分类号： TP311.5 单位代码：

学 号：



硕士学位论文



**中文论文题目 ：团购商品个性化推荐系统的设计与实现**

**英文论文题目：****Group-buying** [**Design**](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E5%9B%A2%E8%B4%AD%E5%95%86%E5%93%81%E4%B8%AA%E6%80%A7%E5%8C%96%E6%8E%A8%E8%8D%90%E7%B3%BB%E7%BB%9F%E7%9A%84%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0)**[And](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E5%9B%A2%E8%B4%AD%E5%95%86%E5%93%81%E4%B8%AA%E6%80%A7%E5%8C%96%E6%8E%A8%E8%8D%90%E7%B3%BB%E7%BB%9F%E7%9A%84%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0" \t "_blank)** **[implementation](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E5%9B%A2%E8%B4%AD%E5%95%86%E5%93%81%E4%B8%AA%E6%80%A7%E5%8C%96%E6%8E%A8%E8%8D%90%E7%B3%BB%E7%BB%9F%E7%9A%84%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0" \t "_blank) of** **[personalized](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E5%9B%A2%E8%B4%AD%E5%95%86%E5%93%81%E4%B8%AA%E6%80%A7%E5%8C%96%E6%8E%A8%E8%8D%90%E7%B3%BB%E7%BB%9F%E7%9A%84%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0" \t "_blank)** **[recommendation](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E5%9B%A2%E8%B4%AD%E5%95%86%E5%93%81%E4%B8%AA%E6%80%A7%E5%8C%96%E6%8E%A8%E8%8D%90%E7%B3%BB%E7%BB%9F%E7%9A%84%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0" \t "_blank)** **[system](http://cn.bing.com/dict/clientsearch?mkt=zh-CN&setLang=zh&form=BDVEHC&ClientVer=BDDTV3.5.0.4311&q=%E5%9B%A2%E8%B4%AD%E5%95%86%E5%93%81%E4%B8%AA%E6%80%A7%E5%8C%96%E6%8E%A8%E8%8D%90%E7%B3%BB%E7%BB%9F%E7%9A%84%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0" \t "_blank)**

申请人姓名： 李朝阳

指导教师： 李际军

合作导师： 朱小军

专业学位类别： 工程硕士

专业学位领域： 软件工程

所在学院： 软件学院

**论文提交日期 年 月 日**

**学位论文书脊示例**

|  |
| --- |
| **题**  **目**  **作**  **者**  **姓**  **名**  **浙**  **江**  **大**  **学** |

3cm左右

3cm左右

**中文论文题目** （小二号仿宋体加黑）



**论文作者签名:**

**指导教师签名:**

论文评阅人1： (姓名\职称\单位,下同)

评阅人2： (隐名评阅学位论文省略)

评阅人3：

评阅人4：

评阅人5：

答辩委员会主席： (姓名\职称\单位)

委员1：

委员2：

委员3：

委员4：

委员5：

答辩日期：

**英文论文题目**（16pt Time New Roman，Bold）



**Author’s signature:**

**Supervisor’s signature:**

Thesis reviewer 1： (姓名\职称\单位,下同)

Thesis reviewer 2： (隐名评阅学位论文省略)

Thesis reviewer 3：

Thesis reviewer 4：

Thesis reviewer 5：

Chair： (姓名\职称\单位)

(Committee of oral defence)

Committeeman 1：

Committeeman 2：

Committeeman 3：

Committeeman 4：

Committeeman 5：

Date of oral defence：

浙江大学研究生学位论文独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得 **浙江大学** 或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名： 签字日期： 年 月 日

学位论文版权使用授权书

本学位论文作者完全了解 **浙江大学** 有权保留并向国家有关部门或机构送交本论文的复印件和磁盘，允许论文被查阅和借阅。本人授权 **浙江大学** 可以将学位论文的全部或部分内容编入有关数据库进行检索和传播，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本授权书）

学位论文作者签名： 导师签名：

签字日期： 年 月 日 签字日期： 年 月 日

摘要

推荐系统是解决信息过载的一种重要方式; 它通过分析用户行为日志预测用户的喜好,行为,兴趣等并为用户做推荐.随着web 2.0 的不断发展, 用户的参与度越来越高, 而早期推荐系统主要基于内容或者静态用户如基于人口统计的推荐, 而用户实时数据无法产生实时推荐.或者根据用户UGC的上下时间因素进行推荐, 在这方面有许多关键点需要解决.

本文基于公开用户行为数据集对推荐系统进行研究, 包括..; 本文的主要工作内容如下:

1. 推荐系统现状, 个性化服务的需求

**关键词：**　推荐系统，协同过滤, 实时响应, 日志系统, 展示系统, 推荐引擎

Abstract

**Key Words：**,

目录

[摘要 i](#_Toc415687926)

[Abstract ii](#_Toc415687927)

[图目录 III](#_Toc415687928)

[第1章 绪论 4](#_Toc415687929)

[1.1 课题背景与意义 4](#_Toc415687930)

[1.2 研究内容 4](#_Toc415687931)

[1.3 公开用户数据集 5](#_Toc415687932)

[1.3.1 亚马孙（z.cn）数据集 5](#_Toc415687933)

[1.3.2 MovieLens电影评分数据集 6](#_Toc415687934)

[1.3.3 CiteULike论文书签数据集 6](#_Toc415687935)

[1.4 论文概要结构描述 6](#_Toc415687936)

[第2章 Top-N推荐用户兴趣预测的模型 7](#_Toc415687937)

[2.1 简介 7](#_Toc415687938)

[2.1.1 7](#_Toc415687939)

[2.2 问题数学定义和概括 8](#_Toc415687940)

[2.2.1 传统Top-N推荐 8](#_Toc415687941)

[2.2.2 基于时间的Top-N推荐 8](#_Toc415687942)

[2.2.3 隐性反馈数据的动态变化 10](#_Toc415687943)

[2.3 本章小结 12](#_Toc415687944)

[2.3.1 Hello 13](#_Toc415687945)

[第3章 评分预测中用户兴趣预测的模型 14](#_Toc415687946)

[3.1 第一节 14](#_Toc415687947)

[3.1.1 14](#_Toc415687948)

[3.2 本章小结 14](#_Toc415687949)

[3.2.1 14](#_Toc415687950)

[第4章 推荐引擎设计与实现 15](#_Toc415687951)

[4.1 第一节 15](#_Toc415687952)

[4.1.1 15](#_Toc415687953)

[4.2 本章小结 15](#_Toc415687954)

[4.2.1 15](#_Toc415687955)

[第5章 用户行为日志系统设计与实现 16](#_Toc415687956)

[5.1 第一节 16](#_Toc415687957)

[5.1.1 16](#_Toc415687958)

[5.2 本章小结 16](#_Toc415687959)

[5.2.1 16](#_Toc415687960)

[第6章 展示系统设计与实现 17](#_Toc415687961)

[6.1 第一节 17](#_Toc415687962)

[6.1.1 17](#_Toc415687963)

[6.2 本章小结 17](#_Toc415687964)

[6.2.1 17](#_Toc415687965)

[第7章 18](#_Toc415687966)

[7.1 第一节 18](#_Toc415687967)

[7.1.1 18](#_Toc415687968)

[7.2 本章小结 18](#_Toc415687969)

[7.2.1 18](#_Toc415687970)

[第8章 19](#_Toc415687971)

[8.1 第一节 19](#_Toc415687972)

[8.1.1 19](#_Toc415687973)

[8.2 本章小结 19](#_Toc415687974)

[8.2.1 19](#_Toc415687975)

[参考文献 20](#_Toc415687976)

图目录

[图 1 数据集所含各种类型物品数目 6](#_Toc415684388)

[图 2 用户-物品关系二分图 9](#_Toc415684389)

[图 3 路径融合算法 10](#_Toc415684390)

[图 3.1 流程图 11](#_Toc415684391)

[表 2.1简单的多描述分配表 2](#_Toc164669160)

# 绪论

## 课题背景与意义

随着web2.0 的快速发展, 无论是用户产生的数据, 还是物品的数量已经从匮乏到了过载的阶段, 那么如何从海量的物品中找到真正用户需要的物品;

以用户的视角来看，自从有了互联网以来，随着网络资源的逐渐增多，用户寻找信息的方式经历了几个重要阶段。早期用户主要靠直接记住感兴趣网站的网址来寻找内容，后来雅虎公司提出了分类目录系统，将网站分门别类，方便用户查询。但随着信息越来越多，分类目录也只能记录少量的网站，而且维护分类信息的费用也随着信息数量的增长也越来越高, 于是产生了搜索引擎。

搜索引擎可以通过关键词找到自己所需要的信息。但是搜索引擎需要用户提供关键字来寻找信息，因此它不能满足用户的其他需求，比如当用户无法准确描述自己需求的关键字的时候，搜索引擎就无法返回用户所需的内容了。和搜索引擎一样，推荐系统也是一种为用户快速查找有用信息的工具。和搜索引擎不同的是，推荐系统不需要提供明确的关键字，而是通过分析用户行为数据来给用户的兴趣建模从而主动推荐出能够满足用户需求的信息。因此从某种意义上说，推荐系统和搜索引擎是两个互补的工具。搜索引擎满足了用户能表达自己的意图时主动查找的需求，而推荐系统是在用户没有目的的时候帮助用户发现让他们感兴趣的内容。这样满足了在互联网有相关信息的基础上, 用户所需信息能够推送给相关用户。

以信息的视角来看，推荐系统可以更好的利用信息的长尾(The Long Tail)长尾理论[[1]](#footnote-1)，电子商务网站相对于传统超市的优点是它可以给用户提供更多的选择。因为电子商务网站货架的成本极为低廉，上架商品只需要在系统中添加一条记录。但同时长尾的出现也给系统或者点子商务带来了新的挑战，即如何将大量的商品推荐给潜在消费用户和用户实时行为系统如何实时为用户推荐。

## 研究内容

推荐系统的主要目的是，将与用户兴趣相关的信息推送给用户；这就涉及两部分：1）如何找到与用户实时行为相关的物品； 2）如何展现相关推荐并与用户交互产生实时数据；本位主要探究以上两个方面。

协同过滤算法是第一代推荐系统也是应用最为成熟的推荐系统，目前有两类主要的协同过滤算法： 基于用户的协同过滤算法和基于项目的协同过滤算法。基于用户的系统过滤算法就好比， 用户A有一批志同道合的朋友， 他们对物品I都有兴趣并做了评分， 而推荐系统就根据A的朋友的评分进行计算判断是否为A推荐物品I； 其核心思想是：利用用户的行为数据计算用户间的相似性；得到与用户相似性较高的邻居-用户好友群， 然后利用这些用户对其物品的评价来预测目标用户的喜好程度， 然后做出推荐； 基于项目的协同过滤算法是： 用户A对物品集合有一个相似的评价，当需要对某物品I进行预测评估的时候，可根据对相似物品的集合的评价来对物品I进行推荐。这种做法的优点在于对推荐对象没有特殊的要求，能处理音乐电影等难以进行文本结构化表示的对象。

基于内容的推荐是在推荐引擎应用最为广泛的推荐机制，它的核心思想是根据推荐物品或内容的元数据，发现物品或者内容的相关性，然后基于用户以往的喜好记录，推荐给用户相似的物品。这种基于内容的推荐机制的好处在于它能很好的建模用户的口味，能提供更加精确的推荐。但它也存在以下几个问题：1. 需要对物品进行分析和建模，推荐的质量依赖于对物品模型的完整和全面程度。在现在的应用中我们可以观察到关键词和标签（Tag）被认为是描述物品元数据的一种简单有效的方法。2. 物品相似度的分析仅仅依赖于物品本身的特征，这里没有考虑人对物品的态度。3. 因为需要基于用户以往的喜好历史做出推荐，所以对于新用户有“冷启动”的问题。

本文相关的推荐算法

## 公开用户数据集

### 亚马孙（z.cn）数据集

从亚马孙网站上抓取的数据， 其中包括548552种不同的产品（书籍，音乐CD，DVD和VHS录像带）的信息。每条记录包括题目， 销售排行， 相似产品列表， 产品的分类以及产品的评价。



图 1 数据集所含各种类型物品数目

### MovieLens电影评分数据集

MovieLens数据集中，用户对自己看过的电影进行评分，分值为1～5。MovieLens包括两个不同大小的库，适用于不同规模的算法．小规模的库是943个独立用户对1682部电影作的10000次评分的数据；大规模的库是6040个独立用户对3900部电影作的大约100万次评分。

### CiteULike论文书签数据集

CiteULike4是一个著名的论文书签网站。它让研究人员把自己喜欢的论文信息提交到它的网站上，同时让用户对这些论文打上标签。CiteULike公布了一个数据集，每一行是一条记录，由一个四元组构成，包括用户，论文，时间和书签。这个数据集不包含评分数据，但包含了时间信息，因此可以作为一个含有时间信息的用户隐形反馈数据集使用。

## 论文概要结构描述

本文除第一章之外可分为六部分：

* 第二章 Top-N推荐
* 第三章 基于用户邻域模型与矩阵分解
* 第四章 主要是推荐系统中推荐引擎具体实现的架构设计， 类图描述等
* 第五站 用户行为日志收集的设计与实现
* 第六章 推荐单子的展示， 与用户日志的收集
* 第七章 论文结束语以及展望

# Top-N推荐用户兴趣预测的模型

## 简介

推荐系统的主要任务是为每个用户提供一个含有N件物品的推荐列表。因此，实际生产中的推荐系统在预测用户行为时通常有两步。第一步是得到一个和用户相关的物品列表，在真正的生产中系统中包含很多件物品， 但是真正和用户相关的物品很少，我们仅仅只要基于与用户相关的物品即可；第二步是在得到用户将会评分的列表之后，预测用户会给该物品多少评分。第一个问题也被称为Top-N推荐问题，该问题是本节研究的主要问题。关于Top-N推荐的研究有很多。其中协同过滤是很多生产系统中的主要算法[[[2]](#endnote-1), [[3]](#endnote-2)]。

用户的行为数据根据是否为用户主动返回数据可将数据分为显性反馈数据， 隐形反馈数据；显性反馈数据是用户主动返回的数据， 比如， 对物品的喜欢/不喜欢，评分， 评价等， 但是返回的数量想回来说比较少； 而对于隐形反馈数据， 隐性反馈行为指的是那些不能明确反应用户喜好的行为。最具代表性的隐性反馈行为就是页面浏览行为。凡是网站大豆有隐性反馈数据(Implicit Feedback)，这些数据存储在日志系统内。比如，在视频网站中，用户可以通过打分表示对一个视频的喜好。但是，大部分用户看完视频后并不会对视频进行评价，相反用户看了什么视频，每个视频看了多长时间，这样的数据是视频网站能够收集的主要数据，而且数据量巨大。但用户观看一个视频，并不能体现出用户对这个视频的兴趣意图，比如用户看完可能觉得不错，或者觉得不好看。文献[[[4]](#endnote-3)]详细对比了隐性反馈数据和显性反馈数据的优点和缺点。隐性反馈数据虽然不如显性反馈明确，但其中也蕴含着用户的兴趣信息，如果能够做出适当的挖掘，可以很好的来预测用户的兴趣和行为。本节主要描述利用隐形反馈数据来过滤和用户有关系的物品列表， 即Top-N推荐。

### 

每个表都应有表说明[2]，表说明包含表序与表题，居中排印在表的上方；表序与表题之间空一字距；独表表示也同样要求；表与表说明不能破页。[[5]](#footnote-2)

表中不设“备注”栏，需要说明的事项可排印在表下方，表内用星号“\*”或圈码“①、②”标注在相应内容的右上角。表中的参数应表明量和单位的符号，如表中所有参数的单位相同，可标注在表的右上方或表说明之后；各栏单位不同则应标注在各栏表头内。表中不能用“同上”、“同左”一类词代替具体数字；无某项目则空白；未发现用“...”；结果为零用“0”；同一栏的数字必须按位次上下对齐。

表随正文，先见文字后见表；需要转页的表，应在续表的右上角或左上角注明“续表×”，并应重复排印表头。

图表与上下文之间各空一行。

于是我们得到表2.1中的两个描述[3]。

表2.1 简单的多描述分配表

|  |  |  |
| --- | --- | --- |
| 描述1 | ζR | (1−ζ)R |
| 描述2 | ζR（重复） | (1−ζ)R |

## 问题数学定义和概括

### 传统Top-N推荐

一般的， 如下的数据集Data = {( user, item)}，其中数据集中的每个二元组( user, item)代表用户 user对物品 item产生过行为。而Top-N推荐的任务，是通过学习数据集Data，建立用户兴趣的模型，用来给每个user推荐 N个与user最相关的物品。

可扩展内容

### 基于时间的Top-N推荐

用户的兴趣是随着时间变化的，并且用户的历史行为也随之增加。因此，推荐结果也需要随之实时更新他们兴趣的变化，来适应用户的新行为，来产生满足用户最新兴趣的推荐内容。因此，考虑了时间因素后，本节主要解决随着时间变化的Top-N推荐：在一个给定的时间段内， 根据用户行为日志行为数据为用户提供满足用户兴趣的N件物品。

#### 推荐方法-基于图模型的协同过滤算法

利用二分图为用户行为建模。用户物品二分图定义为 G(U, I, E, w)，其中I是物品节点集合，U是用户节点集合，若用户 u对物品i 有过行为，就存在一条边e(vu, vi) ∈ E连接用户 u的节点vu和物品i的节点vi，而行为的强度定义为边的权重w(e)。如图2所示， 图中定义了U1, U2, U3 和 I1, I2, I3, I4之间的关系。 从图的左半部分我们可以看到用户和物品之间的关系； 比如 U1 和 I1, I4之间有行为； U2对 I1, I3, I4有行为；U3对I2, I3有行为。

.图 2 用户-物品关系二分图

Top-N推荐的主要任务就是找到和用户有行为的物品集；反映到二分关系模型上就转化为了用户节点到物品节点是否存在路径， 有几条路径， 路径的度为多少； 比如，在图2中用户U1到I2,之间并没有直接关系，但是有 U1 → I4→U3→ I2这条路径到达I2， 也就是说U1和I2,之间有一条长短为3的关系。计算图中顶点之间距离的算法有很多，Fouss在文[[[6]](#endnote-4)]中总结了很多， 我们介绍一种路径融合算法[[[7]](#endnote-5)]

#### 路径融合算法

算法的基本思想为是，图上两个节点相似的判断标准是：

1. 两顶点之间有许多条边相连；
2. 两顶点之间路径比较短；
3. 顶点之间路径不经过出度很大的点。

根据以上三个标准，图中顶点的相似度关键取决于连接顶点的路径。算法在计算顶点的相似度的时候，首先找到顶点间的所有路径，并计算每条路径的权重，并利用全部路径权重的和来衡量顶点之间的相似度。

以图3为例。图中一共有A,B,C,D四个用户和i1,i2,i3,i4四个物品。其中用户A 喜欢物品i1,i3，用户B喜欢物品i1,i3,i4，用户C喜欢物品i1,i4，用户D喜欢物品i2,i4。如果要给用户A 做推荐，可以推荐物品i2和i4，因为这两个物品用户A还没有过行为。但用户A和物品i2,i4之间的相似度却不同。根据上面的三条准则，用户A和物品i4在图上的相似度更高。如图所示，用户A和物品i4之间的最短路径长度是3，并且长度为3的最短路径有2条。而用户A和物品i2之间的最短路径是5。因此根据第二条准则， A对i4的喜欢程度很可能大于A对i2的喜欢程度。此外， A,i4之间长度为3的路径有两条(A,i1,B,i4)和(A,i1,C,i4)。那么根据上面第三条， (A,i1,C,i4)是一条比(A,i1,B,i4)更重要的路径，因为顶点C的出度2，小于顶点B的出度。同理， A,i2之间长度为5的两条路径(A,i1,B,i4,D,i2)的重要程度也小于(A,i1,C,i4,D,i2)，也是因为B的出度值大于C的出度。



图 3 路径融合算法

通过上面简单的例子可知， 路径融合算法在计算用户对物品的喜好程度时通过下面三个步骤. 第一条找出用户顶点和物品顶点之间的最短路径；第二条计算每条最短路径的权重；第三条将所有最短路径的权重线性叠加作为最终用户对物品喜好程度的度量。

详细的路径融合算法

### 隐性反馈数据的动态变化

#### 用户的长期兴趣和短期兴趣(fix)

用户的兴趣可以分为长期兴趣和短期兴趣。比如一个历史爱好者一般阅读历史方面的文章。但是当08年奥运会开始的时候，他也会关注奥运方面的体育新闻。在这个例子中，历史属于这个用户的长期兴趣，而奥运新闻只是该用户在奥运期间受社会热点影响而产生的短期兴趣。由上面的例子可以看出，长期兴趣一般是由于用户自身的兴趣爱好产生的，而短期的兴趣一般是受社会热点的影响而产生的。一个好的推荐算法需要通过同时考虑用户的长期兴趣和短期兴趣来给用户做出推荐。上一节介绍了基于图模型的Top-N推荐算法，这一节将研究如何将用户的长期兴趣和短期兴趣同时建模到图模型中，并在模型的基础上设计推荐算法。

为了表达用户的短期兴趣，我们引入了用户时间节点。在前面的二分图模型中，用户 u会和所有他感兴趣的物品节点通过带有权重的边相连。但是用户不同时刻的兴趣不同。为了表示用户 u在 t时间段的兴趣，我们引入了 vut节点。这个节点将会和用户 u 在 t时段感兴趣的所有物品节点通过边在图上相连。这里每个时间段的时间长度将根据数据集的不同来选择不同的长度。

引入用户时间节点，用户物品二分图模型转变为时间段图模型 Session based Graph Model (SGM)。 SGM是一个二分图，它被定义为 G( U, S, I, E, w)。其中U是用户节点集合， S是用户时间节点集合， I是物品节点集合， w : E →R 定义了边的权重。图4是一个SGM的简单例子，图中包含了2个用户节点，4个session节点以及4个物品节点。图中表明，用户A对物品i1,i2有过行为，其中A在1时段对物品i1有行为， A在2时段对物品i2有行为，用户B对物品i1,i2,c有行为，其中B在1时段对物品i1有行为，在2时段对物品i2,i3有行为。

在SGM中，用户节点vu将和用户 u曾经发生行为的所有物品(这些物品的集合用N(u)表示)对应的节点都有边相连，这表示了用户的长期兴趣。而用户时间节点vut会和用户在t时刻有发生行为的所有物品(这些物品的集合被记为N(u, t))对应的节点有边相连。因此，如果从用户节点vu出发，会经过N(u)中的物品节点而最终到达用户 u没有过行为，但和 N( u)中物品相似的物品对应的节点。而如果从用户时间节点 vut出发，最终则会到达和 N( u, t)中物品相似的物品对应的节点。因此，用户节点和用户时间节点分别代表了用户的长期兴趣和短期兴趣，通过给这两种节点上赋予不同的权重，就可以控制用户的长期兴趣和短期兴趣对最终推荐结果的影响。SGM中一共有三种不同的节点，它们的权重定义如下：

其中β ∈ [0, 1]控制了长期兴趣和短期兴趣对推荐结果的影响。



图 4 时间段图模型

#### 基于SGM的推荐算法

本节主要讨论如何在时间段图模型SGM上度量用户对物品的兴趣并设计个性化推荐算法。 2.2.2.1节中介绍了用户物品二分图模型上的个性化排名算法，本节将对那些算法进行改进，让他们能够运行在SGM模型之上。

路径融合算法

## 本章小结

本章主要讨论如何利用图模型在Top-N推荐中同时考虑用户的短期兴趣和长期兴趣，从而提高推荐结果的准确度。首先本章讨论了传统的用户物品二分图模型以及基于该模型的推荐算法。在介绍相关研究的基础上，本章提出了基于路径的路径融合算法，该算法通过分析图上节点之间的路径来度量节点之间的相似度。然后，我们发现用户的兴趣分为长期兴趣和短期兴趣，并通过在用户物品二分图模型上加入一种新的节点来对这两种兴趣进行建模。两个不同数据集上的实验表明，考虑用户的长期兴趣和短期兴趣能够提高Top-N推荐的准确度，同时路径融合算法相对与其他算法也具有更高的推荐准确度。

### Hello

#### Hello

# 评分预测中用户兴趣预测的模型

## 第一节

正文中要用缩写的，第一次出现时必须用全称，题目中不宜出现缩写。

### 

对外统一的出错处理函数

根据错误的代号确定错误种类

根据错误的文件名以及行号确定

错误位置

打印错误信息

清空连接器前申请资源，退出连接器程序

图 3.1 流程图

## 本章小结

### 

# 推荐引擎设计与实现

## 第一节

### 

数学公式一般另行起排，居中书写，并用阿拉伯数字分章编号。若数学公式前有文字（如"解"、"假定"等），文字空两格写，数学公式仍居中写。数学公式序号按章编排，序号加圆括号，右顶格排。如第1章第1个数学公式序号为"公式（1.1）"。文中引用数学公式时，一般用"见公式（1.1）"或"公式（1.1）"。

## 本章小结

### 

# 用户行为日志系统设计与实现

## 第一节

### 

## 本章小结

### 

# 展示系统设计与实现

## 第一节

### 

## 本章小结

### 

# 

## 第一节

### 

## 本章小结

### 

# 

## 第一节

### 

## 本章小结

### 

参考文献

1. Wiki： http://zh.wikipedia.org/wiki/%E9%95%BF%E5%B0%BE [↑](#footnote-ref-1)
2. []James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor VanVleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, and Dasarathi Sampath. The youtube video recommendation system. In Proceedings of the fourth ACM conference on Recommender systems, RecSys’10, pages 293–296, New York, NY, USA, 2010. ACM [↑](#endnote-ref-1)
3. []Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing, 7:76–80, January 2003. [↑](#endnote-ref-2)
4. Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing, 7:76–80, January 2003. [↑](#endnote-ref-3)
5. 数据来源××××× [↑](#footnote-ref-2)
6. Francois Fouss, Alain Pirotte, Jean-Michel Renders, and Marco Saerens.Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. IEEE Trans. on Knowl. and Data Eng., 19:355–369, March 2007. [↑](#endnote-ref-4)
7. Liang Xiang, Quan Yuan, Shiwan Zhao, Li Chen, Xiatian Zhang, Qing Yang, and Jimeng Sun. Temporal recommendation on graphs via long- and shortterm preference fusion. In Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD ’10, pages 723–732, New York, NY, USA, 2010. ACM. [↑](#endnote-ref-5)