



# 南京大學

## 研究生毕业论文 (申请硕士学位)

论 文 题 目 推荐系统中多源信息融合和隐式反馈挖掘的研究

作 者 姓 名 胡光能

学 科、专 业 名 称 计算机技术

研 究 方 向 自然语言处理

指 导 教 师 戴新宇 副教授

2016 年 5 月 20 日

学        号 : MF1333013  
论文答辩日期 : 2016 年 5 月 30 日  
指 导 教 师 :              (签字)

# **Combining Multiple Information Sources and Incorporating Implicit Feedback for Recommender Systems**

by

**Guangneng Hu**

Directed by

Associate Professor Xinyu Dai

Department of Computer Science and Technology  
Nanjing University

May 20, 2016

*Submitted in partial fulfilment of the requirements  
for the degree of Master of Science in Computer Technology*

# 南京大学研究生毕业论文中文摘要首页用纸

毕业论文题目： 推荐系统中多源信息融合和隐式反馈挖掘的研究  
计算机技术 专业 2013 级硕士生姓名： 胡光能  
指导教师（姓名、职称）： 戴新宇 副教授

## **摘要**

通信技术和网络媒体的发展，产生了大量的信息：新闻，音乐，视频，应用软件等；为克服由此带来的信息过载问题，研究者们就对如何构建支撑用户做出个性化决策的信息系统产生了兴趣，并提出了多种信息过滤算法，这些系统（算法）统称为推荐系统（算法）。由于推荐系统能够为广大用户提供他们感兴趣和有用的物品，给企业带来营收、为平台增强功能同时也提升了用户满意度，使得推荐系统在电子商务、流媒体平台和社交网络与社交媒体等领域有广泛的应用。

由于传统的协同过滤推荐方法只利用了用户对物品的浅层打分信息，使得推荐性能深受冷启动和数据稀疏性的影响。因而目前一个重要方向是增加额外的数据源：一方面，是增加文本信息，如用户对产品的评论数据；另一方面，是增加社交关系信息，如用户在社会网络中形成的信任关系链。然而，如何将文本信息和社交关系信息整合到基于矩阵分解的协同过滤方法中，还缺乏高效的融合方法。另一个重要方向是深度利用有限的信息，如挖掘打分信息中的隐式反馈。但是如何把横向增加额外数据源与纵向深挖有限信息两个重要方向结合起来以进一步提高推荐性能，相关工作据作者所知还没有。

本文对推荐系统中的多源信息融合和隐式反馈挖掘进行了研究，在此基础上提出了两个推荐模型，主要工作包括：

**1. 提出一个基于多源信息融合的高效综合方法，能够同时建模打分信息、文本信息和社交网络信息。**以往利用额外数据源的方法通常只加入了一种额外信息，比如只加入了文本信息，或者只加入了社交关系信息，缺乏一个高效的综合方法。针对这一问题，本文通过基于矩阵分解的协同过滤基本模型，建立了打通文本信息和社交关系信息的融合方法。该方法不仅具有优化目标上的统一性，在两个真实数据集的评估上都有实践上的性能提升。

**2. 提出一个基于隐式反馈挖掘的扩展融合模型，能够在深挖纵向打分信息的同时融合横向的额外数据源。**以往利用隐式反馈信息的方法都只考虑了如何纵向深挖打分信息，而忽略了融合横向的额外数据源。针对这一问题，本文通

过多源信息融合这一综合方法，在此基础上建立了能深挖隐式反馈信息的扩展模型。通过此方法，本文首次实现了横向多源数据融合和纵向隐式反馈挖掘两个克服冷启动和数据稀疏性问题的重要方向的汇合。

所提出的融合模型和扩展融合模型在两个较大规模的数据集上做了实验验证，并对两个模型的所含组件和超参数敏感性进行了详细分析。

**关键词：** 推荐系统，协同过滤，矩阵分解，打分预测，主题建模，社会推荐系统，异构信息源，隐式反馈

# 南京大学研究生毕业论文英文摘要首页用纸

THESIS: Combining Multiple Information Sources and Incorporating Implicit Feedback for Recommender Systems

SPECIALIZATION: Computer Technology

POSTGRADUATE: Guangneng Hu

MENTOR: Associate Professor Xinyu Dai

## **Abstract**

For all the benefits of the information abundance and communication technology, the “information overload” is one of the digital-age dilemmas we are confronted with. Recommender systems (RSs) are instrumental in tackling this problem. They help offer potential interesting information to individual consumers and allow online users to quickly find the personalized information that fits their needs. RSs are nowadays ubiquitous in various domains and e-commerce platforms. They are used to recommend books, point of interest and activities, music, movie, and references.

Collaborative filtering (CF) approaches play a central role in traditional recommender systems, which are extensively investigated in research community and widely used in industry. They are based on the simple intuition that if users rated items similarly in the past, then they will be likely to rate other items similarly in the future. CF models, however, suffer from data sparsity and the imbalance of ratings. They perform poorly on cold users and cold items for which there are no or few data. To overcome these weaknesses, additional sources of information are integrated into RSs. One research thread, which we call topic matrix factorization (Topic MF), is to integrate ratings with item contents or reviews text. Another research thread, which we call social matrix factorization (Social MF), is to combine ratings with social relations. Nevertheless, both Social MF and Topic MF ignore some useful information, either item reviews or social relations. Contrast to integrating more data sources into RSs, another way is to mine the limited data information more deeply; for example, factorizing only the rating matrix and exploiting the implicit feedback from it. Nevertheless, there is no related work on how to merge the two ways of integrating auxiliary sources and of mining implicit feedback.

This thesis focuses on combining multiple information sources and incorporating implicit feedback for recommender systems. The main contributions of the dissertation are outlined as follows.

1. Proposing a novel synthetic approach MR3 to jointly model user-item ratings, social network structure, and item reviews for rating prediction; and along with an extended Social MF method which exploits the ratings and social relations more tightly by capturing the graph structure of neighbors.
2. Extending the proposed approach to obtain a new model MR3++ by incorporating implicit feedback from ratings to enhance its capability and to demonstrate its flexibility; the extension model mines the limited information more deeply by introducing implicit features which captures the influence of rated items.

We demonstrate our methods empirically on two real-life datasets over various state-of-the-art algorithms. Further we analyse the impact of their components and the sensitivity of their hyper-parameters.

**Keywords:** Recommender systems, collaborative filtering, matrix factorization, rating prediction, topic modeling, social recommendation, heterogeneous information sources, implicit feedback

# 目 录

目录 .....	v
<b>第一章 绪论 .....</b>	<b>1</b>
1.1 研究背景 .....	1
1.2 待研究问题 .....	3
1.3 本文工作 .....	3
<b>第二章 相关工作 .....</b>	<b>4</b>
2.1 基于协同过滤的推荐系统 .....	4
2.1.1 基于记忆的协同过滤 .....	5
2.1.2 基于模型的协同过滤 .....	7
2.2 加入文本信息的推荐系统 .....	8
2.2.1 主题建模 .....	8
2.2.2 主题矩阵分解 .....	9
2.3 加入社交关系的推荐系统 .....	10
2.3.1 基于记忆的社会推荐系统 .....	10
2.3.2 基于模型的社会推荐系统 .....	11
2.4 推荐系统的评估 .....	12
2.4.1 基于预测准确性的评估方法 .....	13
2.4.2 基于排序准确性的评估方法 .....	13
2.4.3 决策支撑的评估方法 .....	14
<b>第三章 基于多源信息融合的推荐系统 .....</b>	<b>16</b>
3.1 引言 .....	16
3.2 利用多源信息的各个组件 .....	16
3.2.1 利用打分信息 .....	19
3.2.2 利用社会关系信息 .....	20
3.2.3 利用评论文本信息 .....	21

3.3	多源信息融合的问题描述 .....	23
3.4	一个多源信息融合模型 .....	23
3.5	一个扩展社会矩阵分解法 .....	27
3.6	实验评估及结果分析 .....	27
3.6.1	数据集和评估指标 .....	27
3.6.2	社会推荐方法的比较 .....	29
3.6.3	评估多源信息融合模型 .....	30
3.6.4	额外数据源对推荐性能的贡献 .....	32
3.6.5	多源信息融合模型的超参分析 .....	34
3.7	小结 .....	35
<b>第四章</b>	<b>基于隐式反馈挖掘的扩展融合推荐系统 .....</b>	<b>38</b>
4.1	引言 .....	38
4.2	挖掘隐式反馈的动机 .....	38
4.3	隐式反馈挖掘的扩展融合推荐系统问题描述 .....	39
4.4	一个挖掘隐式反馈的扩展融合模型 .....	40
4.5	关于隐式特征矩阵的进一步说明 .....	42
4.6	实验评估及结果分析 .....	43
4.6.1	评估挖掘隐式反馈的扩展融合模型 .....	43
4.6.2	隐式反馈对推荐性能的贡献 .....	44
4.6.3	挖掘隐式反馈的扩展融合模型的超参分析 .....	47
4.7	小结 .....	47
<b>第五章</b>	<b>模型的学习过程 .....</b>	<b>49</b>
5.1	学习过程 .....	49
5.2	参数的梯度 .....	50
<b>第六章</b>	<b>总结和展望 .....</b>	<b>52</b>
<b>参考文献 .....</b>	<b>53</b>	
<b>致谢 .....</b>	<b>60</b>	

附录：攻读硕士学位期间发表的论文、获得的奖励、参加的学术活动和参与 的科研项目 .....	<b>61</b>
--	-----------

# 第一章 绪论

## 1.1 研究背景

通信技术和网络媒体的发展，产生了大量的信息：文本信息诸如新闻、博客、产品评论；多媒体信息诸如音乐、图片、视频。为克服由此带来的信息过载问题，信息检索（information retrieval, IR）系统通过用户（user）发起的主动信息请求（query）从候选的物品（item）集合中检索出与该请求匹配度（relevance）高的。而信息过滤（information filtering）系统则在用户没有主动发起请求时根据用户的偏好（preference）和隐式行为数据等信息为用户过滤掉无关的信息 [4]。自上世纪九十年代初，研究者们就对如何构建支撑用户做出个性化决策的信息系统产生了兴趣，并提出了多种信息过滤算法，这些系统/算法统称为推荐系统/算法（recommender systems, RecSys）[50]。由于推荐系统能够为广大用户提供他们感兴趣和有用的物品，给企业带来营收、为平台增强功能同时也提升了用户满意度，使得推荐系统在电子商务 [33]、流媒体平台 [1] 和社交网络与社会媒体 [30, 63] 等领域有广泛的应用（见图 1.1）。

协同过滤的推荐系统（collaborative filtering, CF）为用户推荐与他们有着相似偏好的用户所喜欢的物品 [1]。一种方法是计算用户与用户（user-based CF）或者物品与物品（item-based CF）的相关性 [33, 52]，据此来预测一个用户对未知物品的偏好程度。另一种方法是学习用户与物品的向量表示（latent factors CF），这些隐藏因子描述了用户与物品的特性，据此可用他们的内积来预测未知的打分（rating）[32, 42]。尽管协同过滤方法在传统推荐系统中有着重要作用，他们却深受数据稀疏性与打分不平衡的影响：对于打分较少或者没有打分的用户与物品（cold-start problem, cold-user, cold-item），协同过滤方法则无能为力。

克服数据稀疏性的一条研究道路是加入额外的数据源。社交网络、社交媒体与知识共享网站的发展，例如Twitter、微博、Facebook、Epinions、大众点评，为用户之间建立联系、使用户参与到在线活动中、让用户表达他们的观点和分享内容提供了平台。社交网络关系（social relations）和用户对产品的评论（item reviews），是在打分数据之外的两种重要信息源，这给传统的推荐系统既带来了机遇又带来了新的挑战。社会矩阵分解（social matrix factorization, Social MF）将社交关系集成到打分数据中，从而能够同时利用这两种信息 [2, 22, 27, 36, 57]。社会学的经验研究表明，在有相似特性的人群间更易建立

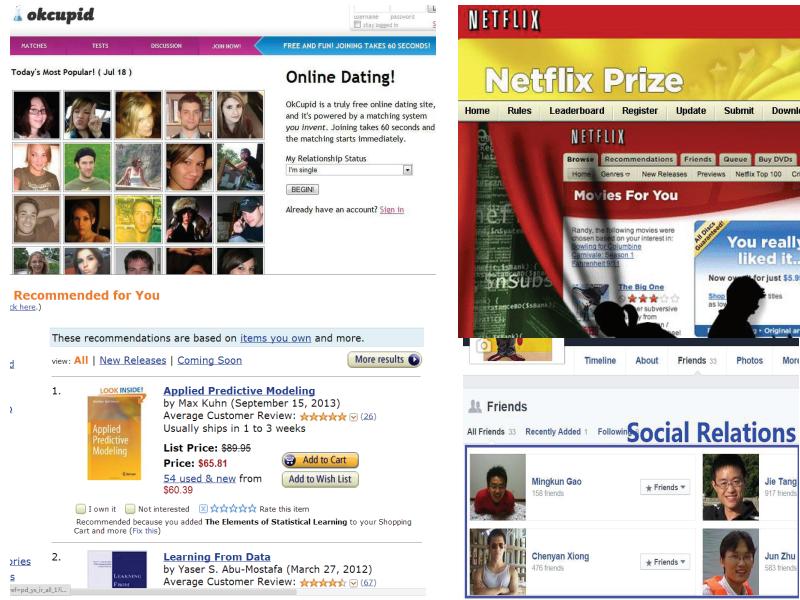


图 1.1: 推荐系统的应用场景举例。约会推荐、电影推荐、书籍推荐和好友推荐。

社交关系，即“物以类聚，人以群分”(theory of homophily)；而同质性的连接成为了社会影响力的重要方式(social influence)，即“近朱者赤，近墨者黑”[55, 56]。社会矩阵分解方法有两个组件，一个是矩阵分解，它分解打分矩阵学习用户与物品的向量表示，另一个是社会分解，它分解社交网络学习用户的社会表示。用户的向量表示与用户的社会表示都是描述用户这一共同的实体，因此通过他们就能桥接打分数据和社交数据。主题矩阵分解(topic matrix factorization, Topic MF)将评论文本集成到打分数据中，从而能够利用这两种信息[3, 17, 26, 39, 59]。物品评论在一定程度上解释了用户打分的行为。主题矩阵分解方法有两个组件，一个是矩阵分解，它分解打分矩阵学习用户与物品的向量表示，另一个是主题模型，它建模文本信息学习物品的主题分布。物品的向量表示与物品的主题分布都是描述物品这一共同的实体，因此通过他们就能桥接打分数据和评论数据。然而社会矩阵分解和主题矩阵分解都忽略了对方有用的信息。

克服数据稀疏性的另一条道路是深挖有限的数据源。对于显式的打分(比如1到5分)，它表明了用户对物品的偏好程度；伴随着显式打分的另一种信息，是用户是否对物品表达过偏好，如果发生了打分行为则取值是1，否则取值是0，这种指示信息称为隐式打分信息(implicit rating)。用户通过投票一个分数来隐式表达它的偏好，因此从先验上来看，有相同打分行为的用户更容易有相似的

偏好。SVD++模型 [32]通过学习一个额外的隐式特征矩阵来利用这种隐式打分信息，而TrustSVD [22]模型则再加入了社交关系；这两个模型都经验性的验证了隐式反馈信息的有用性。最近的工作虽然有集成上述主题矩阵分解和社会矩阵分解以达到同时利用打分、评论和社会关系三种信息 [9, 46, 60]，但他们都没有挖掘隐式信息以进一步提高推荐性能。

## 1.2 待研究问题

随着集成多种额外数据源的尝试日益丰富，凸显高效融合方法的缺乏；另外，如何将加入多种数据源和深挖有限数据源结合起来以进一步提高推荐性能，依然是有待解决的问题：

1. 给定打分矩阵、社交矩阵和评论文本，如何高效地同时利用三种信息提高推荐系统性能。现有的融合框架组合了一个来自主题矩阵分解的方法和一个来自社会矩阵分解的方法，然而他们所采用的两个组件都不是高效的。发展和融合更加高效的组件也就凸显的尤为重要。
2. 如何在融合多源信息的同时，在深度上挖掘隐式反馈信息。现有的方法要么在打分数据的基础上融合社会关系和产品评论信息，要么在显式打分的基础上挖掘隐式反馈信息，然而他们都只是从广度或者深度出发，因而就难免忽略彼此有用的信息。

## 1.3 本文工作

本文对推荐系统中的多源信息融合和隐式反馈挖掘进行了研究，并提出了两个推荐模型，具体来说：

在第三章中，针对现有数据融合框架所使用的两个组件不高效问题，本文先提出一个更加高效的社会矩阵分解方法，它能够考虑到社交网络的局部结构性质，然后在此基础上提出一个能融合三种信息的高效方法。

在第四章中，针对以往研究都要么只从广度上增加额外数据源，要么都只从深度上挖掘有限数据源，从而忽略掉对彼此有用的信息的问题，本文提出一个基于隐式反馈挖掘的扩展融合模型，首次实现了横向多源数据融合和纵向隐式反馈挖掘两个缓解冷启动和数据稀疏性问题的重要方向的汇合。

两个模型的求解过程是一致的，因此单独抽象出来放在第五章。所提出的融合方法及其扩展模型在两个较大规模的数据集上做了实验验证，并对结果做了分析。在提出上述两个模型之前是第二章相关工作的介绍，最后是第六章结论与展望。

## 第二章 相关工作

按照时间的演进，推荐系统的发展可以划分为四个有重叠的阶段。第一阶段是面向系统的探索，它验证了推荐系统的可行性与效力。这一阶段的标志性事件是在1996年于加州伯克利召开的协同过滤（collaborative filtering, CF）专题研讨会，以及随后由美国计算机协会（ACM）通信杂志出版的专栏文集[49]。第二阶段是快速商业化，它在互联网商业中快速发展，包括麻省理工学院和明尼苏达大学都商业化了他们的推荐系统。此外，这一阶段也解决了一定的技术难题，包括2001年提出的处理大规模在线推荐的基于物品（item-based CF）的相关性计算方法[52]；还探讨了若干研究问题，包括前N个物品的推荐（top-N recommendation）、隐式打分（implicit ratings）、冷启动（cold-start）等。这些方法现在仍然使用、这些问题仍然是研究的热点。第三阶段是研究大爆发，推荐系统成为了研究的主流，从人工智能到数据挖掘，从隐私与安全到商务与营销，推荐系统都是重要的主题。在2006年，美国视频租赁公司Netflix公开了超过1亿规模的打分大数据集[5]，并设立了百万美元大奖：奖励打分预测误差性能指标胜过原公司基线方法10%。第四阶段是将推荐系统落地、进一步向前发展。这一阶段包括推荐系统领域专门会议（ACM RecSys'07）的召开，以及对推荐系统在不同场景下的性能评估研究[23]。

本文的工作是基于协同过滤的打分预测，属于协同过滤的研究范畴，以下将介绍相关工作。首先是利用打分信息的基于矩阵分解的隐藏因子协同过滤方法（latent factors CF），然后是加入文本信息的主题矩阵分解方法（Topic Matrix Factorization），接着是加入社交关系的社会矩阵分解方法（Social Matrix Factorization），最后是推荐系统的评估。

### 2.1 基于协同过滤的推荐系统

协同过滤的推荐方法包括基于记忆的协同过滤（memory-based CF）和基于模型的协同过滤（model-based CF）两种类型[1, 15]，他们都假定如果用户（user）在过去的历史记录中对物品（item）的偏好（preference）相似，那么他们在对将来的未知物品的偏好也很可能相似，从直觉上它实现了口碑效应（word of mouth）[14]。偏好的表达形态可以是显式的打分（rating），例如电影评论的1到10分，也可以是用户隐式的行为，例如点击、浏览、购买。基于协同过滤的推荐系统只需要这样的偏好记录，而无需任何内容信息，如用户的性

别、年龄，或者物品的标签、标题。

### 2.1.1 基于记忆的协同过滤

基于记忆的协同过滤根据相关性计算的不同又细分为基于用户的（user-based CF）协同过滤和基于物品的（item-based CF）协同过滤 [8]。为用户预测未知物品的打分时，前者是基于该用户与其他用户的相似性（所以叫基于用户的），后者是基于该物品与其他物品的相似性（所以叫基于物品的），计算出来的相似性作为该用户对该未知物品打分时的权重。

**基于用户的协同过滤** 让 $u, v$ 代表用户， $i, j$ 代表物品，用户对物品的历史记录打分记为 $r_{u,i}$ ，预测/估计的打分记为 $\hat{r}_{u,i}$ ，那么基于用户的协同过滤（user-based CF, user-user CF）对未知打分的预测公式是：

$$\hat{r}_{u,i} = \frac{\sum_{v \in Nei(u)} sim(u, v) \times r_{v,i}}{\sum_{v \in Nei(u)} |sim(u, v)|} \quad (2.1)$$

这里 $Nei(u)$ 记录的是用户 $u$ 的邻居，即也打分过物品 $i$ 的用户集合，而 $sim(u, v)$ 代表两个用户之间的相似性，它作为基于记忆的协同过滤打分预测中的权重。

上述公式的一个变体是考虑到不同用户的打分偏置，例如有的用户打分普遍偏高，而有的用户可能比较苛刻、打分普遍偏低：

$$\hat{r}_{u,i} = \bar{r}_u + \frac{\sum_{v \in Nei(u)} sim(u, v) \times (r_{v,i} - \bar{r}_v)}{\sum_{v \in Nei(u)} |sim(u, v)|} \quad (2.2)$$

相似性的计算常用的有皮尔逊相关（Pearson correlation）和余弦相似度（Cosine similarity）。

$$sim(u, v) = \frac{\sum_{i \in N_u \cap N_v} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in N_u \cap N_v} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in N_u \cap N_v} (r_{v,i} - \bar{r}_v)^2}} \quad (2.3)$$

上述皮尔逊相关系数计算中的求和索引是取两个用户打分物品集合的交集，即 $N_u$ 表示用户 $u$ 的打分物品集合。为了避免只有少数打分交集而相似性却很高的问题，可以对交集的规模取一个阈值，比如将上述已计算的得分乘以因子: $\min\{|N_u \cap N_v|/50, 1\}$ 。

与上述基于统计方法的皮尔逊相关不同，余弦相似度是一种基于线性代数的向量空间法：打分向量的内积除以他们的欧式长度：

$$sim(u, v) = \frac{\mathbf{r}_u \cdot \mathbf{r}_v}{\|\mathbf{r}_u\|_2 \|\mathbf{r}_v\|_2} = \frac{\sum_i r_{u,i} r_{v,i}}{\sqrt{\sum_i r_{u,i}^2} \sqrt{\sum_i r_{v,i}^2}} \quad (2.4)$$

这里我们将未知的打分置为0，而求和项中的索引取全体物品集 $I$ 。如果用户打分已经事先减去了均值的话，那当两个用户的打分物品集相同时余弦相似度等价于皮尔逊相关系数。

**基于物品的协同过滤** 当用户规模在不断增加时，基于用户的协同过滤在为每个用户寻找相似用户时需要遍历整个用户集，即 $\mathcal{O}(|U|)$ 次搜索操作。基于物品的协同过滤（item-based CF, item-item CF）通过寻找物品打分模式之间的相似性，避免了这个问题。基于物品的协同过滤方法首先于2001年在文献[52]中使用，随后用到了电商亚马逊（Amazon.com）中[33]。基于物品的协同过滤的关键步骤也是相似性的计算，只不过此时是计算物品之间的相似度： $sim : I \times I \rightarrow \mathbb{R}$ 。基于物品的协同过滤对未知打分的预测公式是：

$$\hat{r}_{u,i} = \frac{\sum_{j \in TopK(i)} sim(i, j) \times r_{u,j}}{\sum_{j \in TopK(i)} |sim(i, j)|} \quad (2.5)$$

这里 $TopK(i)$ 表示与给定物品 $i$ 最相似的前 $K$ 个物品，一般取30个，而 $sim(i, j)$ 代表两个物品之间的相似性，它作为基于记忆的协同过滤打分预测中的权重。

上述公式的一个变体是考虑到用户与物品的打分偏置，例如有的用户打分普遍偏高，而有的用户可能比较倾向于打低分，或者有的物品所获得的打分普遍偏高，比如是一部好电影的时候：

$$\hat{r}_{u,i} = b_{u,i} + \frac{\sum_{j \in TopK} sim(i, j) \times (r_{u,j} - b_{u,i})}{\sum_{j \in TopK} |sim(i, j)|} \quad (2.6)$$

这里打分偏置（或者叫基线预测器） $b_{u,i}$ 等于全局打分平均加上用户和物品的平均打分： $globalMean + userMean + itemMean$ 。物品之间的相似性 $sim(i, j)$ 可以类比上述基于用户的协同过滤中所使用的皮尔逊相关和余弦相似度。

当用户集合远大于物品集合时，由于物品集合的相对静态性，可以预计算出每个物品的前 $K$ 个最相似的物品列表，然后在对未知打分进行预测时就可以进行快速的查表操作。这种预计算和截断的技巧对于将推荐系统部署到实际的应用中相当重要。只当物品规模或者打分数据有显著性的变化时再线下重新计算物品与物品的相似性。

### 2.1.2 基于模型的协同过滤

基于模型的协同过滤（Model-based CF）从以往的历史记录打分中学习一个模型，然后利用这个模型对未知的打分进行预测。基于聚类的协同过滤 [43]联合了基于记忆和基于模型的协同过滤：先将物品空间聚类为各个更小规模的空间，然后在各个小空间中运用基于记忆的协同过滤。

基于朴素贝叶斯的方法将协同过滤规约为机器学习中的有监督分类问题 [41, 54]；当预测用户 $u$ 对物品 $i$ 的未知打分时，构造一个朴素贝叶斯分类器：每条训练样本 $E_j$ 对应用户已打分的一个物品 $j \in N_u$ ，样本标签 $C_j$ 即是所打的离散化分数，该样本的特征表示为也打过该物品的其他用户（脚标 $o$ 表示已观察数据）：

$$\text{class} = \arg \max_{j \in \text{classSet}} p(\text{class}_j) \Pi_o P(X_o = x_o | \text{class}_j). \quad (2.7)$$

基于奇异值分解（Singular Value Decomposition, SVD）的协同过滤将用户和物品从原始稀疏高维的打分空间降维到低维稠密的维度空间中 [6, 51]，可以学到隐藏的语义，同时还能克服稀疏性问题：

$$R \approx P \Sigma Q^T. \quad (2.8)$$

上述对打分矩阵的分解用三个低维矩阵近似：正交矩阵 $P$ 是学习到的低维用户特征矩阵，描述了用户的偏好；正交矩阵 $Q$ 是学习到的低维物品特征矩阵，描述了物品的特性；对角矩阵 $\Sigma$ 刻画了偏好的权重。上述分解的一个变体是截取前 $k$ 大的奇异值，在弗洛贝尼乌斯范数（Frobenius norm）的误差度量下，这是最优的秩 $k$ 近似（rank- $k$  approximation）。可以看到 $\Sigma Q^T$ 将物品从原始的稀疏高维打分空间变换到了低维稠密的向量空间中。当从历史的打分数据中学习到这三个参数矩阵后，用户 $u$ 对所有物品的打分可以通过先取出其主题-偏好向量然后再乘以变换后的物品特征矩阵而得到：

$$\hat{\mathbf{R}}_u = \mathbf{P}_u \Sigma Q^T. \quad (2.9)$$

基于主成分分析（Principle Component Analysis, PCA）的方法与基于奇异值分解的方法是等价的，它在协同过滤中曾用于笑话推荐 [18]。

基于概率隐藏语义分析（Probabilistic latent semantic analysis, PLSA, probabilistic latent semantic indexing, PLISI）的方法计算用户对物品的偏好概率 $P(i|u)$ 或者打

分概率分布  $P(r_{u,i}|u)$  [24]，核心思想是将用户表示为若干偏好隐变量  $z$  (aspect) 的混合分布。

$$P(i|u) = \sum_z P(i|z)P(z|u). \quad (2.10)$$

**矩阵分解** 包括最大间隔矩阵分解 (maximum margin MF, MMMF)、非负矩阵分解 (nonnegative MF, NMF)、联合矩阵分解 (collective MF)、概率矩阵分解 (Probabilistic Matrix Factorization, PMF) 等在内的基于矩阵分解的协同过滤 [21, 31, 42, 48, 53]，在打分预测任务上已成为标准基线方法，他们的核心思想是学习用户和物品的低维向量表示 (特征学习:feature learning)。

基于协同过滤的推荐系统容易受到数据稀疏性和打分不平衡的影响，对于打分很少的冷启动用户和物品，协同过滤算法无能为力。缓解稀疏性问题的一条道路是加入额外的数据源，例如评论文本信息和社交网路关系信息。

## 2.2 加入文本信息的推荐系统

缓解冷启动 (cold-start, cold-user, cold-item) 问题的一种方法是加入额外的数据源，例如物品评论数据，即融合了文本信息的推荐系统。早期工作是从在线产品评论中抽取预先定义好的细粒度物品打分，然后将他们作为内容特征用到协同过滤中 [17, 26]。例如一个电影的层面 (aspect) 包含演员、风格、特效，一个酒店的层面包括位置、清洁度、屋景色，而一个餐馆的层面包括价格、卫生、服务等。这些层面的抽取需要使用领域知识，以及大量的人工干预。

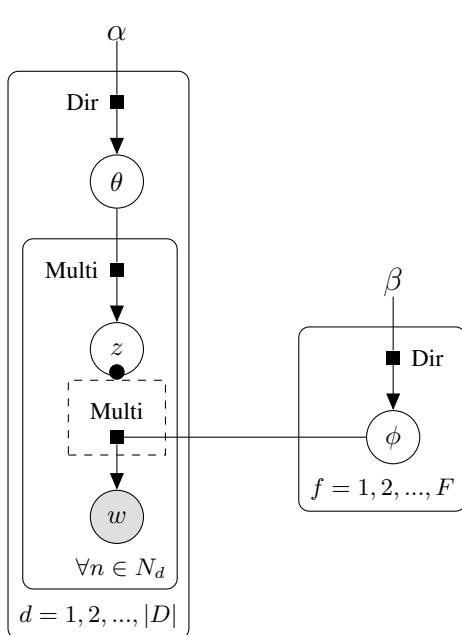
### 2.2.1 主题建模

主题建模方法在早期被用于自动发现科学文章的隐含话题，无监督学习文档-主题分布和主题-词分布。潜在狄利克雷分配 (Latent Dirichlet allocation, LDA) [7]的主题建模方法在文档集合  $D$  上的非负对数似然 (negative log-likelihood, NLL) 函数定义如下：

$$NLL(D) = - \sum_{d=1}^N \sum_{n \in N_d} \left( \log \theta_{z_{d,n}} + \log \phi_{z_{d,n}, w_{d,n}} \right). \quad (2.11)$$

这里参数  $\theta$  和  $\phi$  分别是主题分布 (topic proportions) 和词分布 (topic distributions)，而  $w_{d,n}$  和  $z_{d,n}$  则表示文档中的一个词及其该词的主题。上述公式的含义是对文档集中的每个文档，以及文档中的每个词，生成他们的主题概率要最大化。他们的生成过程如下：

- i). 从狄利克雷先验分布中采样文档-主题分布:  $\theta_d \sim Dir(\alpha)$ , 这里 $d$ 是语料中的文档,  $\alpha$ 是狄利克雷分布的参数。
  - ii). 从狄利克雷先验分布中采样主题-词分布:  $\phi_f \sim Dir(\beta)$ , 这里 $f$ 是主题索引,  $\beta$ 是狄利克雷分布的参数。
  - iii). 对于每个文档 $d$ 中位置 $n$ 的词 ( $d \in \{1, 2, \dots, N\}, n \in N_d$ ):
    - 从多项分布中采样主题:  $z_{d,n} \sim Mult(\theta_d)$ ,
    - 从多项分布中采样词:  $w_{d,n} \sim Mult(\phi_{z_{d,n}})$ 。



框外是先验超参  $(\alpha, \beta)$ , 框内阴影结点是可观察变  $(w)$  量、空白结点是模型参数  $(\theta, \phi)$ 。注意, 虚线框表示以  $z$  为主题索引, 到主题-词分布  $\phi$  中取到对应的多项分布 Multi, 继而从中采样到词  $w$ 。

## 2.2.2 主题矩阵分解

将主题建模集成到协同过滤中从而能自动无监督地利用文本信息的工作有协同主题建模（Collaborative topic modeling, CTR）[59]和泊松分解 [59]，他们推荐科学文献，属于单类协同过滤（One-class collaborative filtering, OCCF）[44]，即打分行为只有两类：推荐或者不推荐。

在打分预测任务上集成主题建模的工作有隐藏因子和隐藏主题模型 (Hidden factors and hidden topics, HFT) [39]，该模型从评论文本中学习产品的

---

<sup>1</sup><https://github.com/jluttine/tikz-bayesnet>

主题分布，同时从打分数据中学习物品的特性向量，然后用一个softmax变换（指数函数）将这两者联系起来；能够联系起来的原因是它们都是同一物品的两个侧面，同时也起到转换概率值的主题分布和实数值的特性向量。也有集成非负矩阵分解以利用文本信息的工作 [3]。这些方法都认为主题分布的维度和特性向量的维度相等。为了避免变换函数的选择困难，[34]将矩阵分解替换为高斯混合（mixture of Gaussian）模型；进一步的，[61]通过共同聚类（co-clustering）用户社区和物品群组来生成打分分布和主题分布，使得用户因子和物品因子可以有不同的维度；而 [12]则通过引入基于层面的模型（aspect-based model）中间表示，来连接用户的兴趣分布和电影的内容分布，也能允许这两个分布的维度不同，但要学习额外的变换矩阵从一个维度空间映射到另一个维度空间。

从最后的参数求解来看，主题矩阵分解的基本思想是将生成文本语料的概率似然函数（或者词现矩阵重构损失）集成到偏好预测的损失函数中得到一个综合的优化目标函数。

加入文本信息的推荐系统虽然能考虑产品评论这一重要数据源，但是忽略了诸如社交关系等另一类能提升推荐性能的信息源。

### 2.3 加入社交关系的推荐系统

缓解冷启动问题的另一种额外数据源是社会关系，即融合了社交关系的推荐系统，可以称为社会推荐（social recommendation, social recommender systems, SocialRec, SocialRecSys）；一个通用的基于协同过滤的社会推荐框架包括如下两部分 [58]：

$$\text{a CF-based SocialRec} = \text{a basic CF} + \text{a social information model} \quad (2.12)$$

第一部分是基本的协同过滤方法，以利用打分信息；第二部分是社会模型，以利用社交关系。根据协同过滤的分类方法，可类似的将社会推荐系统也分为两类：基于记忆的社会推荐和基于模型的社会推荐。

#### 2.3.1 基于记忆的社会推荐系统

基于记忆的社会推荐使用基于记忆的协同过滤利用打分信息，尤其是基于用户的协同过滤，因为这样可以更自然的和社交关系的利用联系起来。当两个用户的共同打分物品集很少时，原始用户-用户相似性计算（皮尔逊相关或者余

弦相似度) 的准确性就不高, 因而很难找到可靠的邻居用户。利用社交关系的方法有多种形式, 其中信任感知的推荐模型 (Trust-aware RecSys) [38] 将原始基于用户的协同过滤中关键的用户-用户相似性权重替换 (或者融合) 为从信任网络中计算得到的信任值, 因此这属于两阶段过程: 预计算用户之间的信任值作为相似性权重 → 基于用户的协同过滤。其中的核心问题是找到给定用户  $u$  的相似邻居  $Nei(u)$ , 例如直接使用从社交网络中获取的直接好友作为相似用户:  $Nei(u) = \{v | T(u, v) = 1\}$ , 这里  $T(u, v)$  指示社交网络中两个用户的社交关系, 如果有关系则取值为1, 否则取值为0。

### 2.3.2 基于模型的社会推荐系统

与上述两阶段过程不同, 基于模型的社会推荐系统使用基于模型的协同过滤作为利用打分信息的组件, 从而能够同时分解打分矩阵和社交关系矩阵, 让两者互相促进 [36]。一个基于模型的社会推荐系统的优化框架可以总结如下 [58]:

$$\begin{aligned} \min_{P, Q, \Omega} & \underbrace{\|W \circ (R - P^T Q)\|_F^2}_{\text{Hadamard product}} + \underbrace{\lambda_{rel} Social(T, S, \Omega)}_{\text{Social model}} \\ & + \lambda_{norm}(\|P\|_F^2 + \|Q\|_F^2 + \|\Omega\|_F^2) \quad (2.13) \end{aligned}$$

优化框架包含三部分内容, 第一部分是基于模型的协同过滤组件, 它利用打分信息学习用户特征矩阵  $P$  和物品特征矩阵  $Q$ ; 第二部分是利用社会信息的社会模型, 它参数化为  $\Omega$ ; 第三部分是防止过拟合的范数正则项。公式中第二部分里的  $\lambda_{rel}$  是控制来自于社交信息贡献的超参。预计算好的可观察变量  $W$  和  $S$  的介绍, 请分别参见第 三章第3.2节公式3.4 和第3.4节公式 3.9。

上述优化框架的一个实例是SoRec模型 [36]:

$$Social(T, S, P, Z) = \min_{P, Z} \sum_{T_{u,v} \neq 0} (S_{u,v} - P_u^T Z_v) \quad (2.14)$$

该模型为每个用户学习了一个额外的特征矩阵  $Z$  以刻画他们的社会属性, 同时在原来基于模型的协同过滤中学习的用户特征矩阵  $P$  也会被来自社会关系的信息所更新, 从而融合打分信息和社交信息以达到相互促进的效果。与额外学习一个社会特征矩阵不同, SoReg (Social Regularization) 模型 [37]是对有好友关系的两个用户的特征向量进行约束, 使他们比不是好友的更相近:

$$Social(T, S, P) = \min_P \sum_{T_{u,v} \neq 0} S_{u,v} \|P_u^T - P_v\|_F^2. \quad (2.15)$$

有社会关系的两个用户之间的相近度是由他们之间的相似度 $S_{u,v}$ 控制的：小相似度则 $u, v$ 特征向量之间的距离要大、大相似度则特征向量之间的距离要小。SocialMF模型 [27, 28]则要求用户的偏好要近似于其所有好友的平均偏好：

$$Social(T, S, P) = \min_P \sum_u \left\| P_u - \sum_{v:T_{u,v} \neq 0} S_{u,v} P_v \right\|_F^2. \quad (2.16)$$

局部和全局模型（local and global, LOCABAL） [57]从全局网络和局部邻居视角利用社交信息：全局信息从整体网络中用PageRank算法计算而来，度量了用户打分的先验权威性；局部信息则揭露了用户与其直接邻居的相关性。信任者和被信任者模型（trustee and truster, TrustMF） [62]从信任和被信任双向关系建模：信任模型建模了其他用户的打分行为是如何影响活动用户的，而被信任模型则建模了其他用户的打分行为是如何受到活动用户影响的。信任分解模型（decomposed trust, DecTrust）将信任关系解剖为四种类型（benevolence, integrity, competence, and predictability），将信任关系值的预测和隐藏用户特征向量之间的相似性组合起来，得到用户之间的总相似值 [2]。

加入社交关系的推荐系统虽然能考虑用户社会网络这一重要数据源，但是忽略了诸如产品评论等另一类可以提升推荐性能的信息源。

## 2.4 推荐系统的评估

当开发出一个推荐系统时，知道该系统的工作效果是很有用的。除了在代价昂贵的真实用户集上运行算法以度量其效果，离线评估是一种常用的重要方式。离线评估的基本过程是基于训练-测试（或者训练-验证-测试）范式，即在训练集（training data）训练推荐算法，然后在新的测试集（test data）上评估算法的预测泛化（generalization）能力。验证集（validation data）通常用来调节模型的超参，即调参过程，所以也叫开发集。为了充分利用数据集，一个折衷的方法是 $K$ 折交叉验证（ $K$ -fold cross-validation,  $K$ -fold CV）：将全体数据集划分为等量的 $K$ 份，轮流保留一份作为测试集、另外的 $K - 1$ 份作为训练集，最后将测试结果平均，一般 $K$ 取5或者10。 [15]

### 2.4.1 基于预测准确性的评估方法

基于预测准确性（Prediction Accuracy）的评估方法是度量推荐算法在未知打分数据集上的预测性能：预测值与真实值得差异。常见的度量有平均绝对值误差（mean absolute error, MAE）及其归一化版本、根均方误差（root mean squared error, RMSE）及其归一化版本。

MAE对预测误差取绝对值：

$$MAE_{\mathcal{T}} = \frac{1}{|\mathcal{T}|} \sum_{(u,i) \in \mathcal{T}} |R_{u,i} - \hat{R}_{u,i}| \quad (2.17)$$

RMSE对预测误差取平方：

$$RMSE_{\mathcal{T}} = \sqrt{\sum_{(u,i) \in \mathcal{T}} (R_{u,i} - \hat{R}_{u,i})^2 / |\mathcal{T}|} \quad (2.18)$$

可以看到，RMSE对较大预测误差值惩罚更厉害。他们的归一化版本是再除以量  $R_{max} - R_{min}$  使得最终的评估结果的值域是[0, 1]。MAE和RMSE的一个变体是考虑时序信息的评估，即对于当前时刻  $t$ ，只对时间在此之前的打分进行预测和评估：

$$TA - RMSE_t = \sqrt{\sum_{(u,i) \in \mathcal{T}_t} (R_{u,i} - \hat{R}_{u,i})^2 / |\mathcal{T}_t|} \quad (2.19)$$

基于预测准确性的评估方法对打分高的和打分低的物品同等对待，然后实际上推荐系统一般只关注打分高的物品，即用户真正感兴趣的物品而不是推荐那些用户不喜欢的物品。

### 2.4.2 基于排序准确性的评估方法

基于排序准确性（Ranking Accuracy）的评估方法是度量推荐算法生成的用户对物品的偏好排序性能：用户偏好高的物品应该排在前面。常见的度量有半衰期效用（half-life utility, HLU）、平均倒排命中排序（average reciprocal hit-rank, ARHR）和归一化累积折扣增益（Normalized Cumulative Discounted Gain, NDCG）。[23, 50]

假设对用户  $u$ ，其偏好物品集是  $Test_u$ ，且按照喜欢程度递降排序，即按照  $idx(i)$  依次为物品赋值  $1, 2, \dots, |Test_u|$ 。HLU 考虑算法推荐的物品列表中命中的偏好物品集  $Hits_u$ ，及其命中时该物品的排序位置  $rank(i)$  且所得效用按位置指数衰减 [8, 50]：

$$HLU_u = \frac{\sum_{i \in Hits_u} \frac{1}{2^{(rank(i)-1)/\alpha}}}{\sum_{i \in Test_u} \frac{1}{2^{(idx(i)-1)/\alpha}}} \quad (2.20a)$$

$$HLU = 100 \frac{\sum_u \sum_{i \in Hits_u} \frac{1}{2^{(rank(i)-1)/\alpha}}}{\sum_u \sum_{i \in Test_u} \frac{1}{2^{(idx(i)-1)/\alpha}}} \quad (2.20b)$$

参数 $\alpha$ 是效用的半衰期参数，例如可以取10；分母是最高效用，起到归一化效果。第一个公式是计算一个用户的效用，第2个公式是在全部用户集上的总体得分（乘以100后）。

与HLU对命中物品的排序位置以指数衰减不同，ARHR则是以倒数（线性）方式作为惩罚：[11]

$$ARHR = \frac{1}{\#users} \sum_{i \in Hits_u} \frac{1}{rank(i)} \quad (2.21)$$

可以看出，ARHR的衰减比HLU慢，但是比下面的DCG要快，因为它以排序位置的对数衰减：

$$DCG@K = \frac{1}{\#users} \sum_u \sum_{i=1}^K \frac{2^{rel_i} - 1}{\log_2(1 + i)} \quad (2.22)$$

这里算法生成的推荐列表长度为 $K$ ， $rel_i$ 表示物品*i*是否用户喜欢，取值是{0, 1}。当按照用户的真实最佳排序时计算出的值叫做最佳值，记为 $DCG^*$ ，则 $NDCG = DCG/DCG^*$ ；这里的归一化方式类似于ARHR。

### 2.4.3 决策支撑的评估方法

决策支撑的评估方法曲线下面积（area under the curve, AUC）是接受者操作特性（relative operating characteristic, ROC）曲线的一个数值度量。

在二元分类问题中，记 $N^+$ 表示正样本个数， $N^-$ 表示负样本个数， $rank(i)$ 表示第*i*个正类在列表中的排序位置，那么AUC计算公式如下 [54]：

$$AUC = \frac{\sum_i rank(i) - \binom{N^-}{2}}{N^- N^+} \quad (2.23)$$

在推荐任务中，类似的，可以为用户*u*构建“正样本”即该用户喜欢的物品集{*i*}、“负样本”即该用户不喜欢的物品集{*j*}，那么性能高的推荐算法应该

把喜欢的物品排在不喜欢的物品之前。记 $Test_u = (u, i) | u \text{ likes } i$ 表示测试集中用户 $u$ 喜欢物品 $i$ 的成对集合， $Train_u$ 表示训练集中的成对集合，不在这两个集合中的物品 $j$ 则认为是该用户不喜欢的，那么AUC计算公式如下[65]:

$$AUC = \frac{1}{\#users} \sum_u \frac{1}{|E(u)|} \sum_{(i,j) \in E(u)} \delta(\hat{R}_{u,i} > \hat{R}_{u,j}) \quad (2.24)$$

这里 $E(u) = \{(i, j) | (u, i) \in Test_u \wedge (u, j) \notin (Test_u \cup Train_u)\}$ 。上述公式的直观含义是推荐算法对用户喜欢物品的预测打分应该比该用户不喜欢物品的预测打分更高，从而才能把喜欢的物品排在不喜欢的物品前面。

由于本文的工作是打分预测，因为一个自然的度量标注是基于预测的评估，我们选择了根均方误差RMSE，它对高预测误差比MAE惩罚更严重（平方vs. 绝对值）。

## 第三章 基于多源信息融合的推荐系统

### 3.1 引言

由于加入额外数据源的方法在缓解协同过滤中的冷启动和数据稀疏性问题上都得到了经验上的证明，各种加入文本信息如产品评论和加入社会信息如好友关系的方法不断涌现：从协同主题建模（CTR）[59]到潜在因子和隐藏主题模型（HFT）[39]，从社会推荐模型（SoRec）[36]、社会正则模型（SoReg）[37]到全局-局部模型（这已经在组件HFT和LOCABAL）[57]。直观的想法，是两者都加入以期待推进性能的进一步提升，而三种信息都融合的方法则还比较缺乏；虽然有一些工作[9]已经融合了打分、文本和社会关系这三种信息，但他们所采用的组件CTR和SoReg并不高效，这已经在组件HFT和LOCABAL分别得到了经验和实验的验证。另外一些融合方法则不是关注于打分预测任务，而是标签推荐[60]、名人推荐[13]和文章推荐[46]。还有一些方法[16]是利用矩阵共同分解（matrix co-factorization）技术融合用户和物品的辅助信息，因而他们利用文本信息的方式是重构文档-词项共现矩阵（等价于latent semantic analysis）而不是挖掘隐藏的主题分布（latent Dirichlet allocation）。

为了充分利用多源异构信息以进一步提升推荐系统性能，本章首次提出了一个基于多源信息融合的高效推荐系统。该推荐模型的核心思想是基于协同过滤技术，利用物品的双重属性和用户的双重属性将评论数据源和社会关系数据源连接起来，从而得到一个综合模型。具体而言，协同过滤中的物品实体也参与到了文本信息中获得用户的评论；协同过滤中的用户实体也参与到了社会信息中建立用户之间的关系，那么这三种信息就可以通过基础的协同过滤融合起来得到一个综合模型。在这一个高效的综合模型基础上，我们还得到了一个高效的社会推荐方法，它能够充分利用打分数据和社会网络信息。

本章后续部分组织如下：首先3.2节给出了融合模型所利用的组件，然后3.3节对多源信息融合这一问题进行描述，接着3.4和3.5节介绍所提出的综合模型及其一个重要模块，第3.6节对所提模型进行实验评估和结果分析，最后3.7节是总结。

### 3.2 利用多源信息的各个组件

对于三类数据源：打分、社会关系和评论文本，它们都有对应的方法可以

利用，这些方法和模型我们称之为融合模型的组件（component）。因此在提出我们的融合模型之前，先来介绍利用多源信息的各个组件。这三种信息在真实世界的数据集上的示意例子见图 3.1，它们来自Ciao<sup>1</sup> 和Epinions<sup>2</sup> 数据集。



图 3.1: 三种推荐信息源。在Ciao和Epinions数据集上，存在三种类型的数据源：数值打分（1-5星），文本信息（产品评论），和社会关系（信任关系）。他们在图中分别用蓝色框标识出来：Rating，Review，Social Relations。

## 记号

假定有 $M$ 个用户 $\mathcal{P} = \{u_1, \dots, u_M\}$ 和 $N$ 个物品 $\mathcal{Q} = \{i_1, \dots, i_N\}$ . 我们将 $u, v, w$ 保留给用户索引，而将 $i, j, k$ 保留给物品索引。令 $R \in \mathbb{R}^{M \times N}$ 标记打分矩阵，其中的元素 $R_{u,i}$ 是用户 $u$ 在物品 $i$ 上的可观察打分，并且当没有打分时置其为0。打分预测任务就是从可观察数据中预测未知的打分。

除了上述基本的打分信息，还存在其他的辅助数据源。一种资源是社会关系信息：用户在社交网络中相互建立关系，他们之间的连接指示了好友或者信任关系的存在。令 $T \in \mathbb{R}^{M \times M}$ 标记用户-用户社会关系矩阵，其中的元素 $T_{u,v} = 1$ 如果用户 $u$ 与用户 $v$ 有社会连接，否则标记为0。另一种资源是评论文本信息：

<sup>1</sup><http://www.ciao.com/>

<sup>2</sup><http://www.epinions.com/>

用户对物品生成的评论，表达了他们的观点和倾向。因此物品附着有内容信息。令 $d_{u,i}$ 表示用户 $u$ 对物品 $i$ 所写的评论，它经常伴随着一个数值打分 $R_{u,i}$ 。<sup>3</sup>

本文所用的记号及其含义列于表 3.1.

表 3.1: 记号

符号	含义
$F$	隐藏因子的维度
$R_{u,i}$	用户 $u$ 对物品 $i$ 的打分
$R_{u,i}^b$	用户 $u$ 对物品 $i$ 的隐式打分
$W_{u,i}$	用户 $u$ 对物品 $i$ 的打分权重
$P_u$	用户 $u$ 的 $F$ -维特征向量
$Q_i$	物品 $i$ 的 $F$ -维特征向量
$Y_j$	物品 $j$ 的 $F$ -维隐式特征向量
$N_u$	用户 $u$ 的已打分物品集
$T_u$	用户 $u$ 所信任的用户集
$T_{u,v}$	用户 $u$ 和 用户 $v$ 之间的社会关系
$C_{u,v}$	用户 $u$ 和 用户 $v$ 之间的社会强度
$S_{u,v}$	用户 $u$ 和 用户 $v$ 之间的社会相似度
$H$	$F \times F$ -维的社会相关矩阵
$d_{u,i}$	用户 $u$ 对物品 $i$ 的评论（文档）
$w_{d,n}$	文档 $d$ 中的第 $n^{\text{th}}$ 个词
$z_{d,n}$	文档 $d$ 中的第 $n^{\text{th}}$ 个词的主题
$N_d$	文档 $d$ 中的词集合
$\theta_i$	物品 $i$ （对应的文档） $F$ -维主题分布
$\phi_f, \psi_f$	主题 $f$ 的词分布，及其对应的未归一化版本
$M, N$	用户数，物品数
$\mathcal{P}$	用户集合，总数为 $M$
$\mathcal{Q}$	物品集合，总数为 $N$
$L$	词表规模
$\ \cdot\ _F$	矩阵的弗罗贝尼乌斯（Frobenius）范数
$ \cdot $	集合的势（cardinality）

<sup>3</sup>我们将评论语料的文档-词项矩阵组织为 $D \in \mathbb{N}^{N \times L}$ ，其中的元素 $d_{n,l}$ 是在第 $n$ 个文档中词项 $l$ 的词现。为了方便起见，我们也将评论记为 $d_{u,i}$ ，即文档 $n$ 和物品 $i$ 是一一对应的。

### 3.2.1 利用打分信息



图 3.2: 用户和物品的打分矩阵示意图。每一行是一个用户，每一列是一个物品，可观察的打分数据构成了部分矩阵（标记为1-5分），打分预测任务就是预测未知的打分（标记为问号“?”）。

利用打分信息的常用方法是基于矩阵分解的隐藏因子模型 (matrix factorization based latent factor models) [31, 42]，他们的主要目标是学习隐藏的用户相关特征矩阵  $P = [P_1, \dots, P_M] \in \mathbb{R}^{F \times M}$ ，这描述了用户的偏好；同时学习隐藏的物品相关特征矩阵  $Q = [Q_1, \dots, Q_N] \in \mathbb{R}^{F \times N}$ ，这描述了物品的特性。特征学习通过近似已观察的打分矩阵，在最小二乘损失下求解一个正则优化问题 (regularized least squares, ridge regression):

$$\min_{P,Q} \sum_{\substack{R_{u,i} \neq 0}} (R_{u,i} - \hat{R}_{u,i})^2 + \lambda(\|P\|_F^2 + \|Q\|_F^2), \quad (3.1)$$

这里  $\lambda$  是正则超参，控制所学参数的范数以防止过拟合。当学习完用户和物品特征表示后，对未知打分的估计则可以用：

$$\hat{R}_{u,i} = \mu + b_u + b_i + P_u^T Q_i. \quad (3.2)$$

这里参数 $\mu$ ,  $b_u$ 和 $b_i$ 分别表示全体打分均值、用户打分均值（称为用户偏置，user bias）和物品打分均值（称为物品偏置，item bias）。维度为 $F$ 的特征向量 $P_u$ 和 $Q_i$ 分别表示用户 $u$ 的偏好和物品 $i$ 的特性，他们之间的内积则度量了两者的交互或者说匹配度。

一个打分矩阵的示意见图 3.2，每一行是一个用户，每一列是一个物品，可观察的打分数据构成了部分矩阵，打分预测任务就是预测未知的打分。

### 3.2.2 利用社会关系信息

对于社会关系信息源，社会矩阵分解（social matrix factorization, Social MF）法 [57, 66]的思想是学习用户的隐藏社会相关（social-specific）特征表示 $P$ ，以此来刻画用户建立社会关系的特性，即近似社会相似矩阵（social similarity matrix） $S \in \mathbb{R}^{M \times M}$ 。类似于打分矩阵分解，社会矩阵分解求解下述的正则化最小二乘损失优化问题（式中 $H \in \mathbb{R}^{F \times F}$ 是社会相关矩阵）：<sup>4</sup>

$$\min_{P,H} \sum_{T_{u,v} \neq 0} (S_{u,v} - P_u^T H P_v)^2 + \lambda(\|P\|_F^2 + \|H\|_F^2), \quad (3.3)$$

这里 $S_{u,v}$ 是用户 $u$ 和用户 $v$ 之间的社会相似度，如下定义为打分向量的余弦相似度：

$$S_{u,v} = \frac{\sum_i R_{u,i} \cdot R_{v,i}}{\sqrt{\sum_i R_{u,i}^2 \cdot \sum_i R_{v,i}^2}} \quad (3.4)$$

此即第二章第2.3.2节中公式2.13里面 $S$ 的具体含义：它是从打分信息和社交网络信息派生出的预算算好的变量，它将原始取离散值的社交关系 $T(u,v) = 1$ 实数化。

社会矩阵分解的工作原理有一个假设，即用户实体扮演了双重身份：一个是实施打分行为，即从打分数据中学习隐藏的偏好表示；一个是建立社会行为，即从社交关系中学习隐藏的社会表示。这两者都记为 $P$ ，作为沟通两类数据源的桥梁。

---

<sup>4</sup>我们略去了打分矩阵分解项即公式(3.1)，以更清晰的突出对社会关系的利用。

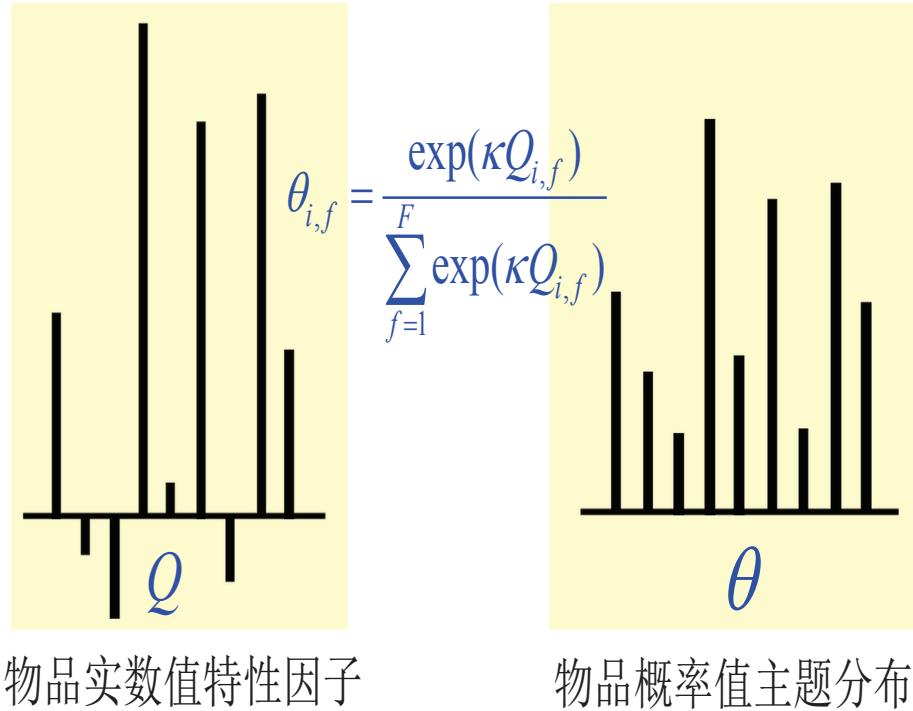


图 3.3: 物品特性因子和主题因子的变换示意图。左侧是从打分数据中学习到的物品实数值特性因子  $Q$ , 右侧是从评论文本中学习到的物品概率值主题分布  $\theta$ , 他们之间的联系通过softmax变换建立起来。

### 3.2.3 利用评论文本信息

对于评论文本信息源, 主题建模用来寻找物品的隐藏属性和隐藏主题 [7, 39]。语料的负对数似然用来参数化语料数据, 作为学习的优化目标; 其定义在章节 2.2 中已由公式 2.11 给出, 这里重列如下并作较为详细的解说:

$$-\sum_{d=1}^N \sum_{n \in N_d} \left( \log \theta_{z_{d,n}} + \log \phi_{z_{d,n}, w_{d,n}} \right),$$

这里要学习的参数  $\theta$  和  $\phi$  分别表示主题分布和词分布, 优化目标是逐词  $w_{d,n}$  遍历文档集。用户的物品评论在某种程度上解释了用户的打分行为, 直观来看, 物品的属性可能会被评论中的一组词提到, 而这可能对应了某个主题。类似于用户实体有双重身份, 物品实体也有双重身份: 物品在打分数据中接受用户的打分, 同时物品在评论数据中得到用户的评论, 因此从来个数据源中学到的物品表示其实是同一实体的不同形态, 存在一种变换将两种表示联系起来。

例如如下的softmax变换（softmax transformation）则是一种可行的变换（变换示意见图 3.3。<sup>5</sup>）：

$$\theta_{i,f} = \frac{\exp(\kappa Q_{i,f})}{\sum_{f=1}^F \exp(\kappa Q_{i,f})} \quad (3.5)$$

这里参数  $\kappa$  控制了变换的峰度（peakiness），可随着其他参数一起更新。上述变换将从打分数据中学习到的实值向量  $Q_i \in \mathbb{R}^F$  与从评论数据中学习到的概率/随机向量  $\theta_i \in \Delta^F$  桥接了起来。

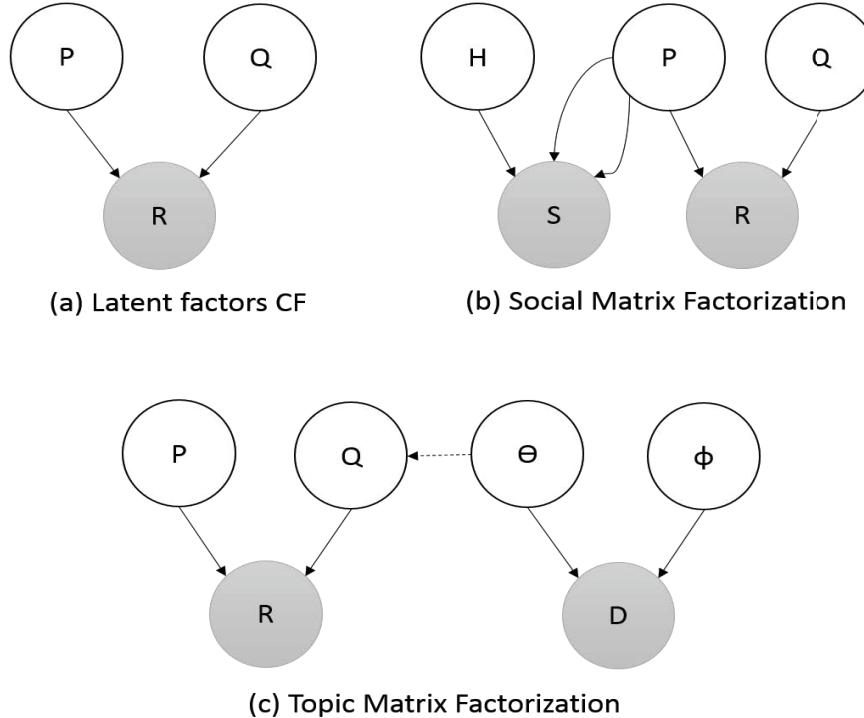


图 3.4：利用各自推荐数据源的三种组件中可观察数据矩阵和模型参数矩阵之间的依赖示例。(a) 利用打分信息的隐藏因子协同过滤法。(b) 集成社会关系的社会矩阵分解法。(c) 集成评论文本的主题矩阵分解，虚线是公式 3.5 所示的softmax变换。其中阴影结点 ( $R, S, D$ ) 表示可观察数据，其他结点表示模型参数 ( $P, Q, H, \theta, \phi$ )。

<sup>5</sup>该示意图修改自HFT [39]模型作者的幻灯片：[http://cseweb.ucsd.edu/~jmcauley/pdfs/slides\\_recsys13.pdf](http://cseweb.ucsd.edu/~jmcauley/pdfs/slides_recsys13.pdf)

上述介绍了三种利用各自推荐信息源的方法：利用打分信息的隐藏因子协同过滤法、集成社会关系的社会矩阵分解法和集成评论文本的主题矩阵分解，这些方法里可观察数据矩阵和模型参数矩阵之间的依赖关系示意于图 3.4。

### 3.3 多源信息融合的问题描述

本节给出多源信息融合问题的问题描述，这是下一节我们提出的综合方法要解决的问题。我们称下述问题为问题 1，该问题需要一个模型同时建模三种推荐数据源。

**Problem 1.** 加入社会关系信息和评论文本信息的打分预测。

**输入：**1) 一个打分矩阵  $R$ ; 2) 一个在用户之间建立的社会网络  $T$ ; 3) 用户对物品打分时同时伴随的评论语料集  $D$ ; 4) 用户集合  $\mathcal{P}$  中的一个用户  $u$ ; 和 5) 物品集合  $\mathcal{Q}$  中的物品  $i$ 。

**输出：**用户  $u$  对物品  $i$  的偏好预测打分  $\hat{R}_{u,i}$ ，这里  $u \in \mathcal{P}$ ,  $i \in \mathcal{Q}$ 。

在问题 1 中，为了预测一个未知的打分，有三种推荐数据源可以利用：打分信息，社会信息和文本信息。

### 3.4 一个多源信息融合模型

在章节 3.2 里我们介绍了如何三种方法是如何各自利用不同数据源的（见示意图 3.4），即利用打分信息的矩阵分解协同过滤法（公式 3.1，图 3.4(a)）、利用社会关系的社会矩阵分解法（公式 3.3，图 3.4(b)）、利用评论文本的主题矩阵分解法（公式 2.11、公式 3.5，图 3.4(c)）。在这些基础条件下，我们在上一个章节 3.3 定义了问题 1。现在我们给出问题 1 里能同时利用三种数据源的一个解决方法。

对于问题 1，主要的挑战在于如何高效的同时利用三种异构的数据源（heterogenous data sources），集成到一个综合模型中。我们的解决方法是基于协同过滤法，组合两个部分：社会矩阵分解法和主题矩阵分解法。通过将这两者学习到的潜在因子和隐藏主题对齐（alignment），就能达到同时利用三种异构信息。

对于融合模型的其中一个部分社会矩阵分解，我们采用全局-局部模型 LOCABAL (*local and global*) [57] 来将潜在的社会因子集成到协同过滤中。建模的优化目标是既能准确预测打分，又能捕捉社会上下文信息：<sup>6</sup>

<sup>6</sup>为了简洁起见，LOCABAL 模型中的全局上下文  $W_{u,i}$  信息在此处忽略，而是在公式 3.8、3.10 中引入介绍。

$$\begin{aligned} \min_{P,Q,H} & \sum_{R_{u,i} \neq 0} (R_{u,i} - \hat{R}_{u,i})^2 \\ & + \lambda_{\text{rel}} \sum_{T_{u,v} \neq 0} (S_{u,v} - P_u^T H P_v)^2 + \lambda_{\text{norm}} \Omega(\Theta) \end{aligned} \quad (3.6)$$

这里超参 $\lambda_{\text{rel}}$ 控制了来自社会信息的贡献度，模型的参数是 $\Theta = \{P, Q, H\}$ ，超参 $\lambda_{\text{norm}}$ 则控制了模型的复杂度以防止过拟合： $\Omega(\Theta) = \|P\|_F^2 + \|Q\|_F^2 + \|H\|_F^2$ 。

我们的融合模型选择LOCABAL，是因为它利用打分信息和社会信息的高效性：在原文章中，作者从理论的变换上和从经验的实验上都表明了该社会矩阵分解法优于SoRec [36]和SoReg [37]，而这是C-CTR-SMF2 [9]、CTRSR [60]和CTRSORec [46]所采用的社会矩阵分解组件。

对于融合模型的另一个部分主题矩阵分解，我们采用隐藏因子和隐藏主题模型HFT (*hidden factors and topics*) [39]来将潜在的物品特性因子集成到潜在主题因子中。建模的优化目标是既能准确预测打分，又能使得生成评论语料的似然最大化：

$$\begin{aligned} \min_{P,Q,\Phi} & \sum_{R_{u,i} \neq 0} (R_{u,i} - \hat{R}_{u,i})^2 \\ & - \lambda_{\text{rev}} \sum_{d=1}^N \sum_{n \in N_d} (\log \theta_{z_{d,n}} + \log \phi_{z_{d,n}, w_{d,n}}), \end{aligned} \quad (3.7)$$

这里超参 $\lambda_{\text{rev}}$ 控制了来自评论文本信息的贡献度，参数 $\Phi = \{\theta, \phi\}$ 。连接打分信息和评论信息的桥梁是物品特性因子 $Q_i$ 和物品主题因子 $\theta_i$ ，如公式 3.5 所示。

我们的融合模型选择HFT，是因为它利用打分信息和评论文本信息的高效性：在原文章中，作者从对打分建模的合理性以及真实数据集的实验验证上都表明了该主题矩阵分解法优于CTR [59]，而这是C-CTR-SMF2 [9]、CTRSR [60]和CTRSORec [46]所采用的主题矩阵分解组件。

基于协同过滤，组合上述两个高效组件LOCABAL和HFT，我们提出如下综合模型从而得到问题 1 的一个解：

$$\begin{aligned}
 \mathcal{L}(\Theta, \Phi, z, \kappa) \triangleq & \sum_{R_{u,i} \neq 0} W_{u,i} (R_{u,i} - \hat{R}_{u,i})^2 \\
 & - \lambda_{\text{rev}} \sum_{d=1}^N \sum_{n \in N_d} \left( \log \theta_{z_{d,n}} + \log \phi_{z_{d,n}, w_{d,n}} \right) \\
 & + \lambda_{\text{rel}} \sum_{T_{u,v} \neq 0} C_{u,v} (S_{u,v} - P_u^T H P_v)^2 + \lambda \Omega(\Theta) \quad (3.8)
 \end{aligned}$$

我们提出的高效综合模型的核心思想是基于协同过滤法，组合社会矩阵分解法和主题矩阵分解法，通过将这两者学习到的潜在因子和隐藏主题对齐(alignment)，就能达到同时利用三种异构信息。我们记该模型为MR3 (*a Model of Rating, Relation, and Review*)。

MR3模型中的打分权重 $W_{u,i}$ :

$$W_{u,i} = \frac{1}{1 + \log r_u}, \quad i \in N_u \quad (3.9)$$

这里 $r_u$ 是用户 $u$ 在整个社交网络中通过PageRank算法计算出的权威性排名，它表示了用户的全局社会上下文信息。此即第二章第2.3.2节中公式2.13里面 $W$ 的具体含义：如果一个用户的社会权威性高，那么从先验上可以认为该用户的打分具有权威性即当算法预测有错误时惩罚更严重；它不再将原始所有用户的打分权重都看做是相等的，即都是1。这里由于仅考虑用户的权威性，因此同一个用户所发出的对不同物品的打分权重都是相同的。当有额外的数据源能够反应出同一用户对不同物品的打分权重时，我们可以调节让 $W_{u,i}$ ,  $i \in N_u$ 有不同的取值。例如一个可能的场景是当物品是视频时，用户观看该视频的时间长度可作为用户对该物品的偏好权重量；让偏好权重正比于观看时间： $W_{u,i} \propto t_{u,i}/t_i$ ，这里 $t_i$ 是该视频总时间长度， $t_{u,i}$ 是该用户观看该视频的时间长度。

MR3模型中的 $C_{u,v}$ 的动机如下。我们虽然了一个社会矩阵分解的高效组件LOCABAL [57]，但是我们也借鉴另一个社会矩阵分解法SoRec [36]中信任值(*trust values*)的概念，以达到更紧密和深入的利用打分信息和社会信息：

$$C_{u,v} = \sqrt{\frac{d_v^-}{d_u^+ + d_v^-}}, \quad (3.10)$$

这里出度(outdegree)  $d_u^+$  表示用户 $u$ 所信任的总用户数，而入度(indegree)  $d_v^-$  表示信任该用户的其他总用户数。信任值的概念能捕捉用户的邻居图结构

(graph structure of neighbors), 从而表达社会影响力局部性 (social influence locality), 即用户的行为主要受到其直接/亲密好友的影响 [64]。

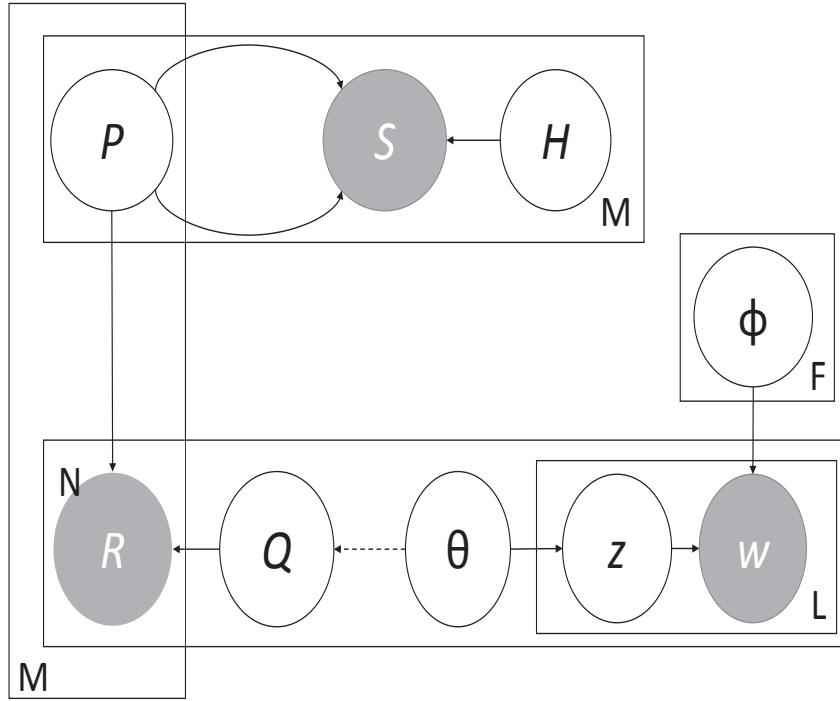


图 3.5: 融合模型MR3中的数据矩阵和参数矩阵的依赖示意图。阴影结点表示可观察数据。 $R$ : 打分矩阵,  $S$ : 社会相似矩阵,  $D$  (由词项 $w$ 呈现) : 评论文本语料的文档-词项矩阵。其他结点表示模型参数。 $P$ : 共享的用户偏好因子和用户社会因子,  $Q$ : 物品特性矩阵,  $H$ : 社会相关矩阵,  $\theta$ : 文档-主题分布,  $\phi$ : 主题-词分布。参数 $P$ 和 $S$ 的双线连接表示公式 3.3 中的项( $S - P^T HP$ ); 参数 $Q$ 和 $\theta$ 之间的虚线表示他们之间的固定softmax变换, 如公式 3.5 所示。

现在我们再来对MR3模型即公式 3.8 做一个解析。公式的右端第一项是打分数据的平方预测误差, 带上用户的全局上下文权重; 第二项是评论数据的语料负对数似然, 建模文本的生成概率; 第三项是局部社会上下文分解, 带上邻居用户的信任值权重; 最后一项是模型参数的弗罗贝尼乌斯范数, 控制模型复杂度防止过拟合。来自文本评论的贡献由超参 $\lambda_{\text{rev}}$ 控制, 来自社会信息的贡献由超参控制 $\lambda_{\text{rel}}$ 。打分数据源和社会关系数据源的联系是用户的共享特征空间 $P$ ; 打分数据和评论数据的联系时物品的潜在因子特征 $Q$ 和物品的隐藏主题特征 $\theta$ 之间的变换公式 (3.5), 且变换的峰度由参数 $\kappa$ 控制; 而三种数据源的联合建模则是基于协同过滤的集成社会因子和主题因子的对齐。MR3模型中可观察数据和

模型参数的依赖关系见示意图 3.5. 注意图中  $Q$  和  $\theta$  的虚线指示了他们之间的固定 softmax 变换，不存在任何分布依赖，因此该图不是描述一个贝叶斯生成概率图模型（Bayesian generative model），而仅仅是展示 MR3 模型中数据矩阵和参数矩阵的关系。

### 3.5 一个扩展社会矩阵分解法

我们将融合 MR3 中的社会矩阵分解组件分割开来，得到下述的扩展社会矩阵分解新模型 eSMF (extended Social MF)：

$$\begin{aligned} \mathcal{L}(\Theta) \triangleq & \sum_{R_{u,i} \neq 0} W_{u,i} (R_{u,i} - \hat{R}_{u,i})^2 \\ & + \lambda_{\text{rel}} \sum_{T_{u,v} \neq 0} C_{u,v} (S_{u,v} - P_u^T H P_v)^2 + \lambda \Omega(\Theta). \end{aligned} \quad (3.11)$$

总结来说，eSMF 模型是通过信任值 [36] 捕捉社会邻居图结构，从而能表示社会影响力局部性 [64]，扩展了 LOCABAL [57] 模型的社会矩阵分解法。

### 3.6 实验评估及结果分析

本章第 3.3 节引入了问题 1，并在第 3.4 节提出了一个可行的解决方法，即模型 MR3，它能效利用额外数据源评论文本和社会关系。

本章在两个较大规模的真实数据集上评估所提出模型的性能，并对实验结果和模型参数进行分析。在 3.6.2 节，我们评估 eSMF 模型，它是一个扩展社会矩阵分解法，通过信任值利用邻居图结构信息，继而捕捉社会影响力。然后在 3.6.3 节，我们评估模型 MR3，它是一个融合多源异构信息的推荐系统。为了进一步理解额外数据源和隐式反馈对推荐性能的影响，我们在 3.6.4 节分析来自额外数据源的贡献。最后在 3.6.5 节，我们分析模型的超参敏感性。

首先我们介绍评估数据集和评估协议。

#### 3.6.1 数据集和评估指标

本节介绍两个用于评估推荐性能的真实世界里的数据集，包括对它们的预处理和数据集的简单统计量。然后我们给出推荐性能的评估指标和评估协议。

## 数据集及其统计量

我们在两个数据集上进行评估：Epinions和Ciao。<sup>7</sup> 他们都是知识共享和评论网站，用户可以对物品进行打分，用户之间可以建立信任关系，用户可以对产品表达他们的观点并写下评论文本（见图3.1）。我们先消除停用词<sup>8</sup>然后按词频选择前  $L = 8000$ 个词作为词表（vocabulary）。为了降低噪音和错误，我们仅保留至少打过4个物品的用户，并消除了那些仅有1或者2个得分的物品。打分矩阵中物品的索引与评论语料中文档-词项矩阵中的文档索引是对齐的，也即我们把属于一个物品的所有用户评论组合起来得到一个该物品的评论。最终数据集的统计量见表3.2。

可以发现，两个数据集上的打分都是很稀疏的，且Epinions数据集上文档的平均长度很短。Ciao数据集上每个物品所含评论词的平均个数是Epinions的42倍，且Ciao的社会关系密度是Epinions的12倍。因此Ciao数据集上所包含的额外外数据源：文本评论和社会信息更丰富也更高质量，可以期望对最终推荐性能的影响也更到。这个观察与我们在第3.6.4节得到的定量评估相一致。

表 3.2: 数据集统计量

统计量	Epinions	Ciao	小记
用户数	49,454	7,340	56,794
物品数	74,154	22,472	96,626
打分/评论数	790,940	183,974	974,914
社会关系数	434,680	112,942	547,622
词数	2,246,837	28,874,000	31,120,837
打分密度	0.00022	0.0011	N/A
社会密度	0.00018	0.0021	N/A
每个物品的平均词数	30.3	1284.9	N/A
每个用户的平均关系数	8.78	15.38	N/A

## 评估协议和评估指标

有三种超参需要设定。对于优化相关的参数，我们使用带动量的微批量梯度下降法（mini-batch gradient descent method with momentum）求解优化目标。参考经验规则和之前文献（如PMF模型[42]），动量=0.8，批大小=训练集大小/

<sup>7</sup> <http://www.public.asu.edu/~jtang20/>

<sup>8</sup> <http://www.ranks.nl/stopwords>

批次数，学习率= 0.0007。在每轮之前都对训练数据进行随机混合（shuffle）。对于正则相关的参数，利用范数正则避免过拟合。参考经验规则和之前文献（如TrustSVD模型 [22]），范数正则 $\lambda = 0.5$ 。对于模型相关的参数，有三个。隐藏因子维度 $F$ ，控制来自评论文本数据的贡献 $\lambda_{\text{rev}}$ ，控制来自社会关系数据的贡献 $\lambda_{\text{rel}}$ ，它们的设置细节在第 3.6.5 节。

我们随机选择 $x\%$ 作为训练集，在剩余的 $1 - x\%$ 作为测试集， $x \in \{20, 50, 80, 90\}$ ；独立重复运行5次，报告平均结果。每个方法都遵循这样的评估协议，这类似于5-折交叉验证（5-fold CV）。

我们关注于打分预测任务，因此一个自然的评估指标是根均方误差RMSE（*root-mean-square error*）

$$RMSE_{\mathcal{T}} = \sqrt{\sum_{(u,i) \in \mathcal{T}} (R_{u,i} - \hat{R}_{u,i})^2 / |\mathcal{T}|}, \quad (3.12)$$

其中 $\mathcal{T}$  是测试集。

与平均绝对误差评估指标MAE（*mean absolute error*）0 [23]比较起来，RMSE对有较大错误的预测惩罚也更严重。例如对RMSE，有2个点的预测错误会导致总惩罚增加4；而对MAE则惩罚只增加2。更小的RMSE和MAE表示有更好的预测性能，一个小的RMSE提升会带来推荐质量的巨大影响，这也是Netflix竞赛的评估指标 [5, 32]。

要注意，对于从MR3模型扩展而来的MR3++模型，他的打分预测计算要从原来的 $\hat{R}_{u,i}$ （见公式 3.2）替换为 $\hat{R}_{u,i}^*$ （见公式 4.1），以考虑隐式反馈的影响。

### 3.6.2 社会推荐方法的比较

本节我们将eSMF模型（见章节 3.5，或者公式 3.11）和一个标准的社会矩阵分解法LOCABAL [57]进行比较，以展示利用邻居图结构信息捕捉社会影响力局部性的益处。评估的动机有两点，一个是展示将打分数据和社会关系数据更加紧密的利用起来能提升社会推荐系统的打分预测结果，另一个是该模型成为了融合模型MR3的高效组件。

我们用网格搜索法确定控制来自社会关系信息贡献度的超参 $\lambda_{\text{rel}}$ 。对于LOCABAL 和eSMF，他们都是 $\lambda_{\text{rel}} = 0.1$ 。参数 $\Theta = \{P, Q, H\}$  按正太分布 $\mathcal{N}(0, 0.01)$  随机初始化。

评估结果展示在图 3.6中，横坐标是变化的训练集百分比 $x = \{20, 30, 40, 50, 60, 70, 80, 90, 99\}$ 。从实验结果可得到如下的观察：

- 更紧凑的利用打分信息和社会信息，即通过信任值利用邻居图结构捕捉社会影响力，能在两个数据集上都能提升RMSE预测性能。例如，eSMF模型相比于LOCABAL模型，在Epinions训练集占比分别为20%, 50%, and 70%时分别有1.18%, 0.89%, 和0.72% 的相对RMSE性能提升。

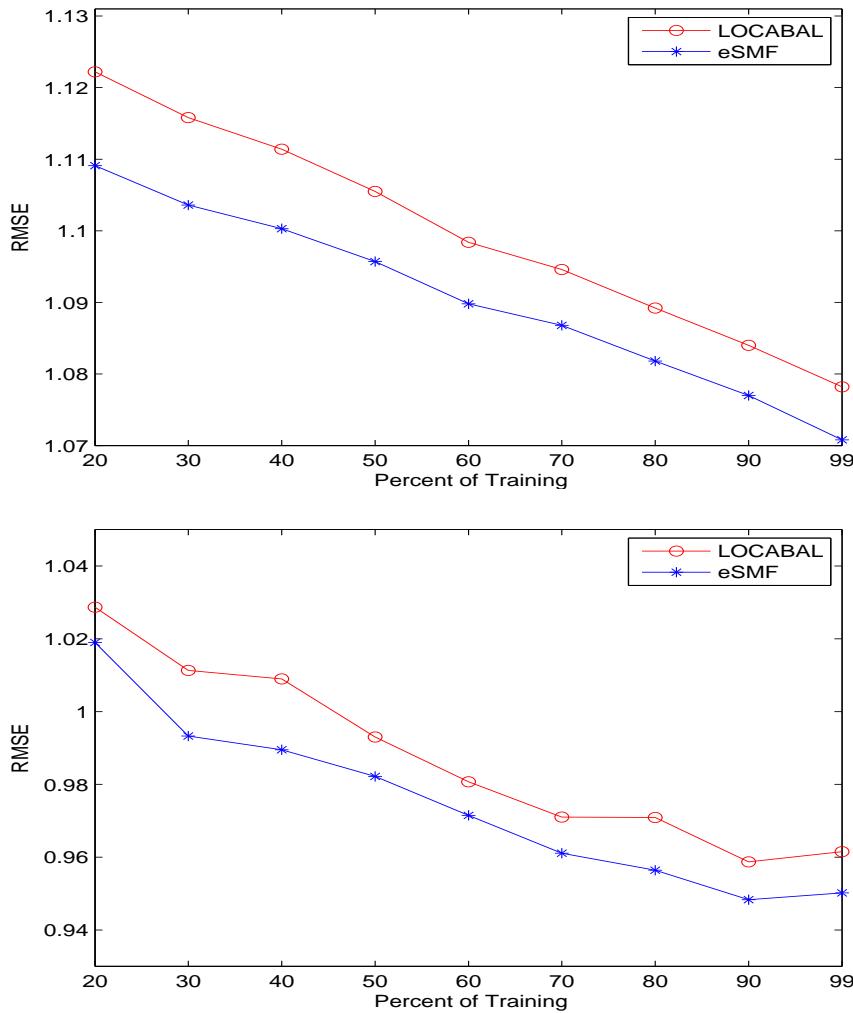


图 3.6: eSMF 模型与一个标准社会矩阵分解模型LOCABAL的比较。上图: Epinions; 下图: Ciao.

### 3.6.3 评估多源信息融合模型

本节评估模型MR3。实验结果将展示MR3模型利用两个高效组件的性能。

我们首先评估MR3模型（见章节 3.4 公式 3.8），以展示高效融合评论数据和社会信息对推荐性能的影响。比较的推荐方法如下：

**Mean.** 这个方法用全体训练集中的打分均值来预测，即公式 3.2 中的项  $\mu$ 。该方法是在 RMSE 度量下的最佳常量预测。

**PMF.** 这个方法分解打分矩阵学习用户和物品特征向量 [42]，如公式 3.1 所示。它是隐藏因子协同过滤法的代表方法，见图 3.4(a)。它只利用的打分数据源。

**LOCABAL.** 这个方法基于矩阵分解，同时利用社会局部和全局上下文 [57]，如公式 3.6 所示。它是社会矩阵分解法的代表方法，见图 3.4(b)。它只利用了打分数据源和社会信息。

**HFT.** 这个方法组合从打分数据中学到的潜在因子和从评论中学到的隐藏主题 [39]，如公式 3.7 所示。它是主题矩阵分解法的代表方法，见图 3.4(c)。它只利用了打分数据源和评论文本信息。

表 3.3: 融合模型 MR3 与不同方法的 RMSE 比较。 $(F = 10)$

Datasets	Training	Methods					Improvement of MR3 vs.		
		Mean	PMF	HFT	LOCABAL	MR3	PMF	HFT	LOCABAL
Epinions	20%	1.2265	1.2001	1.1857	1.1222	1.1051	8.60%	7.29%	1.55%
	50%	1.2239	1.1604	1.1323	1.1055	1.0809	7.35%	4.76%	2.28%
	80%	1.2225	1.1502	1.0960	1.0892	1.0648	8.02%	2.93%	2.29%
	90%	1.2187	1.1484	1.0867	1.0840	1.0634	7.99%	2.19%	1.94%
Ciao	20%	1.1095	1.0877	1.0439	1.0287	1.0142	7.25%	2.93%	1.43%
	50%	1.0964	1.0536	1.0379	0.9930	0.9740	8.17%	6.56%	1.95%
	80%	1.0899	1.0418	0.9958	0.9709	0.9521	9.42%	4.59%	1.97%
	90%	1.0841	1.0391	0.9644	0.9587	0.9451	9.95%	2.04%	1.44%
Average							8.34%	4.16%	1.86%

我们使用 PMF 模型<sup>9</sup> 和 HFT<sup>10</sup> 模型作者公布的代码。通过网格搜索法确定超参如下：对于 HFT 模型， $\lambda_{\text{rev}} = 0.1$ ，对于 MR3 模型， $\lambda_{\text{rel}} = 0.001$ ， $\lambda_{\text{rev}} = 0.05$ 。关于 MR3 模型的超参敏感性分析详见下面的第 3.6.5 节。

RMSE 比较结果见表 3.3<sup>11</sup>，在两个数据集上变化训练集百分比  $x = \{20, 50, 80, 90\}$ 。从实验结果可得到如下的观察：

- 两个数据集上，以 RMSE 为度量，在打分数据之上利用社会关系和产品评论都有助于提升推荐性能。例如，HFT 和 LOCABAL 相比于 PMF，

<sup>9</sup><http://www.cs.toronto.edu/~rsalakhu/>

<sup>10</sup><http://cseweb.ucsd.edu/~jmcauley/>

<sup>11</sup> 标准偏差都小于  $10^{-5}$

在Epinions训练集占比80%上，分别获得了4.95% 和5.60%相对RMSE性能提升。

- 我们的融合模型MR3在两个数据集的不同设置上，都取得了最优的RMSE性能。MR3相比于HFT和LOCABAL方法，在Epinions和Ciao上，分别获得了平均0.0466 和0.0217的绝对RMSE提升。主要原因是MR3同时高效地利用了所有三种数据源。来自额外数据源的贡献分析将在第 3.6.4 节做进一步的详细分析。

#### 3.6.4 额外数据源对推荐性能的贡献

我们已经评估了模型MR3的推荐性能，本章节对其进一步分析，以理解额外数据源（评论文本和社会关系）对推荐性能的影响。

##### 来自社交关系和评论文本的贡献

我们已经通过评估模型MR3展示了融合社会关系和评论文本信息的有效性，我们现在从融合模型MR3中依次消去额外数据源从而考察它们推推荐性能的影响。

**MR3\content:** 从模型MR3即公式 3.8 中通过设置  $\lambda_{rev} = 0$  从而消去来自评论文本信息的影响，这等价于eSMF模型，如公式 3.11 所示。

**MR3\social:** 从模型MR3即公式 3.8 中通过设置  $\lambda_{rel} = 0$  从而消去来自社会关系信息的影响，这等价于HFT模型，如公式 3.7 所示。

**MR3\content\social:** 从模型MR3即公式 3.8 中通过设置  $\lambda_{rev} = 0$  和  $\lambda_{rel} = 0$  从而同时消去来自评论文本信息和社会关系信息的影响，这等价于PMF模型，如公式 3.1 所示。

融合模型MR3与它的三个组件的预测性能展示在图 3.7 中。

从图中可以看到，消去评论数据或者社会信息都会造成推荐性能的下降，且去掉社会信息对预测性能的影响比去掉评论信息的影响更大。具体来看，*MR3\content*, *MR3\social*, 和*MR3\content\social*在Epinions数据集上，与MR3模型比起来，分别平均下降了1.19%, 4.29%，和7.99%相对RMSE性能，说明了评论文本和社会关系对融合的推荐系统包含了必要的信息。

### 进一步分析来自辅助数据源的贡献

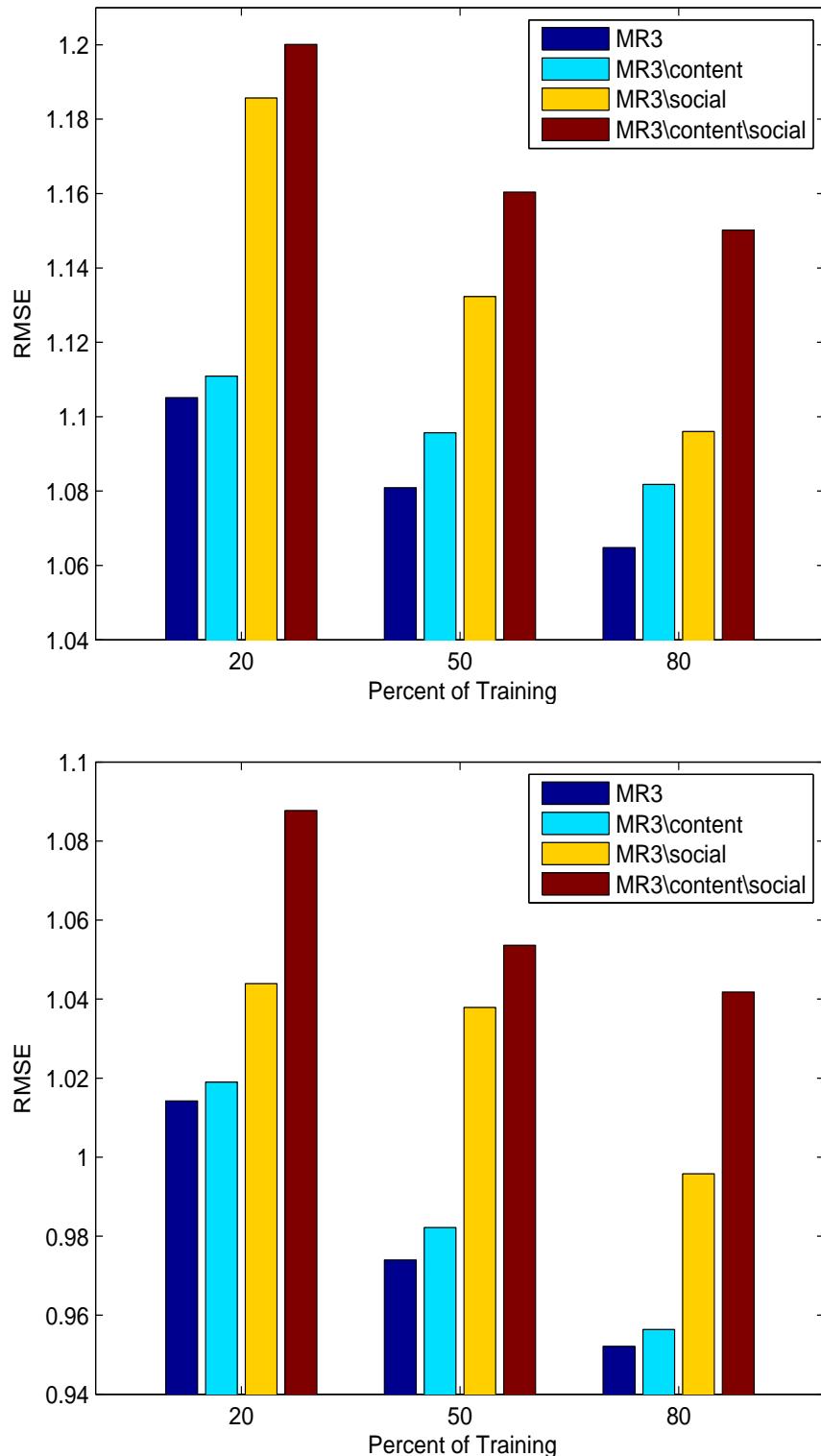


图 3.7: 融合模型MR3与它的三个组件的预测性能。上图: Epinions; 下图: Ciao.

我们已经展示了从融合模型MR3中消去所有辅助数据源 $MR3\backslash content\backslash social$ 会导致总相对RMSE下降7.99%。直觉来看，我们想知道额外数据源是如何帮助MR3模型提升预测性能的。我们从评论数据和社会信息的丰富程度来考察这个问题。具体来说，我们先收集PMF模型预测最差而MR3却预测准确的( $user, item$ )对：将两者的间隔设置为1.例如PMF模型对( $u, i$ )的预测误差是 $e_1 = |\hat{R}_{u,i}^{PMF} - R_{u,i}|$ 而MR3模型的预测误差是 $e_2 = |\hat{R}_{u,i}^{MR3} - R_{u,i}|$ ，如果 $(e_1 - e_2) >= 1$ 那么我们就保留这样的( $u, i$ )对。然后我们计算这些用户 $user$ 涉及的平均社会关系密度，以及这些物品 $item$ 涉及的平均评论长度，用这样的统计量来度量额外数据源的信息质量和丰富程度。

在80%的Epinions训练数据集上，我们收集到了满足上述条件的4,382对( $user, item$ )，其中含用户3,627人和物品2,875个。他们分别覆盖了2.80%，7.33%，和3.88%的总打分数、总用户数和总物品数。用户之间的平均社会关系数目是14.18，物品评论的平均词数1129.23。注意到这两个统计量远大于全部数据集上的相应统计量（分别是8.78和30.3，见表3.2），从而直观上验证了额外数据源对融合模型推荐性能的某种程度的显著影响。

### 3.6.5 多源信息融合模型的超参分析

本节分析融合模型MR3对超参的敏感性，基本方法是控制变量法：即每次变化一个参数而固定其他参数，考察模型的预测性能的相应变化。

融合模型MR3包含三个重要的超参：一个是控制来自社会关系信息的贡献度 $\lambda_{rel}$ ，一个是控制来自评论文本信息的贡献度 $\lambda_{rev}$ ，一个是确定隐藏因子的维度 $F$ 。我们通过每次变化一个参数而固定其他参数的方法来分析MR3的超参敏感性。

首先我们固定隐藏因子的维度 $F = 10$ ，考察与评论信息相关的参数 $\lambda_{rev}$ 和与社会信息相关的参数 $\lambda_{rel}$ 对模型MR3的性能影响。分析实验结果如图3.8所示，我们可以看到：不管哪个参数置为0，也即不论去掉哪种辅助数据源，都会造成预测性能的下降（当两个参数都取值为0时，RMSE=1.1502）。还可以观察到，模型MR3在当两个参数都取值较小时（例如从0.0001到0.1）是处于较稳定的状态，因此我们分别为它们取定默认值 $\lambda_{rel} = 0.001$ ， $\lambda_{rev} = 0.05$ 。

接下来我们固定 $\lambda_{rel} = 0.001$ ， $\lambda_{rev} = 0.05$ ，考察隐藏因子维度 $F$ 对模型MR3的性能影响。我们设置 $F = \{5, 10, 15, 20, 30, 50, 70, 100\}$ ，并且训练集占比分别为20%，50%，80%。分析实验结果如图3.9所示，我们可以看到：模型MR3对隐藏因子维度具有较稳定的性能，因此我们取定默认值 $F = 10$ 。

### 3.7 小结

本章首次提出了一个能同时利用三种异构推荐数据源的高效融合模型MR3。与已有的融合模型不同，我们所采用的两个组件都是业已验证的高效方法。同时，我们还得到了一个高效的扩展社会推荐模型 eSMF。我们提出的这两个模型的高效性在两个较大规模的数据上得到了验证，说明利用邻居图结构和加入额外数据源能进一步提升推荐性能。

为了深挖有限的推荐数据源，接下来第 [四章](#) 我们将探讨如何在融合模型MR3的基础上进行扩展，以挖掘来自打分数据源中的隐式反馈信息。本章的融合模型和下一章的挖掘隐式反馈的扩展融合模型，他们的模型参数学习过程是相似的，因此将其归纳统一在第 [五章](#)。

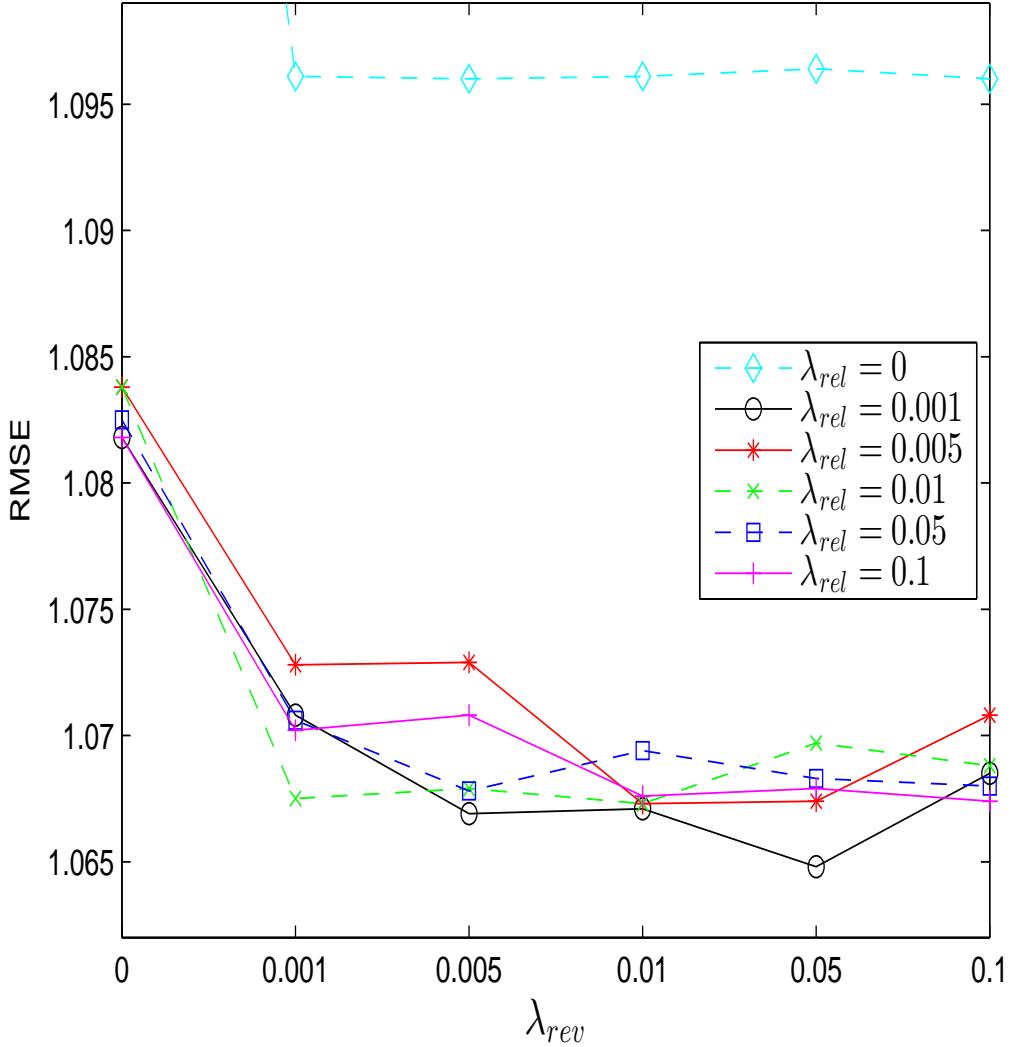


图 3.8: 融合模型MR3随超参社会正则 $\lambda_{rel}$ 和文本正则 $\lambda_{rev}$ 变化的预测性能。社会正则和文本正则两个参数的变化范围都是{0, 0.001, 0.005, 0.01, 0.05, 0.1}，当它们都是0时的RMSE值是1.1502。固定隐藏因子维度 $F = 10$ ，训练集占比80%。评估数据集：Epinions。

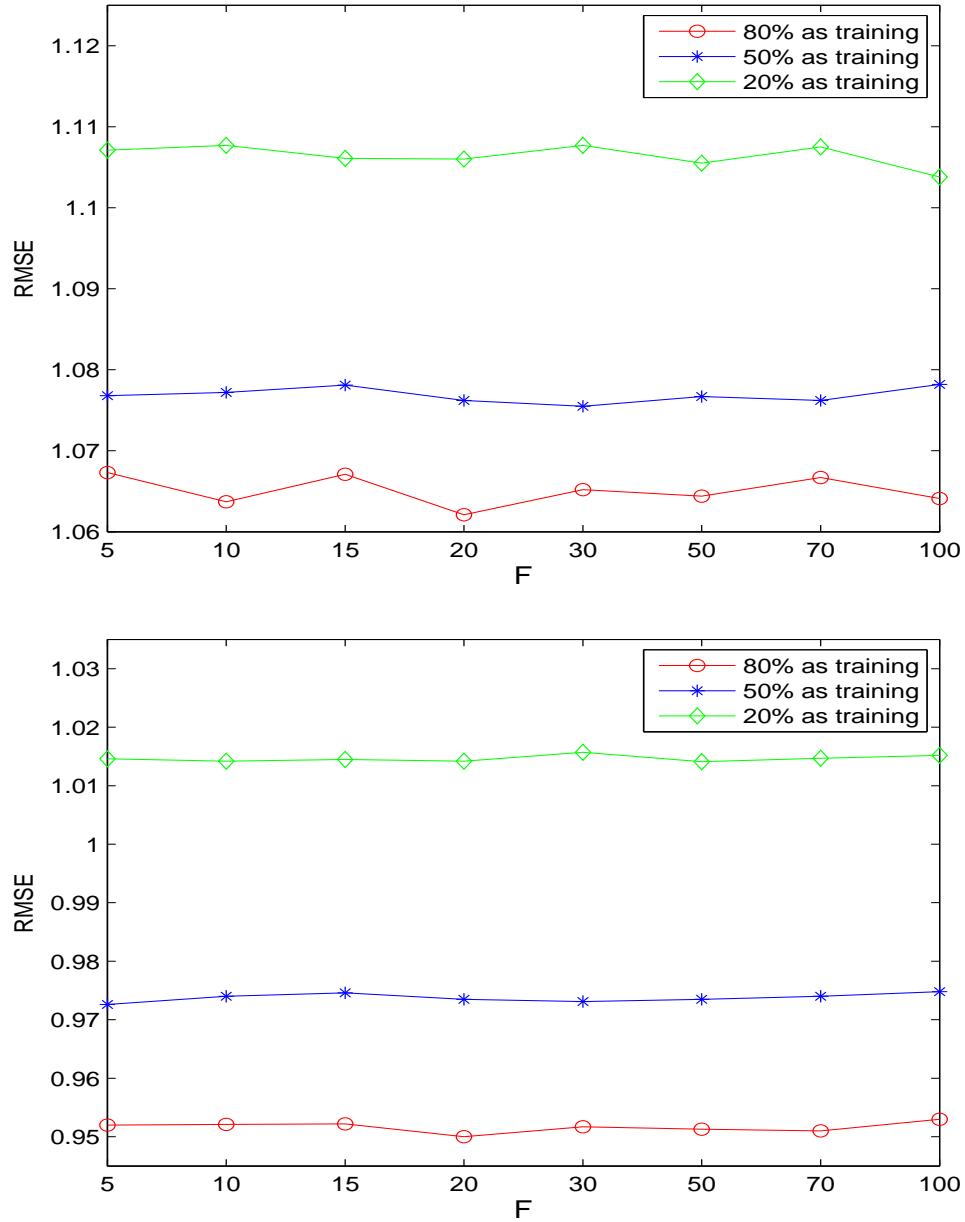


图 3.9: 融合模型MR3随超参隐藏因子维度 $F$ 变化的预测性能。固定社会正则 $\lambda_{\text{rel}} = 0.001$ , 文本正则 $\lambda_{\text{rev}} = 0.05$ 。上图: Epinions; 下图: Ciao.

## 第四章 基于隐式反馈挖掘的扩展融合推荐系统

### 4.1 引言

克服协同过滤中数据稀疏性问题的方法除了加入额外的数据源，例如评论文本和社会关系信息，另一类方法是深度挖掘已有的打分数据源，例如来自于打分数据和社会信息的隐式反馈。一种利用隐式反馈的方式是将其集成到基本的协同过滤中，同时学习显式打分和隐式打分的隐藏因子，例如SVD++和PMF（Probabilistic matrix factorization）模型 [22, 32, 42, 47]。另外一种是从原始打分中推导出偏好-置信对（preference-confidence pairs）[25]：可观察的打分被视为带有置信度的正、负偏好指示器，最终的优化目标变为在满打分矩阵上的遍历，而不是像SVD++和PMF模型中只遍历可观察的部分打分矩阵 [35]。然而这些算法只着眼于深度挖掘有限的数据源，而没有考虑到融合更多的额外数据源，例如物品的评论信息。

为了能同时利用多源异构信息和深度挖掘打分数据，本章首次提出了一个基于隐式反馈挖掘的扩展融合推荐系统。该模型的核心思想是在多源异构信息融合模型MR3（见公式 3.8，或图 3.5）的基础上，将原来的分解打分矩阵的协同过滤增强为能学习隐式特征的矩阵分解法。具体而言，在从打分数据上学习用户偏好因子和物品特性因子时，同时学习一个额外的隐式特征矩阵，该隐式特征矩阵捕捉了来自打分数据的隐式反馈信息。

本章后续部分组织如下：首先 4.2 节给出了挖掘隐式反馈的动机，然后 4.3 节对挖掘隐式反馈的扩展融合推荐系统这一问题进行描述，接着 4.4 和 4.5 节介绍所提出的挖掘隐式反馈的扩展融合模型和其中涉及的隐式特征矩阵，第 4.6 节对所提模型进行实验评估和结果分析，最后 4.7 节是总结。

### 4.2 挖掘隐式反馈的动机

对于物品的打分信息，它们表明了用户如何对物品打一个分，例如从1到5星的显式打分以表达用户的偏好程度。与显式打分伴随的是隐式反馈，他们表明了用户对哪些物品表达了偏好态度，先不用知道该偏好的程度。具体来说，一个二元矩阵 $R^b$ 可以从原始打分矩阵中构造出来，其中的元素取值1当且仅当对应的原始矩阵中的元素是已观察到的，即 $R_{u,i}^b = 1 \text{ iff. } R_{u,i} \neq 0$ 。从先验上来看，当两个用户的打分物品集相同时，我们可推断这两个用户的偏好

要比打分物品集不相同的其他用户的偏好更可能相似。举个例子来说，当用户  $u$  和用户  $v_1$  的打分物品（电影/视频）集都是 {指环王，冰与火之歌，霍比特人，哈利波特}，而用户  $v_2$  的打分物品集是 {来自星星的你，太阳的后裔，失恋33天}，我们可推断用户  $u$  与  $v_1$  的偏好相似度要比  $v_2$  更高。隐式反馈信息的另一个好处是能缓解协同过滤的冷启动性问题，因为隐式特征矩阵能将所有用户联系起来，从而使得从热启动用户中更新的隐式特征矩阵能促进冷启动用户的打分预测准确性。举个例子来说，当用户  $u$  的打分物品（电影/视频）集是 {指环王}，而用户  $v$  的打分物品集是 {指环王，冰与火之歌，霍比特人，哈利波特}，那么从用户  $v$  中更新得到的对应物品 {冰与火之歌，霍比特人，哈利波特} 的隐式特征矩阵能促进用户  $u$  对着三者的打分预测准确性。从真实数据集上来看，例如Netflix打分预测任务，挖掘隐式反馈的SVD++模型比基本的协同过滤法有更好的性能，如图 4.1 所示。<sup>1</sup>

因此鉴于隐式反馈信息的有效性，我们期待将其集成到融合多源异构信息的模型中，从而得到一个挖掘隐式反馈的扩展融合模型。下面我们描述该问题并且给出一个解决方法。

### 4.3 隐式反馈挖掘的扩展融合推荐系统问题描述

现在我们给出扩展上一章融合模型MR3（见公式 3.8，或图 3.5）从而能挖掘隐式反馈信息的扩展融合推荐系统之问题描述。我们称其为问题 2，它是在问题 1 上要求利用来自打分数据源的隐式反馈信息。

**Problem 2.** 带打分隐式反馈的问题 1。

**输入:** 1) 一个打分矩阵  $R$  及其隐式打分矩阵  $R^b$ ；2) 一个在用户之间建立的社会网络  $T$ ；3) 用户对物品打分时同时伴随的评论语料集  $D$ ；4) 用户集合  $\mathcal{P}$  中的一个用户  $u$ ；和5) 物品集合  $\mathcal{Q}$  中的物品  $i$ 。

**输出:** 用户  $u$  对物品  $i$  的偏好预测打分  $\hat{R}_{u,i}$ ，这里  $u \in \mathcal{P}$ ,  $i \in \mathcal{Q}$ 。

在问题 2 中，为了预测一个未知的打分，除了有额外数据源社会信息和文本信息可以利用，还要利用来自显式打分矩阵的隐式反馈信息。

---

<sup>1</sup> 图片截取自 Alex Smola 幻灯片：[http://alex.smola.org/teaching/berkeley2012/slides/8\\_Recommender.pdf](http://alex.smola.org/teaching/berkeley2012/slides/8_Recommender.pdf)

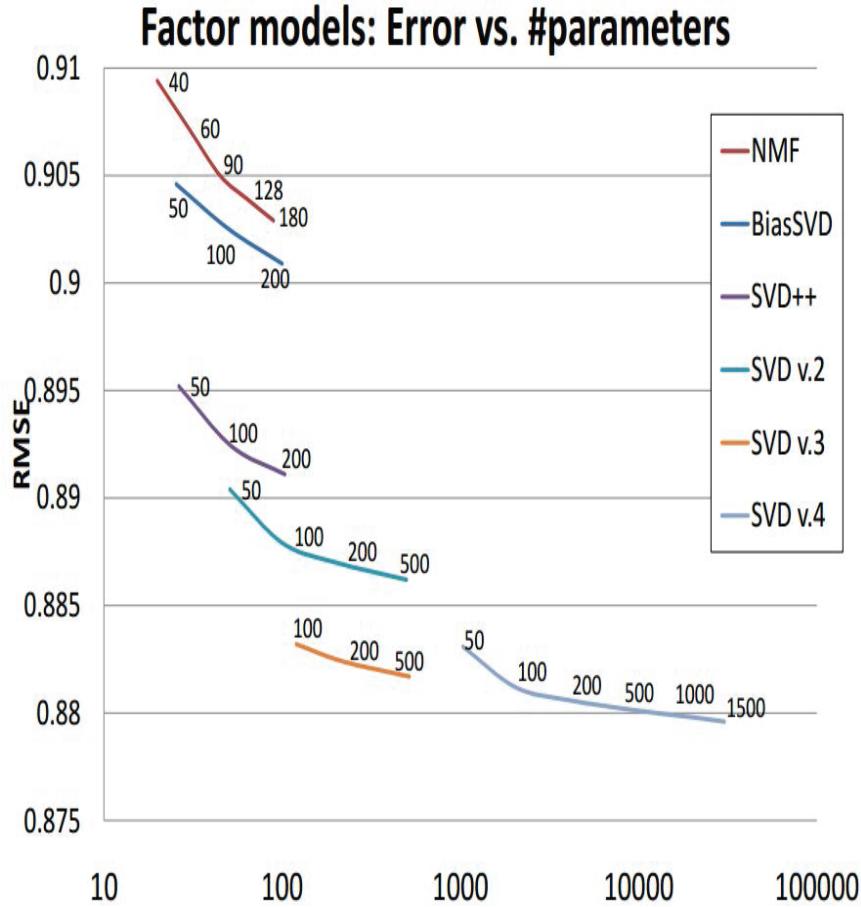


图 4.1: 考虑隐式反馈信息的协同过滤比基本矩阵分解法有更好的推荐性能。横坐标是参数个数，纵坐标是打分预测误差，图中不同的曲线表示不同的协同过滤方法。可以看到，考虑隐式反馈信息的SVD++模型比基本矩阵分解法BiasSVD有更好的推荐性能。

#### 4.4 一个挖掘隐式反馈的扩展融合模型

在第 3.4 章节我们给出了融合多源异构信息的一个模型，即问题 1 的一个解。该解决方法即是我们在提出的MR3模型（见公式 3.8，或图 3.5），通过在基本的协同过滤方法上组合高效的主题矩阵分解法和社会矩阵分解法，它能高效利用三种推荐信息源。为了进一步挖掘隐式反馈信息以提高推荐性能，我们扩展MR3模型，得到一个问题 2 的解。

深度挖掘打分数据的核心思想是从隐式打分矩阵  $R^b$  中学习一个额外的隐式特征矩阵  $Y \in \mathbb{R}^{F \times N}$ 。基本协同过滤法中，由于用户打分数据的稀疏性，即冷启动问题，导致该用户的特征向量被更新次数太少，从而会接近于初始化时的先验均值，即用户平均特征，最终该用户的预测打分就趋近于物品的平均得分。

有了隐式特征，那么用户打过分的物品将会对这些用户的未知打分起到先验性的影响。具体来说，用户  $u$  对物品  $i$  的总偏好可分解为两部分：一部分表达用户的固有偏好（intrinsic preference），由显式特征  $P_u$  反映；另一部分是已打过物品的隐式影响（implicit influence of the rated items），由隐式特征  $Y_j$  反映。这些思想在Constrained PMF [42]和SVD++ [32]模型中得到了具体化。

$$\hat{R}_{u,i}^* = \underbrace{P_u^T Q_i}_{\text{intrinsic preference}} + \underbrace{\left(|N_u|^{-\frac{1}{\alpha}} \sum_{j \in N_u} Y_j\right)^T Q_i}_{\text{implicit influence}} + \mu + b_u + b_i \quad (4.1)$$

SVD++模型中  $\alpha = 2$ ，而Constrained PMF模型中  $\alpha = 1$ 。这里  $N_u$  是用户  $u$  的已打分物品集合，即  $N_u = \{j : R_{u,j}^b = 1\}$ ；  $Y_j$  是物品  $j$  的隐式特征向量。从上述公式中可以清晰的看到，如果两个用户  $u$  和  $v$  的已打分物品集合相同  $N_u = N_v$ ，那么从先验上来看他们有更可能相似的偏好  $\sum_{j \in N_u} Y_j$ ；而来自于他们显式偏好的特征  $P_u, P_v$  可以认为是该先验特征均值的各自偏移（offset）。

有了上述挖掘隐式反馈的组件，并结合上一章提出的多源异构信息融合模型，我们可以得到一个挖掘隐式反馈的扩展融合模型：

$$\begin{aligned} \mathcal{L}(\Theta^*, \Phi, z, \kappa) \triangleq & \sum_{R_{u,i} \neq 0} W_{u,i} (R_{u,i} - \hat{R}_{u,i}^*)^2 \\ & - \lambda_{\text{rev}} \sum_{d=1}^N \sum_{n \in N_d} \left( \log \theta_{z_{d,n}} + \log \phi_{z_{d,n}, w_{d,n}} \right) \\ & + \lambda_{\text{rel}} \sum_{T_{u,v} \neq 0} C_{u,v} (S_{u,v} - P_u^T H P_v)^2 + \lambda \Omega(\Theta^*), \end{aligned} \quad (4.2)$$

其中范数正则  $\Omega(\Theta^*) = \|P\|_F^2 + \|Q\|_F^2 + \|Y\|_F^2 + \|H\|_F^2$ 。从上述公式可以看出，该融合模型包含两个组件：一个是加入额外数据源（评论文本和社会关系），另一个是从打分数据中挖掘隐式反馈。我们将该扩展融合模型记为MR3++，其中数据矩阵和参数矩阵的依赖见图 4.2。与融合模型比起来，多了一个挖掘隐式反馈信息的隐式特征矩阵。

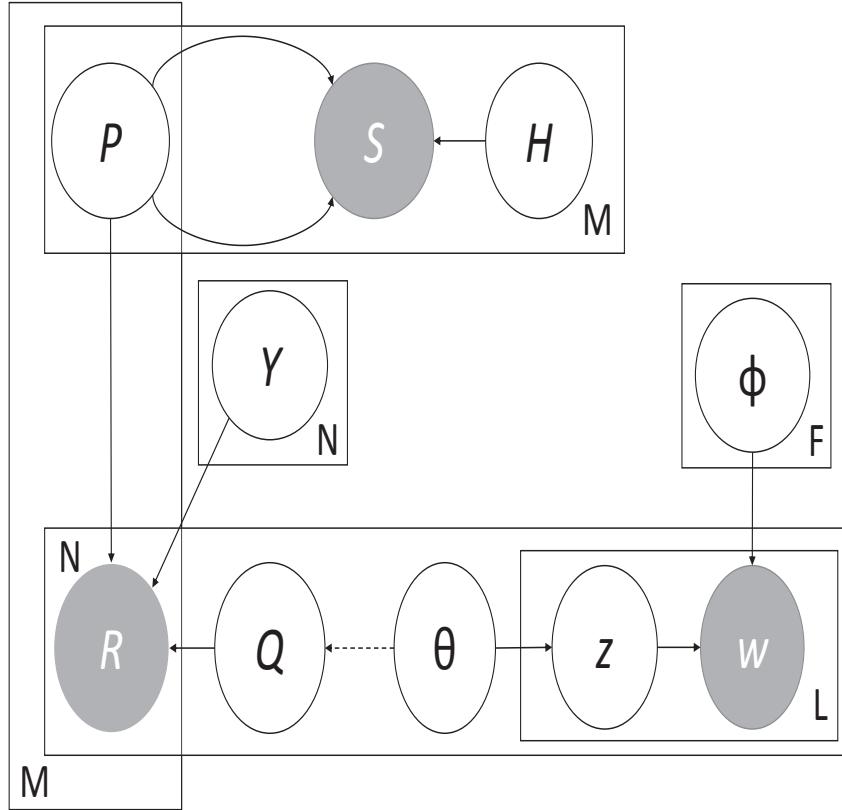


图 4.2: 挖掘隐式反馈的扩展多源信息融合模型。阴影结点表示可观察数据，其他结点表示模型学习参数。新引进隐式特征矩阵  $Y \in \mathbb{R}^{F \times N}$  以考虑来自自己打分物品的隐式影响：用户对物品的总偏好可以分解为两部分，一部分来自从显式打分数据中学习到的固有偏好  $P$ ，一部分来自自己打过物品的隐式影响  $Y$ 。

## 4.5 关于隐式特征矩阵的进一步说明

据我们所知，学习隐式特征  $Y$  的思想来源于KDD Cup 2007时提出的NSVD模型 [45]，初衷是代替用户特征矩阵  $P$  从而能降低模型参数个数继而控制模型复杂性。NSVD模型参数复杂性从  $\mathcal{O}(MK + NK)$  变为  $\mathcal{O}(NK)$ ，使得其不再依赖于用户的规模  $M$ 。

除了能捕捉隐式反馈，隐式特征矩阵  $Y$  还能用于学习物品之间的相似性  $sim(i, j) = Y_j^T Q_i$ ，结合基于记忆的协同过滤法（物品-物品近邻法），被预测的打分可以类似计算为  $R_{u,i} = \sum_{j \in N_u} sim(i, j) + b_u + b_i$ 。这种方法在模型FISM [29]中被修改为  $R_{u,i} = \sum_{j \in N_u \setminus \{i\}} sim(i, j) + b_u + b_i$ ，以消除在相似估计时已有打分信息的影响。

为了挖掘来自显式打分的隐式反馈信息，我们将SVD++模型与MR3融合模

型组合，得到扩展模型MR3++：同时学习物品-物品相似性  $sim(i, j) = Y_j^T Q_i$  和用户-物品相似性  $sim(u, i) = P_u^T Q_i$ ，因而预测打分计算为  $\hat{R}_{u,i}^* = sim(u, i) + |N_u|^{-\frac{1}{2}} \sum_{j \in N_u \setminus \{i\}} sim(i, j) + \mu + b_u + b_i$ ，这正如公式 4.1 所表达的。类似的，如果集成FISM模型，则可以得到如下打分预测公式

$$\hat{R}_{u,i}^* = sim(u, i) + (|N_u| - 1)^{-\frac{1}{\alpha}} \sum_{j \in N_u \setminus \{i\}} sim(i, j) + \mu + b_u + b_i$$

## 4.6 实验评估及结果分析

本章第 4.3 节引入了问题2，并在第 4.4 节提出了一个可行的解决方法，即模型MR3++，它能高效利用额外数据源评论文本和社会关系并在此基础上挖掘隐式打分反馈信息。

本章在两个较大规模的真实数据集上评估所提出模型的性能，并对实验结果和模型参数进行分析。在 4.6.1 节，我们评估模型MR3++，它是一个挖掘隐式反馈的扩展融合推荐系统。为了进一步理解额外数据源和隐式反馈对推荐性能的影响，我们在 4.6.2 节分析来自额外数据源的贡献。最后在 4.6.3 节，我们分析模型的超敏感性。

数据集和评估指标与上一章评估MR3时相同，参加章节 3.6.1。

### 4.6.1 评估挖掘隐式反馈的扩展融合模型

本节评估模型MR3++。实验结果将展示MR3++模型挖掘隐式反馈的效果。

我们评估MR3++模型（见章节 4.4 或者公式 4.2），以展示融合隐式反馈的打分信息对推荐性能的影响。各种类型的推荐方法如下：

**PMF.** 该方法在上一节MR3模型的评估时已经介绍过，这里我们再列出来以和下面的SVD++模型对比分析，更清晰的观察隐式打分反馈的影响。

**SVD++.** 这个方法在分解显式打分矩阵的同时学习隐式特征矩阵从而利用隐式反馈信息 [32]，如公式 4.1 所示。通过与上面的PMF模型对比，能更清晰的观察隐式打分反馈对预测性能的影响。

**MR3.** 该方法即是上一节评估的MR3模型，它没有挖掘隐式反馈信息。这里我们列出来以和下面的扩展融合模型MR3++模型对比分析，更清晰的观察隐式打分反馈的影响。

**MR3++.** 挖掘隐式反馈的扩展融合方法。

表 4.1: 扩展融合模型MR3++与不同方法的RMSE比较。 $(F = 10)$  参见下面的 4.6.2 章节对打星号项目的解释。

Datasets	Training	Methods			
		PMF	SVD++	MR3	MR3++
Epinions	20%	1.2001	1.1159	1.1051	1.1026
	50%	1.1604	1.0816	1.0809	1.0785
	80%	1.1502	1.0655	1.0648	1.0641
	90%	1.1484	1.0601*	1.0634	1.0618
Ciao	20%	1.0877	1.0555	1.0142	1.0132
	50%	1.0536	1.0276	0.9740	0.9711
	80%	1.0418	1.0139	0.9521	0.9464
	90%	1.0391	1.0055	0.9451	0.9425

我们使用Java推荐系统库LibRec.net里提供的SVD++的实现。通过网格搜索法确定超参如下：对于MR3++， $\lambda_{\text{rel}} = 0.001$ ， $\lambda_{\text{rev}} = 0.005$ . 关于MR3++模型的超参分析详见下面的章节 4.6.3。

RMSE比较结果见表 4.1，<sup>2</sup> 在两个数据集上变化训练集百分比 $x = \{20, 50, 80, 90\}$ 。从实验结果可得到如下的观察：

- 融合多源异构信息和挖掘隐式反馈在两个数据集上都能提升RMSE预测性能。例如，SVD++和MR3模型相比于PMF模型，在Epinions训练集占比80%上，分别获得了7.94% 和8.02%的相对RMSE提升。
- 扩展融合模型MR3++基本上都取得了更好的性能。MR3++模型与SVD++和MR3模型比较起来，在Ciao数据集上分别取得了5.94% 和0.31%平均相对RMSE提升。主要原因是扩展融合模型同时利用了额外数据源和隐式反馈信息。来自额外数据源的贡献分析将在第 4.6.2 节做进一步的详细分析。

## 4.6.2 隐式反馈对推荐性能的贡献

我们已经评估了模型MR3++模型的推荐性能，本章节对其进一步分析，以理解隐式打分反馈对推荐性能的影响。

<sup>2</sup>标准偏差都小于 $10^{-5}$ 。虽然表 3.3 和表 4.1 可以合并为一个，但是我们把他分割为两个部分，以分别展示额外数据源（MR3模型）和隐式反馈（MR3++）的各自影响。

## 来自隐式反馈信息的贡献

我们通过逐步从融合模型MR3中消去评论信息和社会信息考察了额外数据源对推荐性能的影响，展示了额外数据源对于以RMSE为评估指标的打分预测任务都有所帮助。我们现在考察扩展融合模型MR3++中来自隐式打分反馈信息的推推荐性能的影响。MR3++模型包括两个组件：一个是融合多源异构信息，另一个是深度挖掘有限数据源。具体来说，类似MR3模型的分析，我们也逐步消去额外数据源和隐式反馈来考察它们对扩展融合MR3++预测性能的影响：

**MR3++\sources:** 从模型MR3++即公式 4.2 中通过设置  $\lambda_{rev} = 0$  和  $\lambda_{rel} = 0$  从而消去来自额外数据源的贡献，这等价于SVD++模型，如公式 4.1 所示。

**MR3++\implicit:** 从模型MR3++即公式 4.2 中通过设置  $Y_j = 0$  即不学习物品的隐式特征从而消去来自隐式反馈信息的贡献，这等价于MR3模型，如公式 3.8 所示。

**MR3++\sources\implicit:** 从模型MR3++即公式 4.2 中通过设置  $\lambda_{rev} = 0$  和  $\lambda_{rel} = 0$  及  $Y_j = 0$ ，即消去来自额外数据源的贡献也不学习物品的隐式特征，这等价于PMF模型，如公式 3.1 所示。

扩展融合模型MR3++与其两个组件在数据集Epinions和Ciao上的预测性能展示于图 4.3。从柱状图上可以清晰的看到，在两个数据集上，消去隐式反馈都会对降低推荐性能。例如  $MR3++\backslash implicit$  在 20% 的 Epinions 数据集和 80% 的 Ciao 数据集上，相对 RMSE 预测性能分别降低了 0.23% 和 0.60%。

## 集成更多数据源还是挖掘更深信息？

回顾表 4.1 和图 4.3，可以看到有时候集成更多数据源比挖掘更深有限数据源提升更大，例如在Ciao数据集上，就平均相对RMSE来说，提升的间隔超过 5.61%。但也不是总这样，例如在Epinions数据集上，就平均相对RMSE来说，提升的间隔小于 0.23%；而且当训练集是 90% 时，也即表中的打星号项目，深度挖掘打分数据源的方法甚至超过了集成评论数据和社会信息的方法。

考虑社会关系和评论文本的质量，参加表 3.2，Ciao数据集上每个物品所含评论词的平均个数是Epinions的42倍，且Ciao的社会关系密度是Epinions的12倍。因此Ciao包含了来自社会关系和评论文本的更高质量和更丰富的信息，从直觉和经验上都可以看到，当额外数据源包含的信息更好更多时，加入额外数据源的方法比只挖掘有限数据源的方法更能提升推荐性能。

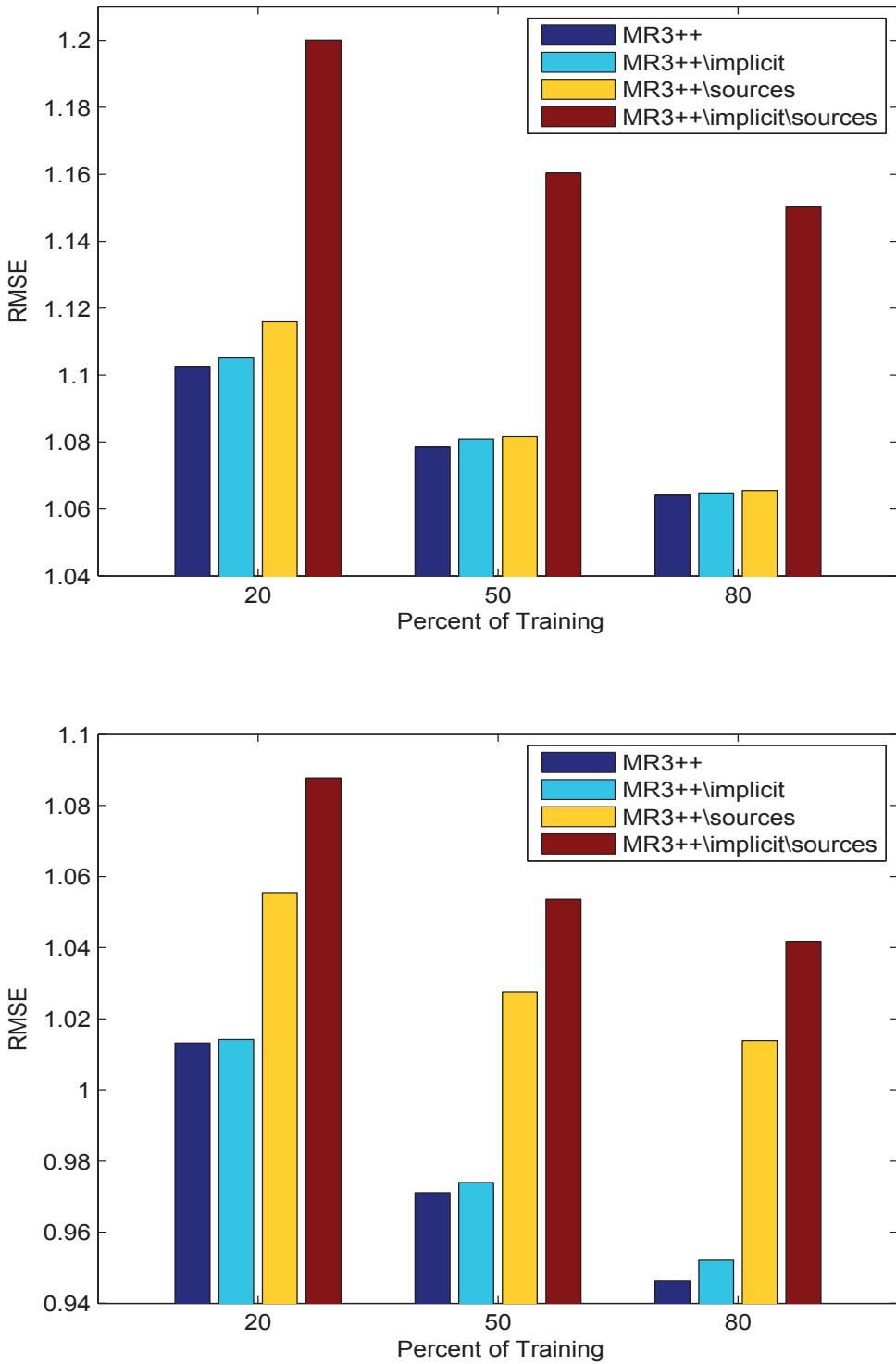


图 4.3: 扩展融合模型MR3++与其两个组件的比较。上图: Epinions. 下图: Ciao.

图 4.3 中在Epninions数据集上的结果值得进一步思考, 此时消去隐式反馈

或者额外数据源都最终模型的打分预测性能都比较小，但是当同时去除时却对推荐性能产生了重大影响。注意这与在Ciao数据集上，甚至于图 3.7 上的观察都不一样。一个可能的解释是，来自额外数据源和来自隐式反馈的贡献可能不是线性可加的，不同数据集上这两者的影响有不同的模式。

### 4.6.3 挖掘隐式反馈的扩展融合模型的超参分析

本节分析扩展融合模型MR3++对超参的敏感性，基本方法是控制变量法：即每次变化一个参数而固定其他参数，考察模型的预测性能的相应变化。类似对MR3模型的分析，我们现在分析MR3++模型对超参的敏感性分析。

扩展融合模型MR3++也包含三个重要的超参：一个是控制来自社会关系信息的贡献度 $\lambda_{\text{rel}}$ ，一个是控制来自评论文本信息的贡献度 $\lambda_{\text{rev}}$ ，一个是确定隐藏因子的维度 $F$ 。我们通过每次变化一个参数而固定其他参数的方法来分析MR3的超参敏感性。

首先我们固定隐藏因子的维度 $F = 10$ ，考察与评论信息相关的参数 $\lambda_{\text{rev}}$ 和与社会信息相关的参数 $\lambda_{\text{rel}}$ 对模型MR3++的性能影响。分析实验结果如图 4.4 所示，我们可以看到：不管哪个参数置为0，也即不论去掉哪种辅助数据源，都会造成预测性能的下降（当两个参数都取值为0时，RMSE=1.0139）。还可以观察到，模型MR3在当两个参数都取值较小时（例如从0.001 到0.005）是处于较稳定的状态，因此我们分别为它们取定默认值 $\lambda_{\text{rel}} = 0.001$ ,  $\lambda_{\text{rev}} = 0.005$ 。注意RMSE的变化区间长度小于0.002。

接下来我们固定 $\lambda_{\text{rel}} = 0.001$ ,  $\lambda_{\text{rev}} = 0.005$ ，考察隐藏因子维度 $F$ 对模型MR3++的性能影响。我们设置 $F = \{5, 10, 15, 20, 30, 50\}$ ，并且训练集占比分别为20%, 50%, 80%。分析实验结果如图 4.4 所示，我们可以看到：模型MR3对隐藏因子维度具有较稳定的性能，因此我们取定默认值 $F = 10$ 。

## 4.7 小结

本章首次提出了一个同时挖掘来自打分数据的隐式反馈信息和加入文本与社交关系的额外数据源的扩展融合模型。据我们所知，深度挖掘隐式反馈的模型没有利用来自评论文本等额外数据源，而加入辅助数据源的模型又没有深度挖掘隐式反馈信息。我们提出的扩展融合模型MR3++的有效性在两个较大规模的数据上得到了验证。

本章的扩展融合模型和上一章的融合模型，他们的模型参数学习过程是相似的，因此将其归纳统一在第 5 章学习过程。

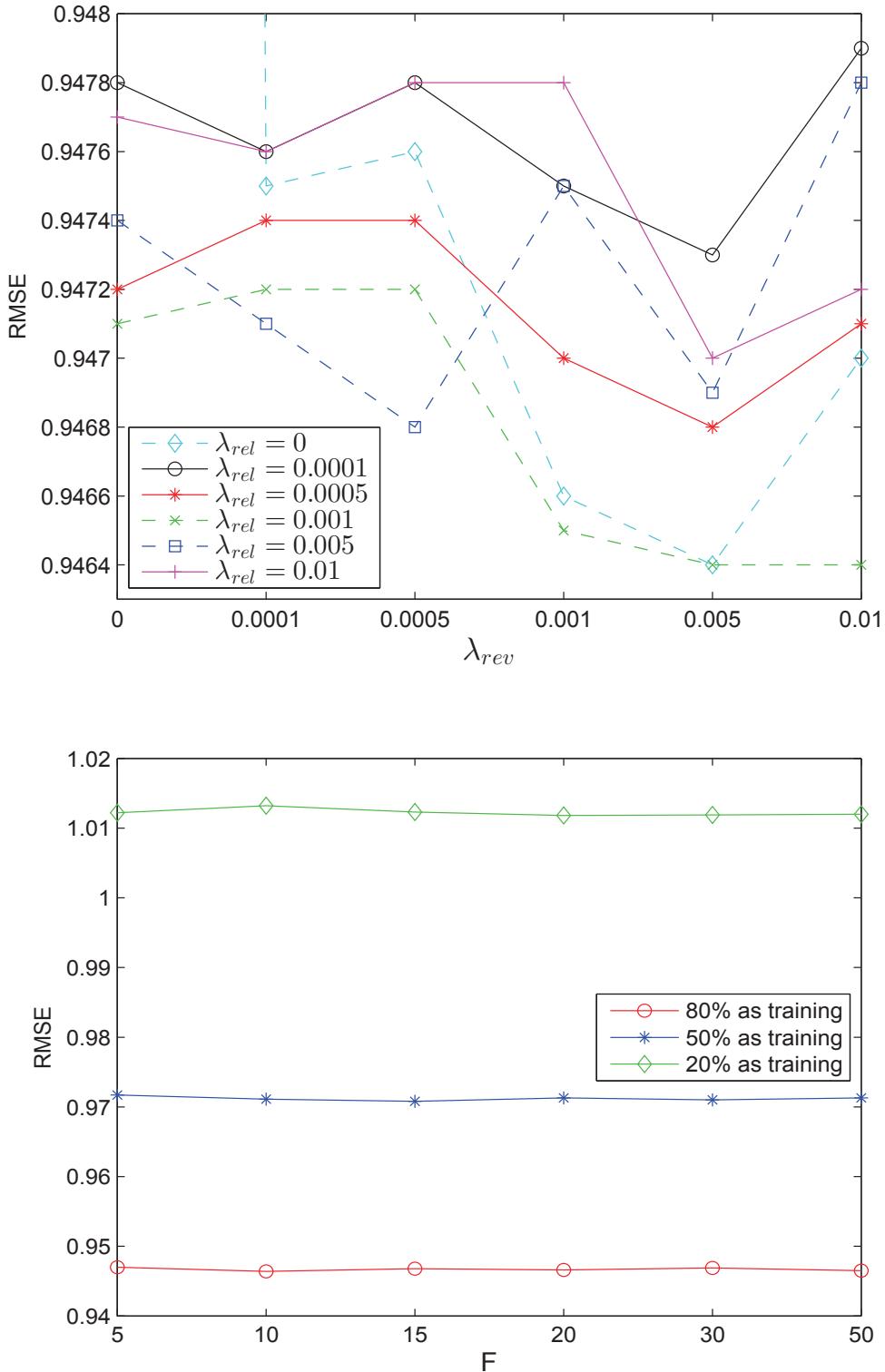


图 4.4: 扩展融合模型MR3++随超参变化的预测性能。上图: 随参数 $\lambda_{rel}$  和 $\lambda_{rev}$  变化的结果, 当两者都是0时的RMSE值是1.0139; 固定隐藏因子 $F = 10$ , 训练集占比80%。下图: 随参数 $F$  变化的结果, 固定 $\lambda_{rel} = 0.001$ ,  $\lambda_{rev} = 0.005$ 。数据集: Ciao

## 第五章 模型的学习过程

本章给出我们提出的MR3和MR3++两个模型的学习过程（见公式 3.8 和 4.2）。他们的学习过程还是相同的，除了在模型参数梯度上有些许差别之外。

### 5.1 学习过程

我们的学习目标是：

$$\arg \min_{\Theta, \Phi, z, \kappa} \mathcal{L}(\Theta, \Phi, z, \kappa). \quad (5.1)$$

注意到参数  $\Theta = \{P, Q, H\}$  和  $\Phi = \{\theta, \phi\}$  的耦合关系是物品特性因子  $Q$  和物品主题因子  $\theta$  之间的softmax变换（见公式 3.5，图 3.5 中的虚线）。前者参数  $\Theta$  与打分和社会信息相关，可以通过梯度下降法学习；后者参数  $\Phi$  与文本信息相关，可以通过吉布斯采样（Gibbs sampling）[20] 学习。类似于社会矩阵分解模型HFT [39]，我们设计一个两阶段交替学习过程如下：

$$\text{更新 } \Theta^{\text{new}}, \Phi^{\text{new}}, \kappa^{\text{new}} = \arg \min_{\Theta, \Phi, \kappa} \mathcal{L}(\Theta, \Phi, z^{\text{old}}, \kappa); \quad (5.2a)$$

$$\text{采样 } z_{d,n}^{\text{new}} \text{ 依概率 } p(z_{d,n}^{\text{new}} = f) = \phi_{f,w_{d,n}}^{\text{new}}. \quad (5.2b)$$

对于上述公式所示的第一个步骤，评论语料中每个词的主题赋值  $z_{d,n}$  被固定下来，然后通过梯度下降更新参数  $\Theta, \Phi$  和  $\kappa$ 。<sup>1</sup> 除了有更多的参数梯度需要计算以外，这个过程同基于梯度的矩阵分解法，如PMF模型的学习过程，我们在下一节单独给出模型参数梯度。

对于上述公式所示的第二个步骤，评论语料相关的参数  $\theta$  和  $\phi$  被固定下来，然后通过采样法更新每个词的主题赋值  $z_{d,n}$ 。<sup>2</sup> 主题赋值的更新是遍历整个语料的文档，以及文档中的每个词，设置  $z_{d,n} = f$  与  $\theta_{d,f} \phi_{f,w_{d,n}}$  成正比采样。

最后上述两个步骤交替重复直到找到一个局部最优点。在我们的实验中，每5次梯度步骤后采样主题赋值，这被称为一趟（pass），通常运行50趟可以找到一个局部最优点。实验设置细节将在实验章节详细描述。

<sup>1</sup>注意参数  $Q$  和  $\theta$  是互相依赖的，我们仅拟合  $Q$  进而通过softmax变换即公式 3.5 确定  $\theta$ 。

<sup>2</sup>这个学习过程与LDA模型 [7]相似，除了文档-主题分布  $\theta$  不是采样于一个Dirichlet先验分布，而是由前一个步骤的  $Q$  确定。

对于MR3++模型，学习过程同MR3，除了它要计算更多的梯度以及修改与隐式特征矩阵 $Y$ 相关参数的梯度公式，我们在下一节给出两个模型的参数梯度公式。

## 5.2 参数的梯度

我们现在给出上述学习过程第一个步骤中公式 5.2a 涉及的参数梯度。<sup>3</sup> 除了表 3.1 中给出的符号，为了学习模型参数需要更多的记号，主要是与吉布斯采样相关的，详见文献 [20]。

- 对于每个物品 $i$ （即文档 $i$ ，把对该物品的所有评论合成为要给文档）：1)  $M_i$  是一个 $F$ -维的计数向量，其中每个元素表示该位置上对应主题出现的次数；2)  $m_i$  是文档所含词数；3)  $z_i = \sum_f \exp(\kappa Q_{if})$  是一个归一化因子。
- 对于每个词 $w$ （在预先确定好的规模为 $L$ 的词表中）：1)  $N_w$  是一个 $F$ -维的计数向量，其中每个元素表示该词被赋予了对应主题的次数；2)  $n_f$  是主题 $f$  在这个词上出现的总次数；3)  $z_f = \sum_w \exp(\psi_{fw})$  是一个归一化因子。

用户相关的特征矩阵  $P$  出现在三个项目中：第一个是打分预测中，第二个是局部社会上下文中，最后是范数罚项中。

$$\begin{aligned} \frac{1}{2} \frac{\partial \mathcal{L}}{\partial P_u} &= \sum_{i: R_{u,i} \neq 0} W_{u,i} (\hat{R}_{u,i} - R_{u,i}) Q_i + \lambda P_u \\ &\quad + \lambda_{\text{rel}} \sum_{v: T_{u,v} \neq 0} C_{u,v} (P_u^T H P_v - S_{u,v}) H P_v \\ &\quad + \lambda_{\text{rel}} \sum_{v: T_{v,u} \neq 0} C_{v,u} (P_v^T H P_u - S_{u,v}) H^T P_v. \end{aligned} \quad (5.3)$$

物品相关的特征矩阵  $Q$  出现在三个项目中：第一个是打分预测中，第二个是评论文本语料中，最后是范数罚项中。

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial Q_i} &= 2 \sum_{u: R_{u,i} \neq 0} W_{u,i} (\hat{R}_{u,i} - R_{u,i}) P_u \\ &\quad - \lambda_{\text{rev}} \kappa \left( M_i - \frac{m_i}{z_i} \exp(\kappa Q_i) \right) + 2\lambda Q_i. \end{aligned} \quad (5.4)$$

---

<sup>3</sup>与用户与物品相关偏置参数梯度略去，不拟合打分均值因为事先已中心化。

社会相关矩阵  $H$  出现在两个项目中：第一个是局部社会上下文中，第二个是范数罚项中。

$$\frac{1}{2} \frac{\partial \mathcal{L}}{\partial H} = \lambda_{\text{rel}} \sum_{T_{u,v} \neq 0} C_{u,v} (P_u^T H P_v - S_{u,v}) P_u P_v^T + \lambda H. \quad (5.5)$$

控制物品特性因子  $Q$  和物品主题因子  $\theta$  变换峰度的参数  $\kappa$  出现在 softmax 变换中（见公式 3.5）。优化目标  $\mathcal{L}$  关于  $\kappa$  的微分遵从链式法则（chain rule）： $\mathcal{L}$  先对  $\theta$  微分，然后  $\theta$  再对  $\kappa$  微分。

$$\frac{\partial \mathcal{L}}{\partial \kappa} = -\lambda_{\text{rev}} \sum_{i,f} Q_{if} \left( M_{if} - \frac{m_i}{z_i} \exp(\kappa Q_{if}) \right). \quad (5.6)$$

注意到主题-词分布  $\phi_f$  是一个随机向量，所以我们先优化它对应的非归一化向量  $\psi_f$ ，然后再归一化得到概率版本： $\phi_{fw} = \exp(\psi_{fw})/z_f$ 。未归一化的主题-词分布存在于评论文本语料的似然函数中。

$$\frac{\partial \mathcal{L}}{\partial \psi_{fw}} = -\lambda_{\text{rev}} \left( N_{fw} - \frac{n_f}{z_f} \exp(\psi_{fw}) \right). \quad (5.7)$$

对于MR3++模型，多了一个隐式特征矩阵  $Y$ ，并且与之相关的物品特性因子  $Q$  的梯度公式也将随之做相应修改；其他参数同MR3模型。

隐式特征矩阵  $Y$  的梯度是：

$$\frac{1}{2} \frac{\partial \mathcal{L}}{\partial Y_j} = \sum_{R_{u,i} \neq 0} W_{u,i} |N_u|^{-\frac{1}{2}} (\hat{R}_{u,i}^* - R_{u,i}) Q_i + \lambda Y_j, \quad \forall j \in N_u. \quad (5.8)$$

原来MR3中的物品特性因子  $Q$  的梯度公式则被修改为：

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial Q_i} &= 2 \sum_{u: R_{u,i} \neq 0} W_{u,i} (\hat{R}_{u,i}^* - R_{u,i}) \left( P_u + |N_u|^{-\frac{1}{2}} \sum_{j \in N_u} Y_j \right) \\ &\quad - \lambda_{\text{rev}} \kappa \left( M_i - \frac{m_i}{z_i} \exp(\kappa Q_i) \right) + 2\lambda Q_i. \end{aligned} \quad (5.9)$$

与公式 5.4 的比较可知，与隐式特征  $Y$  相关的一项被加在了  $P$  中。

## 第六章 总结和展望

除了显式打分数据源，社会关系和产品评论信息等异构推荐数据源，为传统的推荐系统既带来了挑战同时也带来了机遇。我们考察了如何通过对齐隐藏主题和潜在因子高效和紧密地融合三种推荐信息提升推荐性能。进一步的，我们通过物品的隐式特征因子，挖掘来自打分数据的隐式反馈信息。

我们首先提出了一个新颖的融合模型MR3。通过将隐藏主题和潜在因子主题，同时对物品评论学习主题模型和对用户关系学习社会因子，MR3模型能高效地利用三种推荐数据源，提高打分预测性能。另外，我们还得到了一个扩展社会矩阵分解法eSMF。通过信任值利用邻居图结构，捕捉社会影响力局部性，eSMF能够紧密地利用打分信息和社会信息。进一步的，我们提出了一个扩展融合模型MR3++。通过挖掘来自打分数据的隐式反馈信息，考虑已打过分的物品的隐式特征影响，我们扩展了MR3融合模型，得到了一个挖掘隐式反馈的扩展融合模型MR3++。

我们在两个较大规模的真实数据集上评估了我们提出的两个模型，与若干不同类型的推荐方法相比较，我们的模型都获得了良好的预测性能。并且我们考察了额外数据源和隐式反馈信息对推荐性能的影响，从辅助数据源的信息丰富度角度比较了这两者的贡献。最后分析了模型的超参敏感性。

本文关注于打分预测任务，因此我们选择的评估指标是根均方误差；在未来工作中，应该探索更多的评估度量以更完整的理解推荐系统行为 [40]。

## 参考文献

- [1] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.
- [2] Y Bao, H Fang, and J Zhang. Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, 2014.
- [3] Yang Bao, Hui Fang, and Jie Zhang. Topiccmf: Simultaneously exploiting ratings and reviews for recommendation. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.
- [4] Nicholas J Belkin and W Bruce Croft. Information filtering and information retrieval: Two sides of the same coin? *Communications of the ACM*, 35(12):29–38, 1992.
- [5] James Bennett and Stan Lanning. The netflix prize. In *Proceedings of KDD cup and workshop*, 2007.
- [6] Daniel Billsus and Michael J Pazzani. Learning collaborative information filters. In *Proceedings of the Fifteenth International Conference on Machine Learning*, 1998.
- [7] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.
- [8] John S Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, 1998.
- [9] Chaochao Chen, Xiaolin Zheng, Yan Wang, Fuxing Hong, and Zhen Lin. Context-aware collaborative topic regression with social matrix factorization for recommender systems. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

- [10] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, et al. The youtube video recommendation system. In *Proceedings of the fourth ACM conference on Recommender systems*, 2010.
- [11] Mukund Deshpande and George Karypis. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*, 22(1):143–177, 2004.
- [12] Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J Smola, Jing Jiang, and Chong Wang. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014.
- [13] Xuetao Ding, Xiaoming Jin, Yujia Li, and Lianghao Li. Celebrity recommendation with collaborative social topic regression. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 2013.
- [14] Pedro Domingos and Matt Richardson. Mining the network value of customers. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, 2001.
- [15] Michael D Ekstrand, John T Riedl, and Joseph A Konstan. Collaborative filtering recommender systems. *Foundations and Trends in Human-Computer Interaction*, 4(2):81–173, 2011.
- [16] Yi Fang and Luo Si. Matrix co-factorization for recommendation with rich side information and implicit feedback. In *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, 2011.
- [17] Gayatree Ganu, Noemie Elhadad, and Amélie Marian. Beyond the stars: Improving rating predictions using review text content. In *Proceedings of the 12th International Workshop on the Web and Databases*, 2009.
- [18] Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2):133–151, 2001.

- [19] Prem K Gopalan, Laurent Charlin, and David Blei. Content-based recommendations with poisson factorization. In *Advances in Neural Information Processing Systems*, 2014.
- [20] Thomas L Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228–5235, 2004.
- [21] Quanquan Gu, Jie Zhou, and Chris HQ Ding. Collaborative filtering: Weighted nonnegative matrix factorization incorporating user and item graphs. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, 2010.
- [22] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. Trustsvd: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
- [23] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22:5–53, 2004.
- [24] Thomas Hofmann. Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems*, 22(1):89–115, 2004.
- [25] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *Data Mining, Eighth IEEE International Conference on*, 2008.
- [26] Niklas Jakob, Stefan Hagen Weber, Mark Christoph Müller, and Iryna Gurevych. Beyond the stars: exploiting free-text user reviews to improve the accuracy of movie recommendations. In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, 2009.
- [27] Mohsen Jamali and Martin Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*, 2010.
- [28] Mohsen Jamali and Martin Ester. A transitivity aware matrix factorization model for recommendation in social networks. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

- [29] Santosh Kabbur, Xia Ning, and George Karypis. Fism: factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013.
- [30] Younghoon Kim and Kyuseok Shim. Twitobi: A recommendation system for twitter using probabilistic modeling. In *Data Mining , IEEE 11th International Conference on*, 2011.
- [31] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8):30–37, 2009.
- [32] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008.
- [33] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.
- [34] Guang Ling, Michael R Lyu, and Irwin King. Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM Conference on Recommender systems*, 2014.
- [35] Nathan N Liu, Evan W Xiang, Min Zhao, and Qiang Yang. Unifying explicit and implicit feedback for collaborative filtering. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, 2010.
- [36] Hao Ma, Haixuan Yang, Michael R Lyu, and Irwin King. Sorec: social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008.
- [37] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*, 2011.
- [38] Paolo Massa and Paolo Avesani. Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, 2007.
- [39] Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, 2013.

- [40] Sean M McNee, John Riedl, and Joseph A Konstan. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *CHI'06 extended abstracts on Human factors in computing systems*, 2006.
- [41] Koji Miyahara and Michael J Pazzani. Collaborative filtering with the simple bayesian classifier. In *PRICAI 2000 Topics in Artificial Intelligence*, pages 679–689. Springer, 2000.
- [42] Andriy Mnih and Ruslan R Salakhutdinov. Probabilistic matrix factorization. In J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, editors, *Advances in Neural Information Processing Systems 20*, 2008.
- [43] Mark O’ Connor and Jon Herlocker. Clustering items for collaborative filtering. In *Proceedings of the ACM SIGIR workshop on recommender systems*, 1999.
- [44] Rong Pan, Yunhong Zhou, Bin Cao, Nathan N Liu, Rajan Lukose, Martin Scholz, and Qiang Yang. One-class collaborative filtering. In *Data Mining, Eighth IEEE International Conference on*, 2008.
- [45] Arkadiusz Paterek. Improving regularized singular value decomposition for collaborative filtering. In *Proceedings of KDD cup and workshop*, 2007.
- [46] Sanjay Purushotham, Yan Liu, and C-C Jay Kuo. Collaborative topic regression with social matrix factorization for recommendation systems. *Proceedings of the 29th International Conference on Machine Learning*, 2012.
- [47] Steffen Rendle. Factorization machines. In *Data Mining , IEEE 10th International Conference on*, 2010.
- [48] Jasson DM Rennie and Nathan Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning*, 2005.
- [49] Paul Resnick and Hal R Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.
- [50] Francesco Ricci, Lior Rokach, and Bracha Shapira. *Introduction to recommender systems handbook*. Springer, 2011.

- [51] Badrul M Sarwar, George Karypis, Joseph A Konstan, and John T Riedl. Application of dimensionality reduction in recommender system—a case study. In *ACM WEBKDD WORKSHOP*, 2000.
- [52] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, 2001.
- [53] Ajit P Singh and Geoffrey J Gordon. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008.
- [54] Xiaoyuan Su and Taghi M. Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, 2009:421425:1–421425:19, 2009.
- [55] Lei Tang and Huan Liu. Community detection and mining in social media. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 2(1):1–137, 2010.
- [56] Jiliang Tang, Huiji Gao, Xia Hu, and Huan Liu. Exploiting homophily effect for trust prediction. In *Proceedings of the sixth ACM international conference on Web search and data mining*, 2013.
- [57] Jiliang Tang, Xia Hu, Huiji Gao, and Huan Liu. Exploiting local and global social context for recommendation. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 2013.
- [58] Jiliang Tang, Xia Hu, and Huan Liu. Social recommendation: a review. *Social Network Analysis and Mining*, 3(4):1113–1133, 2013.
- [59] Chong Wang and David M Blei. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011.
- [60] Hao Wang, Binyi Chen, and Wu-Jun Li. Collaborative topic regression with social regularization for tag recommendation. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 2013.
- [61] Yingqiang Xu, Wai Lam, and Tianyi Lin. Collaborative filtering incorporating review text and co-clusters of hidden user communities and item groups. In *Proceed-*

*ings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, 2014.

- [62] Bo Yang, Yu Lei, Dayou Liu, and Jiming Liu. Social collaborative filtering by trust. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 2013.
- [63] Zhijun Yin, Manish Gupta, Tim Weninger, and Jiawei Han. Linkrec: a unified framework for link recommendation with user attributes and graph structure. In *Proceedings of the 19th international conference on World wide web*, 2010.
- [64] Jing Zhang, Biao Liu, Jie Tang, Ting Chen, and Juanzi Li. Social influence locality for modeling retweeting behaviors. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 2013.
- [65] Tong Zhao, Julian McAuley, and Irwin King. Leveraging social connections to improve personalized ranking for collaborative filtering. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, 2014.
- [66] Shenghuo Zhu, Kai Yu, Yun Chi, and Yihong Gong. Combining content and link for classification using matrix factorization. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 2007.

## 致 谢

首先感谢我的指导老师戴新宇副教授，是戴老师把我带到了自然语言处理这个令人着迷的研究领域。在学术研究上，戴老师对我严格要求，对研究课题给予深刻的指导意见，对论文都做仔细修改。在日常生活中，戴老师为人和蔼可亲，给我敦敦教诲，也为我排忧解难。

感谢陈家骏老师对论文组织上的指导与帮助，不论是从结构到遣词，都事无巨细。感谢黄书剑老师，维系了自然语言处理实验室这个大家庭的良好学术环境。感谢尹存燕、张建兵和曹迎春老师，在研究生事务上给予了很大的帮助和便利。

感谢Rick Jin, Fandy Wang, 邹远航等，在腾讯（深圳）SPA实习时，得到的指导。感谢Jie Bao, Cong Wang, Qiang Zhang, Qinghua Duan, Dinglish, Yijing Xie, Jing Shi, Sun Yang, 在北京的愉快时光。感谢在远赴阿根廷参加IJCAI-15会议时，一起暴走的蒋庆远，刘倩，吕丹彦，邢佑路。感谢在访问香港科技大学时，与曹绍祖，曾子倩，陈戈，陈雪璐，一起度过的虽然短暂然而难忘的时光。

感谢实验室的牛力强，程川，黄家君，程善伯，周逸初等同学，在学术讨论中收获良多。感谢实验室的邹远航，奚宁，周浩，陈华栋等师兄，张颖杰和潘林林师姐，从他们身上汲取指导。感谢实验室的尚迪，郁振庭，李小婉，汤莲瑞，季红洁等师妹师弟们，在平时相处中快乐满满。

感谢饭友牛力强，蒋哲翎，姜成樾，蒋继东，缪峥，刘瑞奇，程川，黄家君，程善伯，尚迪，享用美食是人生的一大趣事。

## **附录：攻读硕士学位期间发表的论文、获得的奖励、参加的学术活动和参与的科研项目**

### **攻读硕士学位期间发表的论文**

1. Guang-Neng Hu, Xin-Yu Dai, Yunya Song, Shu-Jian Huang, and Jia-Jun Chen. “A synthetic approach for recommendation: combining ratings, social relations, and reviews.” In Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI 2015), pp. 1756-1762, July 25 - August 1, 2015, Buenos Aires, Argentina. AAAI Press, 2015. (CCF-A类会议)

### **攻读硕士学位期间获得的奖励**

1. 国家奖学金（硕士研究生），教育部，2015年
2. 优秀研究生，南京大学，2015年
3. 优秀共青团干部，南京大学，2014年

### **攻读硕士学位期间参加的学术活动**

1. 第24届人工智能国际联合大会：推荐系统分场，布宜诺斯艾利斯，阿根廷，2015年
2. 中文信息学会青年工作委员会系列学术活动：长三角地区自然语言处理青年学者论坛暨顶级国际会议论文报告会，南京大学，2015年
3. IJCAI 2015国内论文报告分享会，清华大学，2015年

### **攻读硕士学位期间参与的科研项目**

1. 国家自然科学基金项目“文本分类中的文本图表示模型和结构化稀疏模型研究”（课题编号：61472183）