单位代码 10144

分类号 TP391.37

编号 _____

硕士学位论文

题　　目 基于改进随机森林的[illegible]

[illegible]研究

研究生姓名 吴迪

（2017届 系统工程 专业）

导师姓名 黄海新

论文完成日期 2016年12月

沈阳理工大学

Shenyang Ligong University

沈阳理工大学
硕士学位论文原创性声明

本人郑重声明:本论文的所有工作，是在导师的指导下，由作者本人独立完成的。有关观点、方法、数据和文献的引用已在文中指出，并与参考文献相对应。除文中已注明引用的内容外，本论文不包含任何其他个人或集体已经公开发表的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

作者（签字）：吴迪

日　　期　：2017年3月9日

学位论文版权使用授权书

本学位论文作者完全了解沈阳理工大学有关保留、使用学位论文的规定，即：沈阳理工大学有权保留并向国家有关部门或机构送交学位论文的复印件和磁盘，允许论文被查阅和借阅。本人授权沈阳理工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，可以采用影印、缩印或其它复制手段保存、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：吴迪　　指导教师签名：黄海新

日　　期：2017.3.9　　日　　期：2017.3.9

分类号：TP391.97　　密级：
ＵＤＣ：621.3　　编号：

工学硕士学位论文

基于改进随机森林的推荐算法研究

硕士研究生：吴　迪
指导教师　：黄海新 副教授
学科、专业：系统工程

沈阳理工大学
2017 年 3 月

分类号：TP391.97　　密级：
ＵＤＣ：621.3　　编号：

工学硕士学位论文

基于改进随机森林的推荐算法研究

硕士研究生　：吴　迪
指导教师　　：黄海新　副教授
学位级别　　：工学硕士
学科、专业　：系统工程
所在单位　　：自动化与电气工程学院
论文提交日期：2016 年 12 月
论文答辩日期：2017 年 3 月 9 日
学位授予单位：沈阳理工大学

Classification Index: TP391.97
U.D.C: 621.3

A Thesis for the Degree of M.Eng

Research on Recommendation Algorithm Based on Improved Random Forest

Candidate : Wu Di
Supervisor : Vice Prof. Huang Haixin
Academic Degree Applied for : Master of Engineering
Speciality : System Engineering
Date of Submission : December ,2016
Date of Examination: March, 9,2017
University: Shenyang Ligong University

摘　　要

随着社会经济的发展，电子商务已经成为生活中不可缺少的一部分。面对电子商务中信息呈几何级数式增长，用户很难在海量的商品信息中快速准确的找到自己感兴趣的商品。个性化推荐算法就是这样的背景下所创建。推荐算法改变了电子商务中从被动接收用户请求到主动为其推荐的方式，同时也为用户解决了从信息过载的网络中找到自己喜欢物品的捷径。本文使用的是基于改进随机森林模型的推荐算法。

随机森林算法是一种包含多个决策树分类器的统计学习理论，采用了特征子空间来构建模型，能较好的处理噪声且避免发生过拟合。本文针对几种典型的决策森林算法，阐述了其原理和算法的特点，并从决策森林的构建过程出发，提出了一种改进随机森林方法。

本文提出一种支持向量机和随机森林算法融合的改进随机森林算法。随机森林中基本弱分类器是决策树，而决策树在进行节点分裂是选择分类能力最强的某个属性。本文在决策树的属性选择中结合支持向量机算法，以特征变量的线性组合（支持向量）构成的超平面进行分裂，比单一属性的分类能力更强，从而在随机森林决策树的建造过程中得到了改进。通过实验分析，充分说明了改进随机森林算法具有更高的准确率。

本论文使用的是阿里巴巴线上的真实用户历史行为数据，通过挖掘用户行为建立改进随机森林算法模型，最终得到了为用户推荐商品列表。实验表明，在对用户历史行为数据的前提下，可以有效地对用户未来购买商品进行预测和推荐，对推荐算法发展具有重要意义。

关键词：随机森林；推荐算法；预测；大数据；机器学习

Abstract

With the development of social economy, e-commerce has become an indispensable part of life. The exponential growth of information in e-commerce makes it difficult for users to quickly and accurately find the goods of interest in the mass of commodity information. The personalized recommendation algorithm is created in this context. Recommendation algorithm changes the way of e-commerce receiving requests from passive users to actively recommending them. It also solves the shortcuts for users to find their favorite items from the information overload network. In this paper, we use improved RandomForest algorithm into recommendation algorithm.

Random Forest algorithm is statistics theory that combins the set of decision tree classification and it has feature subspaces to construct the modelcan deal with noise and avoid over fitting surpassingly. In this paper we mainly introduced the several classic methods of Random Forest algorithm and their characteristics. Researching algorithms in domestic and overseas were analyzed and summarized systematically from the process of the construction of the decision forest, and we propose an improved method for random forest algorithm.

In this paper, we propose an improved stochastic forest algorithm with support vector machine and stochastic forest algorithm. The basic weak classifier is the decision tree in the random forest, and the decision tree is the most powerful one in choosing the classification ability. In this paper, combining the support vector machine (SVM) algorithm with attribute selection of decision trees, the hyperplane of linear combination (support vector) of feature variables is divided, which is more powerful than single attribute classification. It has been improved in the process of random forest decision tree construction. The experimental results show that the improved random forest algorithm has high accuracy.

In this paper, we use the real user behavioral data of Alibaba to establish the improved random forest algorithm model by mining user behavior, and finally get a list of recommended items for users. Experiments show that it can effectively forecast and recommend the future purchase of items, which is of great significance to the development of recommendation algorithm under the premise of user's historical behavior data.

Key words：random forest；recommendation algorithm； prediction；large data；
machine learning

目　录

第1章 绪　　论

1.1 研究背景及意义

随着社会和经济的发展，许多传统行业也搭载上了网络这趟快车，在加快了自身发展的同时也方便了人们的生活。这样的互联网+模式不仅促进了云计算、大数据、物联网和移动互联网等商业的联系，而且推进了电子商务、工业互联网和互联网金融的迅速发展，但这也带来了互联网的主要问题与挑战-信息过载。

信息过载这个难题可以简述为无法从海量数据里准确且短时间内找到用户所需求的有用的信息[1]。而推荐算法（Recommendation Algorithm，RA）就是解决上述难题的有效方法之一，且已经被应用到电子商务、视频网站、交通和生物医疗等多个领域。虽然目前已经提出了多种推荐算法，但是如何使推荐算法更加智能、更加具有鲁棒性仍然有着巨大的挑战。推荐算法[2]是以用户为核心、以数据信息为动力，从这些海量的信息中找到满足用户需求的信息为目标，而用户不仅是数据信息的生产者更是数据信息的消费者。以目前最具代表性的为解决搜索引擎的谷歌和分类目录的雅虎为例，随着互联网规模的幂指数增长，分类目录只能为那些热门的网站进行分类，而无法通过将网站分类为用户得到方便的按类别搜索的网站。在这些大量数据的背景下，搜索引擎诞生了。这样用户只需提供简单的几个关键字即可找到自己需求的信息。但是当用户没有为自己所需求的数据内容给出确切的关键字时，搜索引擎此时无法为用户进行服务了，况且这种根据关键字形式呈现的结果不能为用户解决多场景下个性化需求。在这种背景下创建了推荐算法，因为这种以推荐算法为代表的信息过滤技术的主要思想是通过挖掘海量行为数据，分析和生成用户的需求从而主动向用户提供他们所感兴趣的信息。推荐算法是通过分析用户历史行为数据从而得到用户的感兴趣点，以达到帮用户找到他们所需要的信息的目的。其实和搜索引擎功能相似，推荐算法也是一种能帮助用户快速找到对其有用数据的一种方式。但又和搜索引擎是不同的，因为推荐算法不需要用户写出确切的关键字及需求，而是利用用户自己的历史行为数据

通过分析来构建模型，从而主动为用户推荐那些能让他们满意的需求信息。而对于商品的提供方，利用推荐算法这门技术，可以为用户提供其所感兴趣的商品，以达到用户和商品提供商互利共赢的目标。而在整个过程中，是不需要用户提供产品的关键词就能达到的，实现了智能化。另一方面推荐算法能够挖掘物品的长尾（long tail）从而提高商品的利用率[3]。推荐算法作为新时代的互联网+中不可缺少的为用户提供个性化信息的形式，已被广泛应用到多个领域，最大化了它的学术和商业价值。

近几年来随着电子商务的迅速发展，为推荐算法的研究注入了新的领域的同时也使得推荐算法得到了更广大的应用。用户面对着数以万计的商品而无法选择时，利用传统的搜索方法显然是无法达到精准快速的找到所需物品的目标，且众多商品的排序原因，会使用户丧失很多对其有意义物品的浏览。在用户浏览其查询的目标商品时，会在商品中迷失进而产生选择困难的情况从而体验不到很好的浏览商品选择商品的过程，最后可能就会造成用户对消费失去兴趣导致引起商户经济利润的损失。随着商品的种类的多样性的丰富、物流运送速度的提高和网上支付性财产安全的加强，越来越多的消费者已经开始选择在网上购物的形式，或者说网上购物已经成为现代人们生活中的一部分，所以高质量的个性化推荐不仅会使用户提高对该购物网站的依赖心理和较高的忠诚度外，也对商户在大数据时代的改革中增加了其适应度，提高了社会效率和经济服务[4]。例如淘宝，除了利用高质量、精准的个性化推荐提高营业额外，每年还举办各种类型的推荐算法大赛吸收各高校同学新的想法应用到实践中。产品方面，搜索是一个通用系统，固定的产品形式就是有一个框。被推荐的物品有很多种类，而不同背景不同领域也是不一样的，比如服饰类是基于风格、搭配来推荐，书籍要考虑领域背景、类别、教育水平，而电影、音乐则是另外一种考虑的推荐方式。所以应用数据建立一个个性化推荐平台，为用户分析从而产生更高的价值和利润，这不仅是对现实生活中电商的一个应用，也对学术界、电商、交通和视频网站等领域都有较高的研究意义。

1.2 国内外推荐算法研究及发展现状

斯坦福大学的 MarkoBalabanovic[5]等在 1995 年 3 月的美国人工智能协会上提出了个性化推荐系统；卡耐基梅隆大学的 RobertArmstrong 等也在这个会议上提出了个性化导航系统，这是推荐算法第一次出现在公共视野中。

正是由于推荐算法的应用需求庞大，所以自从被推出就受到了广泛的关注。随着美国计算机协会屡次举办以推荐算法为主题的研讨会，我国学术界对推荐算法的研究也越来越重视，特别是希望推荐算法在商业化的应用上能够一展宏图。

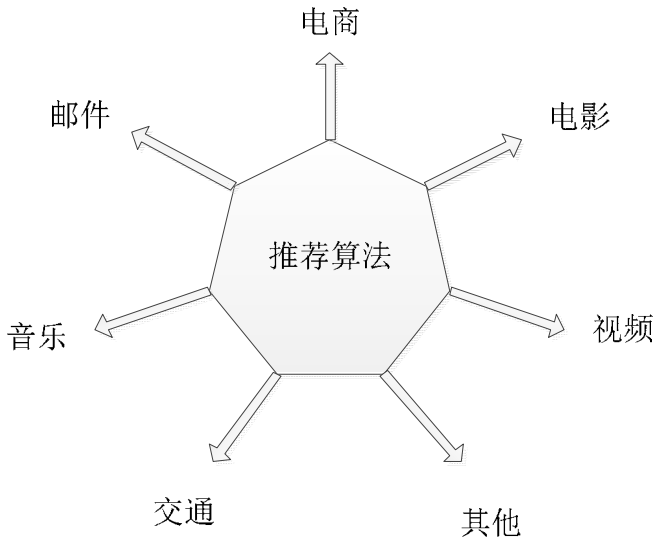

图 1.1 推荐算法的应用领域

Fig.1.1 The applications of recommendation algorithm

由上图可知，推荐算法的应用范围是庞大的。在学术界，是从 2007 年的 ACM（Association for Computing Machinery）组织了推荐年会后，对推荐算法的研究才开始大量的展开[6]。并在 2016 年举行了第 9 届 IEEE/ACM 计算与云计算国际会议，而后 IEEE/ACM 将会与 BDCAT 2016 国际会议共同举办第 3 届大数据应用技术会议。该会议将会以云计算为基础着重讨论大数据领域的不断发展所带来的问题的解决方法和国内外一些顶尖学者对这个问题的处理和解决方法，而推荐算法也会作为一个主要部分在会议中进行提出和讨论。

随着推荐算法的广泛应用，很多学者根据不同的使用领域提出了多种推荐算法，而目前应用最多和最广泛的推荐算法为：混合推荐算法[7]、协同过滤推荐算法[8]、基于内容的推荐算法[9]和基于知识的推荐算法[10]。而协同过滤推荐算法按

照生成方式的不同又分为基于邻域的协同过滤算法（又称基于内存）和基于模型的协同过滤算法[8]。推荐算法的主要思想是通过算法将用户和物品关联起来，让用户和信息提供商在信息过载的条件背景下找到用户自己的需求和为商户找到他们需求的目标用户。对于基于内容的推荐算法，新项目进入后先根据数据特征属性构造特征向量最后根据用户偏好决定是否对其进行推荐。 但是在整个过程中，相似性的计算在面对数据量非常大的情况下效率是及其低的。对于基于知识的推荐，是根据用户各不相同的背景知识提供对其个性化的推荐结果。该算法可以理解成是一种推理方法而不是建立在用户的喜好和需求进行推荐。对于协同过滤推荐算法不能为用户推荐新物品是它主要的缺点,而且依赖于惯用数据(例如评价、购买、下载等用户偏好行为等)，这个缺点又被称为冷启动问题。

表 1.1 各推荐算法优缺点

Table 1.1 Recommendation algorithms' advantages and disadvantages

推荐算法	优点	缺点
基于内容的推荐算法	不需要该领域知识；	容易归档用户；
	推荐结果清晰直观；	缺少多样性；
	没有冷启动问题；	难以联合多个特征；
基于知识的推荐算法	把用户需求映射到物品上	各领域知识难以获取；
协同过滤算法	推荐结果个性化较精确；	可扩展性问题；
	数据复杂非结构对象处理效果好；	数据稀疏性问题；
	不需要该领域知识；	冷启动问题；
	能发现用户新的感兴趣商品；	
混合推荐算法	处理数据稀疏问题；	需大量准备工作；
	能够更有效的利用用户数据；	
	不需要该领域知识	
	推荐结果精度高；	
	无冷启动问题；	
	没有流行度偏见，可推荐罕见物品；	

表 1.1 对比了现有的较主流的个性化推荐的优缺点。

个性化推荐算法是在基于数据的特性基础上生成的具有顺序的物品列表[11]。推荐算法会结合这个列表和用户对于商品的喜好程度以及其它的约束来为用户找到最满足其需求的物品。个性化服务作为一种新兴的服务方式，会在分析用户历史信息的基础上获取用户的行为偏好针对不同用户提供满足用户特点的信息或服务。我们日常接触的电子商务领域是个性化推荐服务应用成功的典范。各电子商务网站会根据用户的商品购买记录、商品访问记录和历史评价信息等作为发掘用户的潜在购买意向，然后搜寻与用户潜在购买意向相符的商品推荐给用户，最后再通过对应推送的用户反馈对推送信息的进行评价和修整，尽力获得用户更高的满意程度[12]。在与用户交换或向用户推荐的过程中，如果推荐系统能很好的引导用户并满足其需求，那么就会留住更多用户从而获得更多的能反应用户偏好的信息而使推荐效果更优。目前亚马逊、京东商城、当当网、天猫等诸多电子商务巨头都已经有了自己推荐系统，这既能向新用户推荐商品吸引其对网站的关注度也可以持续满足老用户的需求，同时还能通过交叉销售产生巨大的利益。

针对现有推荐算法的诸多缺点，如面对数据稀疏时推荐效果不好和推荐精度有待提高等，本文设计一基于改进随机森林算法模型的推荐算法。

1.3 课题的研究内容及创新点

本课题通过对相关文献及各著作细致地阐述和分析，介绍了随机森林算法，重点阐述了国内外现有关于推荐算法的研究现状和应用现状，制定出了基于改进随机森林的推荐算法的具体智能研究方法、研究思路及所研究的具体结果。

本课题实现了对传统随机森林算法的改进。主要利用支持向量机（Support Vector Machine，SVM）和 随机森林（Random Forest，RF）算法融合进行对用户历史行为的分析，实验对比结果证明改进的支持向量机-随机森林算法比基于经典随机森林模型算法模型的推荐算法准确率高。本课题的数据是根据阿里真实用户的历史行为分析，在此背景下的推荐算法在电子商务下的应用。主要的研究工作和创新点如下：

(1) 建立随机森林模型的推荐算法。

随机森林中有众多基本弱分类器,在准确度上高于基于其他模型的推荐算法。如小众的用户，在其他模型的推荐算法中可能会被错认成噪声点，但基于随机森林模型的推荐算法因为森林中众多的决策树，就能为这些用户个性化推荐。

(2) 提出一种 SVM-RF 算法。

在对随机森林改进中，从决策树的建立过程时决策树的每个分支点为单属性划分，因为在多数情况下各属性是相关联的，所以建立多属性组合划分，以提高其决策效果。结合 SVM 原理,对决策树中的单属性进行组合产生新的特征属性,使每次分类时不需要多次迭代就可以达到叶子状态。实现了在决策树的建造过程中实现了对随机森林算法的改进。

(3) 改进随机森林算法与传统随机森林算法的分析及对比。

(4) 建立基于改进随机森林模型的推荐算法。

(5) 应用提出的改进推荐算法对真实的阿里巴巴用户数据进行计算，最后对推荐结果进行分析和对比。

本课题使用的是阿里巴巴线上的真实数据，所以试验结果有很重要的参考价值。具体实验思路为先对数据进行预处理，构造特征集，即是对原始数据用时间区间进行划分、剔除不重要特征。再根据时间划分数据集，然后按比例抽样分为训练集和测试集。最后根据 SVM 算法和随机森林算法的基础，提出基于 SVM 的随机森林算法的混合算法模型，通过挖掘用户行为得到用户行为结果相关的特征集，通过特征选择得到预测模型最终得到了为用户推荐商品列表。

1.4 本文的组织结构

本论文的主要目的是对用户的历史行为进行数据进行处理和计算以获得用户的兴趣模型，应用该模型对用户进行个性化推荐，得到推荐物品列表。

本文总共含有 5 章，每章的具体内容简述说明为：

第一章 绪论。首先介绍了本文的研究背景和研究意义,接着概括地介绍了国内外推荐算法研究现状，说明了现有被广泛应用的几种主要的推荐算法，最后对本文的整体框架进行了阐述，并简述了本文的创新点。

第二章 推荐算法研究。阐述了推荐算法基本原理,详细的说明了目前主要的

几种推荐算法。本章节引入基于模型的混合推荐算法概念，说明了基于模型的混合推荐算法的优点。对于基于模型的推荐算法，主要分为基于回归模型、基于聚类模型和基于分类模型三种，并主要说明了基于分类模型，引出本文研究内容。最后，介绍了现有的推荐算法的评价方法。

第三章 随机森林算法研究。对于本文应用的是基于分类模型的混合推荐算法，分类模型应用的是本文提出的改进随机森林算法，故先介绍随机森林算法。本章基于随机森林算法分析其他几种典型的决策森林算法的特性及不同，进而说明了不同决策森林算法的适合应用范围。通过比较各种算法，根据使用者背景的不同选取能解决问题的最适合的决策森林算法。介绍了面向大数据时，基于决策森林的并行处理数据的方法。

第四章 基于 SVM 的随机森林算法。本章节算法主要描述提出的改进分类算法模型，在本节中对比决策树算法、随机森林算法、adaBoost 算法，说明了提出的改进算法具有较高的准确率。

第五章 基于改进随机森林模型的推荐算法。通过阿里巴巴真实用户的历史行为数据，应用改进随机森林算法分类模型的推荐算法进行个性化推荐。本章节详细说明了对数据预处理的过程以及通过模型预测出用户感兴趣的商品。通过评估标准可知基于改进随机森林模型的推荐算法具有较好的有效性。

第 2 章 推荐算法

2.1 前言

推荐算法[2]是从海量的数据中挖掘有价值的信息提供给用户，为用户提供准确的和实时的个性化服务，为用户提供有价值的物品核心目标。从人工智能方向来看，推荐算法的任务可以理解为利用用户过去知识来学习的问题。用户的多个特征可以反应出他们的偏好。

对于推荐算法的分类方式有很多种。若按照算法构成建立方式分类，可分为基于模型的推荐算法、基于用户和商品自自身的推荐算法（这类方式的推荐算法会生成用户-物品的二维矩阵来描述用户的偏好）和基于关联规则的推荐算法[13]（这类方式最典型的推荐算法是 Apriori）。根据使用不同数据源，推荐算法又可以分为基于内容的推荐算法、协同过滤推荐算法和混合推荐算法[14]。对于基于内容的推荐算法是为用户提供与他们之前喜欢的物品相类似的物品，但是由于难以提取用户和物品之间的特征从而导致用户-物品档案数据不完整，使基于内容的推荐算法使用率不是很高。而协同过滤算法是利用对用户分析后生成的物品喜好模型来对用户进行推荐[15]。混合推荐算法是把几种推荐方法混合使用来弥补每种推荐方法的缺点，这样会得到比单个推荐方法的推荐结果更加准确。

2.2 主要的推荐算法研究

2.2.1 基于内容的推荐算法

基于内容的推荐算法[9]（content-based）是为用户推荐与用户历史行为中感兴趣物品相类似的物品。利用物品的特征属性来计算两个物品间的相似性并把相似性高的推荐给用户，当有新物品时用这样的方式来为用户推荐他可能喜欢的物品。举例说明为：若用户对一首校园歌曲加入试听列表，那么推荐算法就会从同类型的歌曲中为该用户推荐其他歌曲。

基于内容的推荐算法，它主要的思想是与信息检索相同并采纳了很多信息检

索和信息过滤的相关技术。具体算法思路为：通过分析物品的特征或物品的评分数据信息来确定物品间的相似度，将相似度高的物品推荐给用户提供推荐。

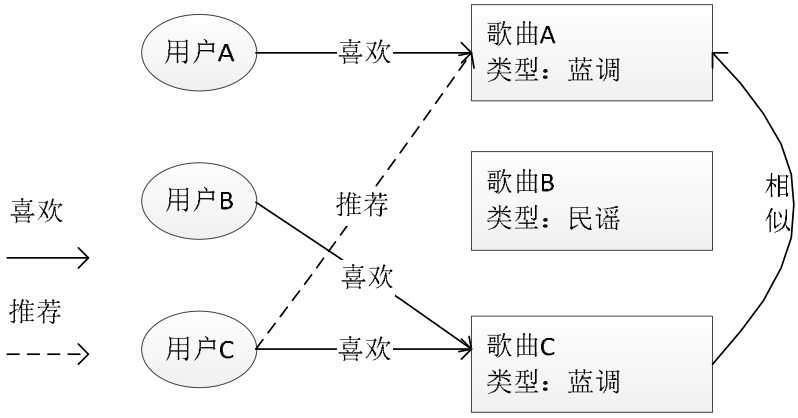

图 2.1 基于内容的推荐算法过程

Fig.2.1 The process of the based on the content recommendation algorithm

由于基于内容的推荐算法本身的性质，这种算法只能为用户推荐与他们信息匹配度高的物品，这样导致了用户被固定在了那些被用户评价过的物品的相似集合中。这样基于内容的推荐方法的缺点是无法为用户提供“惊喜”的物品，因为算法提供的推荐列表中的物品是与用户自己已评分的物品相类似的，这样导致推荐结果范围的限制。

但是对于基于内容的推荐算法有着它一个非常大的限制，就是新用户问题。在分析用户偏好物品类型之前，用户要给出这些物品的评分。但是若对物品的评分不充足只有少量评分可以使用的时候，推荐结果就会不够准确。但是对于新物品，在没有任何评分的情况下基于内容的算法是可以进行推荐。而协同过滤算法却不是，主要是因为基于内容的推荐过分依赖用户喜好而提供推荐的[16]。与协同过滤方法相比，基于内容的推荐方法只使用用户当前已评分物品的数据来构建每个用户的信息特征，而协同过滤算法是需要其他用户对物品的评分去生成这个用户的近邻。另一个缺陷是推荐对象的特征数量和所属类型的限制。但是当物品的特征属性不明确（就是物品的内容不全使得无法判断哪些物品是用户偏爱的哪些物品是用户不需要的）时，基于内容的推荐所给出的结果是十分不准确的。对于基于内容的推荐，它的优点是可以很显示地列出最后推荐结果中物品的特征[17]。

2.2.2 基于知识的推荐算法

基于知识的推荐算法[10]（knowledge-based）是根据背景领域的相关知识为基础的为用户进行推荐。所用到的知识是关于确定物品的哪种特征是可以引起用户的需要和用户的喜好，最终利用这些特征来确定该物品是否对用户有用。

基于知识的推荐方法最大的优点是不用建立用户的偏好模型，这样就不会产生冷启动的问题，影响其推荐精确度的因素是领域知识的获取程度。基于知识的推荐方法可以分为两种：基于约束的推荐和基于实例的推荐。

论文[18]根据实例详细说明了基于知识的推荐算法，主要是根据相似函数来计算用户的需求与最终的推荐结果的匹配度。推荐过程可以简述为：先收集用户的需求信息，利用之前预定义的知识库与用户的需求和物品的特征进行关联，若找不到合理的解决方案系统就会主动给出不太合理的结果。当数据源不进行更新时即不添加新用户或者新的物品时，基于知识的推荐算法会比其他方法效果好。但是更新数据时其他算法会利用用户历史行为数据生成比基于知识推荐更好的结果。

2.2.3 协同过滤算法

协同过滤推荐算法[19]（collaborative filtering，CF）利用它的简单和有效的推荐方式在实际应用时被广泛地使用，因为它会产生比较准确的推荐结果。该算法可以克服基于内容的推荐方法的一些局限性，当物品的特征不完全或者物品的特征难以获得时，它仍可以通过其他用户的行为信息给当前用户生成推荐结果。

按照算法生成方式的不同，协同过滤算法可以分为基于近邻(基于内存)的协同过滤（nearest-neighbor）和基于模型的协同过滤（model-based）[20]。

1. 基于近邻的协同过滤算法

基于近邻的方法又称基于内存的推荐。是在预测过程中直接利用已经存在的数据进行预测，由评分标准化、相似性权重的计算和近邻选择几部分组成。该方法可以通过两种方式进行推荐，分别是基于用户（User-based CF）相似度的推荐和基于物品（Item-based CF）相似度的推荐。基于用户的推荐算法中利用已经对该物品评过分的用户并且这些用户要与目标用户有相类似评价历史行为来估计目标用户对物品的偏好程度。这里，近邻的概念就是在评价过的物品中，两个对该

物品评分相似的用户且这两个用户有相似的其他的评价习惯。这里近邻的概念是被用户评价过的一些相类似的物品且这类物品具有相同的特征。这种算法能够启发式地利用用户已经评过分的用户-物品评分矩阵对新物品进行评分。

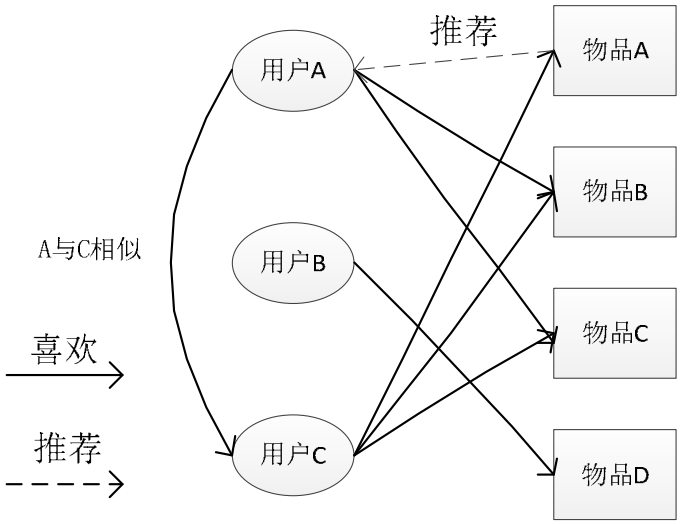

图 2.2 基于用户的协同过滤推荐

Fig.2.2 The process of the user-based collaborative filtering recommendation

论文[21]研究表明，基于近邻的方法在预测分数方面低于基于模型的方法，但是预测精度分数高并不代表用户很满意或用户获得了效率高、速度快的体验。因为评价一个推荐算法好坏的一个重要的因素是惊喜度，惊喜度的概念是会帮助用户找到他喜欢的物品但这个物品之前有可能是用户自己找不到的，这会使得推荐商品列表具有新颖度使得用户具有较高的用户体验。基于近邻的方法能分析出物品的关联性，只要该用户的近邻用户给出了一些物品很强的评分，这个特点使得在为用户推荐的物品列表中存在与该用户平时喜欢不一样的物品或者说是一些会被忽视的物品。

但近邻数量选择的不同对于推荐结果正确率的影响很大。通常，基于近邻的推荐算法可以分为两个步骤[22]，(1)使用全局过滤得到最有可能的近邻；(2）在预测的过程中选择出最佳近邻。在实际应用背景下用户和物品的数量是非常巨大的，所以不会将所有的用户间的或者物品间的相似度都存储在内存中。对于第一步，通过减少近邻数量和降低相似度权重实现。通常有阈值过滤和 Top-N 过滤等方法。当第一步完成了后，就为每个用户（或物品）得到了他们的近邻，之后通过使用 k 近邻方法得到相似权重最大的 k 个近邻。若用户的数量比物品的数量大

的很多，那么选择基于物品的推荐效果会更好，因为这种情况时，更新的频率不会很高且计算的效率会更高。另外当基于近邻的方法遇到数据稀疏问题时，可以使用降维方法解决。

2. 基于模型的协同过滤算法

基于模型的方法[23]是用评价矩阵训练数据，使用用户的显式反馈学习模型或利用其他算法学习出的模型给目标用户推荐物品。使用机器学习模型（如分类模型）去辨别多种多样的用户或者物品，并以此为基础给出推荐预测结果。

推荐算法的推荐效果和输入数据的形式有较大关系。最好的输入是用户直接表达出的喜欢的物品的数据，简单来说就是用户高质量的显式反馈。但是显式反馈一般不容易得到，故隐式反馈就被广大地应用。隐式反馈在解决数据稀疏问题时或在显式反馈信息少量时会突出它的优点。隐式反馈是通过用户的历史行为来得到用户喜欢的物品。通常地，隐式反馈所包含的是用户的购买记录、搜索记录、加入购物车记录和收藏记录等。基于模型的协同过滤算法就是通过模型得到用户隐式反馈信息进而进行推荐。

模型就是在获得用户和物品之间的交互信息，通过这些交互作用得到不同的评分。基于模型的协同过滤推荐是使用隐语义模型得到数据隐藏的特征，而这些特征就会预测出当前的新物品的评分。目前基于模型的协同过滤推荐一般是基于聚类模型、基于回归模型、基于分类模型和基于矩阵因子分解模型[24]（Singular Value Decomposition，SVD）等。各个模型的具体特点、目前所应用的哪种机器学习算法和适用范围会在下一节进行详细说明。

2.2.4 混合推荐算法

从上面的分析可知，每种独立的推荐算法都有他们的优点或者缺点，而这些缺点就是造成他们推荐结果精度的原因。混合推荐算法（hybrid recommendation）综合利用几种推荐算法，利用一种算法的优势去弥补另一种算法的劣势，从而使推荐结果更加精确，更加使用户满意。实验表明[25]，混合推荐算法比单一的推荐算法效果好。组合算法的主要方法有：加权融合、特征组合、变换、多项罗列、级联、特征扩充和元级别混合这 7 种[26]。

混合推荐算法的主要应用方式为：

1.分别运行几种推荐算法，对每种推荐算法的结果分析并将这几种结果混合取并集，这样相当于扩大了推荐结果的范围；

2.分别运行几种推荐算法，对每种推荐算法的结果分析并将这几种结果加权重，这样相当于得到了一个加权的推荐评分；

3.对于数据矩阵稀疏问题，先应用可处理数据稀疏问题的方法将评分矩阵用预测评分填满，之后再运行另一些算法产生推荐结果，提高了算法的利用率。

最常用的就是将基于内容和基于协同过滤这两种方法进行混合。原因在于协同过滤推荐方法在遇到新物品的冷启动问题时，是没有办法推荐那些没有被评分的物品，而基于内容的推荐利用物品的特征得到新物品的预测评分从而没有冷启动问题的限制[27]。在基于内容的推荐算法中融入协同过滤方法最典型的方式是先利用降维技术处理用户的个人信息。混合推荐方法利用每种算法互补的特点，已被广泛应用。

混合推荐算法是将各种算法融合起来使用，这样会带来另一个好处:算法的应用背景不受算法自身的限制。混合推荐算法会利用应用背景的详细分析来解决普通推荐算法解决不了的难题和克服单个算法自身的缺陷。

另外，混合推荐算法的模型训练和独立推荐算法是不同的。一般独立推荐算法的训练过程是将数据集分成训练集和测试集，用训练集训练数据参数然后用测试集计算模型的准确率。可是混合推荐算法是将数据集分成了三个部分，除了上述说的两个部分外还包括了一个训练混合模型参数的数据子集。Toscher 和 Jahrer 提出[28]，在第一步用训练集训练独立的推荐模型后，会选取几组效果好的参数；第二步应用混合算法的训练集训练这组混合参数，选取出每个独立算法中效果最好的那组参数作为混合推荐模型的参数，当误差满足一定条件时训练终止；第三步使用测试集计算混合推荐模型的参数准确率。

2.3 基于模型的推荐算法

基于模型的推荐算法是与基于内存的推荐算法操作过程是完全不同的。基于内存的推荐算法，主要是将所有的用户数据读入内存再进行运算，但是当数据量

特别大时显然这种方法是行不通的，因此出现了基于模型的推荐算法。基于模型的推荐算法是依赖于机器学习算法的模型，工作方式为离线进行训练模型和在线实时推荐。

用户偏爱物品模型建立的根本是特征提取。如果用户的标注形式是离散的，则使用训练分类器的分类或聚类模型来实现。若用户的标注形式是数值，那么使用回归模型实现。基于模型的最大的好处是在提高了算法结果精度的同时可以有效降低基于内存算法的数据稀疏问题。

基于模型的推荐会根据用户的历史行为数据创建一个模型，每次预测时只需将建造好的模型和新数据导入内存中，是不需要每次都将所有数据导入内存中因此基于模型的推荐有很好的可扩展性。

基于模型的推荐技术旨在建立一个模型来表示用户评级数据，并使用该模型来预测用户对特定项目的偏好。论文[21]指出，基于模型的推荐算法的步骤为：(1)只作用在用户对物品评价的矩阵中。(2)使用任何评级之前生成的参考过程（总是对结果进行更新)。通常根据他们彼此间的评分使用相似性度量来获得两个用户直接的距离或两个物品之间的距离。

例如，LDA 模型从用户的文本描述中来推断潜在属性的物品，然后计算用户的喜好或根据用户历史评级推荐出的潜在的物品。简单描述为捕捉隐藏的用户和产品之间的联系。因为一些物品有一个潜在的相关性，有一定概率的项目将会出现在一起。最著名的例子是尿布和啤酒。虽然尿布和啤酒之间没有直接联系，但是发现买尿布的人会够买啤酒的几率是非常大的。LDA 是用概率模型来捕捉潜在的大宗商品之间的联系。吴和孙等人[29]利用奇异值(SVD)模型将潜在语义模型从信息检索到协同过滤的维数降低。奇异值模型在降低了 user-item 矩阵的维度的同时还消除了噪声信息。张和 Iyengar[30]通过使用从不同的领域的数据，对比了基于决策树模型的方法和基于内存的方法去预测用户的喜好。实验结果表明，基于决策树模型比基于内存推荐结果精确得多。

2.3.1 基于回归模型的推荐

在研究基于回归模型的推荐算法中，通常使用的是逻辑回归模型。算法思想

为：使用逻辑回归模型分析数据，研究特征属性在推荐中的作用接着从模型结果中选择最优的特征。逻辑回归函数可以对输入的参数进行参数的自我选优，这样通过最后选择的参数得到较好的特征。具体步骤是，（1）在离线状态下建立回归模型训练参数；（2）将优化选择的参数应用的推荐过程中进行分析。其他的推荐算法不能分析出不同背景下的数据特征不同的影响，而基于逻辑回归的方法可以通过用户购买的自我行为和商品行为来分析出用户和商品的特征。由此可以挖掘出不同数据的不同的特征来得到用户个性化推荐结果。

在实际应用中，用户或物品中都有很多个不同的特征，但如何从这些特征中选择那些会对购买行为起主要作用。基于逻辑回归模型的推荐就是会对这些特征进行分析和选择。对于回归模型很多人会第一时间想到线性回归，其实线性回归和逻辑回归的区别是这两者输入变量的不同。如果数据的连续变量多则用线性回归，但如果是二项分布则使用逻辑回归。论文[31]使用了来自微博网站推特（Twitter）和腾讯微博（Tencent Weibo）的数据对基于回归模型的协同算法进行验证。论文中将用户的历史行为和名人间的关系从原始的数据集中选取出，并在这两个社交网络的应用物品-名人进行对数据分析得出具体的数据集，最后实验结果得出基于回归模型的协同过滤推荐具有很高的有效性。

2.3.2 基于聚类模型的推荐

在推荐算法中，若用户和物品的数量增长很快但用户行为数据很少，在这种情况下会使得用户有历史行为的物品在总的物品数量占的比例很小，就会带来一个推荐算法中难以处理的问题-用户行为矩阵稀疏。这个问题所带来的后果就是推荐效率低而且推荐准确率更低。所以对用户和物品进行聚类，在聚类后的用户和物品中进行推荐会提高效率和准确率。

一般地，基于聚类模型的推荐[32]是使用 K-means 算法。推荐算法中的聚类算法是通过聚类找到有意义的用户群组或者物品群组，然后再在每个群组中找到用户隐藏的喜爱物品最后对用户实现个性化推荐。使用 K-means 模型推荐的一般步骤为：对用户进行聚类使用户归属于一个类别。用户数量为 p 物品数量为 q，用户-物品的评分矩阵为 $A(p,q)$。通过 K-means 聚类后得到 K 个类别。这样相似的

用户只需要在类内进行查找，就会很大程序上降低的计算的复杂度。也就是在 m 和 n 都特别大时造成的评分矩阵过于庞大情况下，这时也可以达到相同的作用效果。聚类算法在协同过滤中的应用是根据每个物品之间的相似度进行聚类从而生成聚类中心，接着找到距离最近的几个类别作为最近邻来进行查询，最后在查询所有物品中找到目标物品最近的物品最终得到推荐结果。

最普遍的进行相似度度量的公式为欧式距离和余弦相似度公式。

（1） 欧式距离公式为：

$$d(x_i, y_j) = \sqrt{(x_{i1} - y_{j1})^2 + (x_{i2} - y_{j2})^2 + ... + (x_{in} - y_{jn})^2} \tag{2-1}$$

其中 $x_i = (x_{i1}, x_{i2}, ..., x_{in})$ 和 $y_i = (y_{i1}, y_{i2}, ..., y_{jn})$ 分别表示 n 维数据点。距离越小则相似度越大。

（2） 余弦相似度公式为：

$$d(x, y) = \frac{x^T y}{\| x \| \cdot \| y \|} \tag{2-2}$$

其中，$\| x \|$ 和 $\| y \|$ 表示向量 x 和 y 的欧几里得范数。

因为欧式距离的度量指标受单位刻度影响所以需要先对数据进行标准化，而余弦相似度不会受指标刻度的影响。例如： (0，1)和(1，0)两个点对电子商务用户做聚类，用用消费次数和平均消费额来区分高价值用户和低价值用户。但是这个例子用余弦夹角是不恰当的，因为它会将(3，12)和(9，36)这两个用户计算成为相似用户，但显然后者的价值高得多。因为此时需要主要关注的是数值上的差异，而不是维度之间的差异。所以余弦相似度衡量的是维度间相对层面的差异，欧氏度量衡量数值上差异的绝对值。

基于聚类模型的推荐的思想是采用聚类算法将喜好相同的用户聚集在一个群组中，使不同簇中的用户的喜好尽可能最大限度的不相同，使一个簇中的用户喜好最大限度相同。不管是对用户或者物品进行聚类，基于聚类模型推荐算法的构建都是离线进行，这样会避免用户反馈的类似于推荐速度过慢这样的问题。论文[33]提出了一种基于改进的 K-means 聚类算法模型的协同过滤算法。该算法对用户和物品分别进行聚类达到了降低数据集稀疏性的问题从而提高了推荐准确度。当

用户或者物品之间没有明确的界限时，使用模糊 C 均值聚类算法。

2.3.3 基于分类模型的推荐

基于分类的推荐，一般使用的分类算法为朴素贝叶斯（Naive Bayes）、决策树（Decision Tree）、支持向量机（Support Vector Machine，SVM）等。

朴素贝叶斯(Naive Bayes)模型的思想是用贝叶斯模型来表示用户的喜好来解决概率化。算法通过计算后验概率来选择概率最大的类别进行分类。该算法的前提条件是假设所有特征之间是没有相关性的，就是相互独立且不依赖。在这种条件独立假设的前提下，算法的时间和空间复杂度都会降低。论文[33]对基于贝叶斯模型推荐研究，他们将每一个物品作为一个节点，评分值是节点的状态。基于贝叶斯模型可以对海量数据进行分布式处理和具有较高的分类准确性等优点被人们广泛使用。

对于基于支持向量机的算法模型，主要是因为 SVM 算法在处理高维稀疏数据分类时会比其他分类算法好很多。SVM 对时间序列的回归问题和分类判别的模式识别问题被广泛地在预测等领域应用。

本课题应用的是基于改进随机森林算法模型的推荐，对于随机森林算法的特征和发展过程研究会在下一章详细说明。

2.4 推荐算法的评价指标

近几年随着推荐算法的关注越来越多,很多学者提出了各种领域的推荐算法。面对种类繁多的推荐算法，应找到客观有效的评价指标去衡量这些推荐算法的效率。目前存在的推荐算法评价指标的数量也很多，但是仍有很多人对评价标准感到迷茫，因为评价好坏的因素不仅仅是局限于某一个单一特性，而是要全面考虑新颖性、覆盖率、多样性、准确度、算法的鲁棒性、用户满意度、测评（包括离线和在线）等多个角度进行全面评估。而在众多因素中，如何进行选取判断某个算法的好坏且这各个因素有些是不可以同时达到的。

在不同的背景环境下对推荐算法进行评价是困难的，原因在于：(1）不同推荐算法虽然都是为用户提供推荐，但是每种推荐算法的侧重点是不一样的，所以每个算法满足的上述的因素是不一样的；(2）数据类型不一样，这样会使得数据

的稀疏程度不用。前文提到，背景不同的算法处理不同数据的方式也不尽相同就会使不同的推荐算法在不同的数据集的处理表现不一样；(3）评价标准太多、推荐算法背景太多、推荐算法种类太多，无法准确选取哪种算法用哪种指标进行评价。

第 3 章 随机森林算法

3.1 前言

随着日常生活和诸多领域中人们对数据处理需求的提高，海量数据分类已经成为现实生活中一个常见的问题。分类算法作为机器学习的主要算法之一，是通过对已知类别的训练集进行分析,得到分类规则并用此规则来判断新数据的类别,现已被医疗生物学、统计学和机器学习等方面的学者提出。同时，近几年大数据时代的到来，传统的分类算法如 SVM、贝叶斯算法、神经网络、决策树等，在实际应用中难以解决高维数据和数量级别的数据。而决策森林在处理类似问题时会有较高的正确率及面对高维数据分类问题时的可扩展性和并行性。Fernandez-Delgado[34]等人通过在 121 种数据集上比较了 14 种决策森林归纳算法的预测效果，8 种改进随机森林算法，20 种改进更新权重抽样算法，24 种改进无更新权重抽样算法和 11 种其他的集成算法，得出结论:随机森林算法比其他分类学习算法效果好很多。目前，决策森林已在人工智能如 AlphaGo、推荐系统、图像和视频检索等中使用。

本章节基于的随机森林算法分析其他几种典型决策森林的特性及不同，进而说明了不同决策森林算法的适用范围。通过比较算法，根据使用者选取能解决问题的最适合的决策森林算法。介绍了面向大数据时，基于决策森林的并行处理数据的方法。

3.2 决策树算法

决策树算法是一种基于实例的算法，常应用于分类和预测。决策树的构建是一种自上而下的归纳过程，用样本的属性作为节点，属性的取值作为分支的树形结构。因此，每棵决策树对应着从根节点到叶节点的一组规则。下图描述决策树的构建过程：

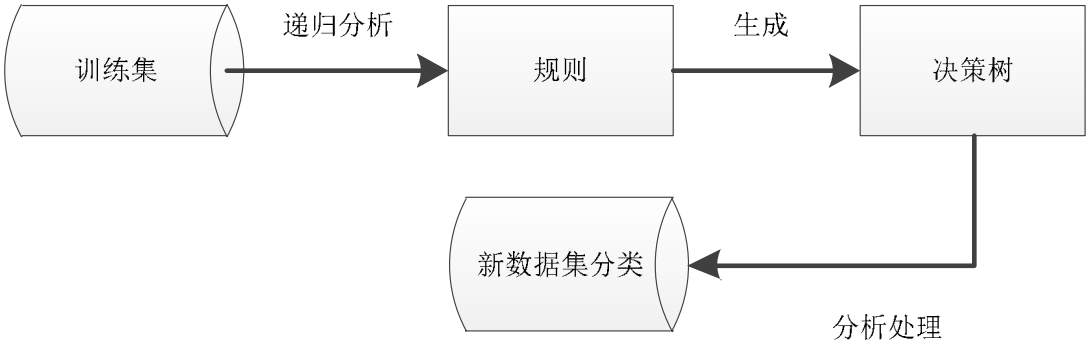

图 3.1 决策树的基本思想
Fig.3.1 The basic idea of decision tree

决策树的可视结构是一棵倒置的树状，结构中包含三种节点，分别为:叶子节点、中间结点、根节点。生成决策树的过程为：从根节点开始生成左右两个中间节点，从这两个中间节点开始又继续生产左右节点，每个中间节点继续递归生成新的中间节点直到得到叶子节点为止。

决策树的分类过程从根本上来讲为把训练集划分为越来越小的子集的过程，最理想的分类结果是每个子集中的叶子节点样本都有相同的标记。而从根节点开始，生成左右两个中间节点的过程中，需要根据不同属性分类后的结果的优劣选择最优的节点，这个过程称为节点分裂。在这个过程，使得决策树造成了局部贪婪的缺点。因为每次只选取一个属性进行分裂构造决策树，所以这些节点所产生的分类规则会特别复杂，对于此问题使用决策树的剪枝可以进行优化。决策树的优点是可以避免特征的归一化，处理大数据量时会有较强的鲁棒性和效率。一般地，决策树的经典算法为 ID3(Iterative Dichotomizer3)算法、C4.5 算法、CART(Classification and Regression Tree)算法等等。

对决策树而言，最主要问题集中在剪枝方法和训练样本数据的处理。相对而言，决策森林在提高了分类精度的同时决解了决策树所面临的问题。决策森林原理是应用集成的思想提高决策树准确率。决策森林，顾名思义由几种决策树的预测组合成一个最终的预测。通过构建森林，一棵决策树的错误可以由森林中的其他决策树弥补。最近的研究[34]证实, 从超过了 121 个数据集的 17 科学习算法的 179 个分类算法得出结论, 决策森林特别是随机森林, 比其他的学习算法分类效果好。而提高预测结果的最主要的原因是森林中每个棵树的互补性。本文主要介绍了最受欢迎的建立决策森林的方法对大数据时决策森林的处理方法。

3.3 抽样方法

抽样方法一般分为有放回抽样和无放回抽样，决策森林中应用的都是有放回抽样，故本章节对有放回详细说明。有放回抽样是一种被广泛应用的统计学习方法，其中有分为更新权重抽样（代表算法是 Bagging Adaptive boosting）和不更新权重抽样(代表算法是 Bagging) 。

决策森林的建立主要通过两种方法：（1）使用普遍的集成方法（例如 AdaBoost)。这个方法可以被应用到任何基础的学习方法，例如决策树；(2）集成方法可以用来建立决策树，例如随机森林。

3.3.1 Bootstrap aggregating

Bagging (bootstrap aggregating)无权重抽样通常用于产生全体模型，尤其是在决策森林中，是一种简单而有效的方法。所有的树都使用相同的学习算法进行训练。最后的预测结果由每棵树的预测结果投票决定。自从有放回抽样被使用，一些原始数据可能出现不止一次而有些数据没有被抽到。为了确保在每个训练实例中有足够数量的训练样本，通常会设置每个样本的原始训练集的大小。无权重抽样的一个最主要的优点是在并行模式下容易执行通过训练不同的处理程序的集成分类器。

通常情况下，bagging 无权重抽样产生一个组合模型，这个模型比使用单一原始数据的模型效果好很多。Breiman[35]指出无权重抽样决策树是随机森林算法中特定实例，而随机森林算法将会在下一节介绍。

3.3.2 Adaptive boosting

Kearns 和 Valiant[36]在 1996 年首次提出了强学习机和弱学习机的概念。他们说明在一个学习过程中，一种方法的正确率很高并且这个方法可以被多项式学习算法分析和学习，就称这个方法是一种可强学习的；相似的如果一种方法的正确率只是比随机好一点，且也是可以被多项式学习算法分析和学习则称它是弱学习的。那么将弱学习的算法训练成强学习算法就应用到了 boosting。

更新权重抽样是一种提高弱学习机性能的方法。先对这些算法进行训练得到强学习机和弱学习机，接着对弱学习机进行组合生成弱分类器，改变这些算法的

权值分布并针对不同的训练集将弱分类器提升为强分类器。这种算法通过反复地迭代不同分布的训练数据中的决策树归纳算法。这些决策树组合成为强集成森林。该算法将预测精度较低的弱学习器提升为预测精度较高的强学习器。与自举法（bootstrapping）相同的是，更新权重抽样方法同样利用重采样原理，然而在每次迭代时却不是随机的选取样本，更新权重抽样修改了样本目的是想在每个连续的迭代中提高最有用的样本。

AdaBoos[37](Adaptive Boosting)是最受欢迎的更新权值抽样算法。AdaBoost 是 Boosting 算法的一种，它能够自适应地调节训练样本权重分布，这个算法的主要思想是给在上一次迭代中错分的树有较高的权值。特别地，这些错分的树的权值越来越大而正确分类的那些树的权值越来越小。这个迭代过程生成了一连串相互补充的决策树。下面对 AdaBoost 算法详细说明。AdaBoost 算法不仅在决策森林中有重要作用，还在生物学中的 DNA 序列分析、计算学中垃圾邮件分类和机器视觉中广泛应用。下面对 AdaBoost 算法详细说明。

设输入：训练数据集 $S=\{(x_1,y_1),(x_2,y_2),...,(x_n,y_n)\}$，其中 $x_i \in \chi \subseteq R^n$，$y_i \in y=\{-1,+1\}$；弱学习算法；

输出：分类器 $F(x)$。

(1) 对训练数据的权值进行初始化；

$$W_1=(w_{11},w_{12},...,w_{1i},...,w_{1n}) \quad , \quad w_{1i}=\frac{1}{n},i=1,2,...,n; \tag{3-1}$$

(2) 对各分类算法进行训练，利用具有权重分布的训练集S_m训练出基本分类器；

(3) 计算分类器在训练集S_m上的误差率；

$$e_m=P(F_m(x_i)\neq y_i)=\sum_{i=1}^{n} w_{mi}I(F_m(x_i)\neq y_i) \tag{3-2}$$

(4) 计算分类器的系数；

$$\partial_m=\frac{1}{2}\log\frac{1-e_m}{e_m} \tag{3-3}$$

(5)更新权重；

$$S_{m+1} = (w_{m+1,1},\ w_{m+1,2},\ldots,\ w_{m+1,i},\ldots,\ w_{m+1,n}) \tag{3-4}$$

$$w_{m+1,\ n} = \frac{w_{mi}}{Z_m}\exp(-\partial_m y_i F_m(x_i)),\ \ i = 1,2,\ldots,n \tag{3-5}$$

（Z_m为规范化因子）

(6)最后的分类器是对这些分类器进行线性组合得到。

$$F(x) = sign(\sum_{i=i}^{m} \partial_i F_i(x)) \tag{3-6}$$

3.4 随机森林算法研究现状

3.4.1 随机森林

随机森林算法(Random forest)是最普通的决策森林，这个算法在二十世纪 90 年代中期被提出。后来被 Breiman[35]完善和推广。该论文截至 2016 年 4 月的谷歌学术已经被引用超过 20,800 次，而这篇论文的受欢迎程度每年都增加的主要一方面是因为随机森林算法的简单，另一方面是因为它的预测能力强。

随机森林算法中有大量的没有修剪的随机决策树，而这些决策树的输出是使用的一种无权重的多数投票。为了保证随机森林的准确性，在决策树归纳算法的建立过程中有 2 个随机的过程：

（1）从训练集中无放回的挑选样本。特别地，虽然样本都是从原始数据集中产生但每棵树的训练数据集是不同的。

（2）不是从所有的特征中选取最佳分裂点，而是随机地从特征子集中取样从中选取最佳分裂点。子集大小 n 是根据式子

$$n = \log_2 N \tag{3-7}$$

其中 N 是特征的数量，近似得到。

第二个随机过程可以有不同的实现过程，因为不是在每个节点选取最好的特征。例如使用信息增益计算，特征被随机选取的概率与它的测量值成正比。相似的算法已经被应用到随机 C4.5 决策树中。在每个阶段不是选择最好特征，而是从最好的 n 个特征集中以同样的概率随机选取。

随机森林是在树的每个节点从不同属性的子集中选择一个特征，其主要思想是替换更广泛的“随机子空间方法”，此方法可以应用于许多其他的算法例如支持向量机。尽管最近对于决策树的随机森林和随机子空间的比较已经表明在精确度方面前者要优于后者[34]。

尤其涉及到数字特征时，也存在将随机性添加到决策树归纳算法中的其他方法。例如代替使用所有的实例去决定为每个数字特性和使用实例的子样本[38]的最佳分裂点，这些子样本的特征是各不相同的。使用这些特征和分裂点评价最优化的分裂标准，而评价标准是每个节点决策选择的。由于在每个节点分裂的样本选取是不同的，这项技术结果是由不同的树组合成的集合体。另一种决策树的随机化的方法是使用直方图[39]。使用直方图一直被认为是使特征离散化的方法，然而这样对处理非常大的数据时能够减少很多时间。作为代表性的是，为每个特征创建直方图，每个直方图的边界被看作可能的分裂点。在这个过程中，随机化是在直方图的边界的一个区间内随机选取分裂点。

由于随机森林的流行，随机森林能被多种软件语言实现，例如：Python、R 语言、Matlab 和 C 语言。在此经典算法的基础上，国内外学者提出了具有较高分类准确性的算法，并被广泛应用到多种学科中。

3.4.2 极端随机森林

随机森林(Extremely randomized forest)选取最佳分裂属性，而极端随机森林[40]的最佳切割点是在随机的特征子集中。比较起来，极端随机森林的随机性既在分裂的特征中又在它相应的切割点中。为了选取分裂特征，该算法随机性表现为在被判断为最好的特征中选取确定数目的特征。除了数字特征以外，该算法还在特征值域中统一地绘制随机切点，即切点选取那些完全随机独立的目标属性。在极端情况下，此算法在每个节点上随机选取单属性点和切割点，因此一棵完全随机化树建立完成。

Geurts 等人[40]指出独立极端随机树往往有高偏差和方差的元素，然而这些高方差元素能通过组合森林中大量的树来相互抵消。

3.4.3 概率校正随机森林

一般地，决策树中的每个节点的分裂标准是使用熵或者基尼指数进行判断，

这些判断标准描述了相应的节点分裂而没有考虑节点的特性。概率校正随机森林[41] (Calibrated probabilities for random forest)算法通过引入一个考虑分裂可能性的第二条件，来提出一个提高随机森林分类算法的分裂标准。提出的方法被直接应用到离线学习的过程，因此分类阶段保留了快速计算决策树特征属性取值的算法特征。

为了得到一个有识别力的可靠的分裂指标，通过使用普拉特缩放(Platt Scaling)方法将 Sigmoid 函数引入特征空间。因此，选取最佳分裂点不再只根据单一的标准，而是一种可靠的必须满足更新最好分裂点的指标。此外，这个指标是可以把随机森林分类器更好的应用到不同阈值的任务和数据集中。

该算法是用交通标志识别的 GTSRB 数据集、手写数字识别的 MNIST 数据集和著名机器学习数据集（美国邮政总局数据集、信件数据集和 g50c 数据集）进行评估。研究结果表明，我们提出的方法优于标准随机森林分类器，尤其适用少量数目的树。下图对概率校正随机森林的构建过程进行说明。

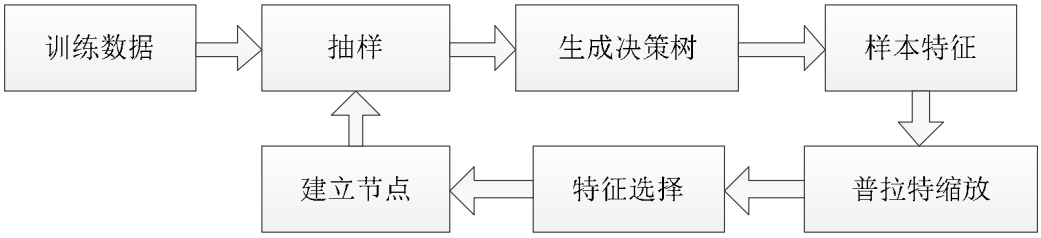

图 3.2 概率校正随机森林基本流程
Fig.3.2 The process of Calibrated probabilities for random forest

3.4.4 梯度提升决策树

梯度提升决策树(Gradient boosted decision trees，GBDT)[42]是更新权重抽样算法的一种，最初用来解决回归任务。与其他更新权重算法相同的是该算法计算一系列的回归树，但却以阶段式方法构建森林。特别地，该算法计算一系列的回归树，而回归树中的每一棵后续树的主要目的是使预测伪残差表现好的树有任意可微的损失函数。树中的每一片叶子在相应区间最小化损失函数。通过使用适当的损失函数，传统的更新权重抽样的决策树可也进行分类任务。

为了避免过拟合，在梯度更新权重抽样森林中选择适当数量的树（也就是迭代的次数）是非常重要的。迭代次数设置的过高会导致过拟合而设置的过低会导致欠拟合。选取最佳值的方法是尝试在不同的数据集中比较不同森林大小的效果。

通过使用随机梯度更新权重抽样方法可以避免过拟合。大体的思路是分别从训练集中选取随机样本并连续地训练树。由于森林中的每棵树是使用不同的样本子集所建立的，所以造成过拟合的概率将会降低。

3.4.5 旋转森林

旋转森林(Rotating forest)[43]在 3.1 节的基础上进行改进，添加了数据轴的一种算法。旋转森林中树的多样性是通过训练整个数据集中旋转特征空间的每棵树得到。在运行树归纳算法之前旋转数据轴将会建立完全不同的分类树。除此之外在确保树的多样性同时，被旋转的树能降低对单变量树的约束，这些单变量树能够分解输入空间到平行于原始特征轴的超平面。

首先随机分离特征集到 K 个相互独立的区间，之后分别在每个特征区间使用主成分分析法[44]（Principal Component Analysis，PCA）。PCA 算法的思想是正交变换任何可能相关的特征到一个线性无关的特征集中。每个元素是原始数据线性组合。且要保证第一主要元素具有最大方差。其他的元素与原来的元素正交的条件下也具有较高方差。

原始的数据集被线性转变为新的训练集，这些主要元素构建一个新的特征集。新的训练集是由新的特征空间所构建的。新的训练集被应用到训练分类树的树归纳算法中。注意的是不同的特征区间将会导致不同的变换特征集，因此建立了不同的分类树。这个旋转森林算法已经被应用到 Matlab 编码的 Weka 工具。

通过对旋转森林的实验研究发现旋转森林要比普通的随机森林算法精度高。然而旋转森林有两个缺点。第一，由于使用 PCA 算法旋转随机森林比普通随机森林计算复杂度高。另一个缺点是在新建立的树中节点是变换后的特征而不是原始特征。这令用户更难理解树，因为树中的每个节点不是审查单一特征，用户需要审查的是树中每个节点上特征的一个线性组合。

3.4.6 其他基于随机森林算法的改进算法

Novi Quadrianto 和 Zoubin Ghahramani[45]提议利用从训练集中随机选取的几个树的平均值进行预测。从一个先验分布中随机选取决策树，对这些选取的树进行加权产生一个加权集合。 与其他的基于贝叶斯模型的决策树不一样的是，需要用马尔可夫链蒙特卡罗算法对数据集进行预处理。这个算法的框架利用数据中相

互独立的树的先验性促进线下决策树的生成。该算法的先验性在查看整体的数据之前从决策树的集合中抽样。此外对于使用幂的可能性，这种算法通过集合的决策树能够计算距离间隔。在无限大的数据的限制下给每一棵独立的决策树赋予一个权值，这与基于贝叶斯的决策树形成对比。

Breiman[46]提出的一种决策森林。该算法中的每棵决策树使用带有随机分类标签的原始训练集。每个训练样本的类标签是根据过渡矩阵改变的。过渡矩阵确定了*i*类被 *j* 类替代的概率。被选择的改变概率是为了保持原始训练集的类分布。

Martínez-Muñoz 和 Suárez[47]指出当森林是非常大的时候改变类的方法能使结果特别精确，而使用多类转变建立的森林是不需要保持原始类的分布的。在不平衡数据集中，原始类分布松弛约束对于使用转变类方法是非常重要的。每次迭代中原始数据集中随机选取一个固定的部分，这些选定实例的类是随机切换的。

Saffari 等人[48]发明了一种在线随机森林算法，这个算法具有处理训练那些按照顺序到达或者具有潜在地连续地改变分布数据的能力。特别地，他们为自适应算法丢弃的树增加了一个暂时的权重方式，这些树根据他们的在给定的时间间隔内的袋外误差决定，之后再产生新的树。

Désir 等人[49]提出一个扩展的随机森林（One Class Random Forests ，OCRF）为解决单分类的分类任务。单分类是一个二项分类任务是专门针对于只有一个类的可供学习的样本。在 OCRF 中，该程序产生人工异常值嵌入到初始的随机森林算法中。这个过程利用两个之前提到过的随机化过程。随机化过程已经被应用到随机森林中建立一个相对小的人工异常值的子集中。把原始的例子作为正值人工的例子作为负值，故现在面临的就是可以用随机森林决策的一个标准的二项分类问题。通过检查随机森林的额外的变化，他们在相关系数的基础上提出了随机森林的特定分类方法，并对比了现在的随机森林的方法。

3.5 算法比较

Dietterich[50]针对构建 C4.5 决策森林已经比较了 3 种算法，分别是随机抽样、无权重抽样和更新权重抽样。实验表明当数据中有少量噪声时，更新权重抽样预测效果最好。无权重抽样和随机抽样有相同的效果。

另一个文献比较了以更新权重抽样为基础的决策树和以无更新权重为基础的决策树[16]。研究表明无更新权重抽样减少了非稳态法样本的方差，而更新权重抽

样方法减小了非稳态法样本的方差和偏差但增加了稳态法样本的方差。

Villalba Santiago 等人[51]为决策森林中建立决策树的根节点对比了七种不同的更新权值抽样算法。他们得出结论，对于二项分类任务来说，大家众所周知的 AdaBoost 算法（通过迭代弱分类器而产生最终的强分类器的算法)的效果更好。然而对于多分类任务来说如 GentleAdaBoost 算法效果更好。

Banfield[52]等人用实验评估无更新权重抽样和其他七种以随机化为基础的决策森林的算法。根据统计测试从 57 个公开的数据集获得实验结果。统计显著性用交叉验证进行对比，得出 57 个数据集中只有 8 个比无更新权重抽样精确，或在整组数据集上检查算法的平均等级。Banfield 等人总结出在更新权重抽样算法的随机森林中，树的数量是 1000 棵时效果最好。

除了预测效果也有其他的标准。根据使用者选取能解决问题的、最适合的决策森林算法：

1）处理数据时适当的对算法进行设置：在处理具体的学习情况时，不同的决策森林方法有不同的适用范围。例如不平衡的高维的多元的分类情况和噪声数据集。使用者首先需要的是描述学习任务的特征并相应地选择算法。

2）计算复杂度：生成决策森林的复杂成本以及实时性，并且对新数据预测的时间要求。通常梯度更新权重抽样的迭代法会有较高的计算效率。

3）可扩展性：决策森林算法对大数据有缩放的能力。因此，随机森林和梯度更新权值抽样树有较好的可扩展性。

4）软件的有效性：现成的软件数据包的数量。这些数据包能提供决策森林的实现方法，高度的有效性意味着使用者可以从一个软件移动到另一个软件，不需要更换决策森林算法。

5）可用性：提供一组控制参数，这些参数是广泛性且易调节的。

3.6 随机森林算法应用

随着通信信息系统收集到的数据数量的增长，这些大规模数据集使得决策森林算法要提高其预测标准。然而对于任何数据学家，这些大规模数据的有效性是至关重要的，因为这对学习算法的时间和存储器提出了挑战。大数据是近几年被创造的专业术语，指的是使用现有算法难以处理的巨量资料集。对于中小型数据集，决策树归纳算法计算复杂度是相对较低的。然而在大数据上训练密集森林仍

有困难。可扩展性指的是算法训练大数量数据能力的效率。

树归纳算法的贪婪天性对大规模数据测量效果不是很好。例如找在根节点上最佳分类点，需要遍历数据集中所有数据。当整个数据集不适合主内存，这个全盘扫面就会变成一种主要的问题。在大数据时代之前，可扩展性主要着力于缓和内存约束。被提出的几种决策树算法，这些算法都不需要将整个训练数据加载到主内存中。

近几年来，可扩展性主要集中在像 MapReduce 和 MPI 的并行技术中。MapReduce 是数据挖掘技术中最普遍的并行编程框架算法之一，由谷歌开创并推广的开源 Apache Hadoop 项目。Map 把一组键值对映射成一组新的键值对，处理键值对来生成过度键值对。指定并行 Reduce 函数，确保所有映射的键值对有相同的键组。对于其他的并行编程架构（例如 CUDA 和 MPI），MapReduce 已经成为产业标准。已经应用于云计算服务，如亚马逊的 EC2 和各类型公司的 Cloudera 服务，它所提供的服务能缓解 Hadoop 压力。

SMRF[53]是一种基于随机森林算法改进的、可伸缩的、减少模型映射的算法。这种算法使得数据在计算机集群或云计算的环境下，能优化多个参与计算数据的子集节点。SMRF 算法是在基于 MapReduce 的随机森林算法模型基础上进行改进。SMRF 在传统的随机森林相同准确率的基础上，能处理分布计算环境来设置树规模的大小。因此 MRF 比传统的随机森林算法更适合处理大规模数据。

PLANET[54]是应用于 MapReduce 框架的决策森林算法。PLANET 的基本思想是反复地生成决策树，一次一层直到数据区间足够小并能够适合主内存，剩下的子树可以在单个机器上局部地生长。对于较高层次，PLANET 的主要思想是分裂方法。在一个不需要整个数据集的特定节点，需要一个紧凑的充分统计数据结构。这些数据结构在大多数情况下可以适合内存。

Palit 和 Reddy[55]利用 MapReduce 框架开发出两种并行更新权重抽样算法：AdaBoost.PL 和 LogitBoost.PL。根据预测结果，这两种算法与他们相应的算法效果差不多。这些算法只需要一次循环 MapReduce 算法，在他们自己的数据子集上的每个映射分别运行 AdaBoost 算法以产生弱集合模型。之后这些基本的模型随着他们权重的减小被排序和传递，被减小的平均权值推导出整体最后的权值。

Del Río 等人[56]提出用 MapReduce 来实现各种各样的常规算法。这些算法用

随机森林来处理不平衡的分类任务。结果表明，多数情况下映射数量的增加会使执行时间减少，而太多的映射会导致更糟糕的结果。

各种分布的随机森林是可以实现的。尤其在 Mahout 中，这只是一个 Apache 项目。Apache 项目是可以提供免费的可扩展的机器学习算法程序包，包括在 Hadoop 框架下实现随机森林的包。MLlib 一个分布式机器学习框架，提供了在 Spark 框架下实现随机森林和梯度提升树的包。分布式版本：第一个被应用到 MapReduce 框架中，第二个应用到 MPI 框架中。为了得到精确的结果，要求在聚类中所有的节点是所有的被发现的潜在分类点。Palit and Reddy[57]利用 MapReduce 框架开发出两种并行更新权重抽样算法：AdaBoost.PL 和 LogitBoost.PL。根据预测结果，这两种算法与他们相应的连续版本的算法效果差不多。这些算法只需要一次循环 MapReduce 算法。在他们自己的数据子集上的每个映射分别运行 AdaBoost 算法以产生弱集合模型。之后，这些基本的模型随着他们权重的减小被排序和传递，利用被减小的平均权值推导出整体最后的权值。

3.7 本章小结

决策森林主要目的通过训练多个决策树来改善单一决策树的预测性能。当前决策森林的研究趋势是：解决大数据而实现分布式开发；改进现有的分类和回归的决策森林算法来处理各种各样的任务和数据集。

目前国内对于决策森林的研究是很多针对随机森林的，但却对决策森林的其他算法研究的比较少。

下图展示了本章节的各种决策森林算法是基于随机森林算法在哪些方面进行改进的。

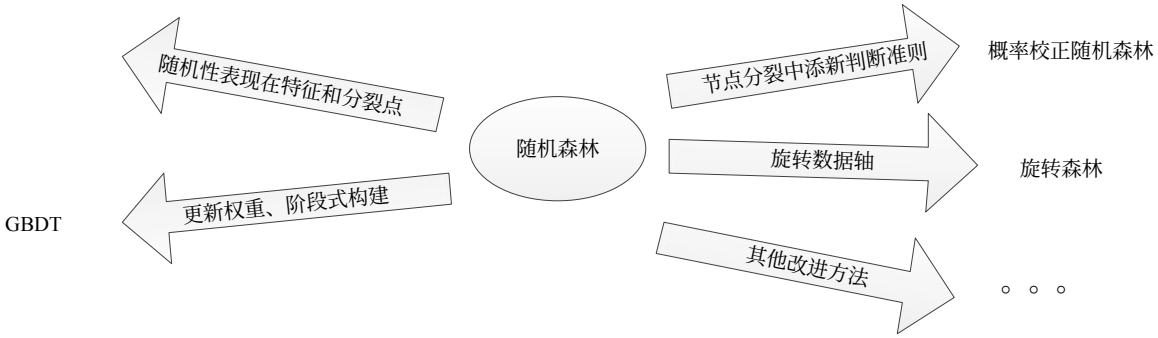

图 3.3 决策森林算法改进过程
Fig.3.3 The improved process of decision forest algorithm

第 4 章 基于 SVM 的随机森林算法

4.1 前言

对于大数据中的优化计算的处理，数据挖掘是一种很重要的方式。随机森林[35]作为数据挖掘中的比较典型的算法，它因为其较高的分类准确度和高效的非平衡数据处理被广泛应用。但是为了能更加精准和高效的计算、处理大数据，提高随机森林的准确率成为一项非常重要的研究方向。

本课题以主要讨论面向大数据集的推荐算法的研究背景，结合基于改进的随机森林模型的推荐算法。对随机森林改进思想是从决策树的建立过程中出发。决策树的每个分支点为单属性的划分，然而在多数情况下各属性是相关联的，所以建立多属性组合划分，以提高其决策效果。结合 SVM 原理[58]，对决策树中的单属性进行组合产生新的特征属性，在决策树的建造过程中实现了对随机森林算法的改进 ;在推荐过程中, 在足够的历史行为数据的前提下对用户、物品进行分析，提取数据特征然后再通过随机森林算法建立分类模型最终获得高效的推荐结果。

本章节主要对改进的随机森林算法进行详细说明。结合第三章已经对随机森林算法进行详细介绍，可知随机森林[35]算法的基本思想为：从训练集中无放回的挑选 N 个训练子集样本。特别地，每个样本子集都是从原始数据集中随机选出，故每棵树的训练数据集是不同的。每个训练子集生成一棵决策树（故生成了 N 棵决策树）。这 N 棵决策树构成随机森林。所有的决策树都使用相同的学习算法进行训练。最后的预测结果由每棵树的预测结果投票决定。在本课题基于 SVM 的随机森林中，将 CART 决策树作为基本的弱分类器。

随机森林算法的基本过程如下图：

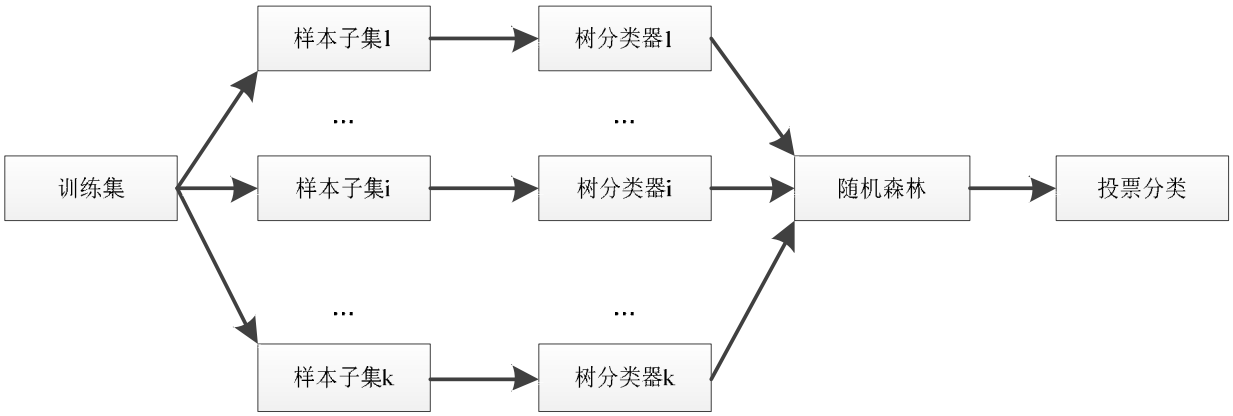

图 4.1 随机森林算法的基本过程

Fig.4.1 The process of random forest

4.2 改进随机森林算法实现

4.2.1 算法介绍

算法的内容主要体现在以下几个方面：

(1)基于 SVM 的决策树模型。随机森林中基本弱分类器是决策树，而决策树在进行节点分裂是选择分类能力最强的某个属性。本文在决策树的属性选择中结合支持向量机算法, 以特征变量的线性组合（支持向量）构成的超平面进行分裂, 比单一属性的分类能力更强且使每次分类时不需要多次迭代就可以达到叶子状态。通过 SVM 算法和决策树算法的融合建立了基于 SVM 的决策树模型。

SVM 的工作原理[59]是找到一个满足分类要求的最优分类超平面，使得该超平面在保证分类精度的同时，能够使超平面两侧的空白区域的集合距离最大化。理论上，SVM 算法可以对线性可分数据实现最优分类。

例如，给定训练样本集 $(x_i, y_i), i=1,2,...,l, x \in R^n, y \in \{\pm 1\}$，超平面记作 $(w \cdot x)+b=0$。为使分类面对所有样本正确分类并且具备最大距离间隔，就必须满足如下约束:

$$y_i[(w \cdot x)+b] \geq 1, i=1,2,...,l \tag{4-1}$$

可得出分类几何距离为 $\frac{2}{\|w\|}$，因此构造最优超平面的问题就转化为在约束条件下求：

$$\min\phi(w)=\frac{1}{2}\|w\|^2=\frac{1}{2}(w'\cdot w) \tag{4-2}$$

为了解决该个约束最优化问题，引入 Lagrange 函数：

$$L\ (\mathrm{w},b,a)=\frac{1}{2}\|w\|-a\{y[(w\cdot x)+b]-1\} \tag{4-3}$$

式中，$a_i>0$ 为 Lagrange 乘数。约束最优化问题的解由 Lagrange 函数决定，并且最优化问题的解在点处满足对w 和 b的偏导，将该 QP 问题转化为相应的对偶问题，即：

$$\begin{aligned}&\max \quad Q(a)=\sum_{j=1}^{l}a_j-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}a_ia_jy_iy_j(x_i\cdot x_j)\\&s.t. \quad \sum_{j=1}^{1}a_jy_j=0(j=1,2,...,l,a_i\geq 0)\end{aligned} \tag{4-4}$$

解得最优解 $a^*=(a_1^*,a_2^*,...,a_l^*)^T$。

计算最优权值向量 w^* 和最优偏置 b^*，分别为：

$$w^*=\sum_{j=1}^{l}a_j^*y_jx_j \tag{4-5}$$

$$b^*=y_j-\sum_{j=1}^{l}y_ja_j^*(x_i\cdot x_j) \tag{4-6}$$

因此得到最优分类超平面 $(w^*\cdot x)+b^*=0$，而最优分类函数为：

$$f(x)=\mathrm{sgn}[(w^*\cdot x)+b^*]=\mathrm{sgn}(\sum_{j=1}^{l}a_j^*y_jx_j+b^*),x\in R^n \tag{4-7}$$

对于线性不可分情况，SVM 的主要思想是将输人向量映射到一个高维的特征向量空间，并在该特征空间中构造最优分类超平面。

将x做从输入空间R^n到特征空间H的变换Φ，得：

$$x\rightarrow\phi(x)=(\phi_1(x),\phi_2(x),...,\phi_l(x))^T \tag{4-8}$$

以特征向量Φ(x)代替输入向量x，则可以得到最优分类函数为：

$$f(x)=\mathrm{sgn}(w^*\cdot\phi(x)+b^*)=\mathrm{sgn}(\sum_{i=1}^{l}a_jy_i\phi(x_i)\cdot\phi(x)+b) \tag{4-9}$$

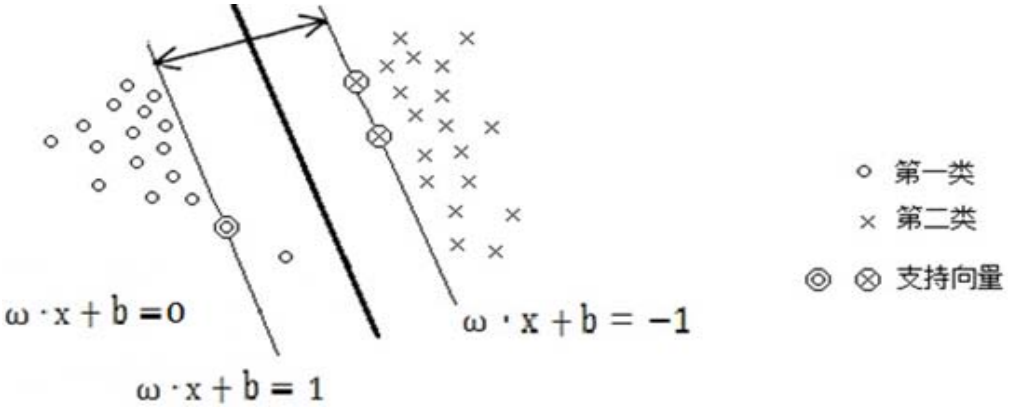

图 4.2 线性的最优超平面$(w\cdot x)+b=0$

Fig.4.2 The linear optimal hyperplane $(w\cdot x)+b=0$

（3）随机森林构建。每一棵决策树都对应一个训练集，要构建 N 棵决策树，那就需要产生对应数量的训练集，从原始训练集中产生 N 个训练子集就涉及到统计抽样技术，主要有 Boosting 和 Bagging。

（4）使用 python 实现模型及证明算法的有效可行性。

对于随机森林算法，很多学者都会用一个比较生动形象的例子来说明。每棵不同特征属性建立的决策树可以看成是不同领域的专家，整个随机森林中就具有了很多个专家，每个专家擅长的领域不同故会有结果的好坏而通过抽样提升的过程会令结果的准确度大大提升。本课题的改进过程，可以理解为每个专家学习的过程会比不改进的专家判断范围更加专攻该领域。

4.2.2 算法步骤

改进随机森林算法的算法描述：

（1）将数据分为训练集和测试集，随机分成 m 个训练子集；

（2）从样本属性中随机选取 p 个属性$(t_1,t_2,...,t_p)$作为决策树的待分裂节点；

（3）将这 m 个属性随机划分 q 个子集，在每个子集中用 SVM 进行操作，计算出w_i和b_i；

（4）用 GINI 指标或信息增益等方法计算 q 个点，选择最优的新特征作为该节点的分裂点；

（5）递归构建每个节点，使决策树完全生长且不需要剪枝；

（6）重复过程 1-5，生成改进的 SVM-随机森林；

（7）通过投票的方式对样本进行分类。

因为随机森林建立过程中的中 2 个随机选取的过程，保证了随机森林算法结果不会出现过拟合的现象。

基于 SVM-RF 模型的构建思路为：

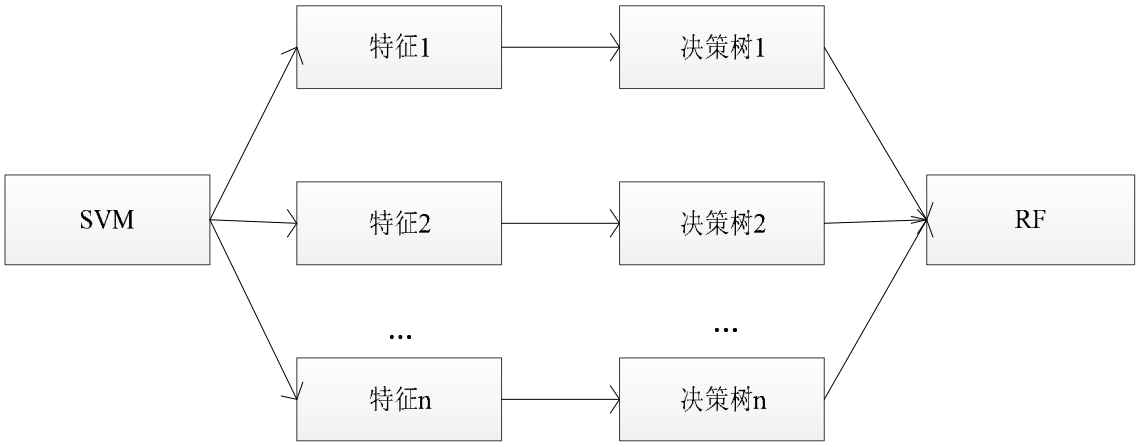

图 4.3 改进随机森林模型建立过程

Fig.4.3 The process of improved random forest

4.2.3 算法对比

针对本课题是对随机森林算法的改进，故先在 10000 个数据集中对随机森林算法和改进算法进行比较,针对森林中弱学习机 CRAT 的数量的不同对比正确率。

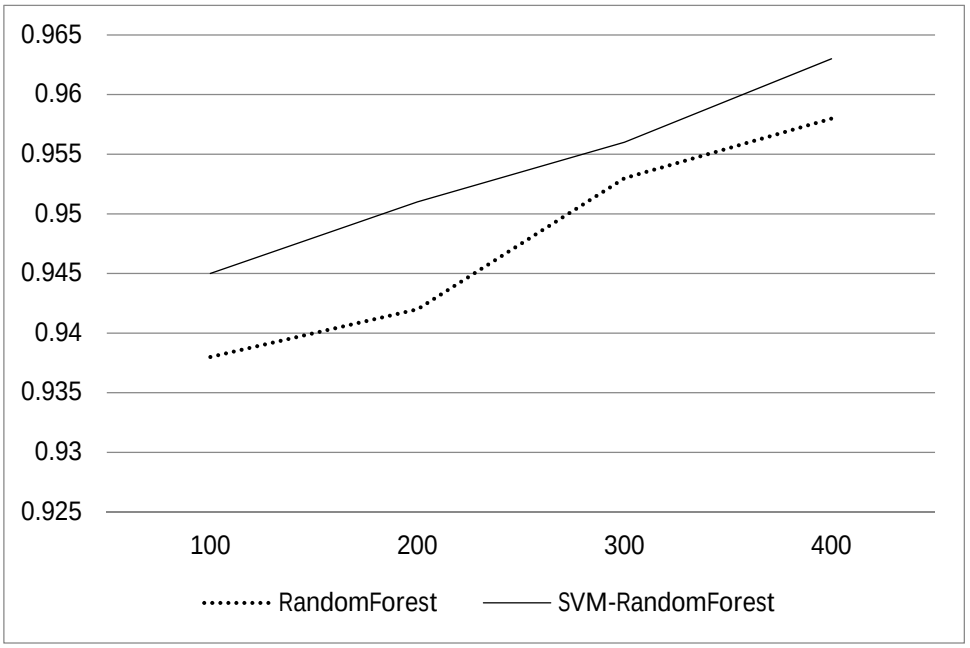

图 4.4 10000 数据集上 RF 和 SVM-RF 准确率对比

Fig.4.4 Accuracy comparison on RF and SVM – RF in 10000 data sets

接着 SVM-RandomForest 对比 Adaboost 算法， RandomForest 算法，DecisionTree 算法，算法中使用的都是分类回归树（Classification And Regression Tree，CART）。在 10000 个随机产生的数据集中，针对个算法的决策树的数量的不同来对比的各算法的错误率。由下图可知, 当随机森林中决策树的数量较小时，算法差距不是很大，但当决策树的数量增加时，本课题提出的改进随机森林算法的正确性比其他算法高的多。

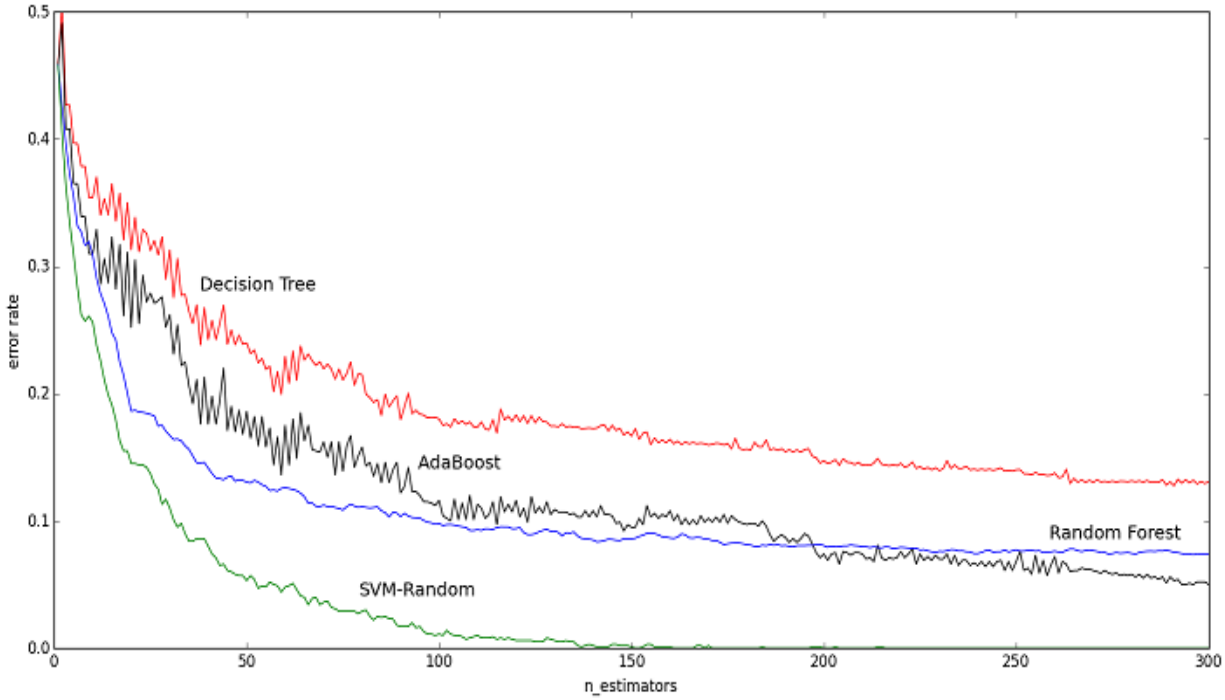

图 4.5 算法误差率对比

Fig.4.5 The comparison of algorithms' error

上述算法的对比的数据集都是用的随机产生的，故本课题又对比了 UCI 数据集中的数据，在很多种的分类算法中都使用这个数据集验证准确率，比较具有代表功能故本课题使用它进行验证。所使用的数据信息如下表：

表 4.1 数据信息

Table 4.1 Data information

name	size	Number of Attributes	class
Iris	150	4	3
wine	178	13	3

对各算法的多次实验的平均正确率进行比较如下图。

表 4.2 随机森林与其他算法正确率对比

Table 4.2 Random forests compare with other algorithms on accuracy

name	iris	wine
decisionTree	0.96038	0.89368
randomForest	0.95384	0.97778
adaboost	0.96732	0.87146
ExtraTreesClassifier	0.96078	0.95556
SVM-randomForest	0.96732	0.98315

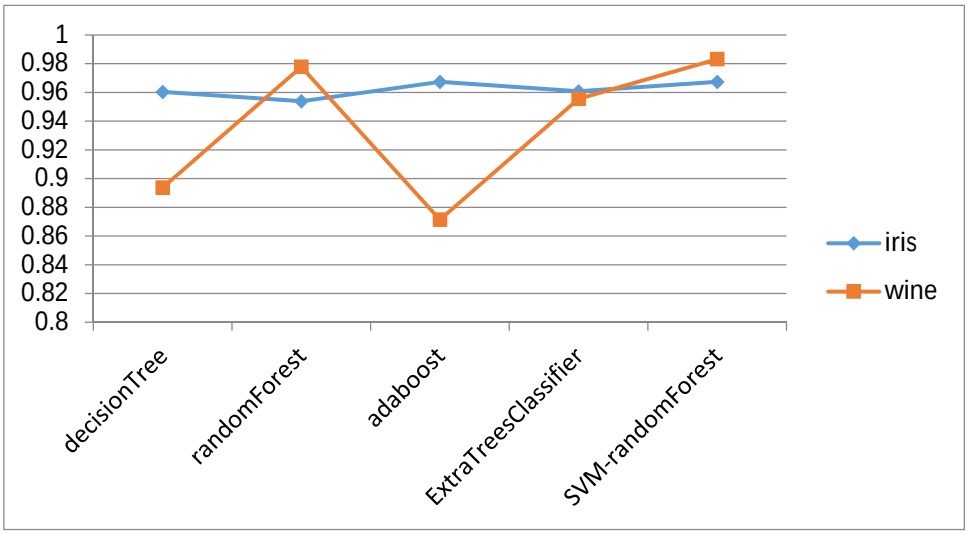

图 4.6 算法的正确率比较

Fig.4.6 Algorithm's compare on correct

最后进行的是袋外误差比较。在随机森林的 bootstrap 的抽样时，研究学者发现在原始数据集中每次大约有 1/3 的样本不会被抽取到, Breiman 将这些没有被抽取到的样本所组成的集合称为袋外误差样本（out-of-bag）。Breiman 通过多次实验指出可以用袋外误差来衡量类估计密度，它是可以代替测试集的误差估计算法，是与同训练集一样大小的测试集得到的精度一样的误差估计方法。

袋外误差估计的计算过程可以描述为[60]：在已经建立好的随机森林算法中，先假袋外数据量是 N，然后用已经生成的随机森林算法将这些袋外数据进行分类并得到正确分类的数量 n，得到袋外数据的正确率为 n/N。多次实验已经表明，袋外数据的测试结果是无偏估计的，是不用使用测试集来测试它的正确率或进行交叉验证来计算它的无偏估计。所以在随机森林算法中不需要再进行单独的测试集来获取测试集误差的无偏估计。

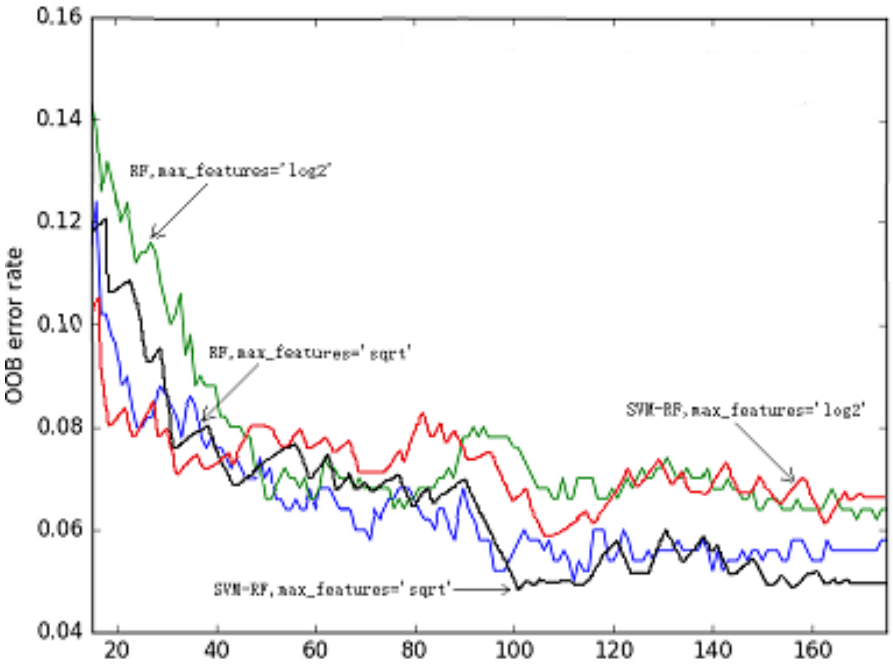

图 4.7 随机森林和改进随机森林袋外误差比较

Fig.4.7 Random forests compared with improved random forest on out-of-bag error

4.3 本章小结

本章节主要对改进的随机森林算法进行详细讲解，即利用 SVM 算法的特征组合选取节点，对随机森林中决策树的建立进行改进。

先是对改进的随机森林算法 SVM-RF 和随机森林算法 RF 在 1000 个随机产生的数据集中，通过决策树数量的不同进行正确率的比较。实验结果表明 SVM-RF 算法较 RF 算法正确率高且在袋外误差方面改进算法具有较小误差。接着用 SVM-RF 算法在误差率方面对比 Adaboost 算法、RF 算法和 DecisionTree 算法，再一次验证了改进算法有较好的正确率。由于上述对比使用的数据是随机产生，实验再一次通过 UCI 数据对比正确率，结果仍是 SVM-RF 算法具有较高的正确率。

第 5 章 基于随机森林推荐算法实现及分析

5.1 前言

随着互联网迅猛发展，电子商务受到人们越来越多的关注，其平台中用户的数量和物品数量激增导致了用户-物品矩阵稀疏。这使得利用相似度进行推荐的算法推荐质量和效率降低，而基于模型的推荐算法却不受此影响。常用的基于模型的推荐算法为基于逻辑回归和基于决策树算法。本文使用随机森林算法创建模型，并为用户购买商品做出预测。由于随机森林算法结合了多个决策树的预测，因此可以提高决策树的预测精度。实验结果表明，该算法有效的提高了推荐算法的准确率。

目前，个性化推荐算法主要有基于内容推荐、协同过滤推荐、基于统计推荐、基于知识推荐和混合推荐算法[61]。其中协同过滤算法[62]的应用最为广泛。在协同过滤算法中，以使用者为基础（User-based）的协同过滤和以项目为基础（Item-based）的协同过滤统称为以内存为基础（Memory based）的协同过滤技术。基于内存的算法精度性能好，但是他们不能处理可伸缩性和的数据稀疏问题。然而在电子商务系统中，用户和物品的数目非常巨大且还在不断增加。这会使得用户对物品评分项的减少，从而导致用户-物品评分矩阵数据稀疏性非常严重而降低了推荐精度。在这样的时代背景下使得以内存为基础的协同过滤技术推荐精度和效率越来越差，因此发展出以模型为基础的协同过滤[63]技术。以模型为基础的协同过滤（Model-based Collaborative Filtering）广泛使用的技术包括基于逻辑回归模型等。该算法是先用历史资料得到一个模型，再用此模型进行预测。基于模型的推荐算法通常比基于内存的推荐算法的速度快，即在线应用时具有更好的实时性。

而基于模型的推荐算法[64]很受欢迎的主要是因为用户-物品模型的信息突出数据并提供一个直观建议。是根据用户的历史行为数据创建一个模型，每次预测时只需将建造好的模型和新的数据导入内存中，而不是每次都调用整个数据库，提高了速度与系统伸缩性。

5.2 随机森林简介

众所周知在处理大规模数据时，随机森林[35]算法不论在分类精度和在对缺失数据和非平衡数据的处理上，都要优于决策树算法。故本文使用基于随机森林模型的推荐算法，并在随机森林的基础上进行改进以生成更好的推荐结果。

随机森林算法主要概述为：从训练集中无放回的挑选 N 个训练子集样本。特别地，每个样本子集都是从原始数据集中随机选出，故每棵树的训练数据集是不同的。每个训练子集生成一棵决策树（故生成了 N 棵决策树）。这 N 棵决策树构成随机森林。所有的决策树都使用相同的学习算法进行训练。最后的预测结果由每棵树的预测结果投票决定。

随机森林在大数据时代中的优势即可显现出，不论是基于 users 还是基于 items, 随机森林处理速度快、精确度高,保证了在推荐算法中的实时性和高效性。故本课题使用随机森林算法模型，但是对于传统的随机森林算法的基础上，本课题在应用随机森林时，对于之前构建随机森林中决策树的方法进行改进。即:结合 SVM 原理，对决策树中的单属性进行组合产生新的特征属性，解决了各个属性之间的相关性问题，以提高其决策效果。

改进随机森林算法建立一个树的递归分区训练数据。建立一个随机决策树的过程中,用户的行为数据信息作为特征属性和是否有购买行为作为决策属性。特征属性为{商品标识、浏览、收藏、加购物车、购买、空间标识、商品分类标识、行为时间}。对于决策属性提取过程和预测过程下节将详细说明。

在随机森林中，特征选择一般用 Gini 增益、信息增益和增益比率等，本文主要说明 Gini 增益[65]公式如下：

索引值定义为：

$$Gini = 1 - \sum_{k=1}^{C} P_k^{\,2} \tag{5-1}$$

其中，若按类别 A 进行分支选择，C 表示数据集 S 中类别 A 的个数，P_k 表示数据集中每个特征属于类别 A 的概率。

Gini 指数为：

$$Gini\_Gain = \frac{|S_1|}{|S|}Gini(S_1) + \frac{|S_2|}{|S|}Gini(S_2) \quad (5\text{-}2)$$

若选择属性 m(b 是按照类别 A 作为分类中其中一个属性)作为分裂点，在集合 S 中对于属性 m 有多个子集（下一个按照类别 B,C,…进行分类），通过计算，选择最小的值作为最后选取的 Gini 指数。

Gini 增益为：

$$\Delta Gini\ (m) = Gini\_Gain(S) - Gini\_Gain_m(S) \quad (5\text{-}3)$$

总体内包含的类别越杂乱，GINI 指数就越大（跟熵的概念很相似）。选择信息增益大的属性作为最佳分裂点。

5.3 算法步骤

对于用户购买预测推荐，本论文的主要思路是一个二分类问题（买或不买）。随机森林中选 CART 作为弱分类器。本课题是基于改进随机森林模型的推荐算法在 2015 年天池推荐算法决赛数据上进行分析。具体算法步骤如下图所示：

图 5.1 算法流程图

Fig.5.1 Algorithm flowchart

5.3.1 数据预处理

数据采用的是阿里巴巴在真实的业务场景下的数据。即 500 万用户的完整行为数据（为保护用户私隐，对用户的 ID 等信息采用了脱敏）以及千万级的商品信息。在真实的业务场景下，通常需要对所有商品的一个子集构建个性化推荐模型。在完成这件任务的过程中，不仅需要利用用户在这个商品子集上的行为数据，往往还需要利用更丰富的用户行为数据，即完整的特征属性为浏览、收藏、加购物车、购买、空间标识、商品分类标识、行为时间。

在数据预处理阶段，先将数据中断码数据格式转换为整数型，即将数据的行

为时间转化为 1-31 的数字；再遍历数据把异常用户删除。

表 5.1 用户在商品全集上的行为数据说明

Table 5.1 The user behavior data on the goods complete specifications

字段	字段说明	提取说明
user_id	用户标识	抽样&字段脱敏
item_id	商品标识	字段脱敏
behavior_type	用户行为类型	浏览 1、收藏 2、加购物车 3、购买 4
user_geohash	用户位置的空间标识，可以为空	经纬度生成
item_category	商品分类标识	字段脱敏
time	行为时间	精确到小时级别

接着对数据进行统计，统计如下图：

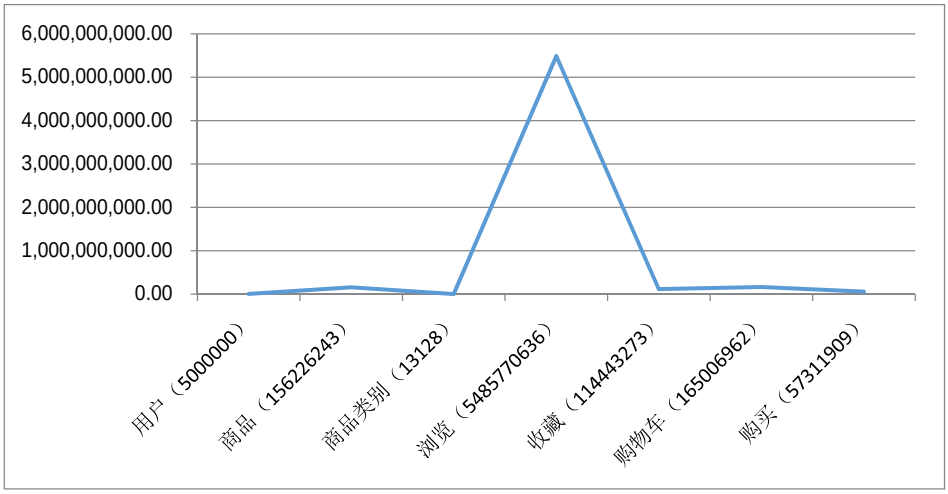

图 5.2 用户行为、商品等数据统计

Fig.5.2 Statistics on user behavior、commodity etc.

由图 5.1 可知，用户的浏览行为相比其他行为过于庞大且对于天池的规则，预测第 31 天的对商品子集购买预测，故删除没有购买行为只浏览的用户；与商品子集中没有交互的用户；浏览次数过大的用户。

5.3.2 数据特征选取

特征属性提取过程为：

1）数据集划分：第 1-7 天作为训练集，第 8 天作为标注正负样本的标注集；2-8 天作为训练集，第 9 天作为标注正负样本的标注集。依次类推，第 23-29 天作为训练集，第 30 天作为标注正负样本的标注，得到正负样本。第 31 天作为预测日并把第 31 天的真实购买数据作为测试集。

2）样本属性标注过程为（结合本文的实验数据）：从第 1-7 天抽象出来特征向量，用第 8 天所有购买记录标注特征向量（发生购买行为的 user-item 对标记为正样本，否则为负样本），构建训练集。

下表显示了特征提取后正样本的数量，共得到正样本数量 187289 个。

表 5.2 数据划分后得到的正样本数量

Table 5.2 Sample size after data partitioning

1-7 天数据量	得到 1-7 天正样本数量	2-8 天数据量	得到 2-8 天正样本数量
4601128	6868	4616452	6626
3-9 天数据量	得到 3-9 天正样本数量	4-10 天数据量	得到 4-10 天正样本数量
4608298	6785	4622959	6051
5-11 天数据量	得到 5-11 天正样本数量	6-12 天数据量	得到 6-12 天正样本数量
4647900	6366	4664561	6975
7-13 天数据量	得到 7-13 天正样本数量	8-14 天数据量	得到 8-14 天正样本数量
4698832	7023	4727193	7285
9-15 天数据量	得到 9-15 天正样本数量	10-16 天数据量	得到 10-16 天正样本数量
4782417	7562	4889277	7226
11-17 天数据量	得到 11-17 天正样本数量	12-18 天数据量	得到 12-18 天正样本数量
4945808	6050	4979681	6720
13-19 天数据量	得到 13-19 天正样本数量	14-20 天数据量	得到 14-20 天正样本数量
5028366	6512	5036871	6543
15-21 天数据量	得到 15-21 天正样本数量	16-22 天数据量	得到 16-22 天正样本数量
5042299	6475	5055842	6695
17-23 天数据量	得到 17-23 天正样本数量	18-24 天数据量	得到 18-24 天正样本数量
5058388	6771	5253060	30237
19-25 天数据量	得到 19-25 天正样本数量	20-26 天数据量	得到 20-26 天正样本数量
5886076	6965	5927768	6768
21-27 天数据量	得到 21-27 天正样本数量	22-28 天数据量	得到 22-28 天正样本数量
5940105	7369	5949356	7382
23-29 天数据量	得到 23-29 天正样本数量		
5930593	7071		

5.3.3 模型训练

对于模型的训练，在任何实验背景下都是特别重要的。在随机森林中树的个

数、深度、内部节点中选取分裂的数量和叶子节点的数量等参数的训练也十分重要，故需要对模型进行模型训练。最后使用训练后的随机森林算法模型训练上述得到的训练集，用训练好的模型预测第 31 天的 user-item 购买情况。用第 31 天的真实购买数据对模型预测结果计算准确率、召回率和 F1 值。

5.3.4 推荐算法评估标准

采用经典的精确度(precision)、召回率(recall)和 F1 值作为评估指标[66]。具体计算公式如下：

$$Precision = \frac{\left|\cap(\Pr edictionSet, \mathrm{Re}\ ferenceSet)\right|}{\left|\Pr edictionSet\right|} \tag{5-4}$$

$$\mathrm{Re}call = \frac{\left|\cap(\Pr edictionSet, \mathrm{Re}\ ferenceSet)\right|}{\left|\mathrm{Re}\ ferenceSet\right|} \tag{5-5}$$

$$F1 = \frac{2\times \Pr ecision\times \mathrm{Re}call}{\Pr ecision + \mathrm{Re}call} \tag{5-6}$$

其中 PredictionSet 为算法预测的购买数据集合，ReferenceSet 为真实的答案购买数据集合。F1 值是统计学中用来衡量二分类模型精确度的一种指标。它同时兼顾了分类模型的精确度和召回率，F1 值可以看作是模型精确度和召回率的一种加权平均，它的最大值是 1，最小值是 0。

5.4 算法实现和分析

32166145	248134056	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0
100743346	287451634	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0
126920490	156092687	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0
61419422	342349049	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	1
138679807	119882802	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
116553650	348028379	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
23067517	180110948	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
132598646	248649812	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0
41368753	129689235	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
112802035	116654304	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
141285359	309323125	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
55503089	400695358	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0
44193530	306301741	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
55793114	249886677	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

图 5.3 部分用户-商品对与商品子集交互后结果

Fig.5.3 The Partial result that user-item interacted with commodity subset

最后，用对最后模型出来的结果，与商品子集取交集得到最后第 31 天的预测。

其中的数字代表行为的操作次数。

表 5.3 处理后各属性总结表

Table 5.3 Summary of Attributes after Processing Table

名称	数目
Total Count	23291027
User Count	20000
All Item Count	4758484
Interact Item Count	422858
Category Count	1054
User Geo Count	7380017

表 5.4 部分对用户所购买商品预测表

Table 5.4 Users' forecasting purchasing merchandises partly

user_id	item_id	user_id	item_id
12979104	139170538	139731337	171312393
31284451	274834255	45561	324535120
133353113	234315653	41283289	107291417
64546497	134456912	49756956	18588382
186960	207249855	127546209	51544540
48064470	364659986	43397713	159286325
121454675	24240356	132626507	207550096
62957267	210574877	12952941	42903965
38045126	134615342	40167073	138913700
11436462	297797143	121492295	165544137
28914539	79723559	17912257	354585058
33939991	66664046	103725799	251345446
29020268	250366251	110151511	106038137
58991721	212171533	127344690	261689343
140349011	210990792	38900936	65016327
101969992	52008082	127367034	144722574
110900914	112286569	31619407	170312952
16693399	355222628	64792062	341970737
51007258	131482274	107609597	374513933
141822218	158479069	13976301	77538967
28862804	312216295	21624663	119586916
110115430	399422732	40553785	266334821
101847145	19797361	120636580	320733429
130762607	291573307	106176029	94230576

24868414	340183194	1945207	16205732
108536069	28918350	55643672	364331235
19000534	329267714	37093106	214684878
40026613	93934920	15305635	388241085
135292392	327917522	42288637	397609620
132025703	54216754	62109026	25104846
37662299	307914100	58703349	168443491
42928165	236329122	42510379	137033752
43289425	161943104	57745434	237805613
60857306	193778345	63051743	100919623
44395163	207959803	37534184	227395041
104644609	380645933	46511333	312853904
137388282	69803517	114139508	101851222
133761989	22329515	46540256	356999260
33253837	117627716	50881771	53131655
128170431	211788388	40755589	25959588
19430771	321071004	34370575	163450674
61211120	368586328	64315314	58671428
6074624	143443181	52800551	51045585
121250534	51376626	57671175	204735978
109130675	279625459	53600723	322464560
60064811	73744013	61283696	80778202
123344315	123293811	132953186	1504296
117346094	139315932	37358249	156335086
59860956	158122891	33984142	222768781
117520661	135426525	17001133	385846964

基于随机森林模型的推荐算法结果与基于逻辑回归模型的推荐算法结果对比。

表 5.5 实验结果对比（单位%）

Table 5.5 Comparison of experimental result(Unit %)

Name	改进随机森林算法	逻辑回归算法	随机森林
Precision	3.78	2.30	2.96
F1	7.16	4.40	5.46

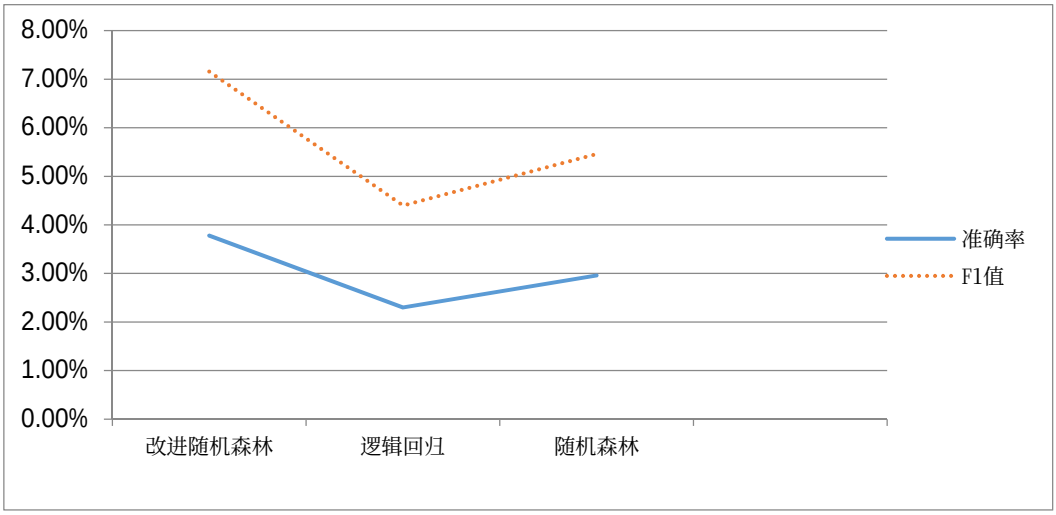

图 5.4 实验结果对比

Fig.5.4 Comparison of experimental result

5.5 本章小结

由于本实验所用的硬件设备是：Intel 奔腾 5（2.4GHz）处理器，内存 8G，Windows 8 操作系统。算法由 Python 语言编写，在 Pycharm4.0 上进行编译运行。

随机森林在大数据时代中的优势即可显现出，不论是基于 users 还是基于 items，随机森林处理速度快、精确度高，保证了在推荐算法中的实时性和高效性。本课题在应用随机森林时，对于之前构建随机森林中决策树的方法进行改进。即：结合 SVM 原理，对决策树中的单属性进行组合产生新的特征属性，解决了各个属性之间的相关性问题，以提高其决策效果。

在本文中，我们提出了一个基于改进随机森林模型的推荐算法建议。通过比较与基于逻辑回归模型和基于随即森林模型的比较，可以得出基于改进随机森林模型的推荐算法效果会好一些。实验表明，在对用户历史行为数据的前提下，可以有效地对用户未来购买商品进行预测和推荐。该算法对海量数据下电子商务个性化推荐算法性能的完善有较为重要的意义。

结　　论

随着电子商务在人们日常生活中的普遍使用，其平台中用户的数量和物品数量激增导致了用户-物品矩阵稀疏。这使得利用相似度进行推荐的算法推荐质量和效率降低，而基于模型的推荐算法却不受此影响。基于模型的推荐技术旨在建立一个模型来表示用户评级数据，并使用该模型来预测用户对特定项目的偏好。基于模型的推荐算法的步骤为：使用评价矩阵训练数据，使用用户的显式反馈学习模型或者是我们利用其他算法学习出的模型给目标用户推荐物品。使用机器学习模型（如分类模型）去辨别多种多样的用户或者物品，并以此为基础给出推荐预测结果。通常根据他们彼此间的评分使用相似性度量来获得两个用户直接的距离或两个物品之间的距离。常用的基于模型的推荐算法为基于逻辑回归和基于决策树算法，本文使用的是基于改进随机森林模型的推荐算法。本文主要完成的工作和创新点是：

（1） 提出一种 SVM 和 RF 算法融合的改进随机森林算法。

随机森林在节点分裂时，先随机地选择特征属性后通过节点分裂指标计算公式比较这些属性进而选择该分裂的节点。本论文主要是结合 SVM 算法原理对随机特征变量的产生过程中加入随机组合输入变量进行改进。本文在决策树的属性选择中结合支持向量机算法，以特征变量的线性组合（支持向量）构成的超平面进行分裂，比单一属性的分类能力更强，从而在随机森林决策树的建造过程中得到了改进。

（2）建立改进随机森林模型的推荐算法。应用提出的改进推荐算法对真实的阿里巴巴用户数据进行计算，最后得出推荐结果。

随机森林在大数据时代中的优势即可显现出，随机森林处理速度快、精确度高，保证了在推荐算法中的实时性和高效性。故本课题使用随机森林算法模型，但是对于传统的随机森林算法的基础上，本课题在应用随机森林时，对于之前构建随机森林中决策树的方法进行改进。即:结合 SVM 原理，对决策树中的单属性进

行组合产生新的特征属性，解决了各个属性之间的相关性问题，以提高其决策效果

通过实验对比和结果分析可知，本文提出的基于 SVM 算法的随机森林算法在准确率和袋外误差等方面都比未改进的随机森林算法效果好。这使得基于改进随机森林算法的推荐算法的预测结果的精准率得到了提高和改善。同时本文也对比了 decisionTree、randomForest、adaboost 和 ExtraTreesClassifier 算法，结果表明改进的随机森林算法在分类准确度都较上述算法好。

最后，本文还是存在很多的不足之处需要完善和解决：

（1）随机森林中的基本弱学习机只是使用 CART 树，没有对别其他决策树如 CH4.5 和 ID3 等；

（2）数据采用的是阿里巴巴在真实的业务场景下 500 万用户的完整的历史行为数据数据，在数据预处理过程中时间消耗较长。

这将是以后进行改善的任务。

参考文献

[1] 梁宇.移动网络个性化信息推荐技术及影响因素分析[J].电子世界,2013(6):11-11.

[2] Rathore S S, Kumar S. A decision tree logic based recommendation system to select software fault prediction techniques[J]. Computing, 2016:1-31.Liu J, He K, Wang J, et al. Service organization and recommendation using multi-granularity approach[J]. Knowledge-Based Systems, 2015, 73:181-198.

[3] 张宜浩.基于半监督学习的个性化推荐研究[D].重庆大学,2014.

[4] Wilson J, Chaudhury S, Lall B. Improving Collaborative Filtering Based Recommenders Using Topic Modelling[C]// IEEE/WIC/ACM International Joint Conferences on Web Intelligence. 2014:340-346.

[5] 郑直.VOD系统应用服务器中Web挖掘技术的研究与应用[D].北京邮电大学,2010.

[6] 萨师煊,施伯乐,童頫,等.数据库技术的发展——第12届国际VLDB会议评介[J].计算机科学,1987(2):19-26.

[7] Lu P Y, Wu X X, Teng D N. Hybrid Recommendation Algorithm for E-Commerce Website[C]// International Symposium on Computational Intelligence and Design. 2015.

[8] Oldale A, Oldale J, Reenen J V, et al. COLLABORATIVE FILTERING: US, US 20040054572 A1[P]. 2004.

[9] Zisopoulos H, Karagiannidis S, Antaris S, et al. Content-Based Recommendation Systems[J]. 2008.

[10] Zhao W, Wang J, Liu G. A Knowledge Recommendation Algorithm Based on Content Syndication[J]. Advances in Information Sciences & Service Sciences, 2012。

[11] 张腾季.个性化混合推荐算法的研究[D].浙江大学,2013.

[12] Zhang Z, Xu G, Zhang P. Research on E-Commerce Platform-Based Personalized Recommendation Algorithm[J]. Applied Computational Intelligence & Soft Computing, 2016, 2016:1-7.

[13] 周祥.基于Web的数据挖掘的研究及其应用[D].同济大学，2006.

[14] 左子叶.电子商务推荐系统与推荐过程研究[D].复旦大学，2004.

[15] 王美玲.基于加权信任关系和用户相似性融合的社会化推荐算法研究[D].山东大学,2015.

[16] 祝奇伟,陈家琪.一种改进相似度计算方法的协同过滤推荐算法[J].信息技术,2015(3):13-16.

[17] 魏欢.基于本体的影视个性化推荐算法研究[D].武汉理工大学,2013.

[18] Ricci F, Cavada D, Mirzadeh N, et al. Case-based travel recommendations.[J]. Destination Recommendation Systems Behavioural Foundations & Applications, 2006.

[19] Boutet A, Frey D, Guerraoui R, et al. Privacy-preserving distributed collaborative filtering[J]. Computing, 2016, 8593(8):1-20.

[20] 杨芳．电子商务系统协同过滤推荐算法研究[D]．河北工业大学，2006.

[21] Takács G, Pilászy I, Németh B, et al. Major components of the gravity recommendation system[J]. Acm Sigkdd Explorations Newsletter, 2007, 9(2):80-83.

[22] 詹尼士.推荐系统[M].人民邮电出版社，2013.

[23] Zhong J, Li X. Unified collaborative filtering model based on combination of latent features[J]. Expert Systems with Applications, 2010, 37(8):5666-5672.

[24] Chen T, Zhang W, Lu Q, et al. SVDFeature: a toolkit for feature-based collaborative filtering[J]. Journal of Machine Learning Research, 2012, 13(1):3619-3622.

[25] Ren X, Lü L, Liu R, et al. Avoiding congestion in recommender systems[J]. New Journal of Physics, 2014, 16(6).

[26] Burke R. Hybrid Recommender Systems: Survey and Experiments[J]. User Modeling and User-Adapted Interaction, 2002, 12(4):331-370.

[27] Breese J S, Heckerman D, Kadie C. Empirical Analysis of Predictive Algorithms for Collaborative Filtering[J]. New Page, 2015, 7(7):43--52.

[28] Töscher A, Jahrer M, Legenstein R. Improved neighborhood-based algorithms for large-scale recommender systems[C]// Kdd Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition. ACM, 2008:1-6.

[29] 吴俊伟,孙国伟,张如,等.基于SVD方法的INS传递对准的可观测性能分析[J].中国惯性技术学报,2005,13(6):26-30.

[30] Schroeder B, Harchol-Balter M, Iyengar A, et al. How to Determine a Good Multi-Programming Level for External Scheduling[C]// International Conference on Data Engineering, ICDE 2006, 3-8 April 2006, Atlanta, Ga, Usa. 2006:60.

[31] 丁雪涛. 基于协同关系主题回归模型的推荐算法研究[D].清华大学, 2013.

[32] 邓晓懿,金淳,韩庆平,等.基于情境聚类和用户评级的协同过滤推荐模型[J].系统工程理论与实践,2013,33(11):2945-2953.

[33] 冀俊忠,沙志强,刘椿年,等.贝叶斯网模型在推荐系统中的应用研究[J].计算机工程, 2005, 31(13):32-34.

[34] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems. J Mach.Learn. Res. 15 (1) (2014) 3133–3181.

[35] Breiman L. Bagging predictors[J]. Machine learning, 1996, 24(2): 123-140.

[36] Kearns M J, Valiant L G. Cryptographic limitations on learning Boolean formulae and finite automata[J]. 1996, 41(1):67-95.

[37] 钱志明,徐丹.一种 Adaboost 快速训练算法 [J].计算机工程,2009.

[38] Ho T K. Random Decision Forests[C]// Document Analysis and Recognition, 1995. Proceedings of the Third International Conference on. 1995:278-282 vol.1.

[39] Amit Y, Geman D. Randomized Inquiries About Shape: An Application to Handwritten Digit Recognition[R]. CHICAGO UNIV IL DEPT OF STATISTICS, 1994.

[40] Geurts P, Ernst D, Wehenkel L, Extremely randomized trees, Mach. Learn. 63(1) (2006) 3–42.

[41] Baumann F, Chen J, Vogt K, et al. Improved Threshold Selection by Using Calibrated Probabilities for Random Forest Classifiers[C]// Computer and Robot Vision. IEEE, 2015:155-160.

[42] Friedman J H. Greedy Function Approximation: A Gradient Boosting Machine[J]. Annals of Statistics, 2000, 29(5):1189--1232.

[43] Rodríguez J J, Kuncheva L I, Alonso C J. Rotation forest: A new classifier ensemble method.[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2006, 28(10):1619-1630.

[44] 杨开睿,孟凡荣,梁志贞.一种自适应权值的PCA算法[J].计算机工程与应用,2012,48(3):189-191.

[45] Novi Q, Zoubin G. A Very Simple Safe-Bayesian Random Forest.[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2015, 37(6):1297-1303.

[46] Breiman L. Randomizing Outputs To Increase Prediction Accuracy[J]. Machine Learning, 2000, 40(3):229-242.

[47] Martínez-Muñoz G, Suárez A. Switching class labels to generate classification ensembles[J]. Pattern Recognition, 2005, 38(10):1483-1494.

[48] Saffari A, Leistner C, Santner J, et al. On-line Random Forests[C]// IEEE, International Conference on Computer Vision Workshops. 2009:1393-1400.

[49] Désir C, Bernard S, Petitjean C, et al. One class random forests[J]. Pattern Recognition, 2013, 46(12):3490-3506.

[50] Dietterich T G. An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization[J]. Machine Learning, 2000, 40(2):139-157.

[51] Villalba S D, Rodr□guez J J, J. Alonso Carlos, An empirical comparison of boosting methods via OAIDTB, an extensible java class library, in: II International Workshop on Practical Applications of Agents and Multiagent Systems – IWPAAMS,2003.

[52] Banfield R E, Hall L O, Bowyer K W, et al. A comparison of decision tree ensemble creation techniques.[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2007, 29(1):173-180.

[53] Han J, Liu Y, Sun X. A scalable random forest algorithm based on MapReduce[C]//Software Engineering and Service Science (ICSESS), 2013 4th IEEE International Conference on. IEEE, 2013: 849-852.

[54] Panda B, Herbach J S, Basu S, et al. PLANET: massively parallel learning of tree ensembles with MapReduce[J]. Proceedings of the Vldb Endowment, 2009, 2(2):1426-1437.

[55] Palit I, Reddy C K. Scalable and Parallel Boosting with MapReduce[J]. IEEE Transactions on Knowledge & Data Engineering, 2012, 24(99):1-1.

[56] Río S D, López V, Benítez J M, et al. On the use of MapReduce for imbalanced big data using Random Forest[J]. Information Sciences, 2014, 285:112-137.

[57] Palit I, Reddy C K. Scalable and Parallel Boosting with MapReduce[J]. Knowledge & Data Engineering IEEE Transactions on, 2012, 24(10):1904-1916.

[58] 薛明东,郭立.基于SVM算法的图像分类[J].计算机工程与应用,2004,40(30):230-232.

[59] Shalev-Shwartz S, Singer Y, Srebro N, et al. Pegasos: primal estimated sub-gradient solver for SVM[J]. Mathematical Programming, 2011, 127(1):3-30.

[60] 董师师,黄哲学.随机森林理论浅析[J].集成技术，2013，2(1):3-9.

[61] Rathore S S, Kumar S. A decision tree logic based recommendation system to select software fault prediction techniques[J]. Computing, 2016:1-31.

[62] Wilson J, Chaudhury S, Lall B. Improving Collaborative Filtering Based Recommenders Using Topic Modelling[C]// IEEE/WIC/ACM International Joint Conferences on Web Intelligence. 2014:340-346.

[63] 庄永龙.基于项目特征模型的协同过滤推荐算法[J].计算机应用与软件,2009,26(5):244-246.

[64] 曹正凤.随机森林算法优化研究[D].首都经济贸易大学,2014.

[65] 韩松来,张辉,周华平.决策树的属性选取策略综述[J].网络新媒体技术,2007,28(8):785-790.

[66] Arwar B, Karypls G, Konstan J, et al. Item-based Collaborative Filtering Recommendation Algorithms [C]. In: Proceedings of the 10th International World Wide Web Conference. 2001.

攻读硕士学位期间发表的论文和获得的科研成果

[1] 黄海新， 吴迪, 文峰．决策森林研究综述[J]．电子技术应用, 2016, 42(12):5-9.

致　　谢

二年半的研究生学习阶段即将完成，在此期间感受到学术研究的博大精深，通过导师的悉心指导、同学们的相互探讨学习与自己的努力拼搏，学习到了很多终身受益的知识。

感谢我的导师黄海新教授在研究生阶段给予我学习研究方面及生活方面的指导与帮助。黄老师为人谦和、处事严谨的教学风格及提倡高效率的工作作风，在我科研项目遇到的难题时指明解决方案思想。将学术理论与实践相结合，比如从天猫推荐大赛中提出问题寻求解决方案，不断探索课题寻找创新性方案，黄老师的学术思想让我受益匪浅。在黄老师的指导下，参与了实验室多个科研项目的研究探索，让我多方面得到锻炼与提高。在导师黄老师的悉心指导下完成了本篇论文的编写，在本论文中选题、整体框架构建、论文的内容、论文修改到最终完成，黄老师都给予了极其宝贵的指导意见。

感谢实验室的邓丽师姐与张路师兄们在研究生阶段我们朝夕相处，共同探讨学术研究，相互合作完成项目研究，相互关心鼓励，感谢你们为我创造轻松愉快的学习环境。同时感谢寝室姐妹王杨、赵艳霞和闫翩翩给予生活中的关怀与帮助，是在我们的共同努力下构建良好的生活环境，共同学习成长。感谢实验室的武枫师弟在工作与生活上的支持与帮助。

感谢我的家人，你们一直给予我无微不至的关怀与照顾，是你们的爱让我变得更加坚强与勇敢。

最后，在此衷心感谢在百忙之中评审论文与参加答辩的各位专家、老师。

论文密级 _____

至 _____ 年解密

是 ____ 否____ 可以复印

如复印，从 _____ 年起可复印全文的 _____ %；

从 _____ 年起可全文复印。

封底填写说明： 横线上的内容涉及作者的知识产权，请作者与指导教师商定后认真填写。学位论文如属保密内容，需由指导教师提供申请，到校保密委员会办理手续，确定密级，并填写全部5行内容。如不涉及保密内容，只需填写第3、4、5行内容。所有涉密论文都需交校科技档案室1份留存。

秘密级的涉密论文可提交有保密措施的本院系资料室留存；机密级以上的涉密论文由课题组按有关规定保存，并决定是否提交本院系资料室留存。

保密期过后由学院送研究生部学位办三份。研究生部学位办按有关规定上交国家科技情报中心、校图书馆和留存研究生部各一份。