报, 2013, (3):454-464.

-
- [34] Gemulla R, Haas P J, Sismanis Y, et al. Large-scale Matrix Factorization with Distributed Stochastic Gradient Descent[J]. In KDD, 2011.
 - [35] Makari F, Teflioudi C, Gemulla R, et al. Shared-memory and shared-nothing stochastic gradient descent algorithms for matrix completion[J]. Knowledge and Information Systems, 2015, 42(3):493-523.
 - [36] Y. Zhuang, W. Chin, Y. Juan, and C. Lin. A fast parallel SGD for matrix factorization in shared memory systems. In Proceedings of the ACM International Conference on Recommender Systems (RecSys), Hong Kong, China, 2013.
 - [37] Linyuan Lü, Matúš Medo, Chi Ho Yeung, et al. Recommender Systems[J]. Preprint Submitted to Physics Reports, 2012, 519(1): 1-49.
 - [38] Song, S, Moustafa, H, Afifi, H. Advanced IPTV Services Personalization Through Context-Aware Content Recommendation[J]. Multimedia, IEEE Transactions on, 2012, (6):1528-1537.
 - [39] Zhang W, Liu C, Wang Z, et al. Web video thumbnail recommendation with content-aware analysis and query-sensitive matching[J]. Multimedia Tools & Applications, 2014, 73(1):547-571.

致谢

能够顺利完成我的毕业设计，在此，我要特别感谢指导老师李杰教授。在我们做毕设期间，李老师多次组织实验室成员召开例会，每次开会不仅起到了督促我们进度的作用，更为关键的是，每次开会李老师会就每个成员毕设中遇到的难点问题给予指导并及时地指出我的毕设工作中存在的不足之处，再次衷心地感谢我们负责敬业的李老师。

此外，我要感谢那些和我一起做毕设的实验室成员们。整个毕设期间，实验室成员经常在一起开会讨论问题，这为解决毕设中遇到的各种难题提供了宝贵的灵感，同时活跃的实验室气氛让毕设进行得更加顺畅，再次衷心地感谢这些可爱的同学们。

北京大学学位论文原创性声明和使用授权说明

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或成果。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。本声明的法律结果由本人承担。

论文作者签名： 日期： 年 月 日

学位论文使用授权说明

（必须装订在提交学校图书馆的印刷本）

本人完全了解北京大学关于收集、保存、使用学位论文的规定，即：

- 按照学校要求提交学位论文的印刷本和电子版本；
- 学校有权保存学位论文的印刷本和电子版，并提供目录检索与阅览服务，在校园网上提供服务；
- 学校可以采用影印、缩印、数字化或其它复制手段保存论文；
- 因某种特殊原因需要延迟发布学位论文电子版，授权学校 ☐ 一年 / ☐ 两年 / ☐ 三年以后，在校园网上全文发布。

（保密论文在解密后遵守此规定）

论文作者签名： 导师签名：

日期： 年 月 日