学 校 代 码 10459
学号或申请号 201412172123
密 级

郑州大学

硕士学位论文

基于形式概念分析的推荐算法研究及应用

作 者 姓 名：陈昊文
导 师 姓 名：王黎明 教 授
张 卓 副教授

学 科 门 类：工 学
专 业 名 称：计算机科学与技术
培 养 院 系：信息工程学院
完 成 时 间：2017 年 5 月

A thesis submitted to

Zhengzhou University

for the degree of Master

Research and Application of Recommendation Algorithms Based on Formal Concept Analysis

By Haowen Chen

Supervisor:Prof. Liming Wang and A.Prof. Zhuo Zhang

Computer Science and Technology

Information Engineering Institute

May,2017

学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的科研成果。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式标明。本声明的法律责任由本人承担。

学位论文作者： 

日期：2017年5月25日

学位论文使用授权声明

本人在导师指导下完成的论文及相关的职务作品，知识产权归属郑州大学。根据郑州大学有关保留、使用学位论文的规定，同意学校保留或向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅；本人授权郑州大学可以将本学位论文的全部或部分编入有关数据库进行检索，可以采用影印、缩印或者其他复制手段保存论文和汇编本学位论文。本人离校后发表、使用学位论文或与该学位论文直接相关的学术论文或成果时，第一署名单位仍然为郑州大学。保密论文在解密后应遵守此规定。

学位论文作者： 

日期：2017年5月25日

摘要

做为处理信息过载的有效手段，推荐系统在近些年得到了广泛的研究与发展，推荐系统在各领域应用的成功案例也不断涌现，但是依然面临着很多问题亟待解决。形式概念分析（Formal Concept Analysis，FCA）的核心数据结构—概念格（Concept lattices），是一种数据分析与规则提取的有效工具。外延与内涵做为概念的组成部分使得形式概念展现出了聚类的特性。概念之间存在的偏序关系也揭示了其泛化与特化的本质。随着其研究的不断深入，形式概念分析开始逐步应用于数据挖掘、信息检索等领域。

协同过滤（Collaborative Filtering，CF）推荐作为应用最广的推荐策略之一，其中经典的基于邻域的协同过滤算法通常只考虑用户间或项目间的相似关系，而忽略了不同对象之间的内在联系。此外越来越多的研究人员发现，推荐系统往往面对的是无法直观反映用户喜好程度的隐式数据，并且随着产品种类的剧增，用户与项目间产生的隐式数据也会变得极为稀疏。所以由于稀疏数据环境下信息的缺失，协同过滤算法往往获取不到充足的邻域信息，从而直接影响了最终的推荐效果。针对以上问题，本文提出了一种面向隐式数据的基于概念邻域的协同过滤推荐算法（Conceptual Neighborhood-based Collaborative Filtering，CNCF）。该算法针对 Top-N 推荐问题，以概念格为载体进行推荐问题求解。首先在用户与项目的关系数据转化而成形式背景的基础上进行概念格的构造，将用户与产品分别以对象与属性的形式聚集在概念中，并基于概念格生成相应的起始概念索引，借助索引结构高效地对对象的起始概念进行定位。之后利用概念之间的偏序关系，以对象（用户）的起始概念为起点探索其近邻概念并获取候选项集。最后结合所提出的全局偏好度与邻域偏好度筛选出用户可能感兴趣的推荐列表。

通过对 CNCF 算法的实现，并在两个真实数据集上进行实验验证，相较于传统基于邻域的协同过滤推荐算法，CNCF 算法在可以在保持较好的推荐效果同时，更适用于数据稀疏环境下的推荐。

关键字：推荐算法；协同过滤；形式概念分析；概念格

Abstract

As an effective mean for dealing with information overload, the recommendation system has been extensively researched and developed in recent years.Recommender system is constantly emerging in the field of cases of the successful applications, but is still facing many problems to be solved . The concept lattices is the core data structure of formal concept analysis (FCA) , and is the effective tool for data analysis and rule extraction. The extension and intension are the component of the concept, which makes the formal concept show the characteristics of clustering. The partial order relations among concepts also reveal the essence of generalization and specialization. With the deepening of the research, formal concept analysis has been gradually applied in the field of data mining, information retrieval and so on.

Collaborative filtering is one of the most widely used recommendation strategy.The classic collaborative filtering algorithms based on neighborhood are usually only consider the similar relations between users or items ,while ignoring the inherent relations between objects. In addition, more and more researchers have found that the recommendation system is often faced with the implicit data which cannot directly reflect the user preference.And with various kinds of products increase, the implicit data generated by users and items will become extremely sparse. Therefore, due to the lack of informations in the sparse data environment, collaborative filtering algorithms can not get enough neighborhood informations, which directly affects the final recommendation. In view of the above problems, this thesis proposes a conceptual neighborhood-based collaborative filtering algorithm(CNCF), which is mainly based on the concept neighborhood for the implicit data. In this algorithm, the concept lattice is used as the data carrier to solve the Top-N recommendation problem. First of all, we construct concept lattice based on the formal context converted from the relational data between users and items. Users and products are respectively in the forms of objects and attributes, and gathered in the concept.Then based on the concept lattices, the initial concept index is

generated and futher improve the efficiency of the location of the initial concept. Afterwards, with the help of the partial order relations among concepts, the initial concepts of objects (users) are used as the starting points to explore the neighbor concepts and obtain the candidate set. Finally, the proposed global preference and neighborhood preference are used to select the final list of recommendation that the users may be interested in.

Through the realization of the CNCF algorithm, and carrying out experiments on two real data sets, compared with the traditional collaborative filtering algorithms based on neighborhood, CNCF algorithm is more suitable for the recommendation under the sparse data environment while maintaining better recommendation effects.

Keywords: Recommendation algorithm; Collaborative filtering; Formal concept analysis; Concept lattices

目录

图清单

表清单

1 绪论

1.1 研究背景

随着近年来互联网的飞速发展，其用户的数量也在以惊人速度增长。为了满足大量用户的各项需求，种类繁多且量级庞大的数据支撑着各项互联网服务的正常运转。作为人类的本能反应，当想要获取任何事物时，首先思维中会形成对需求的特征描述，之后根据这些特征去寻找目标，这就是人们自发产生的搜索行为。而在当今的互联网环境中，正是搜索引擎加快了用户获取信息的效率。一旦用户向互联网表述自己的明确需求，搜索引擎就会根据用户所提交的各项需求信息迅速地将搜索结果反馈给用户。但是，人们不可能任何时候都对需求具有清晰明确的认识。面对这种情况，互联网需要自发地向用户提供信息资源，并且尽量确保它是符合用户潜在需求的，这就是个性化推荐。

互联网企业为了应对基数庞大的人群的各项需求，将不同类型的海量内容以数据的形式通过互联网提供给用户。而用户通过互联网与不同信息产生的各种行为也会以数据的形式记录下来。由此可见，信息过载问题的出现是互联网发展的必然趋势。该问题无论是对运营商还是用户来说都成为了处理或使用过程中的障碍。推荐系统作为处理信息过载问题的一种有效方法，经历了长时间的发展。Amazon 早期将推荐系统应用在了商品的在线销售上，为不同消费者提供可能感兴趣的商品，使得 35%的销售额是与其推荐系统相关的。Netfilx 作为一家较早成立的在线影片租赁公司，由于影片数量巨大，为了让顾客快速方便的挑选影片，将推荐系统应用在了业务流程中，并于 2006 年设立了 Netfilx Prize 推荐大赛[1]，通过设立百万美元奖金，促使研究者在其基础上进一步提升推荐精度，同时也吸引了更多人参与到了推荐领域的研究工作中。在国内，豆瓣网、今日头条、网易云音乐等都是以推荐功能为核心的互联网产品。此外，在各大门户网站，甚至在百度首页，都能轻易地找到个性化推荐的痕迹。

形式概念分析（Formal Concept Analysis，FCA），是德国数学家 Wille 基于序理论提出的理论体系，其核心数据结构概念格已经广泛应用于数据挖掘、信息检索、数据抽取、软件工程等领域[2,3]，是一种强有力的数据分析与规则挖掘工具。其主要研究集中在概念格的构造维护算法与实际领域中概念格的应用两个方面，具体的研究现状及成果会在之后内容中介绍。由于概念格存在特殊的

结构及性质，使得它在多个领域都获得了较好的应用效果。但在个性化推荐领域，形式概念分析及概念格理论的应用仍处于探索阶段，其研究进展和成果也无法与其他领域相提并论。虽然如此，但仅就目前研究进展看，概念格的自身特质以及相关理论是可以一定程度上促进个性化推荐问题的解决，所以具有一定的研究意义及价值。

1.2 国内外研究现状

1.2.1 推荐系统研究现状

推荐系统主要以对与用户相关的各类信息的科学分析为基础，从而向目标用户提供他们可能感兴趣的信息和服务。推荐系统的核心组成部分是推荐算法，而协同过滤（Collaborative Filtering）是目前应用最为广泛的推荐策略之一。它的核心思想是通过分析用户记录，在所有用户中找到与目标用户相似的用户群体，综合相似用户群体对某一项目的评价，系统会对目标用户对此项目的喜好程度进行预测。协同过滤算法主要分为基于模型与基于内存两类[4]。基于模型的方法通过对已有用户数据建模并进行推荐 ，例如最具代表性的 SVD 算法[5]。后来，Koren Y 等人[6]又在此基础上提出了 LFM 算法，只针对已有的评分数据进行训练，而无需预先对缺失数据进行填充。此外，还有基于回归[7]、基于贝叶斯[8]等以数据建模为基础的协同过滤策略。但当面对隐式反馈数据进行建模时，往往需要生成负样本，采用方法的不同直接影响了模型的训练结果，从而导致了模型的不稳定性。基于内存的协同过滤算法主要依赖于邻域的偏好信息，可分为基于用户（user-based）和基于项目（item-based）[9] 两种方式。基于项目的协同过滤是目前业界应用最为广泛的算法，无论是亚马逊网[10]，还是 Netflix，都是以该算法为基础构建的。此外，也有一些算法[11,12]将以上两种方法结合在了一起。但是传统的基于邻域的协同过滤算法仅仅挖掘用户或项目之间的相似关系，而忽略了两者间存在的内在联系。并且在数据稀疏条件下，由于信息的大量缺失通常获取不到充足的邻域信息，直接影响了最终的推荐效果。

1.2.2 形式概念分析研究现状

自从形式概念的理论提出之后，形式概念分析及其核心数据结构概念格的相

关理论作为数据分析领域的重要方法，开始受到国际学术界的广泛关注。时至今日，学术界对于形式概念分析及概念格的相关研究仍在继续，主要集中在概念格的构造、维护算法的研究以及形式概念分析与概念格相关理论在实际问题中的应用两个方面。

（1）概念格构造与维护

目前，概念格的构造方法主要分为批处理和渐进式两种方式。经典的批处理构造算法主要有 Bordat、Chein、Ganter 和 Nourine 算法。而采用渐进式的经典够格算法有 Godin 和 Capineto 等算法。批处理算法的主要思想是一次性构造出所有的概念，随后再根据概念间的偏序关系，填补概念格中的边，也就是为概念之间添加前驱-后继关系。不同于批处理，渐进式够格算法首先将概念格初始化为空，每当有新的对象加入时，与概念格中所有概念进行关系运算，根据结果的不同在原有的格结构上进行相应调整。国内有些研究者在以上经典算法基础上做了改进，其中谢志鹏、刘宗田[13]提出了一种基于树状结构的渐进式构造算法，有效地缩小了新生格节点的父节点和子节点的搜索范围，一定程度上提升了构造效率。沈夏炯等人[14]为了在部分情况下减少够格的时间复杂度，通过减少新生节点间偏序关系确定所需的遍历次数，利用数据库的内部技术优化了 Godin 算法。

另一方面，随着互联网由海量数据时代进入大数据时代，概念格的规模也在逐渐扩大，指数上升的空间与时间复杂度一直是概念格发展所面临的重要问题，并且相关构造算法也整体呈现复杂低效的状态。不只是针对构造算法的优化，对概念格自身结构规模的约简也能有效地缓解以上存在问题，我们也将这类对概念格自身结构进行调整方法称为概念格的维护算法。文献[15]从减少行与列的角度提出了可约简的属性与可约简的对象两个概念。文献[16]则提出了概念格的属性约减理论，通过寻找能够完全确定形式背景上的概念及层次结构的最小属性子集，简化知识的表示与发掘过程。此外，文献[17]提供了一种同步消减多个属性的概念格维护算法。

（2）概念格的应用

随着形式概念分析的发展，其核心数据结构概念格已经广泛应用于数据挖

掘、信息检索、数据抽取和软件工程等领域。(a)概念格利用其自身的结构特点为关联规则提取提供新的思路，不少研究者已经进行了尝试与深入研究。最初Godin在提出渐进式的构格算法同时，也提出了一种基于概念格的关联规则提取方法[18]。Stanford大学的Sahami提出了一种学习分类规则的方法[19]。(b)在粗糙集理论上的应用。粗糙集理论的基础是等价关系，概念格中的格节点恰好具备这种特质，并且每个概念都包含具有最大共同属性集合的对象集。在此基础上，文献[20]提出了一些概念格在粗糙集上的运算方法。Kent 也通过结合粗糙集与概念格理论提出了一种粗糙概念分析的理论。(c)数字化图书馆与文献检索。Neuss与 Kent 运用概念格实现了网络上文档元信息的自动分类与分析[21]。Cole 和Eklund将概念格应用在了医药数据的分析与可视化领域[22],他们还将概念格用于挖掘电子邮件中用户感兴趣的信息。Eklund与Martin将概念格用于Web文档索引与导航[23]。

1.3 主要工作

针对形式概念分析在推荐系统领域的应用与研究，本文主要完成了以下工作：

（1）在起始概念定义的基础上，提出了一种新的起始概念定位方法。通过定义起始概念索引与其构造方法，提升了起始概念的定位效率，避免了遍历格结构时的重复性搜索。

（2）结合形式概念分析及概念格相关理论，对面向隐式数据的Top-N推荐问题进行了形式化描述，并提出了基于概念邻域的协同过滤算法（CNCF）。该方法首先通过在原始数据基础上构造概念格，并根据概念之间的偏序关系通过探索近邻概念来获取推荐候选项集。之后通过本文定义的效用函数计算候选集中各项目对于用户的推荐度，根据推荐度的排序结果形成最终的推荐列表。

（3）在对候选项集进行最终筛选的过程中，本文结合概念格的特殊结构以及概念相似度提出了全局偏好度与邻域偏好度的定义，并在此基础上提出了两种效用函数的计算方法，用于衡量项目对于用户的推荐度。

（4）通过在两个真实数据集上的实验验证，本文提出的CNCF算法，相较于传统的基于邻域的协同过滤算法，能够取得较好的推荐效果，并且能更好地适应数据稀疏环境下的推荐。

1.4 本文组织结构

第一章 引言：主要介绍推荐系统与形式概念分析及概念格相关理论的研究背景，分析了目前形式概念分析与推荐系统的研究现状，并阐述了本文的主要工作和章节安排。

第二章 推荐系统与形式概念分析相关知识：着重介绍了推荐系统与形式概念分析的相关理论知识，并回顾了目前较为重要的研究成果，分析了推荐系统所面临的主要问题。

第三章 概念格中起始概念索引的构造：主要介绍了多值背景向形式背景的转化方法、概念格的渐进式构造方法，以及起始概念索引的相关定义和构造方法。

第四章 基于概念邻域的协同过滤算法：主要介绍了本文提出的基于概念邻域的协同过滤算法，首先通过探索目标用户起始概念的近邻概念构造候选项集。再结合本文提出的全局偏好度与邻域偏好度计算产品对于用户的推荐度，最终，将推荐度最大的 n 个产品形成推荐列表反馈给用户。

第五章 实验结果与分析：首先通过实验验证了起始概念索引的引入相较于传统定位方法效率上的提升。通过各项评价标准对实验结果的评测，进一步验证本文所提出的推荐模型相较于传统的基于邻域的协同过滤能够取得较好的推荐效果，并且更适用于数据稀疏的推荐环境，适用于解决实际问题。

第六章 总结与展望：对本文的研究内容进行了总结，并分析了该方法现阶段存在的不足，对下一步工作进行了展望。

2　推荐系统与形式概念分析相关知识

2.1　常见的推荐模型分析

数据规模的快速增长与用户日益多样化的需求促使了推荐系统的产生。相比于更早起步的搜索引擎，推荐系统做为另一种能够有效处理信息过载问题的手段，虽然其处理方式有所不同，但推荐系统的本质就是发掘用户与产品间的内在联系，最终能够达到对用户需求或兴趣的一种预测。人们最为了解的也是推荐系统较早应用的领域就是电子商务领域。种类繁多的商品信息量早已超出了人们自身的检索能力范围。在搜索引擎与推荐系统的成功应用下，用户不仅可以使用搜索引擎主动过滤出符合需求的有效信息，也可以在推荐系统的作用下，被动地接收到自己可能感兴趣的商品信息。不仅是在电子商务方面，推荐系统的研究发展进一步完善了互联网的使用环境，为用户提供了便利。通常的推荐系统都有几个相对独立的模块组成。

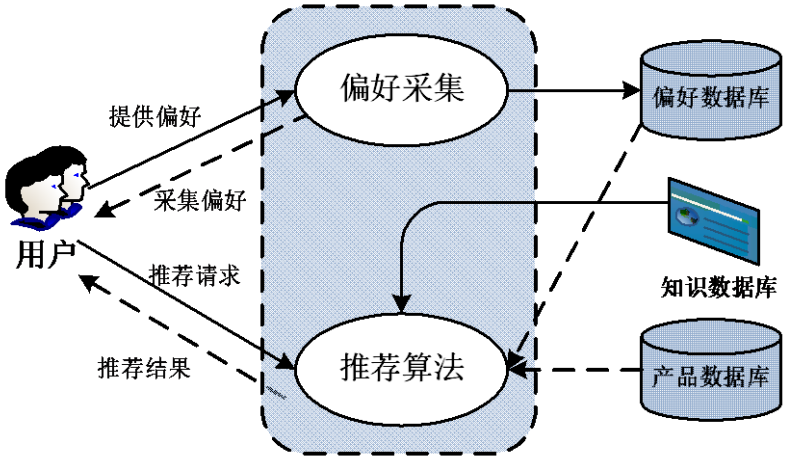

图 2.1　推荐系统通用模型

正如图 2.1 中所示，偏好采集与推荐算法模块组成了完整的推荐系统。下面对该模型中的主要组成部分进行逐一介绍：

（1）偏好采集：于搜集用户与产品相关的数据信息，可以大致分为三类信息：用户信息、产品信息与用户产品交互信息。通过用户的注册、登录等操作，可以直接获取用户的部分特征信息。为每个用户建立能够体现其自身特质的信

息模型。用户偏好数据与产品数据作为最基本的数据类型，表现了用户与产品的本质特征，能够有效地对特定用户或产品进行区分，但无法体现两者之间存在的关系。

（2）知识数据库：其依然是一个与数据相关的模块，有时在进行推荐时并不依赖与用户的偏好或产品的属性信息，而是在预处理阶段，对原始数据进行建模，并将生成的数据模型直接供推荐算法使用。该数据库通常倾向于以某种形式表示用户与产品之间存在的关系。常见的数据模型有用户相似度矩阵、产品相似度矩阵以及用户产品-关系矩阵等。

（3）推荐算法：做为推荐系统的核心，推荐算法直接影响着该系统的推荐水准。它的主要作用就是基于数据模型，采取不同方式对用户的偏好以及需求进行预测，进而将用户可能感兴趣的产品形成推荐列表呈献给用户。推荐算法发展至今，已经得到了大量学者的深入研究，也涌现出了侧重点不同，处理方式不同的推荐方法，下面主要对一些常见的推荐方法进行介绍，并对比分析了方法自身的优缺点。

2.1.1 基于内容的推荐

基于内容（content-based）的推荐方法源自于信息获取领域[24]。作为较早出现的推荐方法，主要以项目的特征信息以及用户的历史信息为基础，进而分析用户对产品各项特征的倾向程度。这样就能得到每一位用户对于产品特征的偏好信息并建立用户偏好模型，通过衡量用户没有接触过的产品特征与其偏好信息的相关程度来决定是否对用户推荐该产品。虽然需要用到用户的历史信息，但整个过程中通常不需要引入用户对项目的意见评价。通过分析用户与产品交互的历史信息，重点在于发掘用户对于产品特征的倾向程度。该方法较早被应用在网页、书籍、饭店和电视节目的推荐过程中。为了得到目标用户感兴趣的产品，需要对该项目的所有描述性内容进行分析，获取其特征与用户的兴趣特征进行匹配，最终将相似度最高的产品项推荐给该用户。整个过程的重点在于选择一种相似度量方式去评估产品特征与用户兴趣特征之间的匹配程度[25]。如下式所示，就是度量相似性的一个函数：

$$u(c,s) = score(ContentBasedProfile(c), Content(s)) \quad (2.1)$$

式（2.1）中 ContentBasedProfile(c)表示用户的资料信息模型，Content(s)表

示产品的特征信息，而 score 函数表示不同类型的相似度计算方法，例如常见的通过计算向量夹角余弦值的余弦相似度[26]：

$$u(c,s) = \cos(\overrightarrow{V_c},\overrightarrow{V_s}) = \frac{\sum_{i=1}^{k} V_{i,c}V_{i,s}}{\sqrt{\sum_{i=1}^{k} V_{i,c}^2}\sqrt{\sum_{i=1}^{k} V_{i,s}^2}} \tag{2.2}$$

产品特征的选取往往取决于多种因素，其中不乏掺杂研究者主观意向的特征，所以对于同一类甚至同一件产品，其特征提取结果具有明显的不确定性。其次，在特征提取的过程中，如果遇到非结构化的数据（例如：新闻、用户评论等），就无法直接获取项目的特征信息，需要首先利用相关文本处理方法对数据进行预处理，并提取特征关键词，采用统计学的方法计算该文本数据中各个特征的权重，例如经典的文本特征 TF-IDF（Term Frequency–Inverse Document Frequency）[27]。此外，ContentBasedProfile(c)表示的用户资料信息模型的建立取决于机器学习过程所采用的方法，常见的有决策树[28]、朴素贝叶斯分类[29]、k 近邻、 Rocchio 算法[30]等。

基于内容的推荐方法的优势在于其推荐过程中未将用户对产品的意见评价考虑其中，所以在推荐结果中不易出现过于集中推荐热门项目的情形，同时也更利于发掘长尾项目。其次，因为最终的推荐来源于产品特征以及用户偏好信息，所以推荐结果具有较强的可解释性，并且能相对较好地应对冷启动问题。另外，大量成熟的机器学习方法也为基于内容的推荐提供充足的算法支持。

但是，特征提取做为基于内容推荐的关键环节，如果采用不同的特征提取方式会直接导致最终特征表示的差异化，从而使该方法具有了一定程度上的不稳定性。甚至对于一些特殊类型的数据，例如视频、图像等数据，特征提取的过程会变得愈发复杂，从而影响了整个推荐算法的表现。

2.1.2 协同过滤推荐

协同过滤推荐作为目前应用最为广泛的推荐技术，它的研究开始于 20 世纪 90 年代，并推动了整个个性化推荐领域的发展与繁荣。协同过滤主要利用邻域的思想，以用户与产品间的交互记录为依据，结合相似度计算方法，构造用户近邻集合或者产品近邻集合，进而对目标用户进行推荐。例如用户在选择产品时，通常会参考其周边好友的历史行为，如果其好友对某件产品具有较高评价，

那么该用户可能就会倾向于选择这件产品。或是通过历史行为记录去发掘选择了与目标用户产生交互的产品的其他用户还与哪些产品进行了交互。当前协同过滤算法主要分为基于内存（memory-based）与基于模型（model-based）两类：

1．基于内存的协同过滤：

对于基于内存的协同过滤，分别从用户与产品的角度出发，又可以将其分为基于用户（user-based）与基于产品(item-based)的两类方法。

（1）基于用户的协同过滤主要通过寻找与目标用户与产品间行为相似的近邻用户，将其近邻用户曾经选择或评价过的产品做为目标用户的候选推荐项，并采用相应的相似度计算方法进一步评估目标用户对推荐候选项集中各项产品的偏好程度，最终将用户最可能感兴趣的一项或多项产品推荐给目标用户。

表 2.1　用户电影评分矩阵

	Movie1	Movie2	Movie3	Movie4
uid001	3	4	5	4
uid002	4	?	?	?
uid003	3	1	?	2
uid004	4	2	5	?

表 2.1 所示为用户对电影的评分评分矩阵，可以看到每位用户并不是都对所有的影片进行了评分，这些未评分的影片自然成为了潜在的推荐项。假设对 uid003 进行推荐，首先从评分矩阵中可以获得该用户的评分向量[3,1,?,2]。之后通过与其余用户的评分向量进行相似度计算，构造目标用户的近邻用户集合，通过其近邻用户的历史评分记录选取评分较高的影片推荐给 uid003。常用的度量用户之间相似性的方法有余弦相似性（Cosine）、相关相似性（Correlatison）以及修正余弦函数弦相似性[31]。

虽然基于用户的协同过滤能够使用用户的历史行为记录进行推荐，但是这种方法也存在着一些问题。对于应用于商业的推荐系统，它所面对的产品数目是巨大的，但是用户选择或者购买产品的数目占产品总数的比重是相当小的。这直接导致了用户-产品数据具有较高的稀疏度，从而使得推荐系统可能无法为某些用户进行推荐。另外，随着用户与产品数量的增长，该方法的计算量也随之增大，尤其是用户相似度矩阵的生成。在实际问题中，面对数百万量级的用

户与产品，毫无疑问，基于用户的协同过滤方法的可扩展性仍需提升。

（2）如果说基于用户的协同过滤方法是基于用户对于朋友的信任，那么基于产品的协同过滤方法则是基于用户对于产品品牌的认可度进行的推荐。通俗的说，在基于产品的协同过滤的使用背景中存在一种假设：大部分人都会去选择的产品，目标用户也会去选择。不同于以上基于用户的协同过滤方式，基于产品的协同过滤方法从目标用户已经选择或评价的产品的角度出发，通过计算产品间的相似度，构造产品相似度矩阵，从而选取与当前产品最为相似的产品作为推荐候选项，最终形成推荐列表反馈给目标用户。在计算产品相似度时，依然可以沿用在基于用户的协同过滤中提到的常用方法，只是参与相似度计算的向量含义发生了略微变化。以表 2.1 为例，影片 Movie1 的评分向量为[3,4,3,4]表示所有用户对这部影片的评分情况。

2．基于模型的协同过滤

基于模型的协同过滤方法将推荐问题视作一个典型的机器学习问题，以样本中的用户与产品之间产生的行为数据作为训练数据集，训练出一个可以预测推荐结果的模型。在这一类方法中，模型的建立是整个算法的核心步骤。例如最具代表性的矩阵分解模型，传统的 SVD 分解同过采取适当措施补全原本稀疏且不完整的用户-产品评分矩阵，采用数学中的奇异值分解方法，作用于补全后的原始评分矩阵，通过选取影响度最大的多个奇异值，将分解后的矩阵映射到低维空间，之后通过用户或产品相似度计算来进行推荐。但由于传统的 SVD 分解的特点，需要面对较大的存储压力，且对大量数据的处理能力较差，并没有获得推荐领域的关注。直到 2006 年，SimonFunk 在其博客上公布了一个称为 Funk-SVD 的算法，重新引起了推荐领域对矩阵分解类方法的关注，也为之后多种衍生算法的产生奠定了基础。Funk-SVD 被后来的研究者也称为 Latent Factor Model（简称 LFM）。LFM 的主旨在于将用户-产品评分矩阵分解为两个低维矩阵相乘的形式，基于训练集中的样本值，通过最小化训练集的预测误差学习 P，Q 矩阵，从而最小化测试集预测误差。LFM 的提出获得了广泛的关注，之后很多基于矩阵分解的推荐模型都是在此基础上改进获得的，例如加入了用户偏置项的 BiasSVD、考虑用户邻域因素的 SVD++[32]以及融合时间信息的 timeSVD 等。

相比基于内容的推荐，协同过滤推荐最为显著的优势就是不需要刻意地对

用户或产品进行特征提取，这也预示着在整个推荐过程中不需要严格具备专业领域的相关知识。另外，协同过滤更适用于非结构化对象的推荐问题中，例如影像、书籍等产品的推荐，并且能够产生更加新颖的推荐结果，进一步发掘用户新的兴趣领域。正因为如此，协同过滤推荐在目前的互联网环境下得到了很好的应用与发展。

然而协同过滤对于数据的稀疏问题是十分敏感的。准确有效的推荐结果需要建立在大量的用户产品交互数据的基础之上。一旦数据缺失导致数据环境稀疏，协同过滤方法就会获取不到充足的邻域信息，同时也会对最终的推荐结果造成消极的影响。此外，协同过滤方法依赖于用户与产品间的交互信息，如果有新用户或新产品的加入，也就是所谓的冷启动问题，对新用户或新产品的相关推荐操作也会由于历史行为数据的缺失而变得困难。

2.1.3 其他推荐方式

1. 基于社交网络的推荐

社交网络被业界广泛地认为是继搜索引擎之后互联网领域的又一座金矿。如今，事实已经证明这种看法的正确性。国外知名的 facebook 与 twitter，国内家喻户晓的微信和微博，虽然它们之间存在着风格或机制上的差异，但无疑提高了人们之间的沟通效率。大量的信息储备也让社交网络成为了利于推荐系统应用的优势平台。所谓基于社交网络的推荐，重点在于利用其雄厚的数据基础（例如用户的位置信息、注册信息和邮件收发记录等）。用户在操作期间会逐步建立起以围绕自己的社交网络，通过分析网络中各节点的相关程度，对用户进行推荐[33,34]。文献[35,36]利用社交网络节点之间的相关度，通过计算它们之间的信任度，相比一般的协同过滤推荐，取得了更好的推荐效果。

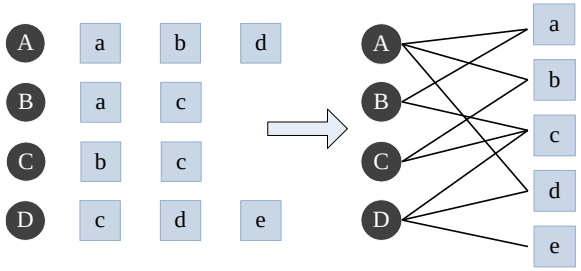

图 2.2 用户-项目二分图

2. 基于网格结构的推荐

在基于网格结构的推荐方法中，不考虑用户或者产品的具体信息，而是将这二类对象看作图中的节点。对于推荐问题中经常涉及的用户与产品二类信息，可以将这二者看做二分图中的隶属于不同集合的顶点，那么一个二元组（u，i）就代表用户 u 与产品 i 之间发生了交互，同时对应二分图中的连接顶点 u 与顶点 i 的一条边。根据此种转化方式，可以将原始数据转化为图的形式。如图 2.2 中所示，圆形节点代表用户，方形节点代表产品，左侧部分表示了每位用户产生过行为的所有产品，右侧部分则表示了按照以上原则转化而成的二分图。从图中可以看出用户 C 与产品 b,e 之间存在连线，那么就代表用户 C 在产品 b,e 上产生过行为。并且，用户与产品的相似度，以及用户对产品的倾向度都能够通过图中顶点间的相关性度量。

2.1.4 推荐系统的应用现状

推荐系统的发展过程一直被实际问题所驱动着，至今已经获得了丰硕的研究成果。此外，推荐系统的研究与发展一直是业界关注的焦点，无论是每年定期举办的专注于推荐系统研究的国际会议，还是互联网企业为了解决运营过程出现的实际问题而举办的各类推荐赛事，都不断地吸引着越来越多的研究者投入到个性化推荐领域的研究当中。毫无疑问，推荐系统已经渗透进了生活中方方面面，其主要应用领域有以下几个方面：（1）视频推荐。最为著名的电影网站 Netflix 较早将推荐系统作为辅助功能应用在了其电影租赁业务中，直至发展成以推荐为核心的在线观影网站。据 Netflix 估算，个性化推荐系统每年为其节约的业务费用可达 10 亿美金。国内的诸多视频网站，如优酷、爱奇艺等也在线提供了相关的个性化推荐服务，使网站更加智能化。（2）音乐推荐。国外的 SongTaste 音乐社区通过分析用户之间的相似度从而将可能喜欢的音乐推荐给每位用户。近几年，网易云音乐做为国内蹿升速度最快的音乐播放平台，以个性化推荐最为其最大特色，吸引了大批的用户。其推荐形式主要分为两类，一种是定时更新的离线推荐，另一种则是随时更新的在线推荐。丰富的推荐模式也使不同用户的兴趣得到了更好的覆盖。（3）社交推荐。在国外的 Twitter 与 Facebook，国内的微博等社交平台上，推荐系统被用来推荐可能与你兴趣相投或者社交关系有重叠的潜在好友。当然社交推荐的方式也有所不同，常见的方法

就是通过分析用户的兴趣特征或者好友关系来评估不同用户是否适合成为好友。此外，集成了地图定位的社交平台通常也会参考用户的位置坐标进行推荐。(4)新闻推荐。如今我们打开百度首页呈现给用户的不再单单是一个搜索框，还有大家都在关注的热点新闻标题，和针对个人浏览习惯的新闻推荐。今日头条作为一款基于数据挖掘的推荐引擎产品，为用户提供个性化的新闻信息，也是目前上升最快的互联网产品之一（5）电子商务推荐。亚马逊对推荐系统的成功应用以及其客观的效益转化率为之后推荐系统普遍应用于电子商务领域的进程拉开了帷幕。现今，无论是已成规模的淘宝、京东还是刚刚兴起的电商平台，毫无疑问都在使用推荐系统提升其市场竞争力。面对着各式各样的互联网产品，推荐系统几乎覆盖了所有互联网能触及到了领域，它所具备的能力和为用户提供的便利是会随着人们需求不断变化与提高而不断进步的，所以在目前这个推荐系统趋于普及化的时代，仍然有很长的路需要研究者们去深入探索。

2.1.5 推荐系统中亟待解决的问题

随着推荐系统应用的进一步深入，面临的问题也随之产生。在上一节介绍常用推荐模型的内容中已经对不同方法所面临的问题作了简要描述，例如基于内容的推荐中产品特征提取问题，基于模型的协同过滤中存在的训练过拟合问题等。推荐系统目前所面临的关键问题主要有两个方面：

（1）数据稀疏问题：推荐系统依赖于用户与数据之间的行为信息，但由于现存于互联网的数据规模庞大，用户群体仅仅与其中一小部分数据建立了联系。例如在电子商务领域，与用户群体存在关联的产品种类数目只占产品种类总数很小比重。以上情形就造成了一个数据稀疏的推荐环境，将这种现象对应到用户-产品矩阵中去，就会发现矩阵中只有很少的元素是非零元素。用户-产品矩阵的稀疏可能无法为推荐系统提供足够的数据量去挖掘用户之间、产品之间以及用户与产品间的潜在联系，直接影响了推荐结果的可靠性。

（2）冷启动问题：与数据稀疏问题类似，冷启动问题也是由于推荐系统对用户群体的历史行为与兴趣高度依赖性引起的。通常在已经聚集了大量用户群体的互联网应用中，推荐系统能够更好地得到应用。由于其数据优势，推荐效果往往也会显著提升。但在一些脱离实际应用的研究环境下，数据的缺乏无疑会成为研究过程中的障碍。冷启动问题描述是面对新的用户或产品的加入，在

其相关数据记录极度缺乏的情况下，如何对新用户进行推荐，如何将新产品推荐给用户。

2.2 形式概念分析及概念格相关理论

在哲学范畴中，概念被理解为由外延与内涵所构成的思想单元。德国数学家 Wille 在 1982 年首先提出了形式概念分析（Formal Concept Analysis，FCA），用于概念的发现、排序和显示[37]。下面首先对形式概念分析与概念格的基本理论进行介绍。概念格作为形式概念分析的核心数据结构,是基于形式背景中对象与属性之间的二元或多元关系建立起的一种概念层次结构。Hasse 图能够清晰地体现概念格中概念之间的泛化与特化关系，因此被看做是进行数据分析的有力工具。下面给出形式概念分析的相关定义：

定义 2.1（形式背景）：形式背景$K=(G,M,I)$是由两个集合 G（对象集）和 M（属性集）以及 G 中元素与 M 中元素之间的关系 I 所构成的。集合 G 中的元素称为对象，集合 M 中的元素称为属性。若$(g,m)\in I$或 gIm，则表示对象 g 具有属性 m，或称属性 m 属于对象 g。

表 2.2 形式背景示例一

	考古遗址	沙滩	欧元	溪流	滑雪区
雅典	1	1	1	0	0
因斯布鲁克	0	0	1	1	1
巴黎	0	0	1	1	0
罗马	1	1	1	1	0

表 2.2 为四座欧洲城市与城市特有属性构成的形式背景，可以看到表中各对象与属性的关系值非 0 即 1。例如，如果巴黎流通欧元，那么它所对应的属性值则为 1,否则为 0。这种类型的形式背影由于其仅用 1 和 0 表示关系的存在与否，所以形式背景仅包含二元关系，基于形式背景下的概念格称为经典概念格。当然现实中对象与属性之间的关系不仅仅只有 0 和 1 的形式，还可能有多个离散值或连续实值的的情况存在。

定义 2.2（伽罗瓦联接）：对于任意的对象集合$A\subseteq G$，定义函数$f(A)=\{m\in M\mid\forall g\in A, gIm\}$(集合 A 中对象共同具有属性的集合)。相应地，

对于任意的属性集$B \subseteq M$，$g(B) = \{g \in G \mid \forall m \in B, gIm\}$(具有 B 中所有属性的对象的集合)。若$A = g(B), B = f(A)$，则称集合 A 与 B 满足迦罗瓦联系，函数$f, g$为迦罗瓦联接。

定义 2.3（形式概念）：形式背景$K = (G, M, I)$上的形式概念是以二元组$C = (A, B)$的形式存在的，其中$A \subseteq G, B \subseteq M$，且集合 A 与 B 满足迦罗瓦联系，则将对象集 A 称作概念$C = (A, B)$的外延，属性集 B 为概念的内涵。

对于形式背景$K = (G, M, I)$，$A_1, A_2, A \subseteq G$，$B_1, B_2, B \subseteq M$存在以下性质：

（1） $A_1 \subseteq A_2 \Rightarrow f(A_2) \subseteq f(A_1)$，$B_1 \subseteq B_2 \Rightarrow g(B_2) \subseteq g(B_1)$

（2） $A \subseteq g(f(A)), B \subseteq f(g(B))$ $A \subseteq g(B) \Leftrightarrow B \subseteq f(A)$

（3） $f(A_1 \cup A_2) = f(A_1) \cap f(A_2)$，$g(B_1 \cup B_2) = g(B_1) \cap g(B_2)$

（4） $f(A_1 \cap A_2) = f(A_1) \cup f(A_2)$，$g(B_1 \cap B_2) = g(B_1) \cup g(B_2)$

定义 2.4（层次序）：$C_1=(A_1, B_1)$ 和 $C_2=(A_2, B_2)$是同一形式背景下的两个概念，若 $A_1 \subseteq A_2$（$B_1 \supseteq B_2$），则称 C_1 与 C_2 之间存在偏序关系，其中 C_1 称为 C_2 的子概念，C_2 则是 C_1 的父概念，记作 $C_1 \leq C_2$，关系≤称为层次序。若不存在中间概念（A，B），使得（A_1，B_1）≤（A，B）≤（A_2，B_2），则 C_1 称为 C_2 的直接子概念，C_2 称为 C_1 直接父概念。

概念格就是由形式背景下的所有概念以及概念之间的层次序构成的，通常以 Hasse 图的形式对其格结构进行可视化，图 2.3 所示为表 2.2 中形式背景下概念格的 Hasse 图。为便于表示我们将表 2.2 所示形式背景中的对象按从上至下的顺序依次标注为 a、b、c、d,属性按从左至右的顺序标记为 i、j、k、l、m。

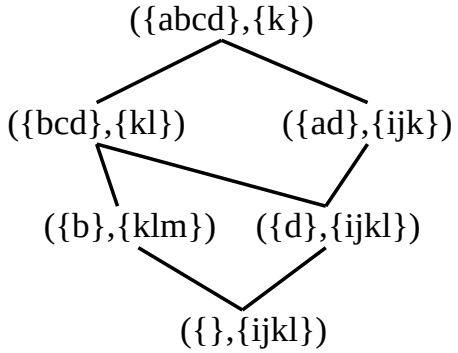

图 2.3 形式背景下的概念格 Hasse 图

除了以上基本概念之外，为了便于之后章节中推荐过程的描述，本文在这

里引入邻域节点和对象的起始概念的定义：

定义 2.5（邻域节点）当不存在概念（A_3，B_3）满足（A_1，B_1）<（A_3，B_3）<（A_2，B_2）时，称（A_1，B_1）是（A_2，B_2）的子节点。当概念（A_1，B_1）与（A_2，B_2）具有相同的直接父概念（或直接子概念）时，则称（A_1，B_1）与（A_2，B_2）存在兄弟节点的关系，一个概念的所有的子节点与兄弟节点构成了该概念的邻域节点。

定义 2.6（对象的起始概念）已知对象 a，且 $a\in A_1$，当不存在概念（A_2，B_2）<（A_1，B_1），并且 $a\in A_2$ 时，我们称概念（A_1，B_1）为对象 a 的起始概念。

根据定义 2.6，本文提出命题如下：

命题 2.1 一个对象具有的属性集即为此对象起始概念的内涵集。

证明:设对象 a 的属性集 $H=\{m_1,m_2,\ldots,m_k\}$，对象 a 的起始概念 C =（A,B），由于 $a\in A$，可知 $B\subseteq H$。当 $B\subset H$ 时，根据定义 2 可以构造概念 C’=(g(H)，H)，再结合条件 $B\subset H$，得出 C’<C，显然这与起始概念的定义相矛盾，所以 B=H，命题 2.1 得证。

由命题 2.1 可知，对象 a 的起始概念的内涵集即为对象 a 的属性集，结合层次序的定义，对象 a 将不再出现在其起始概念的子节点概念的外延集中。所以对象 a 的起始概念是包含对象 a 的概念中唯一能够完整描述对象 a 的特性的。在之后章节中会结合推荐问题详细阐述起始概念在算法中的作用。

2.3 形式概念分析在推荐领域的应用

形式概念分析及概念格相关理论在推荐系统方面的应用研究仍处在探索阶段。2006 年 Boucher Ryan 等人[38]首次提出了将形式概念分析与协同过滤算法结合的思想，该文献将概念格作为用户与产品间关系信息的存储载体，通过利用概念之间的偏序关系，探索性地搜索近邻概念，并从中获取推荐候选项，虽然并未具体提出明确的搜索策略，但为之后的进一步研究提供了方向。文献[39]将概念格应用在了广告业词汇的推荐上，通过构造基于以广告公司为对象，所购买的广告词为属性的形式背景，并对其建立概念格，使用挖掘关联规则的方式来为广告公司提供个性化的广告词推荐服务。Tomohiro Murata 等人[40]结合形式概念分析提出了一种基于知识的推荐模型，该模型的核心结构主要分为三个部分：（1）知识源本体，用于知识表示，该部分描述了产品来源与相关特征的

综合信息；（2）用户配置文件本体，用来有组织的存储用户的历史和行为信息，通过分析使用用户的请求与喜好，从而是搜索更加快速；（3）形式概念本体，它是对所有实体和其属性以及实体间关系的形式化描述，提供了以个捕捉关键区别的通用映射域，加快了推荐候选项集的生成。通过以上三部分的协同工作，最终为用户提供个性化的推荐项。另外，也有研究者为了应对多值背景将模糊形式概念分析应用在了推荐问题中[41,42]。国内也有部分学者将形式概念分析相关理论与推荐系统进行了结合。文献[43]中提出了一种基于概念格的图书协同推荐模型，利用概念之间的偏序关系，寻找与目标用户相近的用户群体，从这些相似用户的阅读记录中挑选书籍推荐给目标用户。文献[44]从大量的社交数据中抽取用户知识，以概念格为载体，构造了用户属性概念格和用户社交概念格结合带重启的随机游走算法，进行朋友推荐。

2.4 本章小结

本章分别介绍了推荐系统与形式概念分析相关理论知识，对主流的个性化推荐方法进行了列举介绍，并对推荐系统中仍普遍存在的问题及各领域中推荐系统的应用情况作了叙述，给出了形式概念分析及概念格理论的相关定义及性质，并详细分析了形式概念分析及概念格理论在推荐系统中的应用现状。

3 概念格中起始概念索引的构造

本章主要介绍了数据处理阶段的相关工作，着重对形式背景的转化、概念格的生成以及本文提出的起始概念索引的定义与构造方法进行了介绍。

3.1 形式背景的转化

推荐问题通常面对的数据主要分为显式（如用户对产品的评分）与隐式（如用户的购买、浏览记录）两类。在隐式数据中，通常用户对项目只发生过购买或浏览类的操作，并未对其进行定量地评价，所以用 0 与 1 来表示用户与项目之间是否存在关系。对照形式背景定义中的描述，可以自然地将隐式数据中的用户看作对象，项目作为属性，完成隐式数据到形式背景的转化，转化过程如表 3.1 所示。

表 3.1 隐式数据向形式背景的转化

	item1→atrr1	item2→attr2	item3→attr3	item4→attr4
user1→obj1	0	1	1	0
user2→obj2	1	0	1	1
user3→obj3	1	1	1	1

不难想象，现实很多应用场景中用户与项目之间不仅只存在表示二元关系的隐式数据。但是在经典概念格的研究范畴中，其所依赖的形式背景中只包含二元关系，即对象与属性的关系值非 0 即 1。如果遇到显示数据的情形，由于用户与项目之间的非二元关系，往往不能直接基于多值背景进行概念格的构造，需要将其转化为只包含二元关系的形式背景。根据文献[45]对多值背景的介绍与分析，对其类型进行了如下区分：

（1）数值型多值背景

在数值型多值背景中，对象与属性之间的关系包含不同数值的有限集合中的某一数值元素唯一表示的。如表 3.2 所示，用户对电影的评分均为取值在 1-5 之间的某个数值。对应于形式背景中的要素，用户可以看做对象，电影作为属性，对象与属性间的关系值即为用户对电影的评分值。

表 3.2 数值型多值背景

	movie1	movie2	movie3	movie4
user1	2	5	1	2
user2	3	4	3	2
user3	1	2	2	4

（2）区间型多值背景

区间型多值背景中，对象与属性之间的关系以区间的形式表示，准确地说是由包含不同区间的区间集合中的某一区间项唯一表示的。正如表 3.3 所示为高中生体育达标的不同的评级标准与其对应的各测试项目之间的成绩区间。将评级标准看做形式背景中的对象，测试项目作为属性，那么区间则代表对象与属性间的关系表示。

表 3.3 区间型多值背景

	立定跳远（米）	50 米跑（秒）	引体向上（次）	握力体重指数
不及格	[1.97，2.10]	[8.4，9.0]	[4，8]	[46，54]
及格	[2.11，2.32]	[7.8，8.3]	[10，14]	[55，71]
良好	[2.35，2.53]	[6.8，7.7]	[15，19]	[73，85]
优秀	[2.55，2.63]	[6.2，6.7]	[20，25]	[87，94]

（3）语言型多值背景

语言型多值背景在某种程度上类似于离散数值型背景，只是将代表关系的离散型数值替换为了一些表示程度的词汇。表 3.4 中所示为学生在四门学科中获得的评价，评价值为由包含表示优劣程度的词集={优，良，中，差}中的元素表示。

表 3.4 语言型多值背景

	语文	数学	物理	英语
学生 1	中	良	良	差
学生 2	良	优	良	优
学生 3	优	差	中	良
学生 4	优	优	优	良

通过分析以上三种类型的多值形式背景可知，如果简单地将多值背景中“对象与属性之间的关系值”理解为形式背景中的“对象存在某种属性”，那么通过这种方式转化而成的形式背景，相较于原始的多值背景将造成部分信息的损失。如表 3.2 所示，<user1,movie2>=5，<user1,movie3>=1,分别表示 user1 对 movie2 的评分为 5，对 movie3 的评分为 3。按上述方法转化为二元关系则为 <user1,movie2>=1，<user1,movie3>=1，表示 user1 观看过 movie2 与 movie3。可以看出，原始的多值背景下，由于 user1 对于 movie2 的评分较高，对 movie3 评分较低，进而推断 user1 相较于 movie3 更对 movie2 感兴趣。但是在转化过后的二元关系中，除了表示 user1 看过这两部电影以外，没有任何其他含义，导致了转换过程中的信息损失。

表 3.5 多值背景通过阈值法转化生成的形式背景

	movie1	movie2	movie3	movie4
user1	0	1	1	0
user2	1	1	1	0
user3	0	0	0	1

对于以上问题，通常的处理方法是设定一个阈值 p。例如在表 3.2 的多值背景下设定阈值 p=2.5，那么当评分值大于 p 时，则转化为二元形式背景后的关系值为 1，否则为 0，转换形成的二元形式背景如表 3.5 所示。这种方式将原始的关系数值范围通过阈值一分为二，实质上是对分值进行了较为粗糙的分类，信

息损失问题虽得到一定程度缓解，但仍明显存在。为了解决这类问题，存在另一种利用概念定标原理的方法[46]，该方法为多值背景中的每个属性赋予一个概念标尺。仍以表 3.1 为例，该用户电影评分表中，分值取值范围为 1-5 分。设定概念标尺来表示用户对于电影的兴趣度，其中包含三个度（High，>=4），（Middle，>=2），（Low，<2），则原始的多值背景可转化为表 3.6 的形式。

表 3.6　多值背景通过概念定标法转换生成的形式背景

	Movie1			Movie2			Movie3			Movie4		
	H	M	L	H	M	L	H	M	L	H	M	L
user1	0	1	0	1	0	0	0	0	1	0	1	0
user2	0	1	0	1	0	0	0	1	0	0	1	0
user3	0	0	1	0	1	0	0	1	0	1	0	0

3.2　形式背景下的概念格的生成

为了够造以形式概念为单位的邻域环境，在确定了形式背景之后，需要在其基础上构造出完整的概念格。为了便于讨论概念格的构造过程，给出表 3.7 所示的形式背景，其中对象表示推荐背景下的用户，属性表示项目。

表 3.7　形式背景示例二

	属性1	属性2	属性3	属性4
对象1	0	1	0	1
对象2	0	0	1	1
对象3	0	0	0	1
对象4	1	0	0	0
对象5	1	1	1	0
对象6	0	0	1	0
对象7	1	1	0	0

虽然表 3.7 能够清楚地反映用户（对象）与项目（属性）之间的关系，但并不能从中直接提取用户或产品的邻域信息。而概念格具有明显的聚类特性，概

念格中的层次序也能清楚地反映对象集间的泛化与例化关系，通过概念之间的关系可以直接获取邻域信息。

由之前章节中对概念格构造方法的介绍可知，其构造方法分为批处理与渐进式两种方式。批处理方式在生成所有概念的同时不会伴随产生概念之间偏序关系，需要在获得所有概念之后重新进行关系的补全，这样一定程度上提高整个构造算法的时间复杂度。由于之后的推荐过程中需要借助概念之间的偏序关系，所以概念格构造方法的确定为更利于概念间关系生成的渐进式构造算法，并选择了 Godin 算法做为最终的构造方案。

Godin 算法是一种经典的渐进式构造方法。该方法首先将概念格初始化为空，之后每当有新的对象加入，就在原有概念格的结构上进行动态调整。随着后续的数据对象加入，如果使用批处理方式，则需要在扩充的形式背景基础上重新进行所有概念与关系的生成，影响了够格效率。

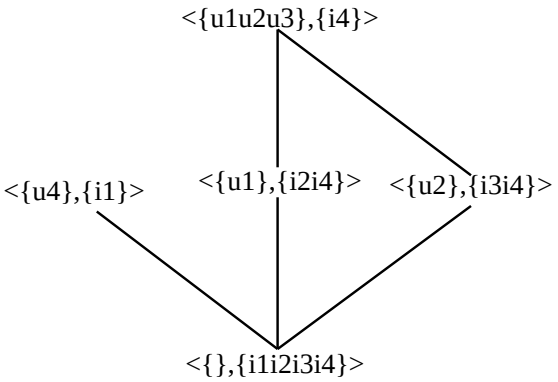

图 3.1　概念格的渐进式构造中间结果

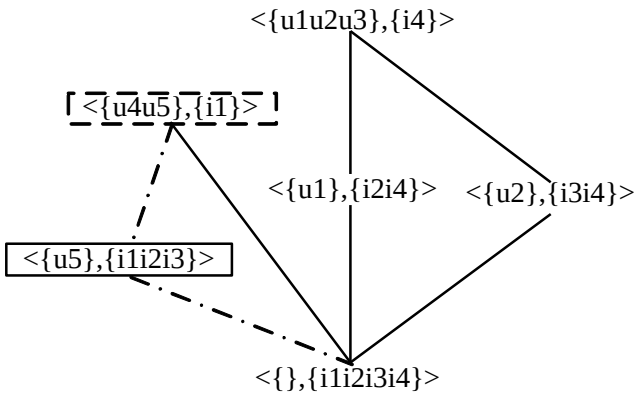

图 3.2　概念格的渐进式构造中间状态

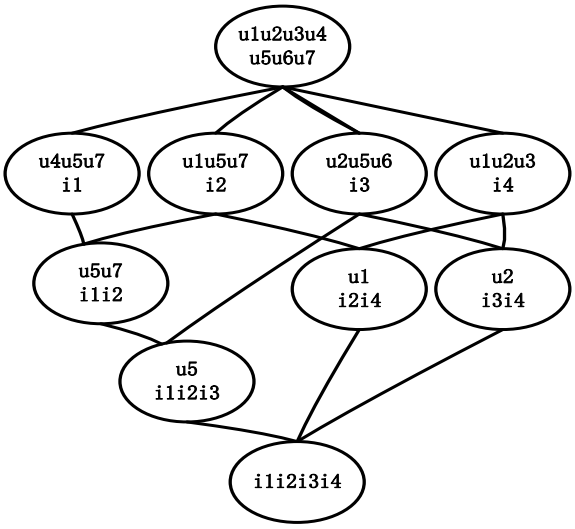

图 3.3　概念格构造完成的结构图

图 3.1 所示为表 3.7 中包含对象 1—对象 4 的渐进式够格中间结构。记对象为 u，属性为 i。当 u5 加入时，对其属性集{i1,i2,i3}与概念格中的现存概念的内涵分别求交集，并根据结果进行分类处理。（1）当与概念<{},{i1,i2,i3,i4}>的内涵求交集，得出集合{i1,i2,i3}.而图 3.1 中的结构中并没有内涵为该集合的概念存在。这时候需要新生成概念<{u5},{i1,i2,i3}>，并加入到原有的格结构中。如图 3.2 中所示，<{u5}，{i1,i2,i3}>为新生成概念，并且与概念<{}，{i1,i2,i3,i4}>之间建立了父子关系。(2)当与图 3.1 中的概念<{u4},{i1}>的内涵进行交集运算后，得到集合{i1}，即与概念<{u4},{i1}>的内涵相同，这也说明了对象 u5 的属性集包含于该概念的内涵中。此时需要对<{u4},{i1}>的外延进行更新，结果如图 3.2 中虚线框中的概念所示。值得注意的是，更新后的概念<{u4,u5},{i1}>与新生成的概念<{u5},{i1,i2,i3}>和原有概念<{},{i1,i2,i3,i4}>之间均存在直接父子关系，出现关系冗余的情形。观察图 3.2 可知，需要将概念<{u4,u5},{i1}>与原有概念<{},{i1,i2,i3,i4}>之间存在的直接父子关系删除，才能得到正确的构造结果。

根据以上讨论分析，可以将新对象加入时的概念格中的节点分为三种类型：

（1）更新节点：将新对象加入该节点的对象集中，属性集不变。

（2）生成节点：该节点为格中原有节点，当新对象加入时，该节点自身所包含信息不发生变化。但会产生新的节点。

（3）新生成节点：与生成节点之间进行集合运算形成的原格结构中不存在

的新节点。

在根据新对象加入时节点的操作方式对节点类型进行区分后，下面将 Godin 算法描述如下：

算法 3.1：概念格的渐进式构造算法

输入：概念格 G，新对象 x，x_attr 表对象 x 的属性

输出：更新后的概念格 G

```
If 概念格 G 的最小上确界 sup(G)=( Ø,Ø)
    then 初始化 sup(G)←({x},{x_attr})；
    else If sup(G)的内涵不包含{x_attr}
            then If sup(G)的外延为空
                    then 更新最小上确界 sup(G)的内涵；
                    else 新节点 node=( Ø,sup(G).intent∪{x_attr})；
                         将新节点添加至概念格 G 中；
                         将 node 连接为原先 sup(G)的子节点，并完善之间父子关系；
                         将 node 作为新的最小上确界 sup(G)；
                End if
        End if
    For node in G
        If node 的内涵集 ⊆{x_attr}
          then 该节点作为更新节点将 x 添至其外延中；
               If node 的内涵={x_attr}
                  then 构造完成；
               End if
          else inte = node 的内涵集 ∩{x_attr}；
               if 内涵为 inte 的节点没有被生成过
                  then 生成新节点 new_node=(node 的外延∪{x},inte)；
                       将 new_node 添加至概念格 G 中；
                       从跟更新节点与新节点集合中内涵基数小于 new_node 的节点
                       中寻找其直接父节点，并建立父子关系；
                       删除冗余关系，即 new_node 直接父节点与其直接子节点间存
                       在的关系；
               End if
               if inte={x_attr}
                  then 构造完成；
               End if
        End if
    End for
End if
```

从算法 3.1 中可以看出，Godin 算法在构造概念格的过程中是按照了之前的

分析与对节点的分类进行的。整个过程中，偏序关系也就是节点之间的连线，也在随着节点的更新和生成进行着调整，最终使加入新对象之后的概念格仍然具有完备性。

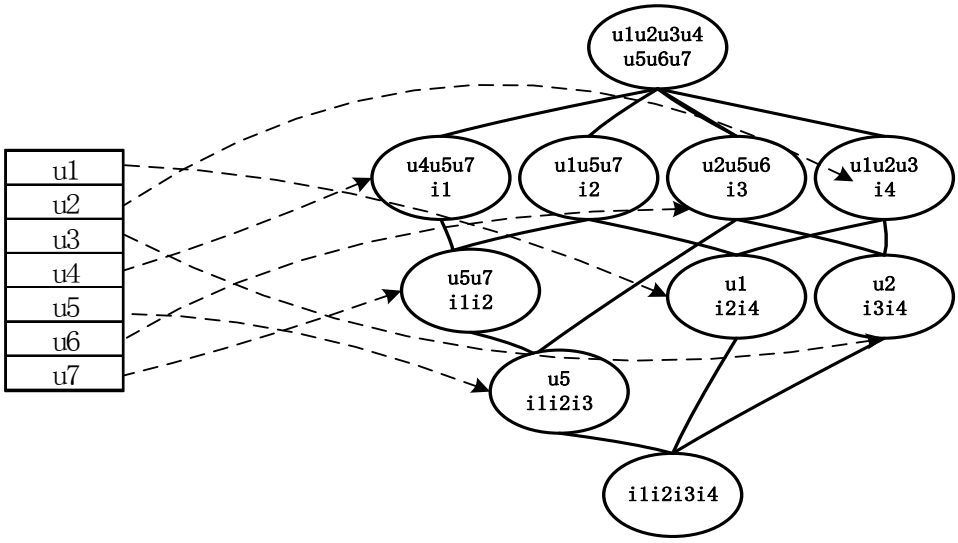

图 3.4　对象与其起始概念的对应关系

3.3　起始概念索引的构造

当需要对目标用户 u 进行推荐时，虽然每个用户都拥有唯一标识，但是在概念格的所有概念中，外延包含用户（对象）u 的概念通常不只存在一个。所以，在概念格范畴中，如果选择从概念的角度出发，则需要确定唯一一个与用户 u 相关的概念作为出发点。由之前定义可知，对象 u 的起始概念的内涵集包含了该对象所具备的所有属性，也是所有外延集包含该对象的概念中，唯一的内涵集包含其所有属性的概念。通过分析以上特性可以看出，任一对象的起始概念是所有概念中唯一能够完整描述该对象特性的概念，所以更适合作为寻找其邻域概念的起点。

然而在整个推荐过程中，如果每一次推荐都需要从概念格中定位目标用户（对象）的起始概念，那么就会产生等同于用户数量的定位次数，这其中必定会出现大量的重复性搜索，也会一定程度上影响整个算法的执行效率。在数据库相关技术中，索引的引入很大程度上提升了数据检索的效率。依据索引的构造思想，假设可以一次性地建立所有用户（对象）与起始概念之间的对应关系，那么当对某一用户进行推荐时，就能够借助已有的对应关系快速获取与之相应

起始概念的位置，有效地减少了大量的重复工作。借助哈希索引的思想，可将对象视为索引键，对象与起始概念间的映射作为哈希函数，具体定义如下：

定义3.1（起始概念索引） 设基于形式背景K=(G，M，I)下的概念格为G，对象a∈G为索引键，则映射I＝{a→c|∀ a∈G，c为对象a的起始概念}为联系对象与起始概念的哈希函数。

由命题2.1可知，任一对象的起始概念的内涵集，即该对象的属性集是唯一的。所以该对象的起始概念在概念格中也具备唯一性。根据以上理论基础，下面以伪代码的形式详细描述包含所有对象的起始概念索引的构造算法：

算法 3.2：起始概念索引构造算法
输入：概念格 G
输出：起始概念索引 Gindex

```
1.  For node in G
2.      If Gindex 包含了所有对象
3.          then 构造完成;
4.      End if
5.      For obj in node 的外延
6.          If 对象 obj 存在于 Gindex 中
7.              then 循环下一个对象;
8.          End if
9.          设置标记变量 flag=true;
10.         For chd in node 的直接子节点
11.             If obj 存在于 chd 的外延中
12.                 then Flag=false;
13.                 跳出循环;
14.             End if
15.         End for
16.     End for
17.     If flag is true
18.         then Add(Gindex,obj→node);
19.     End if
20. End for
```

算法 3.2 中的最外层循环是对概念格中的所有概念的遍历。由于在本文所采用的概念格的存储结构中，每一概念都包含了其父节点及子节点的所有信息。所以，可以直接获取当前概念的子概念，并根据起始概念的定义判定当前概念是否为某对象的起始概念。图 3.4 左侧部分表示起始概念索引中的索引键（对

象），带有指向的虚线则表示对象与相应起始概念之间的对应关系。通过构造起始概念索引，使得在进行推荐时可以以目标用户（对象）作为索引键直接得到其起始概念的信息。

3.4 本章小结

本章从形式背景的转化、概念格的生成与起始概念索引的构造三个方面，对基于概念邻域的推荐的数据处理工作进行了叙述。介绍了不同类型多值背景向单值背景的转化方法，分析了概念格的渐进式构造过程并给出了代码描述，通过分析起始概念定位中存在的问题，提出了起始概念索引的定义与构造方法，并给出了代码描述，提升了起始概念定位的效率，也一定程度提高了整个算法的可行性。

4 基于概念邻域的协同过滤算法

4.1 问题的形式化描述

为了将推荐的应用背景映射到形式背景中，将用户看做形式背景中对象，产品作为属性，那么对特定用户进行产品推荐就可以看做对形式背景中对象进行属性推荐。为了选取候选属性，结合概念间的层次序做如下分析：

(1) 在概念 Ci 的父节点中，其内涵集应包含于概念 Ci 内涵集，所以对概念 Ci 的父节点进行探索无法从中获取对象 u 的属性以外的其它属性。

(2) 概念 Ci 的兄弟节点的内涵集包含其与概念 Ci 共同所属父节点内涵集中的所有属性，同时也包含在概念 Ci 的内涵集中未出现过的属性，所以在概念 Ci 的兄弟节点中可以获取额外的属性作为候选属性。

(3) 概念 Ci 的内涵集包含于其子节点内涵集，显然其子节点的内涵集中可以提供额外的属性作为候选属性。

根据以上分析可知，对于起始概念 Ci，只有在它的兄弟节点与子节点的内涵集中存在未在概念 Ci 的内涵集中出现过的属性，即候选属性。本文定义候选项集来描述以上过程中获取到的属性（产品）所构成的集合，定义如下：

定义4.1（候选项集）设对象u的起始概念为$C=(A，B)$，概念C的所有兄弟节点，子节点为$C_1=(A_1，B_1)$，$C_2=(A_2，B_2)$，…，$C_k=(A_k，B_k)(k\geq 0)$，则对象u的候选项集$CS=B_1\cup B_2\ldots\cup B_k/B$。

结合文献[25]中对推荐系统的形式化定义，可以将基于概念邻域的Top-N推荐算法的形式化定义描述如下：

设C是所有用户的集合，S是所有产品的集合，由原始数据转化生成的形式背景$K=（C，S，R)$，基于形式背景K生成的概念格记为G，CS为探索概念邻域获取的推荐候选项集，且$CS\subseteq S$，设效用函数Pre()可以计算用户c对产品s的偏好度，即$Pre:C\times S\rightarrow R$，R是一定范围内的全序非负实数，则推荐问题所要获取的结果是使偏好度值R最大的那些产品s^*，如式(4.1)所示:

$$\forall c\in C,s^*=argmax_{s\in CS}Pre(c,s) \tag{4.1}$$

由式4.1可知，达到最终的推荐目的需要经历两个阶段。第一阶段在于推荐候选项集的构造，其作用在于将推荐产品的选取范围由整个产品集缩小为包含

少量产品的候选项集，可以视作一次粗略地筛选。第二阶段的关键在于效用函数Pre()的确定，它的作用就是进一步过滤掉候选项集中那些不适合作为最终推荐项的产品，同时将剩余的产品项作为推荐结果反馈给用户。

4.2 构造推荐候选项集

根据概念间的偏序关系，越是处于概念格下层的概念所包含的对象越特殊，因为这些对象具有更多的多属性。在确定推荐对象即目标用户的起始概念之后，本文利用概念格中概念之间的偏序关系，通过探索邻域概念中的内涵集合直接获取推荐候选项。为了构造推荐候选项集，并尽量使候选项构造充分，本文在这里采用控制递归深度的方法探索起始概念的子节点及兄弟节点：

算法 4.1：构造候选项集算法
输入：概念格 G，起始概念索引 Gindex，目标用户 u，递归深度 n
输出：用户 u 的推荐候选项集 Cands

```
1. 通过起始概念索引 Gindex，直接获取用户 u 的起始概念 C(E,I)；
2. 通过子节点获取候选项 Cchd ← GetItemOfChildren(C,n)；
3. 通过兄弟节点获取候选项 Csib ← GetItemOfSiblings(C,n) ；
4. 初始化 Cands；
5. Cands ← Cands ∪ Cchd ∪ Csib；
6. Removehad(Cands,u)，去除与用户 u 存在关系的项目；
```

算法 4.1 为构造候选项集的主体部分,根据定义 4.1,通过合并由子节点获取的候选项集与从兄弟节点获取的候选项集,并除去之前与用户 u 存在关系的产品,得到最终的候选项集。其中函数 GetItemOfChildren 与 GetItemOfSiblings 的执行过程描述如下：

函数 4.1：GetItemsOfChildren ( )
输入：起始概念 Ci，递归深度 n
输出：通过子节点后去的候选项集 CSchd

```
1. n=n-1
2. For c in Ci 的直接子节点
3.     If n!=0
4.         then   CSchd ← CSchd ∪ GetItemOfChild(c,n)
5.         else   CSchd ←CSchd ∪ c 的内涵
6.     Endif
7. Return CSchd
```

函数 4.1 从起始概念出发，通过探索子节点获得候选项集 CSchd。根据定义 2.4、2.5 可知，一个概念的内涵集一定是它任意子节点所包含内涵集的真子集。所以从第 5、6 行中可以看出当递归进入下一层时并未将当前探索到的子节点中内涵集的项目添加到候选项集中。只需如第 9 行所示，在递归进行至最后一层时将所有当前探索到的子节点中内涵集项添加到候选项集中，就能够能保证候选项集充分扩展。

函数 4.2：GetItemsOfSiblings ( )
输入：起始概念 Ci，递归深度 n
输出：通过兄弟节点获取的候选项集 CSsibs

```
1.  n=n-1
2.  For c in Ci 的直接父节点
3.      For d in c 的直接子节点
4.          If n!=0
5.              then CSsib ← CSsib ∪ d 的内涵;
6.                   CSsib ← CSsib ∪ GetItemOfSiblings(d,n);
7.              else CSsib ←CSsib ∪ d 的内涵;
8.          End if
9.      End for
10. End for
11. Return CSsib
```

函数 4.2 的执行过程与函数 4.1 类似。但不同的是，函数 4.2 第 5 行中将当前递归层中探索到的兄弟节点内涵集中的项目添加到了候选项集中，原因是兄弟节点的内涵集之间并不存在包含与被包含的关系，所以在每层递归需要将当前兄弟节点的内涵集项添加至候选项集才能保证构造的充分性。

4.3 用户对产品的偏好度计算

在候选项集生成之后，需要确定效用函数 Pre( )以计算产品对于用户的推荐度，并对每个用户的候选项集进行再次过滤，从而得到最终推荐项。在 Top-N 推荐中需要从中筛选出用户最有可能感兴趣的 N 项产品，传统的基于邻域的协同过滤算法通常利用用户相似度或物品相似度的方法来计算产品对于用户的推荐度。本文以概念格作为数据载体，结合概念相似度提出的全局偏好度与邻域偏好度两种偏好度定义，定义了两种产品对于用户推荐度的计算方法。首先介绍概念相似度的相关定义及度量方法。

4.3.1 概念相似度

在推荐问题的研究过程中经常涉及到用户相似度或产品相似度等概念。通过计算相似度，能够量化同类事物间的相关程度,便于比较与推荐结果的生成。。但是在概念格中，每个概念都是由对象和属性共同构成的，所以本质上表现的是用户与产品间的关系，而不能单独用户或产品的角度去思考问题。如果只考虑二者其一，无异于传统的相似度度量方法，更无法通过挖掘概念间的内在联系去改进推荐效果。为了将问题的求解过程置于概念格的结构背景下，本文引入概念相似度的相关理论及度量方法。

与形式概念分析相比，领域本体中的概念相似度问题已经得到了更为广泛的关注与研究[47-51]，例如基于字符、基于图以及基于知识的相似度度量方法。但在领域本体问题中，概念是作为一种数据标签的表现形式，而形式概念分析中的概念是不包含数据标签的，它可以看做是由对象与属性所构成的双聚类结构（对象聚类与属性聚类）。下面给出相似性度量的形式化定义[52]：

定义 4.2 一种相似度测量方法 S 就是定义在集合 X 的笛卡尔积上的非负实值函数，形式如下：

$$\mathrm{S}: \mathrm{X} \times \mathrm{X} \rightarrow \mathrm{R}$$

S 满足以下条件：

1. $\exists \mathrm{s0} \in \mathrm{R}: -\infty < S(\mathrm{x},\mathrm{y}) \leq \mathrm{s0} < +\infty, \forall x, y \in X$
2. $\mathrm{S}(\mathrm{x},\mathrm{x}) = \mathrm{s0}, \forall \mathrm{x} \in \mathrm{X}$
3. $\mathrm{S}(\mathrm{x},\mathrm{y}) = \mathrm{S}(\mathrm{y},\mathrm{x}), \forall \mathrm{x},\mathrm{y} \in \mathrm{X}$

如果 S 又同时满足：

1. $\mathrm{S}(\mathrm{x},\mathrm{y}) = \mathrm{s0} \leftrightarrow \mathrm{x} = \mathrm{y}$
2. $\mathrm{S}(\mathrm{x},\mathrm{y})\mathrm{S}(\mathrm{y},\mathrm{z}) \leq [\mathrm{S}(\mathrm{x},\mathrm{y}) + \mathrm{S}(\mathrm{y},\mathrm{z})]\mathrm{S}(\mathrm{x},\mathrm{z}), \forall \mathrm{x},\mathrm{y},\mathrm{z} \in \mathrm{X}$

那么相似度测量函数 S 又称为度量相似度函数。

在模式识别与数据挖掘中，相似性函数通常是基于实数值与离散值的向量集合定义的。对于离散值向量的相似度度量方法来说，通常都是基于集合间的关系运算与基数构造而来的。常见的几种与集合相关的离散值向量的相似度测量方法有：

$$\text{Jaccard index } S_{Jac} = \frac{|x \cap y|}{|x \cup y|} \tag{4.2}$$

$$\text{Sorenesen coefficient } S_{Sor} = \frac{2*|x \cap y|}{|x| + |y|} \tag{4.3}$$

$$\text{Symmetric difference } S_{Xor} = 1 - \frac{x \ominus y}{|x \cup y|} \tag{4.4}$$

其中$x \ominus y$表示集合 x 与集合 y 的对称差：

$$x \ominus y = (x \backslash y) \cup (y \backslash x) \tag{4.5}$$

为了将这些基于集合运算的相似性计算方法推广到形式概念中去，需要首先对形式概念的构成进行分析。由之前定义可知，形式概念本质上是由两个集合所构成，分别为对象集与属性集。所以，从形式概念的结构特点可以容易地想到通过分别计算对象集之间与属性集之间的相似度，并分别加权求和的方式来构造一种基于概念的相似度计算方法。按照这种方式就得到了一种称为加权概念相似度（weighted concept similarity）的概念相似度度量方法，定义如下：

定义 4.3 已知形式背景 K=（G，M，I）下的概念 C_1=（A_1，B_1），C_2=（A_2，B_2），那么概念 C_1 与 C_2 的加权相似度为：

$$S_s^w(C_1, C_2) = w * S(A_1, A_2) + (1 - w) * S(B_1, B_2) \tag{4.6}$$

其中 $0 \le w \le 1$，且 S 为 Jaccard index、Sorensen coefficient 或 Symmetric Difference 等基于集合的相似性度量方法。

表 4.1 形式背景示例三

	m1	m2	m3	m4
g1	0	1	0	1
g2	0	0	1	1
g3	0	0	0	1
g4	1	0	0	0
g5	1	1	1	0
g6	0	0	1	0
g7	1	1	0	0

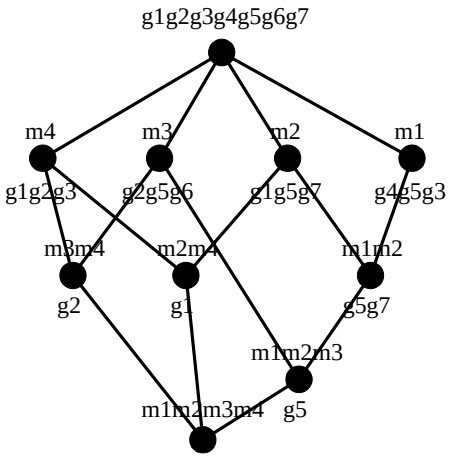

图 4.1 形式背景下的概念格

基于集合的相似度度量方式是在较为完善的相似度度量方式的基础上构建而成的，并且具有较好的计算效率。集合的交、并、差集运算都可以在时间复杂度为 O（min{|x|,|y|}）的时间内完成。令（A_1，B_1）、（A_2，B_2）为两个已知的

概念，那么在所有基于集合的相似性中最坏的情况下，时间复杂度为 O（$\min\{|A_1|,|B_1|,|A_2|,|B_2|\}$）。从加权概念相似度的定义可以看出，这种相似性度量方式仍存在着一些不足：（1）权值 w 的设定直接影响着这种相似性度量方式的计算结果，使得这种方式在度量相似度的效果上存在着必然的不稳定性；（2）另外，加权的概念相似度仅仅考虑了每个单独的概念中的集合的基数，并没有关注两个概念之间所共享的信息量，这在一定程度上淡化了概念间的内在联系。表 4.1 为一个包含七个对象与四个属性的二元形式背景，图 4.1 为基于表 4.1 中形式背景构造而成的概念格。为了对图中概念格中概念的相似度进行分析，另概念 C1 = ({g5}, {m1, m2, m3}), C2 =({g2, g5, g6}, {m3}), C3 = ({g4, g5, g7}, {m1})，权值 w=0.5,那么我们采用加权概念相似度对其进行计算：

$$S_{\mathrm{Jac}}^{0.5}(C_1,C_2)=S_{\mathrm{Jac}}^{0.5}(C_1,C_3)=0.333$$

$$S_{\mathrm{Sor}}^{0.5}(C_1,C_2)=S_{\mathrm{Sor}}^{0.5}(C_1,C_3)=0.5$$

可以看到，在分别采用了两种 S 函数，参数 w=0.5 的加权相似度对（C_1，C_2），（C_1，C_3）的概念相似度进行计算之后，所得的结果均表示两组概念具有相同的相似度。从图 4.1 中概念之间的结构关系作如下分析：（1）相比概念 C_1，C_2 缺少了属性 m1 与 m2,但是多出了对象 g2 与 g6。结合形式背景中的信息，可知对象 g2 与 g6 并不具备属性 m1 和 m2。（2）此外，相比概念 C_1，C_3 则在属性集中缺少了 m2、m3，而在对象集中多出了 g4、g7，同样对照形式背景中对象与属性的二元关系，我们发现对象 g7 具备属性 m2。

从以上两点可以得出，相比概念 C_2，概念 C_3 中的对象包含了更多概念 C_1 中的属性。直观的想，概念 C_3 应该与 C_1 之间具有更高的相似度，而不是等同于 C_2 与 C_1 之间的相似度。

为了在相似性中引入两个概念之间的共同存有的信息，我们可以由原始的形式背景 K 得到矩阵 mat(K)，合并两个概念 C_1=（A_1，B_1），C_2=（A_2，B_2）后可得到子矩阵 D=($A_1 \cup A_2, B_1 \cup B_2$)，并且矩阵 D 中一定存在零元素，也就是 D 中一定存在对象不具备某种属性的情况。将所得子矩阵中零元素的个数作为影响概念之间相似度的重要因素，得到以下定义[52]：

定义 4. 4 给定形式背景 K 下的两个概念 C_1=（A_1，B_1），C_2=（A_2，B_2），那么由概念 C_1，C_2 引入的零元素定义为 zero(C_1,C_2),它代表由行（$A_1 \cup A_2$），列（B_1

∪ B_2）构成的 mat(K)的子矩阵 D 中零元素的个数。

计算 zero(C_1，C_2)需要将子矩阵中每一行的零元素个数进行累加，公式如下：

$$\mathrm{zero}(C_1,C_2)=\sum_{a\in A_1\cup A_2}|(B_1\cup B_2)\backslash a'| \tag{4.7}$$

定义 4.5 已知概念 C_1=（A_1，B_1），C_2=（A_2，B_2），那么基于零元 zero（C_1，C_2）的相似度计算方法为：

$$\mathrm{Sim}_z(C_1,C_2)=\frac{|A_1\cup A_2|\times|B_1\cup B_2|-\mathrm{zero}(C_1,C_2)}{|A_1\cup A_2|\times|B_1\cup B_2|} \tag{4.8}$$

使用以上相似度公式计算后,图 4.1 中概念 C1 = ({g5}, {m1, m2, m3})与 C2 =({g2, g5, g6}, {m3}), C3 = ({g4, g5, g7}, {m1})之间的相似度为：

$$\mathrm{Sim}_z(C_1,C_2)=\frac{9-4}{9}=\frac{5}{9}$$

$$\mathrm{Sim}_z(C_1,C_3)=\frac{9-3}{9}=\frac{2}{3}$$

通过引入这种概念相似度度量方式，在加权概念相似度的基础上进一步提高了概念相似性度量的精度，更深层地挖掘了形式概念之间的共有信息与内在联系。

4.3.2 全局偏好度

设 C=（U，I），C=（U’，I’）为同一概念格中的两个概念，并且 i∈I，i’∈I’，那么基于全局偏好度计算方法如下：

$$G(u,i')=\frac{\sum_{i\in I}w(i,i')}{|I|} \tag{4.9}$$

在这里我们只考虑概念中的所包含的内容，用户 u 对产品 i’的全局偏好度是基于内涵集 I 中属性（产品）与 i’之间的平均相似度定义的。公式 4.9 中| I |表示概念 C 内涵集中的属性(产品)个数，w(i，i’)表示 i 与 i’的 Jaccard 系数，Jaccard 系数主要用于计算符号度量或布尔值度量的个体间的相似度，计算方法如下式所示：

$$w(i,i')=\frac{|U_i\cap U_{i'}|}{|U_i\cup U_{i'}|} \tag{4.10}$$

Ui，Ui’分别表示与产品 i，i’存在关系的用户集合，在全局偏好度中，通过计算起始概念中与目标用户相关的产品与推荐项的平均相似度定义了用户 u 对产品 i’偏好度。

4.3.3 邻域偏好度

设 C=（U，I），C’=（U’，I’）为同一概念格中的两个概念，且 i∈I，i'∈I’，基于概念格的邻域偏好度计算公式如下：

$$\mathrm{N}(\mathrm{u},\mathrm{i}') = \frac{1}{|N_{i'}|}\sum_{C'\in N_{i'}} sim(C,C') \tag{4.11}$$

公式 4.11 中 Ni’代表概念 C 的子节点和兄弟节点中内涵中包含属性 i’的概念集合，sim(C，C’)表示概念 C 与 C’之间的相似度，其计算方式如式 4.8 所示，该计算公式可以化简为下式：

$$\mathrm{sim}(\mathrm{C},\mathrm{C}') = 1 - \frac{\mathrm{zero(D)}}{|D|} \tag{4.12}$$

其中$|\mathrm{D}| = |\mathrm{U}\cup\mathrm{U}'|\times|\mathrm{I}\cup\mathrm{I}'|$，D 表示 U∪U’为对象集，I∪I’为属性集的子形式背景，zero(D)表示了形式背景 D 中零元的个数。在邻域偏好度的计算中，所涉及的信息均为邻域概念（子节点或兄弟节点）中的内容。结合以上对全局偏好度与邻域偏好度的定义，下面给出两种效用函数 Pre( )的定义方式：

$$\mathrm{Pre1}(\mathrm{u},\mathrm{i}') = \mathrm{G}(\mathrm{u},\mathrm{i}') \tag{4.13}$$

$$\mathrm{Pre2}(\mathrm{u},\mathrm{i}') = \mathrm{G}(\mathrm{u},\mathrm{i}')*\mathrm{N}(\mathrm{u},\mathrm{i}') \tag{4.14}$$

可以看出，Pre1 仅使用了公式 4.9 即全局偏好度来计算产品对用户的推荐度，Pre2 则通过全局偏好度与邻域偏好度的乘积来计算推荐度。

为了区分使用这两种效用函数的 CNCF 算法，将使用式 4.13 作为效用函数的 CNCF 算法记为 CNCF-1，使用式 4.14 作为效用函数的 CNCF 算法记为 CNCF-2。

以表 3.6 中的形式背景为例，使用 CNCF-2 算法对对象 3 进行推荐，u_3 的属性集为 B={i_4}，如果要向该用户进行推荐:

(1) 根据起始概念定义，在图 1 右侧概念格中找到 u3 的起始概念{(u_1，u_2，u_3)，(i_4)}

(2) {(u_4，u_5，u_7)，(i_1)}，{(u_1，u_5，u_7)，(i_2)}，{(u_2，u_5，u_6)，(i_3)}为兄弟节点，通过这些节点的内涵集对候选项集扩充，得到 CSsib={i_1，i_2，i_3}

(3) 通过子节点{(u_1)，(i_2，i_4)}，{(u_2)，(i_3，i_4)}的内涵集扩充扩选项集得到

CSchd={i_2，i_3，i_4}

(4)根据(2)，(3)的结果得到最终的候选项集 CS=CSsib∪CSchd/B={i1，i2，i3}

(5)用 Pre2 来计算推荐度：

$$\mathrm{Pre1}(u_3,i_1)=\mathrm{G}(u_3,i_1)\times\mathrm{N}(u_3,i_1)=0\times\left(\frac{6}{12}\right)=0$$

$$\mathrm{Pre2}(u_3,i_2)=\mathrm{G}(u_3,i_2)\times\mathrm{N}(u_3,i_2)=\frac{1}{5}\times\frac{1}{2}\times\left(\frac{6}{10}+\frac{4}{6}\right)=0.1267$$

$$\mathrm{Pre2}(u_3,i_3)=\mathrm{G}(u_3,i_3)\times\mathrm{N}(u_3,i_3)=\frac{1}{5}\times\frac{1}{2}\times\left(\frac{6}{10}+\frac{4}{6}\right)=0.1267$$

通过计算结果可以看出，i_2，i_3 对于 u_3 的推荐度要高于 i_1，所以采用 CNCF-2 算法的推荐系统最终会向该用户推荐产品 i_2，i_3。CNCF-1 与 CNCF-2 的算法流程如图 4.2 所示

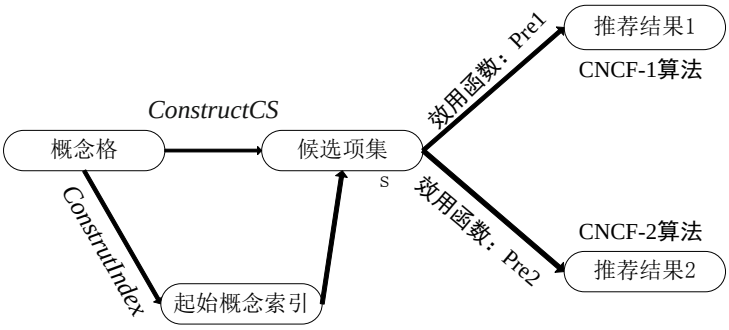

图 4.2 CNCF 算法流程图

4.4 本章小结

本章将概念相似度引入到效用函数 Pre 的构造过程中，提出了用户对产品的全局偏好度与邻域偏好度，分别从不同角度衡量了用户对于产品的兴趣度。并通过对两种偏好度进行组合，定义了两种效用函数的计算方式。最后通过举例简要展示了 CNCF 算法的执行过程，进一步说明了该算法的工作原理。

5 实验结果与分析

5.1 实验设计

实验部分首先针对构造起始概念索引对其定位效率的影响进行分析评估。然后对本文所提出的基于概念邻域的协同过滤算法（CNCF），选取了基于邻域的协同过滤算法中的 userkNN(基于用户)和 itemkNN(基于项目)作为对比算法进行实验，并使用召回率和准确率等推荐算法评估标准对其进行分析比较。

5.2 实验环境与实验数据

实验在处理器为四核 3.40GHz，内存 4GB 的计算环境下进行。所有算法均由 python 语言所实现。实验数据来自 Movielens100k 与 Bookcrossing 两个公共数据集。前者包含 943 个用户对 1682 部电影的 100000 条用户评分记录，后者则是 bookcrossing 图书社区 278858 个用户对 271379 本图书的 1149780 条评分记录。实验采用 5 折交叉验证法，为方便表述，标记 Movielens 数据集为 T1，而 Bookcrossing 数据集本身数据量较大，从中随机抽取了 5064 个用户对 36145 本图书的评分记录构成数据子集，记为 T2。

5.3 推荐系统评测指标

Top-N 推荐通常会生成一个包含 N 项的推荐列表作为最终结果反馈给目标用户。这种推荐模式的测评方法主要通过准确率（precision）和召回率（recall）来度量。令 result(u)为推荐系统最终生成的针对某一用户的推荐列表，test(u)为该用户在测试集上的行为列表，则准去率与召回率的定义如下：

$$Precision = \frac{\sum_{u \in U} |result(u) \cap test(u)|}{\sum_{u \in U} |result(u)|} \tag{5.1}$$

$$Recall = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |T(u)|} \tag{5.2}$$

覆盖率（coverage）能够描述一个推荐算法对物品长尾的发掘能力，本文通过以下方法计算覆盖率:

$$Coverage = \frac{\left|\bigcup_{u \in U} R(u)\right|}{|I|} \tag{5.3}$$

5.4 实验结果分析

5.4.1 起始概念索引对于定位效率的影响

为了验证起始概念索引的有效性，将通过起始概念索引定位起始概念的方法称为索引定位法，而传统的重复 n（n 为推荐次数）次定位起始概念的方法称作重定位法。通过将这两种起始概念定位方法应用在不同规模的同一概念格上，去比较两种方法的时间开销。实验使用 Godin 够格法随机构造了五个概念数量依次递增的概念格作为操作对象。并通过模拟推荐过程中的起始概念定位操作来获取起始概念定位所带来的总的时间消耗。

表 5.1 索引定位法与重定位方法的时间对比

	索引定位(ms)	重定位(ms)	概念数
G_1	1016	1966	88572
G_2	1247	2044	95748
G_3	1279	2184	99310
G_4	1279	2167	100399
G_5	1295	2399	105486

如表 5.1 所示，实验结果中列出了在不同规模概念格下索引定位法与与重定位法的时间消耗。$\{G_1, G_2, G_3, G_4, G_5\}$代表包含不同数目概念的概念格，可以看出，索引定位方法在时间效率上要明显高于重定位法，通过一次性构造包含所有对象与其起始概念对应关系的起始概念索引，直接从中获取特定对象的起始概念位置，从而提高了定位效率。也进一步印证了之前对于重定位法包含大量重复性探索的观点。

5.4.2 召回率与准确率

为了全面评测 CNCF 算法推荐的召回率和准确率，选取不同的推荐列表长度 N=1,5,10,并分别在每个推荐列表长度下进行实验。

表 5.2 四种算法在 T1 与 T2 数据集中的召回率

数据集	N	userkNN	itemkNN	CNCF-1	CNCF-2
T1	1	0.032	0.025	0.028	0.034
	5	0.119	0.098	0.111	0.121
	10	0.197	0.161	0.183	0.199
T2	1	0.0078	0.0075	0.0078	0.0076
	5	0.0261	0.0246	0.0258	0.0264
	10	0.0369	0.0343	0.0381	0.0395

表 5.3 四种算法在 T1 与 T2 数据集中的准确率

数据集	N	userkNN	itemkNN	CNCF-1	CNCF-2
T1	1	0.284	0.214	0.248	0.292
	5	0.2112	0.17	0.194	0.21
	10	0.176	0.1398	0.1598	0.1732
T2	1	0.0277	0.0265	0.0275	0.0268
	5	0.0184	0.0173	0.0182	0.1086
	10	0.0139	0.0121	0.0134	0.0139

表 5.1、表 5.2 所示为四种算法在 T1、T2 上的召回率与准确率。从整体来看，T1 上进行实验得出的召回率与准确率远高于 T2 上的实验结果，这也体现了两个数据集的区别。之所以 T2 上的实验结果整体低于 T1，是因为 T2 所代表的数据集具有更高的稀疏度。实验中涉及的方法都依赖于邻域信息的获取，无论是用户邻域、产品邻域或是概念邻域，均不同程度受原始数据稀疏程度的影响，所以在较为稠密的数据集中的推荐效果通常要优于稀疏数据集中的表现。依据实验结果，分别给出四种算法在 T1、T2 上召回率与准确率的折线图，如图 5.1 至图 5.4 所示。通过比较可以看出，itemkNN 算法的召回率与准确率均为最低，这也与其近邻产品的选择方式有着密切关系。由于本文集中讨论基于邻域的推荐

方法，并未考虑产品属性的相关信息，而主要通过共同访问过两个产品的用户数来定义产品间的相似度，加上用户-产品矩阵通常都具有一定程度的稀疏度，导致了其近邻产品选择的局限性，影响了推荐结果。同样是基于邻域的方法，从用户角度出发的 userkNN 则取得了较高的召回率，通过聚集具有相似访问的用户，将其中用户访问较多的产品进行推荐。由于 T1 与 T2 分别为电影与书籍类的数据集，较为容易在具有相似特质的用户群体内传播共享，所以 userkNN 的效果要明显优于 itemkNN。

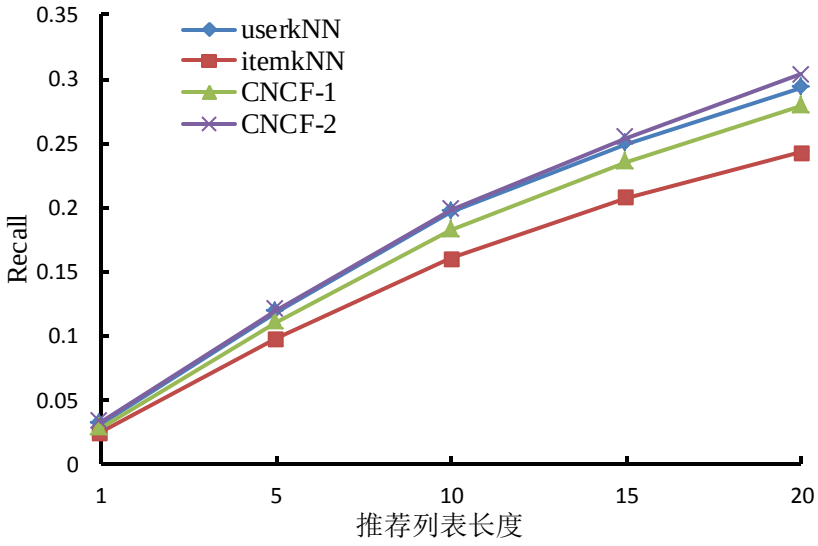

图 5.1　四种算法在 T1 中的召回率

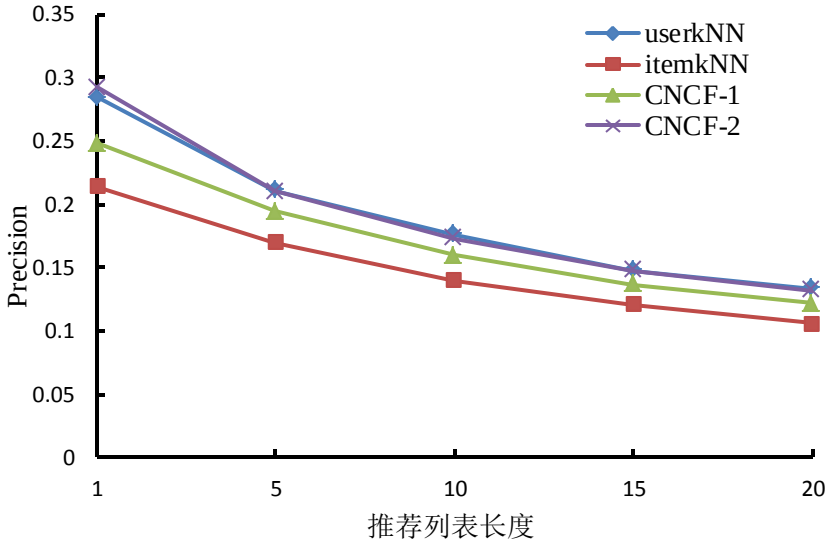

图 5.2　四种算法在 T1 中的准确率

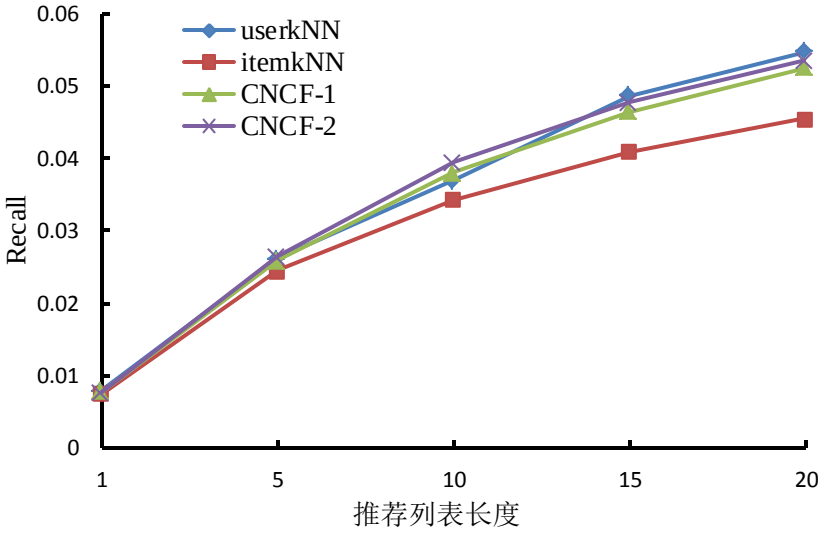

图 5.3 四种算法在 T2 中的召回率

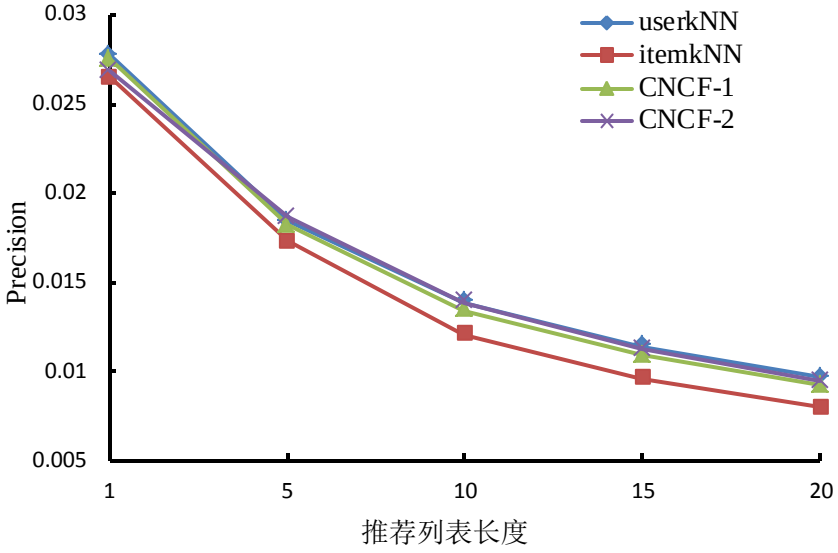

图 5.4 四种算法在 T2 中的准确率

对于本文提出的 CNCF-1 与 CNCF-2 算法，其两项指标均优于 itemkNN 算法，相比 CNCF-1，CNCF-2 算法在召回率与准确率上更具优势，与 userkNN 处于同一水准。CNCF-1 算法在计算用户对于产品的偏好度时没有将概念相似度考虑其中，仅基于格结构获取候选产品集，并使候选产品与用户已访问产品间的 jaccard 系数来定义其推荐度，再加上用户访问数据的缺失，对最后的推荐效果还是产生了一定的影响。而 CNCF-2 在综合考虑了产品之间与概念之间的相似度

后，明显取得了更好的推荐效果。可见在利用概念格进行推荐时，不仅需要考虑概念间的层次结构，还需要进一步挖掘概念之间的内在联系，例如本文中对概念相似度的引入。

表 5.4 四种算法在 T1 与 T2 中的覆盖率

	userkNN	itemkNN	CNCF-1	CNCF-2
T1	0.189	0.256	0.239	0.248
T2	0.184	0.228	0.192	0.219

5.4.3 覆盖率

覆盖率能够描述一个推荐系统对物品长尾的发掘能力。一个好的推荐系统不仅需要有较高的用户满意度，也要有较高的覆盖率，这样才能避免出现越是热门的产品越容易被推荐，而越是冷门的物品越无人问津的情况出现。如表 5.3 所示 ，userkNN 算法的覆盖率在两个数据集中都是最低的，这也反应了 userkNN 算法的根据相似用户进行推荐的核心思想，往往推荐的是在这个相似用户群体中比较热门的物品，从而一定程度忽略了对长尾产品的发掘。相反，itemkNN 的覆盖率在两个数据集中都维持在较高水准。CNCF-1 与 CNCF-2 算法的覆盖率均高于 userkNN 算法，而 CNCF-1 算法的覆盖率均低于 itemkNN 与 CNCF-2 算法。可以看出，CNCF-1 与 CNCF-2 算法在获得较高的准确率和召回率的同时，也能保持较高的覆盖率。

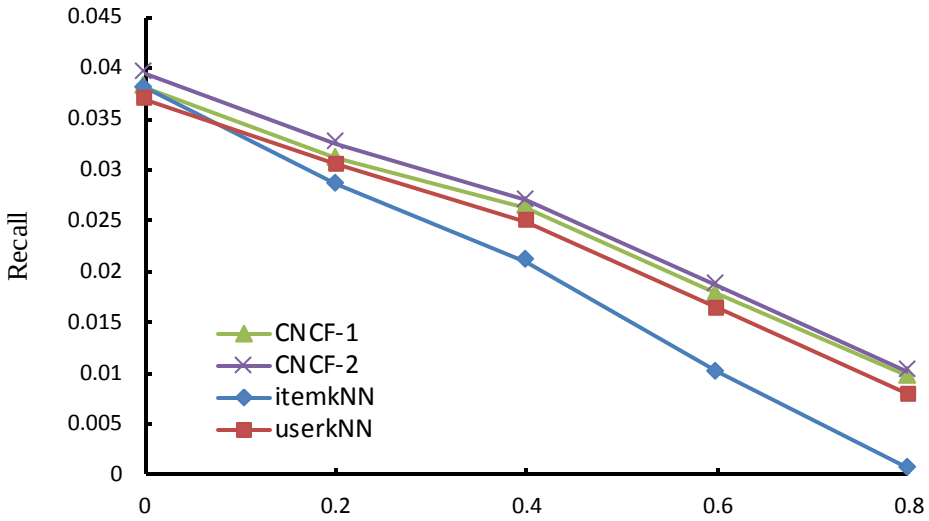

图 5.5 T2 中不同消减概率下的召回率

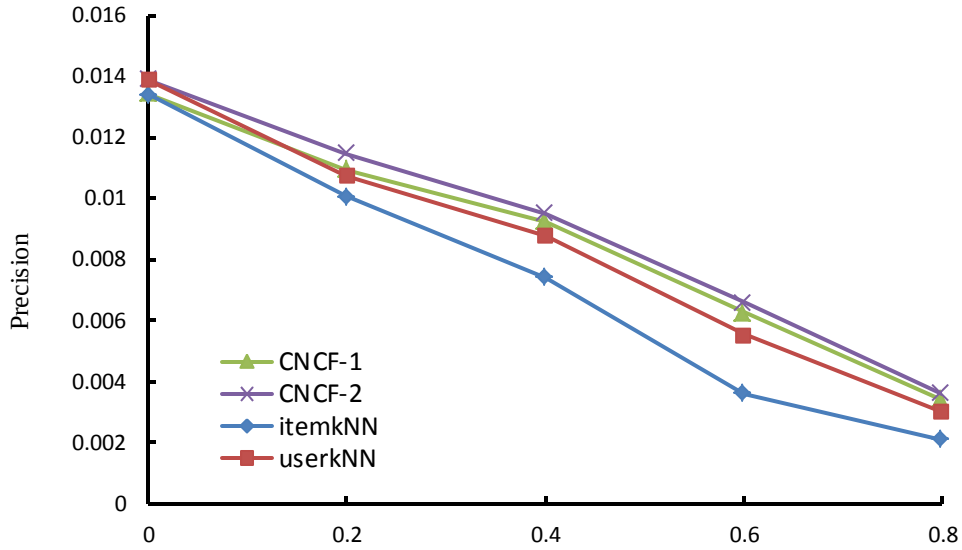

图 5.6　T2 中不同消减概率下的准确率

5.4.4　稀疏环境下的性能评测

为了评测 CNCF 算法在数据稀疏情况下的推荐性能，本文设计实验在数据稀疏度较高的 T2 数据集中进行，通过以一定概率删除数据集中用户产品交互记录的方式来模拟数据稀疏性逐渐增大的推荐环境。设定消减概率 p=[0.2，0.4，0.6，0.8]，消减概率表示删除一条记录的可能性，随着概率 p 的增加，越来越多的记录从数据集中剔除，数据稀疏度逐渐增大，在固定推荐列表长度为 10 的前提下，对四种算法推荐结果的召回率与准确率进行评估。

图 5.5 与图 5.6 展示了 T2 数据集上的四种算法的在不同消减概率下的召回率与准确率。可以明显看出 itemkNN 算法对于稀疏数据的适应性是最差的，userkNN 在 p=0.2 之后的准确率与召回率的下降速率虽然减缓，但其召回率与准确率均低于在 CNCF-1 和 CNCF-2 算法。随着消减概率的增加，传统的 itemkNN 和 userkNN 算法都会因为数据愈发稀疏而获取不到足够的邻域信息，从而直接影响了最终的推荐效果。由之前章节中的概念格示意图可以看出，如果把格中的概念看做顶点，则其 Hasse 图为连通图，从其中任意一点出发，都能到达其他任意各点，再结合函数 4.2、函数 4.3 的递归探索过程，能够很容易访问到邻域概念，并构造候选项集，保证了在相对稀疏的形式背景下，依然能够获取足够的候选项进行推荐。综上所述，本文提出的 CNCF-1 与 CNCF-2 算法在数据稀疏

的情况下能够取得更好地推荐效果。

5.5 本章小结

本章首先对实验设计及比对算法、实验数据、实验环境与推荐系统评测指标进行了逐一介绍。之后分别从起始概念索引对于定位效率的影响、召回率与准确率、覆盖率以及稀疏数据环境下的推荐性能评测四个方面对本文所提出的的起始概念索引构造算法与 CNCF 算法进行了实验分析。通过分析实验结果，起始概念索引的构造明显提升特定对象起始概念的定位效率。而在召回率与准确率上，CNCF 算法的效果明显好于对比算法中的 itemkNN，其中结合了概念相似度的 CNCF-2 与 userkNN 的推荐结果基本处于同一水准。另外，在覆盖率的评测中，CNCF 算法的实验结果位于两种对比算法之间，并较为接近覆盖率较高的 itemkNN 算法。当在模拟的稀疏环境下进行推荐时，CNCF-1 与 CNCF-2 推荐结果的指标值均高于两种比对算法，所以整体来看，相较于传统的基于邻域的协同过滤算法，CNCF 算法能够保持较高的推荐水准，并且对稀疏数据具有更好地适应性。

6 总结与展望

6.1 总结

协同过滤算法作为较早出现推荐算法之一，如今依然在个性化推荐领域发挥着重要作用，本文首先对研究背景进行了描述，通过对推荐系统以及形式概念分析领域的整体介绍，说明了本文研究的必要性。同时对国内外的研究现状进行了阐述与分析，并对现阶段研究中的欠缺与不足进行了说明。之后分别对推荐系统与形式概念分析及概念格的相关理论进行了介绍。从推荐角度的不同对常见的推荐模型展开了分类说明。由于本文所提出 CNCF 算法也是协同过滤方法的一种扩展，所以着重对协同过滤方法进行了详细讲述，为之后内容的开展做好铺垫。在形式概念分析部分，进一步对其理论进行详细描述，并介绍了目前形式概念分析在个性化推荐领域的研究状况。

在文章主体部分，首先从形式背景的转化、概念格的生成与起始概念索引的构造三个阶段介绍了概念格中起始概念索引的完整构造过程。起始概念索引作为本文提出的一种辅助于推荐过程的索引结构，通过一次性构造对象与其起始概念之间的映射关系。在将原始数据转化为对应的概念格并生成了起始概念索引后，本文在形式概念分析的背景下，对推荐问题进行了形式化描述。并依次从构造候选项集与用户偏好度计算对 CNCF 算法的核心部分做了详细描述。构造候选项集中之所以采用递归方式是为了充分扩充候选属性（产品）集合。为了在候选项集中进一步筛选，提出用户对产品的全局偏好度与邻域偏好度，通过单独使用全局偏好度与将全局与邻域偏好度结合衍生出了两种效用函数的计算方法，并将分别使用这两种效用函数的 CNCF 算法标记为 CNCF-1 与 CNCF-2。

最后在实验部分，首先对实验整体设计、实验环境与实验数据以及推荐系统评测指标进行了介绍，之后通过在不同规模概念格上的实验，验证了索引定位法对起始概念定位的效率的提升，并对 userkNN、itemkNN、CNCF-1 与 CNCF-2 四种算法在 movielens 与 bookcross 数据集上分别进行了实验，主要对推荐结果的召回率与准确率、覆盖率以及稀疏环境下的推荐效果进行了评估。整体来说，CNCF 算法的推荐结果在召回率与准确率上明显高于 itemkNN，并且与 userkNN 同处于较高水准，在覆盖率上也有较好表现。尤其通过在数据稀疏环境下的比

较，CNCF 的两种算法均具备更好的推荐效果。

6.2 展望

下一步的工作主要集中在两个方面：（1）由于形式背景中只包含二元关系，所以 CNCF 算法对于结构类似的隐式反馈数据具有较好的操作性。而现实中却也存在着一些直接体现用户对产品喜爱程度的显式数据，例如用户对项目的评分。虽然存在着多值背景向单值背景转化方式，但是转化过程不可避免地存在着信息的损失。后续会尝试将 CNCF 算法推广到多值的模糊背景下，结合模糊概念格理论，直接在多值背景进行推荐问题求解。（2）在用户与产品间的交互数据的基础上，引入更多影响因素（例如用户信息、产品属性，时间因素等），进一步探索该算法模型的可扩展性，提升推荐效果。

参考文献

[1] R Bell,Y Koren,C Volinsky. Modeling relationships at multiple scales to improve accuracy of large recommender systems[C].Acm Sigkdd International Conference on Knowledge Discovery & Data Mining, 2007:95-104

[2] Wang Li-ming , Zhang Zhuo.An algorithm for mining closed frequent itemsets based on apposition assembly of iceberg concept lattices[J].Journal of Computer Research and D evelopment, 2007,44(7): 1184- 1190.

[3] Zhang Zhuo,Du Juan,Wang Li-ming. Formal concept analysis approach for data extraction from a limited deep-web database[J].Journal of Intelligent Information Systems, 2013, 41(2):211-234.

[4] John S.Breese,David Heckerman,Carl Kadie. Empirical Analysis of Predictive Algorithms for Collaborative Filtering[J].Fourteenth Conference on Uncertainty in Artificial Intelligence,2013,7(7):43-52.

[5] Polat H,Du W.SVD-based collaborative filtering with privacy[C].Acm Symposium on Applied Computing,2005,1:791-795.

[6] Koren Y,Bell R,Volinsky C.Matrix factorization techniques for recommender systems[J], Computer,2009,42(8):30–37.

[7] Vucetic S,Obradovic Z.Collaborative filtering using a regression-based approach[J],Knowl Inf Syst .2005.7(1):1–22.

[8] Rendle S, Freudenthaler C, Gantner Z,Schmidt-Thieme- L.Bpr: Bayesian personalized ranking from implicit feedback[C]. In: Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence,2009:452-461.

[9] Deshpande M, Karypis G. Item-based top-n recommenddation algorithms[J].ACM Trans Inf Syst , 2004, 22(1):143–177.

[10] Linden G,Smith B,York J.Amazon.com Recommendations Item-to-Item Collaborative F i-ltering [J].IEEE Inter-net Computing,2003,7(1):76-80.

[11] Koren Y.Factorization meets the neighborhood:a multif-aceted collaborative filtering m -odel[C]. In:Proceedings of the 14th ACM SIGKDD international conference on knowledg -e discovery and data mining,2008,pp 426–434.

[12] Xue G-R, Lin C,Yang Q,et al. Scalable collaborative filtering using cluster-based s-m oothing[C].Sigir: Internati-onal Acm Sigir Conference on Research & Development in In-fo rmation Retrieval,2005, 114–121.

[13] 谢志鹏, 刘宗田.概念格的快速渐进式构造算法[J].计算机学报,2002,25(5):490-496.

[14] 沈夏炯, 韩道军, 刘宗田, et al.概念格构造算法的改进[J].计算机工程与应用,2004,40(24):100-103.

[15] Ganter B.Formal Concept Analysis:Mathematical Foundations[M].Springer-Verlag New York,Inc,1997.

[16] 张文修, 魏玲, 祁建军.概念格的属性约简理论与方法[J].中国科学:技术科学,2005,35(6):628-639.
[17] 姜琴, 张卓, 王黎明.基于多属性同步消减的概念格构造算法[J].小型微型计算机系统,2016,37(4):646-652.
[18] Godin R.Incremental concept formation algorithm based on Galois(concept) lattices.[J]. Computational Intelligence,1995,11(2):246-267.
[19] Sahami M. Learning Classification rules using lattices[A].European Conference on Machine Learning, 1995, 912:343-346.
[20] 王志海, 黄厚宽.概念格上粗糙集合运算与函数依赖生成[C].全国机器学习研讨会,1998.
[21] Neuss C,Kent RE.Conceptual Analysis of Resource Meta-Information.Computer Networks & Isdn Systems,1995,27(6):973-984.
[22] Cole R,Eklund P W.Scalability in formal concept analysis[J].Computational Intelligence,1999.15(1).
[23] Eklund P W,Martin P.WWW indexation and document navigation using conceptual structures[A].2nd IEEE Conference on Intelligent Information Processing Systems(ICIPS'98)[C],IEEE Press,1998,217-221.
[24] 刘玮. 电子商务系统中的信息推荐方法研究[J]. 情报科学. 2006，24(2):300-303.
[25] Adofnavicius G，tuzhilin A.Toward the next generation of recommender systems:A survey of the state-of-the-art and possible extensions[J].IEEE Trans on Knowledge and Data Engineering,2005,17(6):734-749.
[26] Resnick P, Varian H R. Recommender systems[J]. Communications of the Acm, 1997, 40(3):56–58.
[27] 徐海玲.互联网推荐系统比较研究[J].软件学报,2009,20(2):350-362.
[28] Mooney R J, Bennett P N, Roy L. Book Recommending Using Text Categorization with Extracted Information[J]. Recommender Systems Papers from Workshop, 1999:49-54.
[29] Pazzani M, Billsus D. Learning and Revising User Profiles: The Identification of Interesting Web Sites[J]. Machine Learning, 1997, 27(3):313-331.
[30]Schapire R E, Singer Y, Singhal A. Boosting and Rocchio applied to text filtering[C], International ACM SIGIR Conference on Research and Developmentin Information Retrieval. ACM, 1998:215-223.
[31] Sarwar B, Karypis, J Konstan, et al.Item-based collaborative filtering recommendation algorithm[C]. International Conference on World Wide Web, 2001, 4(1):285-295.
[32] Koren Y. Factor in the neighbors: Scalable and accurate collaborative filtering[J]. Acm Transactions on Knowledge Discovery from Data, 2010, 4(1):1.
[33] 邓爱林. 电子商务推荐系统关键技术研究[D]. 复旦大学, 2003.
[34] Zheng R, Wilkinson D, Provost F. Social Network Collaborative Filtering[J]. Social Science Electronic Publishing, 2008.
[35] Kautz H, Selman B, Shah M. Referral Web: combining social networks and collabor-

ative filtering[J]. Communications of the Acm, 1997, 40(3):63-65.

[36] Golbeck J. Generating Predictive Movie Recommendations from Trust in Social Networks[C],Trust Management, International Conference, Itrust 2006, Pisa, Italy, May 16-19, 2006, Proceedings. DBLP, 2006:93-104.

[37] Wille R. Restructuring lattice theory: An approach base-d on hierarchies of concepts [M].Berlin: Springer,2009.

[38] Boucher-Ryan PD,Bridge D,et al. Collaborative Recommending using Formal Concept Analysis [J]. Knowledg-e-Based Systems,2006,19(5):309-315.

[39] Dmitry I. Ignatov,Sergei O. Kuznetsov,et al. Concept based Recommendations for Internet Advertisement[C]. 6-th International Conference on Concept Lattices and Their Applications,2008.

[40] Li X, Murata T,et al. A Knowledge-based Recommendation Model Utilizing Formal Concept Analysis [J]. International Conference on Computer & Automation Engineering,2010,4:221-226.

[41] Maio CD, Fenza G, Gaeta M,et al.Rss-based e-learning recommendations exploiting fuzzy FCA for knowledge modeling [J].Applied Soft Computing,2012,12(1):113-124.

[42] Fang P, Zheng S. A Research on Fuzzy Formal Concep-t Analysis Based Collaborative Filtering Recommendation System[J]. 2nd International Symposium on Knowledge Ac-quisition and Modeling,2009, 3:352-355.

[43] 李云华, 李新广.基于概念格的图书协同推荐研究[J].图书情报工作吗,2012,56(17)：131-135.

[44] 李宏涛, 何克清.基于概念格和随机游走的社交网朋友推荐算法[J]。四川大学学报（工程科学版）,2015,47(6):131-138.

[45] 杨丽, 徐扬. 概念格中不同类型多值背景的研究[J]. 计算机应用研究, 2008, 25(7):2033-2034.

[46] 杨丽. 基于动态多值背景的概念格及其约简方法的研究[D]. 西南交通大学, 2006.

[47] Ichise R. Evaluation of Similarity Measures for Ontology Mapping[C], New Frontiers in Artificial Intelligence, JSAI 2008 Conference and Workshops, Asahikawa, Japan, June 11-13, 2008, Revised Selected Papers. DBLP, 2008:15-25.

[48] Zhu L J, Tao L, Liu H. Calculation of the concept similarity in domain ontology[J]. Journal of South China University of Technology, 2004.

[49] Ichise R. Machine Learning Approach for Ontology Mapping Using Multiple Concept Similarity Measures[C],Ieee acis International Conference on Computer and Information Science. IEEE Computer Society, 2008:340-346.

[50] Giunchiglia F, Shvaiko P, Yatskevich M. S-Match: an Algorithm and an Implementation of Semantic Matching[J]. Proceedings of Esws, 2004, 3053:61-75.

[51] Melnik S, Garciamolina H, Rahm E. Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching[C].International Conference on Data E

ngineering, 2002. Proceedings. IEEE, 2002:117-128.

[52] Alqadah F, Bhatnagar R. Similarity measures in formal concept analysis[J]. Annals of Mathematics and Artificial Intelligence, 2011, 61(3):245-256.

个人简历

陈昊文，男，1990 年 4 月生，河南省新乡市人

2008年9月—2012年7月　郑州大学　信息工程学院　工学学士

2014年9月—2017年7月　郑州大学　信息工程学院　工学硕士

攻读硕士学位期间参与的项目及完成的工作成果：

陈昊文，王黎明，张卓.基于概念邻域的 Top-N 推荐算法，小型微型计算机系统（已录用）

参与项目：

国家自然科学基金(61303044)：有限内存空间下大规模模糊概念格的快速构造理论与方法研究

致谢

三年的硕士研究生生活转瞬即逝，还记得当初入学时对于硕士毕业时的憧憬，而现如今当时的憧憬已即将变为现实。自己十分有幸能够进入这么优秀的学校，并能够在具有丰富研究及教学经验的老师指导下在进行研究生阶段的学习。在王黎明老师、张卓老师的悉心指导下，我踏入了科研的领域，并且体会到了研究过程中的点滴心酸与乐趣。

读研期间，我的导师王黎明老师无论是在科研学习还是生活上都对我提供了莫大的指导与帮助。王老师即使我硕士阶段的导师，也是我本科阶段程序设计基础课程的授课教师，当时通过整个课程中老师对我们的严格要求，我就深深感受到王老师治学严谨的态度。每当我遇到问题向他请教时，王老师都会热心细致地对我进行指导与解疑。他教导我们要让硕士阶段的生活变得有意义，要能够切实的在这三年中提升自己的科研能力。为了提升我们的动手实践能力，为之后的科研工作夯实基础，王老师主动为我们提供的亲身参与项目实际开发的实践机会。也通过那段时间的磨练，是自身的实操能力得到了明显的加强。最初接触科研与论文写作，王老师为了使我们能够快速进入角色，通过定期召开组会与我们探讨交流来了解工作开展情况，从整体研究方法到研究内容的细节逐一进行指导，对我们的研究开展产生了极大地帮助。此外，我也要感谢张卓老师对我科研及论文工作细致入微的指导与帮助，将我带入到了形式概念分析的理论世界，使我深切体会到了研究过程中的酸甜苦辣，也为我的硕士阶段的生活填上了浓墨重彩的一笔。当张老师第一次将他审阅和修改后的论文发给我时，密密麻麻且详细严谨的批注瞬间震撼到了我，我也将他作为我的榜样不断地推动自己在科研的道路上前行。

同事还要感谢柴玉梅老师、南晓斐老师、李晓宇老师、申丰山老师对自己学术及生活上的指导与帮助。

三年的时光，少不了实验室每一位同学的陪伴。在于他们的相处中，不但使我的生活变得不再单调，而且从每位同学身上学习到了很多自己欠缺的素质。首先感谢三年中和我一同并肩奋斗的杨浩宇、李超、李昆、高光、王春月、王冰洁、吴凯敏、李赛、王涛以及我的室友王旭、朱静阳对我生活、学习上的支持与帮助。感谢姜琴师姐、唐同龙师兄在生活、学习上对我的照顾。也感谢师弟师妹们：张菲菲、王红敏、张肖、王文凯、王家南等。

最后要感谢的是我的家人和朋友，三年的硕士生活中五味俱全，你们的理解与支持成为了我迷茫、气馁时的强心剂，是你们给我了坚持下去的勇气。我也要向百忙之中参与审阅、评议本论文的各位老师、向参与本人论文答辩的各位老师表示由衷的感谢!我会更加勤奋学习、认真研究，努力做得更好。在这里把最美好的祝福献给你们，愿永远健康、快乐!