摘 要

粗糙集理论是一种新的刻画不完整性和不确定性的数学工具。知识约简是粗糙集理论研究的核心问题之一。目前，粗糙集理论正在被广泛应用于人工智能、模式识别等很多领域。本文对属性约简与决策树规则简化进行了深入研究：

针对不一致决策表，为克服区分矩阵方法时间复杂度随系统大小增加而指数增长的缺陷，以知识的包含度为基础，将一致与不一致对象分开，给出分布约简的数学判定定理，设计了一种求分布约简的启发式算法。实践表明该方法能够获取较小的约简。

为弥补现有信息论方法的局限性，定义了一种新的条件熵概念，并且给出了以不等式为条件的约简判定定理，提出了一种相对属性约简的启发式算法。实例分析的结果表明，该方法提高了运行效率，有助于搜索最小或次优知识约简。基于上面的思想又提出了基于决策熵的约简算法，实验结果表明该算法也能取得较好的效果。

分析了基于正区域方法的不足，提出了决策强度的代数定义，并证明了知识的决策强度随信息粒度变小而非单调递减的规律，设计了基于决策强度的约简算法。UCI 离散数据集实验比较的结果表明，该算法计算直观有效。

针对现有值约简算法提取规则仍存在冗余与计算复杂度较大等问题，引入决策树分类规则学习方法，定义了一种能反映决策能力实质的新的条件熵，对传统启发式方法中选择属性的标准进行改进，构造决策树，设计规则约简过程。该方法的优点在于构造决策树与提取规则之前不进行属性约简，也能获取简洁有效的规则。为弥补知识粗糙熵的局限性，提出决策熵概念，以条件属性子集的决策熵来度量其对决策分类的重要性，自顶向下递归构造决策树，简化规则。通过实例分析说明了该算法的有效性。

关键词：粗糙集，决策表，属性约简，规则提取，决策树

Abstract

Rough sets theory, introduced by Z.Pawlak, is a new mathematical tool to deal with vagueness and uncertainty. Knowledge reduction is one of the main topics in the study of rough sets theory. It has received much attention of the researchers around the world. At present, rough sets theory has been applied to many areas successfully including artificial intelligence, pattern recognition and so on. The research and innovative results are focused on attribute reduction and rules extraction of decision tree as follows:

In inconsistent decision table, to overcome the disadvantage of ordered reduction which is based on the discernibility matrix as the temporal complexity is increscent exponential along with the size of decision tables, a new significance of attribute is defined, which is on the basis of the inclusion degree with separating consistent objects form inconsistent objects, so the judgment theorem with respect to distribution reduction is obtained, and a heuristic algorithm is proposed. Finally, the experimental analysis of this algorithm shows that it can obtain meaningful and small relative reduction.

To eliminate the limitations of the current conditional entropy, a new conditional entropy is defined with separating consistent objects form inconsistent objects, and the judgment theorem with respect to knowledge reduction is obtained from inequality. A heuristic algorithm is proposed. The example is given and the analyses show that the proposed heuristic information is better and more efficient than the others, and the method here reduces the temporal complexity and improves the operating efficiency. Experimental results prove the validity of this reduction method in searching the minimal or optimal reduction. So it enlarges the application area of rough sets theory. Based on the fore mentioned ideal, a new reduction algorithm about decision information entropy is proposed. The experimental result shows that this method is very effective and useful.

To eliminate the disadvantages of classical rough reduction algorithms based on positive region, a new decision power definition of algebra is proposed, and the new significance of an attribute is defined. The conclusion that decision power of knowledge decreases non-monotonously as the information granularities become finer is obtained, and a heuristic algorithm is proposed. Finally, the reduction comparison results of UCI discrete databases using four algorithms show that it is direct and practical.

To remedy some deficiencies of the current value reduction algorithms with attribute redundancy, rules

redundancy, and large computational complexity, the latest decision tree classification rule method is introduced, and a new heuristic function to build decision trees is proposed to extract decision rules.

To make up the shortcoming of the current information entropy for estimating decision ability, a new conditional entropy is defined, and the attribute selection metric of traditional heuristic algorithm is modified, so the new improved significance of an attribute is proposed. Finally, a heuristic algorithm for rules extraction of decision tree is designed. The benefit of this reduction method is that it needn't attribute reduction before extracting decision rules. The experiment and comparison show that the algorithm provides more precise and simple decision rules. On the base of the fore mentioned ideal, a new decision information entropy is proposed. In the process of decision tree building step by step bottom–up, condition attributes are considered to estimate the significance for decision classes. A procedure for reduction of traversing decision rules is also constructed, and helps to get more precise rules.

Key words: rough sets, decision table, attribute reduction, rules extraction, decision tree

独 创 性 声 明

本人郑重声明：所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，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的研究成果，也不包含为获得河南师范大学或其他教育机构的学位或证书所使用过的材料。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了谢意。

签名：王小林 日期：2007.6.13

关于论文使用授权的说明

本人完全了解河南师范大学有关保留、使用学位论文的规定，即：有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权河南师范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）

签名：王小林 导师签名：张文彤 日期：2007.6.13

第一章 绪 论

1.1 研究的目的与意义

由于计算机科学与技术的发展，特别是计算机网络的发展，每时每日都为人们提供了大量的信息，一方面人们可以从中获取更多更有用的知识，另一方面，也使大量知识掩盖在海量信息之中，人们不易从中辨析。在这种矛盾中，智能信息处理成为当前信息科学理论与应用领域中一个崭新的研究热点。特别是最近 20 年间，人们在专家系统、知识工程、人工神经网络、模糊集等领域不断实践与探索，取得了很多很好的成绩。随着数据库技术的不断发展及数据库管理系统的广泛应用，数据库中存储的数据量急剧增大，然而，人们处理与分析数据的能力却是相当有限的。如何从大量的、杂乱无章的、强干扰的数据（海量数据）中挖掘潜在的有利用价值的信息（知识），是人类的智能信息处理能力面临的前所未有的挑战，与此同时数据库知识发现（Knowledge Discovery in Database, KDD）技术应运而生。目前，知识发现（知识约简、规则提取、数据挖掘、机器学习）受到人工智能（Artificial Intelligence, AI）学界的广泛重视，如何高效地处理分布、实时海量数据成为当前知识发现中各种不同方法的研究热点，其中粗糙集方法是主要方法之一。

1965 年，当美国控制论专家 L.A.Zadeh[1]提出模糊（用英文词 Fuzzy 翻译德文 Vague）集的概念后，不少计算机科学家和逻辑学家试图依此概念来解决 1904 年谓词创始人 G.Frege 提出的含糊（德文 Vague）概念问题，但遗憾的是模糊集理论是不可计算的，即模糊集没有给出数学公式描述这一含糊概念，因此无法计算出其中包含的含糊元素数目。1982 年，波兰科学家 Z.Pawlak 针对 G.Frege 的边界线区域思想提出粗糙（Rough 是波兰人对 Vague 的翻译）集理论[2]，粗糙集有确定的数学公式来描述含糊概念，它认为边界线区域是上近似集与下近似集之间的差集，无法确认的个体都归属于边界线区域，而上近似集与下近似集都可以通过等价关系给出确定的数学公式描述，所以含糊元素数目可以被计算出来，即在真假二值之间的含糊程度可以计算，从而实现了 G.Frege 的边界线思想。

本研究得到河南省自然科学基金项目(0511011500)和河南省高校新世纪优秀人才支持计划(2006HANCET-19)资助。

粗糙集理论是建立在分类机制的基础上的，它将分类理解为在特定空间上的等价关系，而等价关系构成了对该空间的划分[3]。其将知识理解为对数据的划分，每一被划分的集合称为概念，并从新的角度认识知识，认为概念的不精确性是由知识的粒度过粗引起的。粗糙集理论认为知识即是将对象进行分类的能力，假定全域里的元素具有必要的信息或知识，通过这些知识能够将其划分到不同的类别；若对两个对象具有相同的信息，则它们就是不可区分的（即根据已有的信息不能够将其划分开），显然这是一种等价关系[4]。不可区分关系是粗糙集理论最基本的概念，在此基础上引入了成员关系、上近似与下近似等概念来刻画不精确性与模糊性，使得粗糙集理论能够有效地逼近这些概念。

粗糙集理论的主要思想是利用已知的知识库，将不精确或不确定的知识用已知知识库中的知识来近似刻画[5]。其要点是将分类与知识联系在一起，作为一种数学理论，它使用等价关系来形式化地表示分类，这样，知识就可以理解为使用等价关系集对离散表示的空间进行划分，即知识就是等价关系集对空间划分的结果[6]。粗糙集理论的主要兴趣在于它恰好反映了人们用粗糙集方法处理不分明问题的常规性，即以不完全信息或知识去处理一些不分明现象的能力，或依据观察、度量到的某些不确定结果进行分类数据的能力[4]。并能对不完整数据进行分析、推理，发现数据间的关系，提取有用的特征，简化信息处理，进而研究不精确、不确定知识的表达、学习与归纳等。

粗糙集理论作为一种新的处理不精确、不确定与不完全数据的数学方法，与其他处理不确定与不精确问题理论最显著的区别是它无需提供问题所需处理的数据集合之外的任何先验信息，所以可以说对问题不确定性的描述或处理是比较客观的。由于该理论未能包含处理不精确或不确定原始数据的机制，其与概率论、模糊数学及证据理论等处理不确定或不精确问题的理论有很强的互补性。由于它在机器学习与知识发现、数据挖掘、决策支持与分析、专家系统、归纳推理、模式识别等方面的广泛应用，现已成为一个热门的研究领域[6]。

粗糙集理论似乎特别适合于数据简化、数据近似分类和数据相关性、数据意义、数据相似或差别及数据模式的发现等[7]。粗糙集理论是对信息进行分析推理、发现数据间关系，提取特征、进行知识约简的有力工具。所以，研究粗糙集理论在知识发现中的应用，将会大大促进知识发现技术的进步，该研究有着广阔的发展前景。而经典的属性约简定义对噪声数据的抗干扰能力十分薄弱，往往导致提取的规则集丢失许多有用的信息；且目前多数学者都将精力集中在属性约简算法的提出和改进上，但属性值约简也是

决策规则挖掘过程中的关键步骤。因此，寻求快速的值约简算法可以尽快地获取最小决策规则集，以便高效地做出高质量的决策。高效的属性约简与规则提取技术是粗糙集理论应用于知识发现领域的基础，也是当前粗糙集理论研究的主要方向。

作为一种新的智能计算方法，粗糙集理论已在许多科学与工程领域中得到了广泛的应用，其中属性约简与决策规则提取技术仍是粗糙集理论研究的核心内容，也是河南省自然科学基金项目研究的一部分内容。

1.2 粗糙集理论研究的历史与现状

20 世纪 70 年代初，波兰学者 Z.Pawlak 和波兰科学院、华沙大学的逻辑家们组成了研究小组，对信息系统逻辑特性进行了长期的基础性研究。他们针对从实验中得到的以数据形式表述的不精确性、不确定性和不完整性的信息与知识，进行了分类分析，为粗糙集理论的产生奠定了基础。到了 80 年代，粗糙集理论引起了各国学术界的重视，许多数学家、逻辑学家与计算机研究人员对粗糙集理论及其应用产生了极大兴趣，并进行了广泛深入的研究。

从 1992 年至今，每年都召开以粗糙集为主题的国际会议，推动了粗糙集理论的拓展与应用。国际上成立了粗糙集学术研究会，参加的成员来自波兰、美国、加拿大、日本、挪威、俄罗斯、乌克兰与印度等国家[8]。目前，粗糙集理论已成为人工智能领域中一个较新的学术热点，引起了越来越多的科研人员的关注。在国内也成立了“中国Rough集与软计算学术研讨会（CRSSC）”，从 2001 年至今已经连续召开六届“Rough 集与软计算学术研讨会”。随着 CRSSC 系列研讨会在规模与质量上均呈良好的增长趋势，此领域的研究工作发展也很快。2003 年成立了中国人工智能学会粗糙集与软计算专业委员会，粗糙集理论的研究队伍更加壮大，研究成果在深度与广度上有了更大的发展。

下面给出粗糙集理论研究的历史与现状大事记：

1991 年，Z.Pawlak 教授出版了第一本关于粗糙集的专著[3]，成为粗糙集理论研究的第一个里程碑，奠定了粗糙集理论的基础。

1992 年，R.Slowinski 主编的关于粗糙集应用及其与相关方法比较研究的论文集出版[9]，对这一时期的工作成果作了很好的总结，推动了国际上对粗糙集理论与应用领域的深入研究，掀起了研究粗糙集理论的高潮。

在国际上，1992 年波兰 Kiekrz 召开了第一届国际粗糙集学术研讨会，这次会议着

重讨论了集合近似定义的基本思想及其应用。

1993年，在加拿大Banff召开了第二届国际粗糙集与知识发现研讨会。这次会议的主题是粗糙集、模糊集与知识发现。这次会议积极推动了国际上对粗糙集理论与应用的研究。由于当时正值数据库知识发现成为研究的热门话题，一些著名知识发现学者参加了这次会议，并且介绍了许多扩展粗糙集理论的知识发现方法与系统。

1994 年，在美国 San Jose 召开了第三届国际粗糙集与软计算研讨会，这次会议主要探讨了粗糙集与模糊逻辑、神经网络、进化理论等的融合问题。

1995 年，Z.Pawlak 等在 Communications of the ACM 上发表了“Rough Sets”[10]，文章概括性介绍了作为目前人工智能应用新技术之一的粗糙集理论的基本概念以及它在知识获取、机器学习、决策分析和知识发现等领域的具体研究项目和进展。

1995 年，ACM Communications 将粗糙集列为新浮现的计算机科学的研究课题。

1995 年，在美国 Wilmington 召开了第四届国际粗糙集研讨会，在这次会议上，对粗糙集合与软计算方法的基本观点与关系展开了激烈的探讨。

1996 年，在日本东京召开了第五届国际粗糙集研讨会，这次会议推动了亚洲地区对粗糙集理论与应用的研究。

1998年，国际信息科学杂志（International Journal of Information Sciences）为粗糙集理论的研究出了一期专辑。

1998 年，在波兰召开了第一届国际粗糙集与计算的当前趋势学术会议。

1999 年，在日本召开了第七届国际粗糙集、模糊集、数据挖掘和粒度——软计算会议，主要阐述了当前粗糙集、模糊集的研究现状与发展趋势。

2000 年，在加拿大召开了第二届国际粗糙集与计算的当前趋势学术会议。

2003 年，在重庆邮电大学召开了第九届国际粗糙集、模糊集、数据挖掘与粒计算学术会议。

2004 年，国际粗糙集协会主办的第一本粗糙集国际期刊《Advances in Rough Sets》出版发行。

2004 年，在瑞典召开了第四届国际粗糙集与计算的当前趋势学术会议。

2005 年，在加拿大召开了第十届国际粗糙集、模糊集、数据挖掘与粒计算学术会议。

2006 年，在美国 Georgia State University 召开了第二届粒计算国际会议。

2006 年，在日本 Kobe 召开了第五届国际粗糙集与计算的当前趋势学术会议。

波兰华沙大学、工业大学、信息技术与管理大学和加拿大Regina大学、圣玛丽大学以及英国Edinburgh大学、Ulster大学、Cardiff大学等对粗糙集理论都有深入的研究。

在国内，2001 年重庆邮电大学召开了第一届中国 Rough 集与软计算学术研讨会，以便国内学者共同理解和探讨粗糙集理论及应用研究的新内容与新方法，推动了粗糙集理论及其应用在国内的研究与发展。

2002 年，在苏州大学召开了第二届中国 Rough 集与软计算学术研讨会。

2003 年，在重庆邮电大学召开了第三届中国 Rough 集与软计算学术研讨会。

2004 年，在浙江海洋学院召开了第四届中国 Rough 集与软计算学术研讨会。

2005 年，在清华大学召开了第一届粒计算国际会议。

2005 年，在鞍山科技大学召开了第五届中国 Rough 集与软计算学术研讨会。

2006 年，在浙江师范大学召开了第六届中国 Rough 集与软计算学术研讨会。

2006 年，在南昌大学召开了 Rough 集前景——粒计算理论国际论坛。

2006 年，在重庆邮电大学召开了第一届粗糙集与知识技术国际会议。

2007 年，拟定在山西大学召开第七届中国 Rough 集与软计算学术会议，第一届中国 Web 智能学术研讨会和第一届中国粒计算学术研讨会。

目前，仍有许多重要的国际国内学术会议继续把粗糙集理论研究列入主要内容之一。在中国几乎所有重要的计算机学术期刊均刊登有粗糙集理论的学术论文。从研究地域来看，欧洲学者比较注重理论研究，北美学者比较注重应用，日本学者在粗糙集与概率论相结合以及在医学的应用方面比较突出，国内在知识约简、与信息论结合、粗糙逻辑、粒计算、知识的不确定性研究方面取得了较大的成果。对粗糙集理论的知识表示与处理不确定性问题数学方法的关系，近年来国内的研究也发展迅速，出现很多综述性报告及中文著作[4-8,11-18]。粗糙集理论已成为当前信息科学最为活跃的研究领域之一。

粗糙集理论经过国内外众多研究人员 20 多年的共同努力，不但为信息科学和认知科学提供了新的科学逻辑与研究方法，还为智能决策提供了有效的处理技术。作为一种新的知识发现方法，粗糙集理论不仅在数学理论上不断得到完善，而且在其它研究领域中也得到了成功的应用[3,9-23]，如机器学习、决策分析、近似推理、图象处理、医疗诊断、金融数据分析、专家系统、冲突分析、过程控制和数据库知识发现等领域。目前，粗糙集理论自身已成为完整、独立的科学领域。粗糙集理论模型也得到不断的完善和发展，并逐渐渗透到很多学科。此外，粗糙集理论与其他软计算理论形成了共同发展和优势互

补的局面，诸如与 Fuzzy 集、Dynamic Fuzzy 集、粒计算、遗传算法、神经网络等软计算理论[24-30]，已经成为当前国内外计算机及相关专业的研究热点。国内外学者也公认粗糙集理论是研究数据挖掘、知识约简与粒计算的理论基础。

1.3 粗糙集理论的优点与特点

采用粗糙集理论作为研究知识发现工具具有许多优点[4]：

（1）粗糙集理论包括了知识的一种形式模型，这种模型将知识定义为不可区分关系的一个族集，这就使知识具有了一种比较清晰的数学意义，并且很方便用数学方法来分析处理。

（2）粗糙集理论在数学上非常严密，有一套处理数据分类问题的数学方法，尤其是当数据具有噪声干扰、不完全性或不精确性时。

（3）粗糙集仅仅分析隐藏在数据中的事实，没有校正数据中所表现的不一致性，一般只将所生成的规则分为确定与不确定规则。

（4）粗糙集理论的实用性非常强，它是为开发自动规则生成系统而提出的，因而它的研究完全是应用的驱动。

（5）基于粗糙集的计算方法非常适合于并行处理，粗糙集计算机的研制工作已在进行之中，并取得了一定成果。

（6）粗糙集理论与模糊逻辑、神经网络、概率推理、信度网络、连接计算、遗传算法、混沌理论一起形成了软计算方法的基础，为问题的处理提供了鲁棒性强、成本较低的解决方案。

粗糙集理论具有很多自己的特点[8,31]，归纳如下：

（1）粗糙集不需要任何附加信息或先验知识。模糊集与概率统计方法都是处理不确定信息时常用的方法，但这些方法需要一些数据的附加信息或先验知识，如模糊隶属函数、概率分布等，这些信息有时并不容易得到。粗糙集分析方法仅利用数据本身提供的信息，无须任何先验知识，这是和模糊理论及证据理论最主要的区别。

（2）粗糙集是一个强大的数据分析工具。它能表达和处理不完备信息，能在保留关键信息的前提下对数据进行化简并求得知识的最小表达，能识别并评估数据之间的依赖关系，揭示出概念的简单模式，并能从经验数据中获取易于证实的规则知识，特别适合于智能控制。

（3）粗糙集与模糊集分别刻画了不完备信息的两个方面。粗糙集以不可区分关系为基础，侧重分类，模糊集基于元素对集合隶属程度的不同，强调集合本身的含混性（Vagueness）。从粗糙集的观点看，粗糙集合不能清晰定义的原因是缺乏足够的论域知识，但可以用一对清晰集合逼近。文献[24]阐述了粗糙集与模糊集的内在联系及模糊粗糙集（Fuzzy-Rough Set）的概念。粗糙集与证据理论也有一些相互交叠之处，在实际应用中可以相互补充。

粗糙集理论所具有的独特分析视角不仅可以克服传统不确定性处理方法的不足，而且与其它分析方法有机结合，有望进一步增强对不确定问题的处理能力。粗糙集理论对于人工智能与认知科学是十分重要的，自提出以来一直受到模糊数学创始人 L.A.Zadeh 的重视，并给与很高的评价。近年来，粗糙集理论凭借自己独特的优势，开始逐渐应用到知识发现的各个领域中，在对大型数据库中不完整数据进行分析学习方面具有广泛的应用前景及实用价值。粗糙集理论不仅为信息科学与认知科学提供了新的科学逻辑和研究方法，而且为智能信息处理提供了有效的处理技术。

1.4 粗糙集理论的研究方向

1.4.1 粗糙集理论的理论研究

目前，粗糙集在知识发现中的理论研究主要集中在数学性质、模型拓展、有效性算法、与其它多种不确定智能分析方法的融合、多 Agent 中的粗糙集、粒计算等方面。

（1）粗糙集理论数学性质方面的研究主要是对粗糙集理论中知识的不确定性问题进行理论研究，包括讨论粗糙集代数结构[32]、拓扑结构[33]、粗糙逻辑[34]、粗糙集的收敛性[35]以及信任函数（Belief Functions）[36]问题。随着粗糙结构与代数结构、拓扑结构、序结构等各种结构的不断整合，必将推动粗糙集理论的快速发展[12]。

（2）粗糙集理论模型拓展方面的研究包括可变精度模型（Variable Precision Rough Sets, VPRS）[37]、相似模型（RST Based on Similarity Relation）[38]和连续属性离散化模型[39]。主要解决粗糙集理论应用于数据分析时，遇到的数据噪声、数据不完备和连续数据离散化等问题。

（3）粗糙集理论中有效性算法的研究是粗糙集合在 AI 方向上研究的一个主要方向。目前，该研究主要集中在导出规则的增量式算法[40]、约简的高效启发式算法[41-43]、粗糙集合基本运算的并行算法[44]以及现有算法的改进[45]。

（4）在粗糙集理论与其他不确定智能分析方法之间关系的研究中，目前主要讨论它与模糊集理论[24-27]、D-S（Dempster-Shafer）证据理沦[46]、神经网络[47]、统计方法[48]和信息论的相互渗透与补充，研究怎样将粗糙集与其他不确定分析方法结合起来以取得更好的效果。

（5）在多 Agent 系统中粗糙集研究的焦点是多 Agent 系统基于粗糙集的推理和规则合成策略[49]。

（6）粒度计算也是粗糙集的一个新的发展方向[28-30]。

1.4.2 粗糙集理论的应用研究

粗糙集是发现知识、辅助决策的有效工具，具有坚实的理论基础。粗糙集理论自提出以来，已在许多领域中得到了应用。目前，随着对粗糙集理论研究的不断深入，粗糙集的应用领域不断得到了扩展。

近年来，在粗糙集理论发展的基础上，粗糙集应用方法大体有如下几个方面[50-51]：

（1）与其它研究方法相结合。例如与模糊集理论、模糊逻辑推理、模态逻辑、神经网络、遗传算法等处理不确定问题与软计算方法的有机结合，产生了粗模糊理论、粗神经网络等新的理论和研究方法。

（2）应用于规则学习和决策表推导。在保证简化后的决策系统具有与原决策系统相同分类能力的前提条件下，通过使用知识约简和范畴约简，将决策系统简化并找到最小（最短）决策规则集合，以达到最大限度泛化的目的。

（3）进行知识约简。约简和相对约简在粗糙集中十分重要，它反映了一个决策系统的本质。对条件属性集合的约简，可以保证简化后的决策系统具有与原决策系统相同的分类能力。从数据预处理的角度看属性约简能去掉冗余属性，提高系统的效率。

（4）进行属性相关分析。粗糙集方法中属性重要程度可以用来衡量该属性对分类的影响程度，进而对关键属性和次要属性分别进行处理，以得到较好的分类效果。

（5）进行数据离散化。将粗糙集理论引入数据离散化，可以避免离散化的盲目性，在保持原来数据分类能力不变的情况下进行有效的离散化。

（6）进行增量式学习。从粗糙集理论的差别矩阵出发，利用与/或逻辑关系求取规则描述。新的对象只需在差别矩阵上增加相应的列，即可获得增量后的规则。

粗糙集理论从诞生到现在虽然只有几十年的时间，但由于它具有较强的实用性，已经在许多领域获得了令人鼓舞的成果[6,8]：

（1）股票数据分析。利用粗糙集方法通过分析股票的历史数据，研究股票价格与指数之间的依赖关系，从而获得预测规则，这一研究成果已得到了华尔街证券交易专家的认可。

（2）模式识别。应用粗糙集方法研究语音识别、手写字符识别等问题，并提取特征属性，从而为计算机的进一步智能化打下基础。如邮政系统中的信件发送，信件的分类是一个十分繁琐的问题，如果利用粗糙集方法识别出手写字符，则信件的分类将变得十分简单，进而大大提高邮政系统的效率，降低费用。

（3）地震预报。利用粗糙集方法研究震前的地质、气象数据与里氏地震级别的依赖关系，从而为地震预报提供一定的依据。

（4）冲突分析。应用粗糙集方法已建立了反映以色列、巴勒斯坦、约旦、埃及、叙利亚和沙特阿拉伯等六国关于中东和平问题的谈判模型。

（5）数据库中的知识发现。数据库知识发现（KDD）又称数据挖掘（Data Mining），是当前人工智能与数据库技术交叉学科的研究热点之一。粗糙集方法现已成为 KDD 的一种重要方法，其导出的知识精练且更便于存储和使用。

（6）专家系统（Expert System, ES）。粗糙集抽取规则的特点，为构造 ES 知识库提供了一条崭新的途径。

（7）粗糙控制（Rough Control）。粗糙集理论根据观测数据获得控制策略的方法被称为从范例中学习（Learning from Examples），属于智能控制的范畴。文献[52]应用粗糙控制研究了“小车倒立摆系统”这一经典控制问题，取得了较好的结果。在过程控制领域，文献[53]应用粗糙集方法成功地提取出了水泥窑炉的控制规则。粗糙控制的优点是简单迅速、实现容易，不需要像模糊控制那样进行模糊化和去模糊化。因此在特别要求控制器结构与算法简单的场合，采取粗糙控制较为合适。另外，由于控制算法完全来自观测数据本身，其决策与推理过程可以很容易被检验和证实。一种新的有吸引力的控制策略“模糊粗糙控制（Fuzzy-Rough Control）”正悄然兴起，其主要思路是利用粗糙集获取模糊控制规则。

（8）医疗诊断。粗糙集方法根据以往的病例归纳出诊断规则，用来指导新的病例。如现有的人工预测早产准确率只有 17%-38%，应用粗糙集理论可以提高到 68%-90%。

（9）人工神经元网络（Artificial Neural Network, ANN）。训练时间过于漫长的固有缺点是制约 ANN 实用化的因素之一。但可以利用粗糙集方法化简神经网络训练样本数

据集，在保留重要信息的前提下消除多余数据，使训练速度提高4至5倍。如果将粗糙集与ANN结合起来，充分利用粗糙集处理不确定性的特长，就可以增强ANN的信息处理能力。

（10）决策分析。粗糙集的决策规则是在分析以往经验数据的基础上得到的。粗糙集允许决策对象中存在一些不太明确、不太完整的属性，弥补了常规决策方法的不足。如希腊工业发展银行ETEVA应用粗糙集理论协助制订信贷政策，是粗糙集多准测决策方法的一个成功范例。

粗糙集理论的应用领域还包括：近似推理、软件工程数据分析、图像处理、商业金融分析、硬件实现、材料科学中的晶体结构分析、新材料发现、预测建模、结构建模、过程控制、投票分析、电力系统、破产估计、飞行员评价等。

1.5 粗糙集理论存在的问题

作为一种新事物，粗糙集在实际应用中也遇到了许多困难，存在很多问题。关键问题主要表现在以下几个方面：

（1）不适合处理大规模数据。粗糙集本身特点决定了它在处理大规模问题时的低效性。因此，需要首先把大型数据进行有效处理，而如何进行有效处理有待进一步研究。

（2）不能有效地描述数据的不精确性或不确定性。粗糙集理论在处理数据时也有许多局限性，粗糙集理论对知识的不完整处理是有效的，但它未包含处理不精确或不确定原始数据的机制，因此，单纯的粗糙集理论不一定能有效地描述数据不精确或不确定的实际问题，需要其它方法来补充。该理论与概率论、模糊数学及证据理论等处理不精确或不确定问题的理论有很强的互补性。

（3）粗糙集方法存在容错能力与推广能力相对较弱的问题。当利用粗糙集方法提取规则时，它实质上就是对数据集进行约简，然后归并相同的数据。经过约简后获得的数据集就是所谓的被提取的规则。由此可见，粗糙集方法是通过去除“冗余”数据来获得规则的，然而，数据在一定程度上的冗余却能够提高其容错能力与推广能力。因此，粗糙集方法的容错能力与推广能力相对较弱。

（4）不能直接处理连续属性值数据。在利用粗糙集进行数据约简的时候，数据必须以知识表达系统的形式表示，因此粗糙集方法一般只能处理离散值的数据，对连续值的数据必须先进行离散化，而且离散化得好坏对最终结果有很大影响，这样，在一定程

度上限制了粗糙集方法的应用。目前还没有一种通用的、合适的数据离散化方法。

（5）现已证明寻找决策表最小约简是一个 NP-Hard 问题[47-48,54]。导致 NP-Hard 的主要原因是属性的组合爆炸问题。所以约简中的关键问题就是找出一种有效的搜索方法。启发式搜索算法是可行的方法，但这些算法属于串行搜索，结果的最优性与算法的效率有待深入研究。

解决这些局限性的方法首先是拓广基本的粗糙集理论，再者是将基本的粗糙集理论与其他数学方法有效结合。尤其是基于粗糙集理论开发的知识发现系统，促进了粗糙集理论的进一步发展，使其理论与应用的研究在国际上日益受到广泛重视。

这些问题虽然已引起了人们的重视，但仍有许多问题需要深化。目前基于粗糙集在下述方面有待进一步深化[55]：

（1）粗糙集与其它软计算方法的进一步结合问题。粗糙集理论与模糊集理论、模糊逻辑推理、模态逻辑等有着丰富的内在联系。粗糙集理论究竟是一种与这些理论并行的不确定性知识处理方式，还是可以由这些理论综合表示的复合体？它们如何来结合？

粗糙集理论与模糊集理论及神经网络方法的结合和交叉应用正广泛进行，新的理论与方法层出不穷。这样的结合是否一定有效？尽管它对用于实践的数据有较好的效果，但对于海量数据呢？非线性方法与非线性方法的反复结合，对系统的有效性是否有一定的影响？这是粗糙集的生长点，也是其发展的动力。

（2）粗糙集理论的精髓在于知识约简，且这种知识约简主要是针对决策表等离散化后的分明概念，能否将这种知识约简直接拓展到连续或模糊概念？并且，粗糙集的基本理论，如连续属性的离散化、决策表属性约简等问题都是 NP-Hard 难题，目前还缺乏普遍适用的算法。这是制约粗糙集实用化的重要方面。

（3）粗糙集基本运算的并行算法及硬件实现，将大幅度提高数据开采的效率，已有的粗糙集软件适用范围还很有限。

（4）现代信息系统具有分布异构的特点，解决的办法之一是分解：即先将用户提出的全局开采要求分解为不同节点的子数据库开采要求，然后在各个点上单独开采，最后集成，这可借助 Agent 技术。粗糙集与多智能体的结合也是一个崭新的课题。

（5）目前，粗糙集理论主要是对条件属性值不含有空值的完全决策表的研究，而实际上很多情况我们只能得到不完全决策表，即某些条件属性值为空值的决策表。因此对于不完全决策表的最小决策算法获取问题仍需深入研究。

（6）当前，只是对内容已初始给定的不变静态知识库的特征提取进行了研究，实际上随着人们知识不断更新与发展，知识库的内容可能发生变化。因此，找到一种对动态知识库特征提取的有效算法将具有重要意义。

综上所述，这些客观存在的问题，需要我们根据具体情况找到合适的解决方法。因而在本文中，我们重点对下面的问题进行研究：

（1）针对决策表最小约简求解的 NP-Hard 问题，引入有效的启发式搜索方法。

（2）借助决策树构造方法，进行属性值约简，进而获取决策规则。

1.6 论文研究内容与结构安排

1.6.1 论文主要研究内容

本文针对数据中存在冗余的情况，研究了基于粗糙集理论的知识约简，提出了新的去除冗余数据的属性约简方法，并采用目前归纳学习中最有效的决策树分类规则学习方法[56-58]，构造决策树，设计决策规则提取算法，这些方法能够有效地减少数据冗余，并且尽可能不损失数据的原始信息，部分实现了从数据中挖掘有用信息的目的。

本文重点研究以下内容：

（1）基于包含度的不一致决策表约简新方法

（2）基于新的条件熵的决策表约简方法

（3）一种新的基于决策熵的决策表约简方法[59]

（4）决策强度的决策表约简设计与比较

（5）基于新的条件熵的决策树规则提取方法[60]

（6）基于决策熵的决策树规则提取方法

1.6.2 论文结构安排

第一章主要是绪论。其包括以下六部分内容：首先阐述了本文研究的目的和意义；粗糙集理论研究的历史与现状；粗糙集理论的优点与特点；粗糙集理论的研究方向，包括理论研究与应用研究；粗糙集理论存在的问题；最后介绍了论文研究内容和结构安排。

第二章主要是粗糙集理论基础知识。首先介绍粗糙集理论基本概念，包括信息系统、上近似集与下近似集、分类精度、分类质量、决策表等；然后在知识约简中详细讨论代数观点与信息论观点下的属性约简；最后对决策规则进行概述。

第三章主要研究基于粗糙集的启发式属性约简方法。首先概述了现有四种不同的知识约简算法，并对其进行了详细比较与分析；然后针对比较分析的结果，设计了四种启发式属性约简算法：*a*）基于包含度的不一致决策表约简新方法；*b*）基于新的条件熵的决策表约简方法；*c*）一种新的基于决策熵的决策表约简方法；*d*）决策强度的决策表约简设计与比较。

第四章主要研究基于粗糙集的决策树规则提取。首先介绍了现有值约简算法，指出其存在的不足；接着概述了决策树技术，指出基于粗糙集构造的决策树比基于信息熵的ID3算法生成的决策树更简洁；然后引入决策树分类规则学习方法，结合更优的启发式函数来构造决策树，提取规则；最后设计了两种决策树规则提取算法：*a*）基于新的条件熵的决策树规则提取方法；*b*）基于决策熵的决策树规则提取方法。

第五章对全文所做的工作进行了总结。首先对粗糙集理论的前景做了初步展望，然后指出本文的不足之处，最后对下一步的研究工作给出进一步的探讨与设想。

第二章 粗糙集理论基础知识

粗糙集理论作为智能信息处理技术的一个新成果，在 KDD 分类规则发现与学习研究中有特殊作用。它比原来的模糊集与概率统计方法具有更大的优越性，是处理不精确信息的有力工具。因此研究如何用粗糙集方法提取本质特征，具有重要的实际意义和理论价值。下面我们先熟悉一下粗糙集理论的基础知识，关于其基本概念的详细定义，可参阅有关文献[12-14]。

2.1 粗糙集理论基本概念

在粗糙集理论中，“知识”被认为是对研究对象进行分类的能力。知识是用信息系统（即属性—值对表）来表示的。基于粗糙集理论进行数据分析的全部对象的数据集合称为信息系统（Information System, IS），用一个四元组来定义。一般情况下，表中的列标记不同的属性，行标记论域的对象。

定义 2.1 设四元组 $IS = (U,R,V,f)$为信息系统，其中 U 是一个非空集合，$U = \{x_1,x_2,\ldots,x_n\}$是一个有限对象构成的论域；$V = \cup\{ V_a \mid a\in R \}$，$V_a$为属性 a 的值域；f是一个信息函数，它对一个对象的每一个属性赋予一个信息值，其中

$$f: U\times R\rightarrow V，\text{即}\forall a\in R，x\in U，\text{有} f(x,a)\in V_a。$$

定义 2.2 在信息系统 $IS = (U,R,V,f)$中，任意属性子集 $P\subseteq R$ 决定了一个二元不可区分关系记为 $IND(P)$：

$$IND(P) = \{ (x,y)\in U\times U \mid f(x,a) = f(y,a), \forall a\in P \}, \tag{2-1}$$

等价关系 $IND(P)$可确定 U 的一个划分，用 $U/IND(P)$表示，简记为 U/P。

定义 2.3 在信息系统 $IS = (U,R,V,f)$中，属性子集 $P\subseteq R$，则对任意 $x\in U$，等价类（划分块）记为$[x]_P$：

$$[x]_P = \{ y \mid f(x,a) = f(y,a), \forall a\in P \}。 \tag{2-2}$$

定义 2.4 在信息系统 $IS = (U,R,V,f)$中，任意属性子集 $P\subseteq R$，$X\subseteq U$ 为论域的一个子集，记 $U/P = \{P_1,P_2,\ldots,P_m\}$，定义两个子集：

$$P\_(X) = \cup\{ P_i \mid P_i\in U/P, P_i\subseteq X \}, \tag{2-3}$$

$$P^{-}(X) = \cup\{ P_i \mid P_i\in U/P, P_i\cap X\neq \varnothing \}, \tag{2-4}$$

其中 $i = 1,2,\ldots,m$，则分别称它们为 X 关于 P 下近似集（Lower Approximation）和 P 上近似集（Upper Approximation）。同时也将 X 的 P 边界域、X 的 P 正区域（简称为正域）及 X 的 P 负域分别定义为：

$$BN_P(X) = \overline{P}(X) - \underline{P}(X), \tag{2-5}$$

$$POS_P(X) = \underline{P}(X), \tag{2-6}$$

$$NEG_P(X) = U - \underline{P}(X)。 \tag{2-7}$$

由此可以将 $\underline{P}(X)$与 $POS_P(X)$理解为根据知识 P 判断肯定属于 X 的 U 中的元素组成的集合；将 $\overline{P}(X)$看作由知识 P 判断可能属于 X 的 U 中元素组成的集合；那么边界域 $BN_P(X)$就是依据知识 P 无法确定属于 X 还是属于~X 的 U 中元素组成的集合，$BN_P(X)$在某种意义上是论域的不确定域，边界域中的元素既不能肯定地属于集合 X，也不能肯定地属于~X。负域 $NEG_P(X)$是由知识 P 判断肯定不属于 X 的 U 中元素组成的集合。

命题 2.1　（1）当且仅当 $\overline{P}(X) = \underline{P}(X)$，称集合 X 是 P 可定义集；

（2）当且仅当 $\overline{P}(X) \neq \underline{P}(X)$，称集合 X 是 P 粗糙集。

我们也可以将 $\underline{P}(X)$描述为 X 中的最大可定义集，将 $\overline{P}(X)$描述为含有 X 的最小可定义集。根据 X 的上近似集和下近似集的不同情况，可定义四种不确定程度的粗糙集：

（1）如果 $\underline{P}(X) \neq \varnothing$ 且 $\overline{P}(X) \neq U$，则称 X 为 P 粗糙可定义；

（2）如果 $\underline{P}(X) = \varnothing$ 且 $\overline{P}(X) \neq U$，则称 X 为 P 内不可定义；

（3）如果 $\underline{P}(X) \neq \varnothing$ 且 $\overline{P}(X) = U$，则称 X 为 P 外不可定义；

（4）如果 $\underline{P}(X) = \varnothing$ 且 $\overline{P}(X) = U$，则称 X 为 P 全不可定义。

这种划分的直观意义是：

如果集合 X 为 P 粗糙可定义，则意味着我们可以确定 U 中的某些元素属于 X 或~X。

如果集合 X 为 P 内不可定义，则意味着我们可以确定 U 中的某些元素是否属于~X，但不能确定 U 中的任一元素是否属于 X。

如果集合 X 为 P 外不可定义，则意味着我们可以确定 U 中的某些元素是否属于 X，但不能确定 U 中的任一元素是否属于~X。

如果集合 X 为 P 全不可定义，则意味着我们不能确定 U 中的任一元素是否属于 X 或~X。

定义 2.5　令 $F = \{X_1, X_2, \ldots, X_n\}$ 是基于 P 对 U 的一个分类或划分，P 是一个属性子集，则 P 对 U 近似分类的精度定义为：

$$\alpha_P(F)=\frac{\sum_{i=1}^{n}|P\_(X_i)|}{\sum_{i=1}^{n}|P^{-}(X_i)|}, \tag{2-8}$$

则 P 对 U 近似分类的质量定义为：

$$\gamma_P(F)=\frac{\sum_{i=1}^{n}|P\_(X_i)|}{|U|}, \tag{2-9}$$

其中$|X|$表示集合 X 的基数。

二者的区别是：近似分类的精度描述的是当使用知识 P 进行分类时，所有可能的决策中正确决策的百分比；近似分类的质量描述的是应用知识 P 能确切地划入 F 类的对象的百分比。

在信息系统中，属性的重要性是指属性对分类的重要性。在讨论不同的问题时属性具有不同的重要性，这种重要性可在辅助知识的基础上事先定义，并用“权重”表示。在粗糙集理论中，可以不使用任何先验信息给出一种基于数据的客观度量。

为了找出某个属性或属性集的重要性，需要从属性集中去掉该属性或属性集，再来考察分类会发生什么变化。若分类情况改变较大，则说明该属性或属性集重要性高，反之重要性低。

定义 2.6（属性的重要性定义）对于属性集 C 导出的分类属性子集 $B'\subseteq B$ 的重要性，可以用两者依赖程度的差来衡量，即

$$\gamma_B(C)-\gamma_{B-B'}(C)。 \tag{2-10}$$

这表示当从属性集合 B 中去掉某些属性的集合 B'时，对属性集合 B 在 U/C 中的正域受到的影响。这里属性的重要性也可以用正域来衡量。

定义 2.7 在信息系统 $IS=(U,R,V,f)$中，属性子集 P、$Q\subseteq R$，属性集合 Q 的 P-正域记为 $POS_P(Q)$，定义为：

$$POS_P(Q)=\cup\{P\_(X)\,|\,X\in U/Q\}。 \tag{2-11}$$

当然属性的重要性可以用 $POS_{B-B'}(C)$与 $POS_B(C)$商的形式来表示。

2.2 知识约简

众所周知，知识库中知识（属性）并不是同等重要的，其中某些知识甚至是冗余的。

在保持知识库分类能力不变的前提下，删除其中不相关或不重要的知识，导出问题决策或分类规则的过程，称为知识约简，包括属性约简与属性值约简。

定义 2.8 设 P 为一个属性集合，且 $r\in P$，当 $IND(P)=IND(P-\{r\})$，称 r 为 P 中可省略的，否则 r 为 P 中不可省略的。若 $\forall r\in P$ 都为 P 中不可省略的，则称 P 为独立的。

定义 2.9 若 Q 为独立的，且 $IND(Q)=IND(P)$，$Q\subset P$，则称 Q 为 P 的约简。

定义 2.10 属性集合 P 中所有不可省略关系的集合称为 P 的核，记作 $CORE(P)$，且 $CORE(P)=\cap RED(P)$，其中 $RED(P)$是 P 的所有约简。

如果将信息系统中的属性进一步分成条件属性与决策属性，则称该信息系统为决策表。在知识约简中，经常遇到的是决策表约简问题。决策表约简，又称为知识的相对约简，其最终结果是将决策表中的知识化成少量的决策规则。决策表的简化一般有属性约简（它等价于从决策表中消去一些不必要的列）和属性值约简（它等价于从决策表中消去一些无关紧要的属性值）。因此，条件属性子集形成的分类与决策属性集形成的分类两者的关系十分重要。下面讨论与决策表约简相关的一些重要概念。

定义 2.11 设五元组 $S=(U,C,D,V,f)$为决策表，其中 U 为论域；C 为条件属性集；D 为决策属性集，且 $C\cap D=\varnothing$；$V=\cup\{V_a\mid a\in C\cup D\}$，$V_a$为属性 a 的值域；f是一个信息函数：$U\times(C\cup D)\rightarrow V$。

定义 2.12 在决策表 $S=(U,C,D,V,f)$中，条件属性子集 $P\subseteq C$，决策属性集 D 的 P-正区域 $POS_P(D)$定义为：

$$POS_P(D)=\cup\{P\_(X)\mid X\in U/D\}。\tag{2-12}$$

$POS_P(D)$表示论域 U 中所有通过分类 U/P 表达的知识能够确定地划入 U/D 类的对象集合。当 $POS_P(D)=POS_{P-\{r\}}(D)$时，称 $r\in P$ 为 P 中相对于 D 可省略的，否则 r 为 P 中相对于 D 不可省略的。若 $\forall r\in P$ 都为 P 中相对于 D 不可省略的，则称 P 为 D 独立的。

定义 2.13 在决策表 $S=(U,C,D,V,f)$中，称 $POS_C(D)$为决策表 S 的一致对象集，$U-POS_C(D)$为决策表 S 的不一致对象集；若 $POS_C(D)=U$，则称决策表 S 为一致决策表，否则称决策表 S 为不一致决策表。

定义 2.14（代数观点和信息论观点下的属性约简定义）

（1）代数观点下两种属性约简定义

① 设 $B\subset P$，若 B 是 P 相对于 D 的独立子集，且 $POS_B(D)=POS_P(D)$，则称 B 为 P 的 D 相对约简。条件属性子集 P 中所有 D 不可省略的原始关系称为 P 的 D 核，记为

$CORE_D(P)$，且有 $CORE_D(P)=\cap RED_D(P)$，其中 $RED_D(P)$是 P 中所有 D 约简。

② 设条件属性子集 $P\subseteq C$，若 $\gamma_P(D)=\gamma_C(D)$，且不存在 $P^*\subseteq P$，使得 $\gamma_{P^*}(D)=\gamma_C(D)$，则称 P 为 C 相对于 D 的一个属性约简。

（2）信息论观点下属性约简定义

假设 U 为一个论域，P 与 Q 为 U 上的两个等价关系，可以认为 U 上任一等价关系簇（属性集）是定义在 U 上的子集组成 σ 代数上的一个随机变量，其概率分布可通过如下方法确定。

设 P 与 Q 在 U 上导出的划分分别为 $X=\{X_1,X_2,\ldots,X_n\}$，$Y=\{Y_1,Y_2,\ldots,Y_m\}$，则 P 与 Q 在 U 上的子集组成 σ 代数上的概率分布为：

$$(X:p)=\begin{bmatrix} X_1 & X_2 & \cdots & X_n \\ p(X_1) & p(X_2) & \cdots & p(X_n) \end{bmatrix}, \tag{2-13}$$

$$(Y:p)=\begin{bmatrix} Y_1 & Y_2 & \cdots & Y_m \\ p(Y_1) & p(Y_2) & \cdots & p(Y_m) \end{bmatrix}, \tag{2-14}$$

其中 $p(X_i)=|X_i|/|U|$，$i=1,2,\ldots,n$，$p(Y_j)=|Y_j|/|U|$，$j=1,2,\ldots,m$。

P 与 Q 的联合概率分布为：

$$(XY:p)=\begin{bmatrix} X_1\cap Y_1 & \cdots & X_i\cap Y_j & \cdots & X_n\cap Y_m \\ p(X_1Y_1) & \cdots & p(X_iY_j) & \cdots & p(X_nY_m) \end{bmatrix}, \tag{2-15}$$

其中 $p(X_iY_j)=|X_i\cap Y_j|/|U|$，$i=1,2,\ldots,n$，$j=1,2,\ldots,m$。

有了知识的概率分布定义后，根据信息论可以定义知识的熵与条件熵的概念。

知识（属性集合）P 的熵记为 $H(P)$，定义为：

$$H(P)=-\sum_{i=1}^{n}p(X_i)\log(p(X_i)), \tag{2-16}$$

知识（属性集合）Q（$U/Q=\{Y_1,Y_2,\ldots,Y_m\}$）相对于知识（属性集合）$P$（$U/P=\{X_1,X_2,\ldots,X_n\}$）的条件熵记为 $H(Q|P)$，定义为：

$$H(Q|P)=-\sum_{i=1}^{n}p(X_i)\sum_{j=1}^{m}p(Y_j|X_i)\log(p(Y_j|X_i)), \tag{2-17}$$

其中 $p(Y_j|X_i)=|X_i\cap Y_j|/|X_i|$，$i=1,2,\ldots,n$，$j=1,2,\ldots,m$。

在此基础上，我们可以得到信息论观点下的属性约简定义：

设 U 是一个论域，P 为一条件属性集合，条件属性子集 $Q\subseteq P$ 是 P 相对于决策概念

D 的一个约简的充要条件为：

（1）$H(D|Q)=H(D|P)$；

（2）对于 Q 中任意一个属性 r 都有 $H(D|Q)\neq H(D|Q-\{r\})$。

2.3 决策规则

粗糙集理论的一个主要应用是经过归纳学习获取有用的规则。归纳学习是通过对大量实例进行推理归纳与对其共性的分析，抽象出一般的概念与规则，使得这些新概念与新规则能够蕴含更多的实例。规则是对原始数据泛化的结果，可以理解为其是一种高度抽象，代表其中最有价值、最直观又可直接为人所用的信息，得到的规则主要用来在将来的决策过程中对未知的观察实例进行决策判定。

规则中很重要的一类是决策规则又称分类规则，其基本形式是 M→N，或者 if M then N，其中 M 是规则的前件，用于指出该规则是否可用的条件；N 是一组结论或操作，用于指出当前件 M 所指的条件被满足时，应该得出的结论或应该执行的操作。整条规则的含义是：如果前件 M 被满足，则可推出结论 N 或执行 N 所规定的操作。

在决策表 $S=(U,R,V_f)$ 中，$R=C\cup D$，C 为条件属性集，D 为决策属性集。下面给出应用于决策表的决策规则的形式化描述。

定义 2.15（公式定义）

（1）(a,v)（或写为 a_v，其中 $a\in R$，$v\in V_a$）是原子公式，原子公式是公式。

（2）如果 A 和 B 是公式，那么¬A，A∧B，A∨B，(A)，A→B 都是公式。

（3）只有按定义（1）和（2）所组成的式子是公式。

定义 2.16 公式 A→B 的逻辑含有称为决策规则。A 是规则的前件，B 是规则的结论，表示若 A 成立则 B 也成立。

定义 2.17 公式 $(a_1,v_1)\wedge(a_2,v_2)\wedge\ldots\wedge(a_n,v_n)$ 称为 P 基本公式，其中，v_i 是属性 a_i 可能的取值，$\{a_1,a_2,\ldots,a_n\}\in P$，$P\subseteq C$。

若一个规则的前件与结论分别是 P 基本公式和 Q 基本公式，则称该规则为 PQ 规则。

定义 2.18 公式 A→B 为决策规则，如果 A 是基本公式且 $B=(d,d_i)$，$d\in D$ 是决策属性，则 A→B 为基本决策规则。

在实际应用中，决策规则可以是基本决策规则的逻辑组合，任何决策规则都可以分解为一个或多个等价的基本决策规则。因此我们主要研究基本决策规则，若没有特殊说

明，本文所提到的规则都是基本决策规则。决策规则用于对象的分类时也称其为分类规则。由多条规则所组成的集合称为规则集或规则库。

定义 2.19 在决策表 $S=(U,C,D,V,f)$中，决策规则为隐含式，记为$(a_1,v_1)\wedge(a_2,v_2)\wedge\ldots\wedge(a_n,v_n)\rightarrow(d,d_j)$，其中$\{a_1,a_2,\ldots,a_n\}\in C$，$v_i\in Va_i$，$i=1,2,\ldots,n$，$\{d\}\in D$，$d_j\in V_d$。

用记号 $M(r)$表示所有能匹配规则前件实例的集合，$S(r)$表示所有满足该规则实例的集合，即集合中的实例同时能匹配规则的前件与结论，且 $S(r)\subseteq M(r)$。

定义 2.20 在决策表 $S=(U,C,D,V,f)$中，决策规则 r: A→B，则该规则的支持度为 $SP(r)=|S(r)|/|U|$，覆盖度为 $CV(r)=|M(r)|/|U|$，可信度为 $CF(r)=|SP(r)|/|CV(r)|$。

第三章　基于粗糙集的启发式属性约简方法研究

粗糙集是通过不可区分关系对论域进行划分，用上近似集与下近似集对给定概念进行逼近，进而确定粗糙的概念表示形式。目前，这种方法正被广泛应用于数据挖掘的各个阶段。在粗糙集理论及其应用中，属性约简是粗糙集研究的核心内容之一。

一般情况下，信息系统中的条件属性并不是同等重要的，有些条件属性是多余的，删除这些多余的条件属性并不影响原来的系统。属性约简就是在不影响原来系统的情况下，删除不相关或不重要的条件属性，使原有的系统得到简化。在不同的决策系统中，或者在不同的条件环境下，人们对属性约简的要求与期望是不一致的。如果在某个决策系统中，存在一些属性，它们的属性值难以得到（测量这些属性值所需要的花费代价很高），这种情况下我们所需要的就是获取代价较少的属性约简。而一般情况下，我们希望得到的约简是包含条件属性数目尽可能少的情况。本章讨论的就是一般情况，而所说的最优属性约简就是所获得约简的条件属性个数最少。

获取属性约简的方法有很多，但无论什么方法，其有效性较好与效率较高才是设计约简算法的最终目标。

3.1 现有属性约简算法

3.1.1 基于属性重要性的属性约简算法

在决策表中，人们更关心的是哪些条件属性对决策更重要，因此有必要对一般属性的重要性进行度量。属性重要性算法首先由 Hu Xiao Hua 提出，它主要利用属性的重要度，将“属性的重要性”这一度量作为算法的启发式信息[61]。

这种算法的基本思路是：先计算出核，用核作为计算的初始约简；而后根据其它属性重要程度的大小依次在核的基础上添加属性，或者根据决策属性对条件属性依赖程度的大小自顶向下逐步剔除那些对分类不产生影响的属性，直到所得的属性集与原信息系统（或决策表）的分类能力相同为止；然后对该约简属性集合中的属性逐一检查，若发现去掉该属性不会改变集合对决策属性的依赖度，则将其删除，否则检查下一属性。

算法 3.1（Hu Xiao Hua 算法）

输入：信息系统 $IS=(U,A,V,f)$；

输出：　$B\subseteq A$ 且 $IND(B)=IND(A)$。

步骤 1：求核 $CORE(A)$；

步骤 2：初始化 $B=CORE(A)$；

步骤 3：对任意 $a\in A-B$，计算 $Sig(c,B,A)$，

如果 $Sig(c',B,A)=\max\{Sig(c,B,A)\mid c\in A-B\}$存在，那么 $B=B\cup\{c'\}$；

步骤 4：如果 $IND(B)=IND(A)$，那么输出 B，算法结束，否则转步骤 3；

步骤 5：结束。

其中属性重要性函数记为 $Sig(c,B,A)$，定义为：

$$Sig(c,B,A)=\frac{|IND(B\cup\{c\})|-|IND(B)|}{|IND(A)|}。\tag{3-1}$$

基于Pawlak属性重要性的算法与Hu Xiao Hua算法的区别在于属性重要性函数的定义不同，其中 Pawlak 属性重要性函数记为 $Sig(c,B)$，定义为：

$$Sig(c,B)=\frac{|IND(B\cup\{c\})|-|IND(B)|}{|IND(U)|}。\tag{3-2}$$

经过简单计算，可以证明这两种算法对属性重要性的排序完全相同，而影响约简结果的只是属性间的序关系，与具体值无关，所以它们是等价的。

通常情况下，决策表的知识约简不是唯一的，但人们关心的是寻找知识库中的最小约简。然而，现在已经证明，寻找决策表所有约简或最优约简是 NP-Hard 问题，解决这一问题的一般方法是采用启发式搜索方法，求出最优或次优约简[42]。因此，采用启发式方法寻求快速高效的决策表知识约简方法，仍是粗糙集理论研究的主要方向之一。至今，许多学者用不同的方法从不同的角度对知识约简做了深入的研究，提出属性重要性的度量方法主要有：根据依赖度（即正区域基数）的变化来定义[41-42,62-63]、根据信息熵来定义[43,64-67]以及根据区分矩阵中属性出现频度来定义[68]。

现阶段对属性重要性的定义存在着不一致性，与其相关的属性约简算法主要有：文献[69]通过区分矩阵求得属性约简集，该算法利用条件熵计算属性约简集中属性间的相关性，并将平均值最小的属性集作为最佳属性约简的结果。文献[66]研究了在属性约简过程中决策属性集相对于条件属性集的条件熵的变化规律，并在此基础上提出了一个新的约简算法。苗夺谦、胡桂荣在文献[64]中从信息的角度，以决策表中添加某属性所引起的互信息的变化大小作为该属性重要性的度量，提出了一种基于互信息的知识相对约简的启发式算法。另外，还有多篇文献[67,70]就属性重要性约简算法进行了探讨，以上

这些算法的侧重点虽不同，但其中属性重要性的确定多是利用本章中提到的几种度量方法，各算法分别在不同程度上对度量方法进行了改进。然而，这些知识约简算法仍存在一定的不足[43,71]：

（1）由于不一致对象的存在，基于正区域与现有信息论的方法无法等价地表示知识约简；

（2）没有完全客观地反映决策表“决策能力”的真实变化情况；

（3）约简算法的时间复杂度比较高。

因此，简单直观而又有效的约简算法还需要在理论和实践上进一步研究与完善。

3.1.2 基于区分矩阵的属性约简算法

区分矩阵（又称可辨识矩阵或差别矩阵）是通过引入代数知识，把对数据库的约简问题转化为对矩阵的化简问题。Skowron 已经严格证明，区分矩阵原理与 Pawlak 粗糙集理论是等价的[63]。区分矩阵是粗糙集理论中的重要概念之一，使用区分矩阵方法可以计算决策表的核与所有约简[12]。

这种算法的基本思路是：利用区分矩阵导出区分函数，然后求解区分函数的析取范式，该范式中的每一个析取项即为系统的一个约简。这种算法直观，易于理解，能够计算出核与所有约简。不足之处是在区分矩阵中会出现大量重复元素，降低了属性约简算法的效率，其时间复杂度随决策表大小的增长而指数增长。

针对基于区分矩阵约简算法时间复杂度较大的问题，文献[14]提出相应的简化方法：从信息系统中提取属性值是分明的属性，并构造分明合取范式，同时对这种逻辑公式做等价变换，直接得到最小析取范式。该范式对应信息系统的多项约简，减少了生成区分矩阵的中间环节，节省了搜索的空间与时间，提高了计算机执行约简算法的运行效率。

下面以信息系统为例来描述基于区分矩阵约简的一般算法。

设信息系统 $IS=(U,A,V,f)$，该信息系统的差别矩阵可以定义为：

$$M(IS)=(c_{ij})_{n\times n}, \tag{3-3}$$

其中，$c_{ij}=\{a\in A \mid a(x_i)\neq a(x_j)\}$，$i,j=1,2,\ldots,n$，$n=|U|$。

同理，决策表 $S=(U,C,D,V,f)$的区分矩阵可以定义为：

$$M(S)=(c_{ij})_{n\times n}, \tag{3-4}$$

其中，$c_{ij}=\begin{cases}\{a\in C \mid a(x_i)\neq a(x_j)\} & d(x_i)\neq d(x_j)\\ \varnothing & \text{其他}\end{cases}$。 (3-5)

这样，对信息系统（或决策表）的研究就转化为对差别矩阵的研究。一个属性在差别矩阵中出现的次数越多，表明它能区分的对象也越多，越重要。

算法 3.2（基于区分矩阵的信息系统约简算法）

输入：信息系统 $IS=(U,A,V,f)$；

输出：$B\subseteq A$ 且 $IND(B)=IND(A)$。

步骤 1：构造 $M(IS)$；

步骤 2：求核 $CORE(A)$（差别矩阵中只含单个属性的元素的集合）；

步骤 3：初始化 $B=CORE(A)$；

步骤 4：$M(IS)=M(IS)-\{c_{ij}\mid c_{ij}\cap B\neq\varnothing\}$；

步骤 5：如果 $M(IS)=\varnothing$，那么输出 B，算法结束；

步骤 6：计算每个 $c\in A-B$ 的重要性，记 $Sig(c,B,A)$为统计 c 在 $M(IS)$中出现的次数，

若 $Sig(c',B,A)=\max\{Sig(c,B,A)\mid c\in A-B\}$成立，

那么 $B=B\cup\{c'\}$，转步骤 4；

步骤 7：结束。

需要指出的是差别矩阵算法对独立约简而言是不完备的。到目前为止有多位学者对区分矩阵作了深入研究，Hu 等根据 Skowron 教授提出的可辨识矩阵得出了一个确定决策表信息系统核属性集合的方法[72]。而叶东毅在文献[73]中指出：利用这种差别矩阵求不一致决策表的核是错误的，并通过对可辨识矩阵的改进提出了一种计算核属性的方法，利用这个新的差别矩阵可以正确计算决策表的核。张文修在文献[12]中给出差别矩阵的另一定义，并指出它能够计算决策表的核与约简。胡可云在文献[74]中提出一种基于区分矩阵的启发式约简计算方法。代建华、李元香在文献[75]中基于 Hu 区分矩阵，采用“自底向上”的方法，在逐步减小的区分矩阵中挑选出现最频繁的属性加入，直到构成一个约简为止。以上这些算法在大多数情况下都能找到最小约简，胡可云提出的算法在没找到最小约简的情况下，能找到次优约简，算法时间复杂度较低且计算速度较快，但出现数据集的核很小或为空时，该算法可能会出现无休止地运行，最终无法得到约简结果的情况。代建华、李元香提出的算法通过处理较小的区分矩阵得到约简，节省了空间，利于处理数据量大和属性繁多的数据集，但算法的完备性还有待于进一步研究。王国胤在文献[76]中进一步分析 Hu 的差别矩阵的不足，指出 Hu 方法错误的根本原因在于决策表的不一致性，即 Hu 方法只能适用于计算一致决策表在代数定义下的核属性，而

对不一致决策表是不正确的。由此分析可见，导致现有差别矩阵多种定义的原因是决策表的不一致性。

3.1.3 遗传算法

遗传算法是一种借鉴生物界自然选择与自然遗传机制的高度并行、随机、自适应搜索算法，它主要用在最优化问题与机器学习中。隐含并行性与对全局信息的有效利用能力是遗传算法的两大显著特点。基于遗传算法求取粗糙集约简的各种算法不同之处，主要在于编码与适值函数的选择不同，由 Bjorvand 提出的遗传算法具有一定的代表性[77]。

在复合系统的约简与扩展法则方面，Kryszkiewicz 与 Rybinski 研究了在复合信息系统中寻求约简的问题，即怎样利用已有子系统的约简求取复合系统的约简[78]。Starzyk、Nelson 与 Sturtz 提出了一种称为强等价（Strong Equivalence）的新概念[79]，进而发展为扩展法则，用于快速简化区分函数。

从某种程度上讲，遗传算法、复合系统与扩展法则都具有同步并行的思想，但当前在大数据集合与多表信息处理问题上仍未找到合理的解决方案，三者是否都适于大规模并行计算机，如何更合理的解决并行计算问题，有待于进一步深入研究。

3.1.4 动态约简

动态约简[80]通过对原决策系统的预处理，首先提取出有强烈概率因素的子表族作为约简对象，再将每个次子表约简结果进行取交集处理，进而得出所有子表的相对稳定约简。动态约简的计算过程主要是对决策系统进行采样，然后对采样后的决策系统计算所有约简，在所有子表中保持不变或近似保持不变的约简就是所获取的动态约简。显然，这些存在于大多数子决策系统中的约简最能描述原决策系统以及部分决策系统的特性，也最能体现动态约简的优势，从而增强了约简的抗噪能力。

Bazan、Skowron 与 Synak 在文献[81]中提出动态约简的概念，在理论上为决策表最稳定约简奠定了初步的基础，将动态约简定义为属于决策表的约简集合，且在各个子表约简集合中出现的约简。在某种意义上动态约简是给定决策表中最稳定的约简，动态约简能有效地增强抗噪音能力。在 Bazan 的基本思想基础上，对精度系数进行调整，更加突出体现了动态约简的特征，使理论体系更为完备，由此为探讨动态约简的更深层次问题打下了坚实的基础。徐燕、怀进鹏在文献[82]中提出了一种处理噪音的有效约简算法（*Reduce-β* 算法）。针对数据库动态建立与粗糙集约简存在的问题，韩斌在文献[83]中提

出了一种动态属性约简算法。

3.2 基于包含度的不一致决策表约简新方法

在粗糙集理论及其应用中，不一致决策表是现实决策分析问题中经常遇到的决策信息系统，也是决策信息系统约简处理研究的重点之一。目前，许多学者对知识约简做了深入的研究，并取得了大量的成果[84]。文献[85]给出了等价矩阵的定义，将粗糙集中的计算转化为矩阵计算，该方法直观有效，但没有全面考虑决策表的情形。文献[69]中求核属性集时用的区分矩阵对于不一致决策表来说是错误的[76]，它不一定能正确地求出核属性集；再者其求出多个约简后，在实际应用中一般并不比较各个约简的优劣。文献[86]中的算法用区分矩阵求核属性集，对于不一致决策表也是错误的[76]。文献[87]给出了不一致决策表分布约简与分配约简两种定义，并讨论了它们的等价形式，但并没有对这两种知识约简方法进一步研究。文献[84]从理论上证明了文献[67]中条件信息熵约简与文献[15]中分布约简是等价的，它们不仅能保证一致决策规则的决策能力不变，而且也能保证不一致决策规则的决策能力不变，其中分布约简可以求出决策表的所有约简。我们知道这些研究大多是在矩阵中进行的，其时间复杂度随决策表大小的增长而呈指数增长，并且在实际问题中也没有必要求出所有约简，因为人们通常关心的是寻找知识库中所含条件属性最少的约简，即最优或最小约简。

本节为了解决以上问题，在不一致决策表中，首先分析了现有矩阵方法的局限性，以知识的包含度为基础，将一致与不一致对象分开，给出分布约简的数学定义与判定定理，设计了求分布约简新的启发式方法。由于该方法不用计算矩阵和与它对应的最小简化的析取范式，从而节省了空间和时间，提高了运行效率，克服了区分矩阵方法计算时间复杂度过高的缺陷。实例验证表明，该约简方法在效率上较现有的约简方法有一定的提高，有助于搜索最小或次优约简。

3.2.1 现有矩阵方法的局限性

在经典知识约简方法中，采用区分矩阵方法求出核属性集，然后从区分矩阵中删除含有核属性的矩阵元素，将不含核属性且以析取形式表示的矩阵元素变成合取形式的表达式，最后对这个表达式进行化简，化简后转化为析取范式。但是合取范式转化为析取范式的过程是相当复杂的，往往导致区分矩阵方法的时间复杂度随系统大小增加而指数增长。析取范式中的每一项加上核属性集是决策表的一个约简，这样就可以求出所有约

简。然而在实际问题中没有必要求出所有约简，人们通常关心的是寻找最小约简。因此，我们认为经典粗糙集理论中的矩阵方法不能有效地搜索最小或次优约简。

3.2.2 分布约简

文献[15]中基于包含度的分布约简可以求出决策表的所有约简，对于一致与不一致决策表，都可以使用区分矩阵法求出核属性集。由于核与约简是决策表知识约简中最重要的概念，而且区分矩阵的主要目标也是为了计算核与约简，但现有基于区分矩阵的求核方法，时间与空间都不理想。为克服通过区分矩阵求核与约简方法的局限性，提高运行效率，我们需要寻求一种新的启发式方法。然而经典粗糙集理论中的基于正区域的属性重要性只对正区域基数进行定量描述，基于现有条件信息熵的属性重要性只描述了条件属性子集等价类中属于不同决策类的对象分离情况，而没有考虑其决策属性值相同的一致与不一致对象的分离。正因为如此，在不一致决策表中，由于不一致对象的存在，使用正区域与现有条件信息熵的方法，无法等价地表示知识约简[43]。那么若将所有不一致对象从一致对象中分离出来，就有助于搜索最小或次优约简。因此，我们在知识包含度的基础上，将一致与不一致对象分开，来寻求一种新的启发式信息。

定义 3.1[15] 在决策表 $S=(U,C,D,V,f)$中，$B\subseteq C$，D 在 U 上导出的划分 $U/D=\{D_1, D_2,\ldots,D_m\}$，记 $D(D_i/[x]_B)=|D_i\cap[x]_B|/|[x]_B|$，其中 $i=1,2,\ldots,m$，$x\in U$，则称 $D(D_i/[x]_B)$是 U 的幂集 $P(U)=\{X\mid X\subseteq U\}$上的包含度。

在决策表$S=(U,C,D,V,f)$中，令$D_0=U-POS_C(D)$，显然有D_0关于C的下近似集$C_D_0=D_0$。由定义2.12知，若S为一致决策表，则$C_D_0=\varnothing$。

为了将等价类逐步细化和分离，我们引入下面的定理与推论。

定理3.1[43] 在决策表$S=(U,C,D,V,f)$中，条件属性子集A、$B\subseteq C$，则$POS_A(D)=POS_B(D)$的充要条件是$A_D_i=B_D_i$，其中$i=1,2,\ldots,m$。

推论3.1[43] 在决策表$S=(U,C,D,V,f)$中，条件属性子集$A\subseteq C$，则$POS_A(D)=POS_C(D)$的充要条件是$A_D_i=C_D_i$，其中$i=1,2,\ldots,m$。

在决策表 $S=(U,C,D,V,f)$中，若集簇$\{A_D_0,A_D_1,A_D_2,\ldots,A_D_m\}$中没有空集，则该集簇是 U 上的一个划分。若该集簇有空集，则去掉空集后仍是 U 上的一个划分。为叙述方便，若没有特殊说明，不妨假设该集簇中没有空集。如果条件属性子集 A 是 C 的一个约简，则 A 在 U 上导出的一个划分是$\{A_D_0,A_D_1,A_D_2,\ldots,A_D_m\}$，它不仅把属于不同决

策类的一致对象分离成不同的划分块，而且把所有不一致对象从一致对象中分离出来，作为一个单独的划分块。这样，在不一致决策表 S 中，条件属性集 C 在 U 上导出划分 $\{C\_D_0,C\_D_1,C\_D_2,\ldots,C\_D_m\}$ 也就是把 U/D 中的等价类逐步细化和分离的过程，在这一过程中，可将所有不一致对象从一致对象中分离出来，由此划分得到的等价关系记为 R_C，即 $U/R_C=\{C\_D_0,C\_D_1,C\_D_2,\ldots,C\_D_m\}$。

在不一致决策表 $S=(U,C,D,V,f)$ 中，若记 $\mu_B(x)=((C\_D_0/[x]_B),(C\_D_1/[x]_B),(C\_D_2/[x]_B),\ldots,(C\_D_m/[x]_B))$，其中 $\varnothing\neq B\subseteq C$，$x\in U$，则称 $\mu_B(x)$ 为对象 x 关于 B 的决策分布函数。显然，$\mu_B(x)$ 是 U/R_C 上的条件概率分布。

这样，在不一致决策表 $S=(U,C,D,V,f)$ 中，我们可以得到分布约简的数学定义。

定义 3.2 在不一致决策表 $S=(U,C,D,V,f)$ 中，条件属性子集 $B\subseteq C$。若 $\forall x\in U$，$\mu_B(x)=\mu_C(x)$，则称 B 是分布一致集。若 B 是分布一致集，且 B 的任何真子集不是分布一致集，则称 B 为分布约简。

若条件属性子集 B 为不一致决策表 S 的分布约简，则由 B 产生的规则与由 C 产生的规则有相同的可信度。因此，分布约简是保持原决策表中条件属性确定的等价类对决策属性等价类的隶属度（条件概率分布）不变的最小条件属性子集。

定义 3.3 在不一致决策表 $S=(U,C,D,V,f)$ 中，条件属性子集 $B\subseteq C$。若 $\forall x\in U$，有 $\mu_B(x)=\mu_{B-\{a\}}(x)$，则称属性 $a\in B$ 在 B 中是不必要的；否则称属性 $a\in B$ 在 B 中是必要的。

定义 3.4 在不一致决策表 $S=(U,C,D,V,f)$ 中，条件属性子集 $B\subseteq C$，任意属性 $a\in B$ 在 B 中的重要性定义为：

$$SGF_{B-\{a\}}(a)=|\{\forall x\in U\mid \mu_{B-\{a\}}(x)\neq\mu_B(x)\}|/|U|。\tag{3-6}$$

由定义 3.4 可以得到下面的性质。

性质 3.1 属性 $a\in B$ 在 B 中的重要性 $SGF_{B-\{a\}}(a)$ 满足 $0\leq SGF_{B-\{a\}}(a)\leq 1$。

性质 3.2 属性 $a\in B$ 在 B 中是必要的，当且仅当 $SGF_{B-\{a\}}(a)>0$，特别地，属性 $a\in C$ 在 S 中是必要的，当且仅当 $SGF_{C-\{a\}}(a)>0$。

性质 3.3 属性核 $CORE_D(C)=\{a\in C\mid SGF_{C-\{a\}}(a)>0\}$。

定义 3.5 在不一致决策表 $S=(U,C,D,V,f)$ 中，条件属性子集 $B\subset C$，任意属性 $a\in C-B$ 关于 B 的重要性定义为：

$$SGF_B(a)=|\{\forall x\in U\mid \mu_{B\cup\{a\}}(x)\neq\mu_B(x)\}|/|U|。\tag{3-7}$$

$SGF_B(a)$ 的值越大，说明在已知 B 的条件下，属性 $a\in C-B$ 关于知识 B 就越重要。

因此，可把 $SGF_B(a)$作为搜索最小或次优属性约简的启发式信息。

由定义 3.2 与性质 3.2，我们可以得到分布约简的判定定理。

定理 3.2 在不一致决策表 $S=(U,C,D,V,f)$中，条件属性子集 $B\subseteq C$，那么 B 是 C 相对于决策 D 的一个分布约简的充要条件为：

（1）对 $\forall x\in U$，有 $\mu_B(x)=\mu_C(x)$；

（2）对任意属性 $a\in B$，有 $SGF_{B-\{a\}}(a)>0$。

3.2.3 基于包含度的决策表约简

定理 3.2 从代数角度提供了分布约简的判定方法，这是求决策表知识约简的基础。由性质 3.3 可以很容易地求出条件属性集 C 的核 $CORE_D(C)$。由于核是唯一的，且是任何约简的子集，因此，核可作为求最小约简的起点。依据定义 3.5 中定义的属性重要性，逐次选择最重要的属性添加到核中去，直到其决策分布函数等于条件属性集 C 的决策分布函数为止。

根据上述分析可知，基于包含度的决策表约简新方法可以保持决策表的决策能力完全不变。我们知道以 $SGF_B(a)$为启发式信息的约简方法，必须计算 $\mu_B(x)$与 $\mu_{B\cup\{a\}}(x)$。为了降低该方法的时间复杂度，首先需要研究计算 $\mu_B(x)$与 $\mu_{B\cup\{a\}}(x)$的高效方法。用文献[42]中计算划分与文献[41]中计算正区域方法，可得计算 $\mu_B(x)$与 $\mu_{B\cup\{a\}}(x)$的具体步骤如下：

算法 3.3（计算 $\mu_B(x)$与 $\mu_{B\cup\{a\}}(x)$的算法）

输入：不一致决策表 $S=(U,C,D,V,f)$，$B\subseteq C$，$x\in U$；

输出：$\mu_B(x)$和 $\mu_{B\cup\{a\}}(x)$。

步骤 1：用基数排序方法计算 U/C，U/D，U/B 和 $U/(B\cup\{a\})$；

步骤 2：用渐增式方法计算 $POS_C(D)$，获得 U/R_C；

步骤 3：计算 $\mu_B(x)$和 $\mu_{B\cup\{a\}}(x)$；

步骤 4：输出 $\mu_B(x)$和 $\mu_{B\cup\{a\}}(x)$；

步骤 5：结束。

经分析计算可知，步骤 1 最坏的时间复杂度为 $O(|C||U|)$，步骤 2 最坏的时间复杂度为 $O(|U|\log|U|)$，步骤 3 最坏的时间复杂度为 $O(|U|)$，那么算法 3.3 总的最坏时间复杂度为 $O(|C||U|)$。

在算法 3.3 中，步骤 2 已将等价类逐步细化和分离，由此以属性核为起点，自底向

上逐步增加属性以获取最小或次优属性约简。

算法 3.4（基于包含度的决策表约简）

输入：不一致决策表 $S=(U,C,D,V,f)$，$x\in U$；

输出：一个相对最小分布约简。

步骤 1：计算 S 的属性核 $CORE_D(C)$和决策分布函数 $\mu_C(x)$；

步骤 2：初始化 $B=CORE_D(C)$，如果 $B=\varnothing$，

那么选择 $U/\{a\}$（$a\in C$）不同等价类的决策分布函数非零值最多的属性 a；

步骤 3：如果 $\mu_B(x)=\mu_C(x)$，那么转步骤 7；

步骤 4：对任属性 $a\in C-B$，计算 $SGF_B(a)$；

步骤 5：选择使 $SGF_B(a)$最大的属性 a，

如果有多个属性同时使 $SGF_B(a)$达到最大值，那么从中选取一个属性 a 使其与 B 的等价类数$|U/(B\cup\{a\})|$最大，且 $B=B\cup\{a\}$；

步骤 6：如果 $\mu_B(x)\neq\mu_C(x)$，那么转步骤 4，

否则{ $B=B-CORE_D(C)$；

$t=|B|$；

$for(i=1;i\leq t;i++)$

{ $a_i\in B$；

$B=B-\{a_i\}$；

如果 $\mu_{B^*}(x)\neq\mu_C(x)$，其中 $B^*=B\cup CORE_D(C)$，那么 $B=B\cup\{a_i\}$；

}

$B=B\cup CORE_D(C)$为相对最小分布约简；

}

步骤 7：输出 B 为一个相对最小分布约简；

步骤 8：结束。

算法 3.4 中步骤 6 保证该决策表约简方法是完备的，即条件属性子集 B 不能再约简。但是很多算法是不完备的，不能保证一定能得到约简，文献[64,67]提出的约简算法是不完备的[41]。由算法 3.3 与文献[41]中计算核的方法，经分析得到算法 3.4 总的最坏时间复杂度为 $O(|C||U|)+O((|C|-1)|U|)+O((|C|-2)|U|)+\ldots+O(|U|)=O(|C|^2|U|)$，低于基于区分矩阵方法[69,85-86]的时间复杂度。

3.2.4 应用实例分析与比较

表 3-1 给出不一致决策表 $S=(U,C,D,V,f)$，其中 $U=\{x_1,x_2,\ldots,x_{10}\}$，$C=\{a_1,a_2,\ldots,a_5\}$，$D=\{d\}$。

表 3-1　不一致决策表 S

U	a_1	a_2	a_3	a_4	a_5	d
x_1	1	1	1	1	0	1
x_2	1	0	0	0	1	0
x_3	0	0	1	0	0	0
x_4	1	0	0	0	1	1
x_5	1	1	0	1	0	1
x_6	0	0	1	0	1	0
x_7	1	0	0	0	0	0
x_8	0	1	0	0	0	0
x_9	0	0	1	0	0	1
x_{10}	1	0	0	0	0	1

用算法 3.4 可求出决策表 3-1 的属性约简结果为$\{a_1,a_2,a_5\}$。

通过上面的分析可以看出，本节从包含度的角度构造了一个新的属性重要性算子，而以代数观点[41]和信息论观点[64,67]定义的属性重要性没有区分一致与不一致对象，并且以它们为启发式信息的约简算法[41,64,67]，均获取属性约简$\{a_1,a_3,a_4,a_5\}$，而不是最小属性约简$\{a_1,a_2,a_5\}$。所以，对于不一致决策表来说，本节提出的启发式约简新方法能有效地获取最小或次优相对约简。

3.2.5 小结

现有基于区分矩阵方法的时间复杂度通常随系统大小增加而呈指数增长，在一定程度上限制了粗糙集理论的广泛应用，因此寻求快速高效的粗糙集约简算法具有重要的实际意义。本节在深入研究知识包含度理论的基础上，给出了分布约简的数学定义与判定定理，从而设计了一种求分布约简的启发式新方法。理论分析与实例验证表明，该方法较现有基于区分矩阵法，时间复杂度更低，为从不一致决策表中搜索最小分布约简提供了一种有效算法。

3.3 基于新的条件熵的决策表约简方法

在决策应用中，属性约简的目的是在保持决策表决策能力不变的前提下，约简属性。本节针对现有属性约简算法中存在的不足[43,71,76]，充分考虑衡量决策表决策能力的两个重要指标：决策规则的可信度（对象的条件概率分布）与对象覆盖度；在此基础上把一

致与不一致对象分开，对现有的信息论方法进行了改进，定义了一种新的条件熵；然后在文献[71]决策表属性约简思想的基础上，给出知识约简的判定定理，使其能够等价地表示知识约简；最后提出了一种基于新的条件熵的启发式属性约简方法。实验比较与分析的结果表明，基于新的条件熵的属性重要性是一种更优的启发式信息，与现有约简算法相比，该方法提高了运行效率，节省了搜索空间与时间。

3.3.1 现有条件熵的局限性

在决策表 $S=(U,C,D,V,f)$中，属性约简的最终目标是在保持决策表 S “决策能力”不变的前提下，去除多余条件属性。由文献[43,67]中基于条件信息熵的决策表约简算法分析可知，一个条件属性是否可以约简，取决于删除该条件属性后决策表 S 产生的条件熵是否改变。由于决策表 S 中一致对象集 $POS_C(D)$产生的条件熵为 0，所以决策表 S 的条件熵改变是由不一致对象集 $U-POS_C(D)$产生的，那么对于决策表 S，增加新的不一致对象或原有不一致对象属于各个决策属性分类的条件概率分布改变，均会引起条件熵发生变化。因而，文献[43,67]的决策表约简算法对决策表“决策能力”衡量的标准表现在以下两个方面：

（1）产生的确定性决策规则数目不变；

（2）产生的不确定性决策规则可信度不变。

若决策表产生的确定性决策规则数目不变，则意味着这些决策规则的可信度不变（可信度仍为 1）。因此，在经典粗糙集理论中文献[43,67]的约简算法只考虑所有决策规则在约简后其可信度是否发生变化。

然而，在决策应用中，决策规则除了其可信度外，规则的对象覆盖度也是衡量其决策能力的重要指标[71]。因此，现有基于条件信息熵的约简算法存在局限性，不能客观地反映决策能力的实质。

3.3.2 新的条件熵与约简定理

为客观有效地反映知识约简后决策表决策能力的真实变化情况，本节提出新的条件熵概念以弥补现有条件熵的局限性。

定义 3.6[88] 设 U 是一个论域，属性集合 $P(U/P=\{X_1,X_2,\ldots,X_n\})$的信息熵记为 $H(P)$，定义为：

$$H(P)=\sum_{i=1}^{n}p(X_i)(1-p(X_i))。\tag{3-8}$$

为了研究能够体现对象覆盖度的知识信息熵，我们引入下面的引理。

引理 3.1[63] 设 P 与 Q 为论域 U 上的两个等价关系集合，则有 $U/(P\cup Q)=U/P\cap U/Q$ 成立。

引理 3.1 的证明参考文献[63]。

这样，在决策表 $S=(U,C,D,V,f)$中，属性集合 $P\cup D$（$P\subseteq C$）的信息熵有如下定义。

定义 3.7 设 U 是一个论域，P（$U/P=\{X_1,X_2,\ldots,X_n\}$）为一个条件属性集合，$D=\{d\}$（$U/D=\{Y_1,Y_2,\ldots,Y_m\}$）为决策属性集，则属性集合 $P\cup D$ 的信息熵 $H(P\cup D)$定义为：

$$H(P\cup D)=\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{|X_i\cap Y_j|}{|U|}(1-\frac{|X_i\cap Y_j|}{|U|})。\tag{3-9}$$

在 $P\cup D$ 的信息熵定义中，$|X_i\cap Y_j|/|U|$代表了该决策规则的对象覆盖程度，而在现有条件熵的定义中，$p(Y_j|X_i)=|X_i\cap Y_j|/|X_i|$代表了决策表产生某一决策规则的可信度。因而，条件熵 $H(D|P)$定义与信息熵 $H(P\cup D)$定义分别反映了决策表决策能力的变化情况。为了更好地研究知识的粗糙性，我们可以把两种熵的定义结合起来，使其客观地反映决策表决策能力的两个重要指标及其真实变化情况。

然而文献[41-42]中基于正区域的属性重要性只对正区域基数进行了定量描述[43]；文献[65,67]中基于信息论的属性重要性，只描述了条件属性子集等价类中属于不同决策类的对象分离情况，而没有考虑决策表中决策属性值相同的一致与不一致对象的分离。正因为如此，在不一致决策表中，由于不一致对象的存在，使用正区域与现有信息论的方法无法等价地表示知识约简。那么将所有不一致对象从一致对象中分离出来，就有助于搜索最小或次优知识约简。

在决策表$S=(U,C,D,V,f)$中，令$Y_0=U-POS_C(D)$，显然$C_Y_0=Y_0$。由定义2.12知，若S为一致决策表，则$C_Y_0=\varnothing$。为了将等价类进一步细化和分离，我们需要引入3.2节中的定理3.1。

定理3.1 在决策表$S=(U,C,D,V,f)$中，条件属性子集A、$B\subseteq C$，则$POS_A(D)=POS_B(D)$的充要条件是$A_Y_i=B_Y_i$，其中$i=1,2,\ldots,m$。

由定理3.1可推出，在决策表S中，条件属性子集$A\subseteq C$，则$POS_A(D)=POS_C(D)$的充要条件是$A_Y_i=C_Y_i$。

在决策表 $S=(U,C,D,V,f)$中，如果集簇$\{A_Y_0,A_Y_1,A_Y_2,\ldots,A_Y_m\}$中没有空集，则该

集簇是 U 上的一个划分。如果该集簇有空集，则去掉空集后仍是 U 上的一个划分。为叙述方便，若没有特殊说明，不妨假设该集簇中没有空集，也就是说我们是针对不一致决策表进行讨论。

显然，如果条件属性子集 A 是 C 的一个约简，则 A 在 U 上导出的一个划分是$\{A_Y_0, A_Y_1, A_Y_2, \ldots, A_Y_m\}$，它不仅把属于不同决策类的一致对象分离成不同的划分块，而且把所有不一致对象从一致对象中分离出来，把它作为一个单独划分块。这样，在不一致决策表 S 中，条件属性集 C 在 U 上导出划分$\{C_Y_0, C_Y_1, C_Y_2, \ldots, C_Y_m\}$也就是把 U/D 的等价类逐步细化和分离的过程，在这一过程中得到的等价关系记为 RD，即 $U/RD = \{C\_Y_0, C\_Y_1, C\_Y_2, \ldots, C\_Y_m\}$。

由上述理论分析可知，我们不仅能够得到反映决策表决策能力变化情况的两种熵的定义，而且还能找到有助于搜索最小或次优知识约简的方法。在此基础上我们定义一种新的条件熵概念。

定义 3.8（新的条件熵） 设 B 是论域 U 上的一个条件属性集合，$D = \{d\}$为决策属性集，则 B 关于等价关系 RD 的新的条件熵记为 $H(RD; B)$，定义为：

$$H(RD;\ B) = H(RD|B) - H(B \cup RD)。 \tag{3-10}$$

有了知识的条件熵定义，我们可以得到与其相应的属性重要性度量方式。

定义 3.9 在决策表 $S=(U,C,D,V,f)$中，条件属性子集 $B \subseteq C$，任意属性 $a \in C-B$ 的属性重要性定义为：

$$SGF(a,B,D) = H(RD;\ B) - H(RD;\ B \cup \{a\})。 \tag{3-11}$$

特别当 $B=\varnothing$ 时，$SGF(a,\varnothing,D) = -H(RD; \{a\})$。

$SGF(a,B,D)$的值越大，说明在已知 B 的条件下，属性 $a \in C-B$ 关于知识 B 就越重要。

通过属性约简可从决策表中删除冗余属性，而且能保持决策表在约简前后的决策能力完全相同。而文献[71]决策表属性约简采用不等式条件，从符合条件的约简结果中选择最优解，同时约简过程中可能有意破坏整个决策表的一致性，增加新的不一致对象，使条件熵发生变化，从而使约简后的决策表决策能力增强。但当删除某一属性后，在条件熵不变的前提下，熵会变小或不变。所以满足现有信息论方法约简条件的情况，也一定满足文献[71]属性约简的条件。在此基础上我们引入文献[71]属性约简的思想，给出以不等式为条件的决策表约简判定定理。

定理 3.3 在决策表 $S = (U,C,D,V,f)$中，条件属性子集 $B \subseteq C$，则 B 是 C 相对于 D 的

一个约简的充要条件：

（1）$H(RD; B) \le H(RD; C)$；

（2）对于任意属性 $a \in B$ 都有 $H(RD; B-\{a\}) > H(RD; C)$。

对于决策表 S，文献[43,67]的决策表约简算法要求条件属性子集与条件属性集 C 的条件熵相等时才可以约简，采用的是等式条件，约简结果没有优劣之分；而定理 3.3 要求条件属性子集的条件熵不大于条件属性集 C 的条件熵时就可以约简，采用的是不等式条件，这样有助于搜索最小或次优知识约简。

3.3.3 基于新的条件熵的决策表约简

在以 $SGF(a,B,D)$为启发式信息的约简方法中，每次循环时条件属性子集 B 的 $H(RD; B)$均不变，这使得 $SGF(a,B,D)$最大的属性 a 就是 $H(RD; B\cup\{a\})$最小的属性。因此，在计算 $SGF(a,B,D)$的过程中，只需计算 $H(RD; B\cup\{a\})$，这样就避免计算 $H(RD; B)$，减少了计算量，进而减小了搜索时间，提高了运行效率。

根据上述分析，对于决策表 S，以 $SGF(a,B,D)$为启发式信息的约简方法，必须计算 $H(RD; B\cup\{a\})$。为降低该方法的时间复杂度，我们先来研究计算 B 关于决策 D 的 $H(D; B\cup\{a\})$的高效算法，由文献[67]中的定理 1 可得到算法 3.5 的具体步骤如下：

算法 3.5（计算 $H(D; B\cup\{a\})$的算法）

输入：决策表 $S=(U,C,D,V,f)$和 $B\subseteq C$；

输出：划分 $U/(D\cup B\cup\{a\})$和 $H(D; B\cup\{a\})$。

步骤 1：计算划分 $U/(B\cup\{a\})$和 U/D，从而得到 $U/(D\cup B\cup\{a\})$；

步骤 2：计算 $H^*(B\cup\{a\}) = -\sum_{i=1}^{n} p(X_i)\log(p(X_i))$ 和 $H(B\cup\{a\}\cup D)$，

其中 $X_i \in U/(B\cup\{a\})$；

步骤 3：计算 $H(D; B\cup\{a\}) = H^*(B\cup\{a\}\cup D) - H^*(B\cup\{a\}) - H(B\cup\{a\}\cup D)$；

步骤 4：输出划分 $U/(D\cup B\cup\{a\})$和 $H(D; B\cup\{a\})$；

步骤 5：结束。

用文献[42]中计算划分的方法，步骤 1 的时间复杂度为 $O((|B|+2)|U|)$，步骤 2 的时间复杂度为 $O(|U|)$，因而算法 3.5 总的最坏时间复杂度为 $O(|C||U|)$。

在算法 3.5 的基础上，下面给出属性约简的具体算法步骤。首先将等价类逐步细化和分离；其次以属性核为起点，自底向上逐步选择最重要的属性添加到核中去，直到获

取最小属性约简为止。

算法 3.6（基于新的条件熵的决策表约简算法）

输入：一个决策表 $S=(U,C,D,V,f)$；

输出：决策表 S 的一个相对约简 $B\subseteq C$。

步骤 1：计算决策表 S 的正区域 $POS_C(D)$和不一致对象集 $U-POS_C(D)$，

由此得到划分 U/RD；

步骤 2：计算条件属性集 C 相对于决策 D 的属性核 $CORE_D(C)$和条件熵 $H(RD; C)$；

步骤 3：初始化 $B=CORE_D(C)$，如果 $B=\varnothing$，那么转步骤 5；

步骤 4：如果 $H(RD; B)\le H(RD; C)$，那么转步骤 8；

步骤 5：对任意属性 $a\in C-B$，计算 $H(RD; B\cup\{a\})$；

步骤 6：选择使 $H(RD; B\cup\{a\})$最小的属性 a，

如果有多个属性同时使 $H(RD; B\cup\{a\})$达到最小值，那么从中选取一个属性 a，使其与 B 的等价类数$|U/(B\cup\{a\})|$最大，且 $B=B\cup\{a\}$；

步骤 7：如果 $H(RD; B)>H(RD; C)$，那么转步骤 5，

否则{ $B=B-CORE_D(C)$；

$s=|B|$；

$for(i=1; i\le s; i++)$

{ $a_i\in B$；

$B=B-\{a_i\}$；

如果 $H(RD; B\cup CORE_D(C))>H(RD; C)$，那么 $B=B\cup\{a_i\}$；

}

$B=B\cup CORE_D(C)$；

}

步骤 8：输出 B 为一个最小相对属性约简；

步骤 9：结束。

算法 3.6 中步骤 7 保证该约简算法是完备的，即 B 不能再约简。但是很多算法是不完备的，不能保证一定能得到约简，其中文献[41,43,68]提出的约简算法是完备的，文献[67]提出的约简算法是不完备的[41]。

用文献[41]计算正区域和核的方法，经分析算法 3.6 总的最坏时间复杂度为 $O(|C||U|)+O((|C|-1)|U|)+O((|C|-2)|U|)+\ldots+O(|U|)=O(|C|^2|U|)$，低于文献[67-68,71]中约简算

法的时间复杂度。

3.3.4 实验比较与分析

表 3-2 给出不一致决策表 $S=(U,C,D,V,f)$，其中 $U=\{x_1,x_2,\ldots,x_{10}\}$，$C=\{a_1,a_2,\ldots,a_5\}$，$D=\{d\}$。

表 3-2　不一致决策表 S

U	a_1	a_2	a_3	a_4	a_5	d
x_1	0	0	1	0	1	0
x_2	1	0	0	0	0	1
x_3	1	0	0	0	1	1
x_4	1	0	0	0	0	0
x_5	0	1	0	0	0	0
x_6	1	1	1	1	0	1
x_7	0	0	1	0	0	1
x_8	1	1	0	1	0	1
x_9	1	0	0	0	1	0
x_{10}	0	0	1	0	0	0

为了验证算法 3.6 的有效性，选用表 3-2 与 UCI 机器学习数据库[89]中 4 个离散数据集，分别用文献[41]的约简算法（简称算法 A）、文献[67]的 CEBRKCC 算法（简称算法 B）和算法 3.6 相比较，结果如表 3-3 所示。

表 3-3　约简算法比较（m 为条件属性基数，n 为对象基数）

数据集	约简前 m	算法 A	算法 B	算法 3.6
		约简后 m		
表 1	5	4	4	3
Liver-disorders	6	3	3	3
Zoo	17	10	11	10
Vehicle	19	4	4	4
Mushroom	22	5	4	4
算法时间复杂度		$O(m^2n\log(n))$	$O(mn^2)+O(n^3)$	$O(m^2n)$

由表 3-3 分析可知，算法 3.6 的时间复杂度相对较低。对于表 3-2 所示的不一致决策表，算法 3.6 可求出属性约简结果为$\{a_1,a_2,a_5\}$，而文献[41,67]的约简算法均获取属性约简$\{a_1,a_3,a_4,a_5\}$。另文献[41,67]的属性重要性求得属性 a_2 相对核的重要性较小，而算法 3.6 对于表 3-2 求得 $SGF(a_2,\{a_5\},D)$最大。所以，本节基于新的条件熵的属性重要性更能准确有效地描述属性 a_2 的重要性，以其为启发式信息的决策表约简方法能够有效地搜索最小或次优知识约简。这说明在不一致决策表中，本节提出的方法能够弥补基于正区域和现有信息论决策表约简算法的不足。

3.3.5 小结

本节在决策表中，引入文献[71]决策表属性约简的思想，提出一种基于新的条件熵的决策表约简方法。该方法具有如下特点：

（1）弥补了经典的知识约简方法反映决策表决策能力的局限性；

（2）能够等价地表示知识约简；

（3）时间复杂度较低。

实例分析的结果表明，该方法弥补了文献[71]属性约简算法时间复杂度比较高的缺陷，为从决策表中搜索最小或次优知识约简提供了一种简单有效的算法。虽然该方法具有这些优点，但它并没有考虑到在大型数据集分析中，由于人为测量误差或噪声可能导致某些数据被错误分类，致使抗噪声干扰能力较差，这在一定程度上制约了其处理复杂应用问题的有效性。在今后的工作中，我们将对此进行深入研究。

3.4 一种新的基于决策熵的决策表约简方法[59]

本节在文献[43,71,76]分析现有知识约简算法存在不足的基础上，在不一致决策表中，把一致与不一致对象分开，定义一种新的信息论形式——决策熵，并给出该约简的判定定理，由此提出一种基于决策熵的启发式属性约简方法。实验结果表明，该约简方法在效率上较现有的属性约简方法有一定提高。

在决策应用中，决策规则的可信度与对象覆盖度都是衡量决策能力的重要指标，但是经典粗糙集理论的知识约简方法并没有真实地反映决策表决策能力的变化情况。在此基础上我们对现有的熵与粗糙熵概念进行改进，提出一种新的信息论定义形式——决策熵，使其能完全客观地反映决策表决策能力的两个重要指标；然后在文献[71]中决策表属性约简思想的基础上，给出知识约简的判定定理，使其能够等价地表示知识约简，这样就节省了搜索空间与时间，提高了运行效率。

3.4.1 知识的决策熵

定义 3.10[88] 设 U 是一个论域，属性集合 R 在 U 上导出的划分 $U/R=\{R_1,R_2,\ldots,R_m\}$，则 R 在 U 上导出划分 U/R 的熵记为 $E(R)$，定义为：

$$E(R)=-\sum_{i=1}^{m}\frac{|R_i|}{|U|}\log(\frac{|R_i|}{|U|}), \tag{3-12}$$

其中$|R_i|/|U|$表示R_i在论域U上的概率。

为了研究能够体现对象覆盖度的知识粗糙性，我们引入3.3节中的引理3.1。这样，在决策表$S=(U,C,D,V,f)$中，属性集合$P\cup D$（$P\subseteq C$）的熵可有如下定义。

定义3.11 设U是一个论域，条件属性集合P在U上导出的划分$U/P=\{X_1,X_2,\ldots,X_n\}$，$D=\{d\}$（$U/D=\{D_1,D_2,\ldots,D_t\}$）为决策概念集，则属性集合$P\cup D$的熵定义为：

$$E(P\cup D)=-\sum_{i=1}^{n}\sum_{j=1}^{t}\frac{|X_i\cap D_j|}{|U|}\log(\frac{|X_i\cap D_j|}{|U|})。\tag{3-13}$$

在属性集合$P\cup D$熵的定义中，$|X_i\cap D_j|/|U|$代表了某一决策规则的对象覆盖度，所以该熵定义就反映了决策表决策能力变化的一个重要指标。

定义3.12[90] 设U是一个论域，P（$U/P=\{X_1,X_2,\ldots,X_n\}$）为一个条件属性集合，$D=\{d\}$（$U/D=\{D_1,D_2,\ldots,D_t\}$）为决策概念集，则决策概念集$D$关于属性子集$P$的粗糙熵记为$E(D_P)$，定义为：

$$E(D_P)=-\sum_{i=1}^{n}\sum_{j=1}^{t}\frac{|X_i|}{|U|}\log(\frac{|X_i\cap D_j|}{|X_i|})。\tag{3-14}$$

在定义3.12中，$U/P=\{X_1,X_2,\ldots,X_n\}$存在两种情况：$x\in X_i\subseteq D_j$与$x\in X_i\not\subset D_j$，其中$i=1,2,\ldots,n$，$j=1,2,\ldots,t$。前一种情况是完全可以确定的，因此，我们只需考虑$x\in X_i\not\subset D_j$的情况，则决策概念集$D$关于属性子集$P$的粗糙熵可简化为：

$$E(D_P)=-\sum_{i=1}^{n}\sum_{j=1}^{k}\frac{|X_i|}{|U|}\log(\frac{|X_{ij}|}{|X_i|}),\tag{3-15}$$

其中$X_{i1},X_{i2},\ldots,X_{ik}$（$k\le t$）是$X_i$与$D_1,D_2,\ldots,D_t$的非空交集。

根据定义3.12可知，$|X_i\cap D_j|/|X_i|$可以代表决策表产生某一决策规则的可信度。为了更好地研究知识的粗糙性，我们可以把两种熵的定义结合起来，使其完全客观地反映决策表决策能力的两个重要指标及其变化情况。

由3.3.2节对基于正区域与现有信息熵的属性重要性分析可知，在不一致决策表中，使用现有知识约简方法将无法等价地表示知识约简。针对这种情况，我们设计一种新的启发式方法使其能够等价地表示知识约简，下面引入3.2节中的定理3.1和推论3.1。

定理3.1 在决策表$S=(U,C,D,V,f)$中，条件属性子集A、$B\subseteq C$，则$POS_A(D)=POS_B(D)$的充要条件是$A_D_i=B_D_i$，其中$i=1,2,\ldots,t$。

推论3.1 在决策表$S=(U,C,D,V,f)$中，条件属性子集$A\subseteq C$，则$POS_A(D)=POS_C(D)$的

充要条件是$A_D_i = C_D_i$，其中$i = 1,2,\ldots,t$。

在决策表S中，如果条件属性子集A是条件属性集C的一个约简，则A在U上导出的一个划分是$\{A_D_0,A_D_1,A_D_2,\ldots,A_D_t\}$，它不仅把属于不同决策类的一致对象分离成不同的划分块，而且把所有不一致对象从一致对象中分离出来，作为一个单独划分块。这样，条件属性集C在U上导出划分$\{C_D_0,C_D_1,C_D_2,\ldots,C_D_t\}$也就是把$U/D$的等价类逐步细化和分离的过程，在这一过程中得到的等价关系记为RD，即$U/RD = \{C_D_0,C_D_1,C_D_2,\ldots,C_D_t\}$。在此基础上将一致与不一致对象分开，我们可以提出一种新的信息论定义形式——决策熵。

定义 3.13 设U是一个论域，P是U上的一个条件属性集合，$D = \{d\}$为决策概念集，则P关于等价关系RD的决策熵记为$E(RD|P)$，定义为：

$$E(RD|P) = E(RD_P) + E(P \cup RD)。 \tag{3-16}$$

有了知识的决策熵定义，我们就可以得到与其相应的属性重要性度量方式。

定义 3.14 在决策表$S = (U,C,D,V,f)$中，条件属性子集$B \subseteq C$，任意属性$a \in C - B$的属性重要性定义为：

$$SGF(a,B,D) = E(RD|B) - E(RD|(B \cup \{a\}))。 \tag{3-17}$$

特别当$B = \varnothing$时，$SGF(a,\varnothing,D) = -E(RD|\{a\})$。

$SGF(a,B,D)$的值越大，说明在已知B的条件下，属性$a \in C - B$关于知识B就越重要。

在计算$SGF(a,B,D)$的过程中，每次循环时条件属性子集B的$E(RD|B)$均不变，这使得$SGF(a,B,D)$最大的属性a就是$E(RD|(B \cup \{a\}))$最小的属性a。因此，把$SGF(a,B,D)$作为搜索最小或次优知识约简的启发式信息时，只需计算$E(RD|(B \cup \{a\}))$，这样就可以避免计算$E(RD|B)$，减少了计算量，进而提高了效率。

在决策表$S = (U,C,D,V,f)$中，经典属性约简方法的最终目标是在保持决策表S决策能力不变的前提下，进行属性约简。由 3.3.2 节对文献[71]中决策表属性约简算法分析可知，采用不等式条件，给出决策表知识约简的判定定理，将会使约简结果更能客观地反映决策表决策能力的真实变化情况。

定理 3.4 在决策表$S = (U,C,D,V,f)$中，条件属性子集$B \subseteq C$，则B是C相对于决策D的一个约简的充要条件为：

（1）$E(RD|B) \leq E(RD|C)$；

（2）对于任意属性$a \in B$都有$E(RD|(B - \{a\})) > E(RD|C)$。

3.4.2 基于决策熵的决策表约简

根据上述理论，我们首先将等价类逐步细化与分离；其次以属性核为起点，自底向上逐步选择最重要的属性添加到核中去，直到获取最小约简为止。具体操作步骤如下：

算法 3.7（基于决策熵的决策表约简算法）

输入：一个决策表 $S=(U,C,D,V,f)$；

输出：决策表 S 的一个相对约简 B。

步骤 1：计算 S 的正域 $POS_C(D)$与不一致对象集 $U-POS_C(D)$，从而得到 U/RD；

步骤 2：计算条件属性集 C 相对于决策 D 的属性核 $CORE_D(C)$与决策熵 $E(RD|C)$；

步骤 3：初始化 $B=CORE_D(C)$，如果 $B=\emptyset$，那么转步骤 5；

步骤 4：如果 $E(RD|B)\le E(RD|C)$，那么转步骤 8；

步骤 5：对任意属性 $a\in C-B$，计算 $E(RD|(B\cup\{a\}))$；

步骤 6：选择使 $E(RD|(B\cup\{a\}))$最小的 a，

如果有多个属性同时使 $E(RD|(B\cup\{a\}))$达到最小值，从中选取一个属性 a，使其与 B 的等价类数$|U/(B\cup\{a\})|$最大，且 $B=B\cup\{a\}$；

步骤 7：如果 $E(RD|B)>E(RD|C)$，那么转步骤 5，

否则{ $B=B-CORE_D(C)$；

$l=|B|$；

$for(i=1;i\le l;i++)$

{ $a_i\in B$；

$B=B-\{a_i\}$；

如果 $E(RD|(B\cup CORE_D(C)))>E(RD|C)$，那么 $B=B\cup\{a_i\}$；

}

$B=B\cup CORE_D(C)$；

}

步骤 8：输出 B 为一个最小相对属性约简；

步骤 9：结束。

算法 3.7 中步骤 7 保证该知识约简是完备的，即 B 不能再约简。使用文献[42]计算划分与文献[41]计算正区域及核的方法，经分析得到算法 3.7 总的最坏时间复杂度为 $O(|C|^2|U|)$，低于现有知识约简算法的时间复杂度[67-68,71]。

3.4.3 实验比较与分析

表 3-4 给出不一致决策表 $S=(U,C,D,V,f)$，其中 $U=\{x_1,x_2,\ldots,x_{10}\}$，$C=\{a_1,a_2,\ldots,a_5\}$，$D=\{d\}$。

表 3-4　不一致决策表 S

U	a_1	a_2	a_3	a_4	a_5	d
x_1	0	1	1	1	1	0
x_2	1	1	0	1	0	1
x_3	1	1	0	1	1	1
x_4	1	0	1	1	1	1
x_5	1	1	0	1	1	0
x_6	0	1	1	1	0	1
x_7	0	1	1	1	1	1
x_8	0	1	1	1	0	0
x_9	0	0	1	0	1	0
x_{10}	0	0	0	0	1	0

为了验证算法 3.7 的有效性，对于表 3-4 所示的不一致决策表，表 3-5 依次给出了文献[41]中算法 4、文献[67]中算法 1、文献[43]中算法 2 与算法 3.7 求得的条件属性相对核的重要性与属性约简结果。

表 3-5　约简算法比较

$C-\{a_5\}$及约简	算法 4	算法 1	算法 2	算法 3.7
a_1	0.100	0.165	0.565	2.595
a_2	0	0.014	0.689	3.144
a_3	0.100	0.089	0.365	1.195
a_4	0.200	0.204	0.604	2.473
约简	$\{a_1,a_3,a_4,a_5\}$	$\{a_1,a_3,a_4,a_5\}$	$\{a_1,a_2,a_5\}$	$\{a_1,a_2,a_5\}$

由表 3-5 分析可知，我们用文献[41]中算法 4 与文献[67]中算法 1 的属性重要性分别求属性 a_1 与 a_2 的重要性相对较小，获取约简结果为$\{a_1,a_3,a_4,a_5\}$，而不是最小约简结果$\{a_1,a_2,a_5\}$；文献[43]中算法 2 与算法 3.7 的搜索结果是最小约简$\{a_1,a_2,a_5\}$，算法 3.7 对于表 3-4 求得 $SGF(a_2,\{a_5\},D)$最大，$SGF(a_1,\{a_5\},D)$次之。所以，本节基于决策熵的属性重要性更能准确地描述属性 a_1 与 a_2 的重要性，以其为启发式信息的决策表属性约简方法能够有效地搜索最小或次优知识约简。这说明在不一致决策表中，算法 3.7 能够弥补基于正区域与条件信息熵决策表约简算法的不足。

3.4.4 小结

本节为弥补知识粗糙熵的局限性，提出了决策熵的概念，同时给出了一种相应的属

性约简算法。实验验证结果表明，该方法为从决策表中搜索最小或次优约简提供了一种快捷有效的算法。

3.5 决策强度的决策表约简设计与比较

针对经典粗糙集知识约简方法仍存在的问题，在决策应用中，为简化计算，本节首先对现有平均决策强度概念进行改进，在把一致与不一致对象分开的基础上，定义了一种新的代数形式——决策强度，以便于更好地获取最优或次优知识约简；然后证明了知识的决策强度与信息粒度之间的关系；在此基础上设计了一种基于决策强度的启发式属性约简方法。理论分析与实验比较的结果表明，基于决策强度的属性重要性是一种更准确、更有效的启发式信息，该方法比现有方法更容易搜索到最优或次优约简，节省了搜索时间与空间。

3.5.1 现有基于正区域约简方法的局限性

在决策表中，经典属性约简的最终目标是在保持决策表“决策能力”不变的前提下，去除多余条件属性。由文献[41]中基于正区域的约简算法分析可知，在决策表 $S=(U,C,D,V,f)$中，一个条件属性是否可以约简，取决于删除该条件属性后，决策表 S 中对应决策集合的下近似集是否改变。也就是说，条件属性子集 $P\subseteq C$ 为条件属性集 C 的一个属性约简的充要条件是：$\gamma_P(D)=\gamma_C(D)$，且不存在 $P^*\subseteq P$，使得 $\gamma_{P^*}(D)=\gamma_C(D)$。因而，文献[41]的约简算法只考虑决策表是否会产生新的不一致对象，而没有考虑原有不一致对象属于各个决策分类的概率分布是否发生变化。所以，文献[41]衡量决策表决策能力不变的标准是产生确定性决策规则的数目（一致对象的数目）不变；若产生的确定性决策规则数目不变，就意味着这些决策规则的可信度不变。由此可见，文献[41]只考虑确定性决策规则在约简后其可信度是否发生变化。

然而，在决策应用中，规则的对象覆盖度也是衡量其决策能力的重要指标[71]。因此，文献[41]基于正区域的约简算法存在局限性，不能客观地反映决策能力的真实变化情况。

3.5.2 知识的决策强度

定义 3.15[40] 在决策表 $S=(U,C,D,V,f)$中，属性集合 C 与 D 在 U 上导出的划分分别为 $U/C=\{X_1,X_2,\ldots,X_n\}$，$U/D=\{Y_1,Y_2,\ldots,Y_m\}$，且 $U/(C\cup D)=\{Z_1,Z_2,\ldots,Z_k\}$，则

$$\sigma_S = \frac{1}{k}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\frac{|X_i \cap Y_j|}{|X_i|}\times\frac{|X_i \cap Y_j|}{|U|}\right) \tag{3-18}$$

为决策表 S 的平均决策强度。其中，k 为等价划分$\{Z_1,Z_2,\ldots,Z_k\}$中划分块的数目。

在平均决策强度的定义中，$|X_i \cap Y_j|/|X_i|$代表了决策表产生某一决策规则的可信度，$|X_i \cap Y_j|/|U|$代表了该决策规则的对象覆盖程度，k 为决策表化简前对应属性集合下所产生的决策规则数目。而文献[40]根据决策表新增对象与原有最简规则集的关系，快速推出了新决策表满足平均决策强度条件的最简规则集。这样，平均决策强度条件在求最简规则集方面就有了显著效果；但在求决策表属性约简方面，比照经典约简算法，若不考虑原始决策规则集数目，则不仅减少了计算量，也不影响搜索最优或次优约简。但基于正区域的属性重要性只对正区域基数进行定量的描述[41-42]，而没有考虑决策表中决策属性值相同的一致与不一致对象的分离。由此可以得出，在不一致决策表中，由于不一致对象的存在，使用基于正区域与信息论的方法无法等价地表示知识约简[43]。所以，在不一致决策表中，将所有不一致对象从一致对象中分离出来，有助于搜索最优或次优约简。

在决策表$S=(U,C,D,V,f)$中，令$Y_0=U-POS_C(D)$，显然有$C_Y_0=Y_0$。由定义2.12知，若S是一致决策表，则$C_Y_0=\varnothing$。这样，为了将等价类逐步细化和分离，我们引入3.2节中的定理3.1。

定理3.1 在决策表$S=(U,C,D,V,f)$中，条件属性子集P、$Q\subseteq C$，则$POS_P(D)=POS_Q(D)$的充要条件是$P_Y_i=Q_Y_i$，其中$i=1,2,\ldots,m$。

由定理3.1可推出，在决策表S中，条件属性子集$P\subseteq C$，则$POS_P(D)=POS_C(D)$的充要条件是$P_Y_i=C_Y_i$。

在决策表 $S=(U,C,D,V,f)$中，若集簇$\{P_Y_0,P_Y_1,P_Y_2,\ldots,P_Y_m\}$中没有空集，则该集簇是 U 上的一个划分。若该集簇有空集，则去掉空集后仍是 U 上的一个划分。为叙述方便，若没有特殊说明，不妨假设该集簇中没有空集，也就是说我们主要针对不一致决策表进行研究和讨论。显然，若条件属性子集 P 是条件属性集 C 的一个约简，则 P 在 U 上导出的一个划分是$\{P_Y_0,P_Y_1,P_Y_2,\ldots,P_Y_m\}$，它不仅把属于不同决策类的一致对象分离成不同的划分块，而且把所有不一致对象从一致对象中分离出来，把它作为一个单独划分块。这样，条件属性集 C 在 U 上导出划分$\{C_Y_0,C_Y_1,C_Y_2,\ldots,C_Y_m\}$也就是把决策表 S 中 U/D 的等价类逐步细化与分离的过程，由此得到的等价关系记为 RD，即 $U/RD=\{C\_Y_0,C\_Y_1,C\_Y_2,\ldots,C\_Y_m\}$。

综上所述，在不影响约简的前提下对平均决策强度进行改进可以简化计算；对一致与不一致对象的分离有助于搜索最优或次优约简。由此我们提出决策强度的代数定义。

定义 3.16 设 U 是一个论域，条件属性子集 $P\subseteq C$（$U/P=\{X_1,X_2,\ldots,X_t\}$），决策属性 $D=\{d\}$（$U/D=\{Y_1,Y_2,\ldots,Y_m\}$），且 $U/RD=\{C\_Y_0,C\_Y_1,C\_Y_2,\ldots,C\_Y_m\}$，则等价关系 RD 关于知识 P 的决策强度记为 $S(RD;P)$，定义为：

$$S(RD;P)=\sum_{i=1}^{t}\sum_{j=0}^{m}\left(\frac{|X_i\cap C\_Y_j|}{|X_i|}\times\frac{|X_i\cap C\_Y_j|}{|U|}\right)$$

$$=\sum_{i=1}^{t}\sum_{j=0}^{m}\left(\frac{|X_i\cap C\_Y_j|^2}{|X_i||U|}\right)。\tag{3-19}$$

定理 3.5 设 U 是一个论域，条件属性子集 $P\subseteq C$（$U/P=\{X_1,X_2,\ldots,X_t\}$），删除 P 中任意属性 a 后，可以通过将 U/P 中的部分划分块合并得到新划分 $U/(P-\{a\})$，故假设 $U/(P-\{a\})=\{X_1,X_2,\ldots,X_{p-1},X_{p+1},\ldots,X_{q-1},X_{q+1},\ldots,X_t,X_p\cup X_q\}$ 是将 U/P 中的两个划分块 X_p 与 X_q 合并为 $X_p\cup X_q$ 得到的新划分，且 $U/RD=\{C\_Y_0,C\_Y_1,C\_Y_2,\ldots,C\_Y_m\}$，则 $S(RD;P)\geq S(RD;P-\{a\})$。

证明：由公式(3)可知，$S(RD;P)=\sum_{i=1}^{t}\sum_{j=0}^{m}\left(\frac{|X_i\cap C\_Y_j|^2}{|X_i||U|}\right)$，

$$S(RD;P-\{a\})=S(RD;P)-\sum_{j=0}^{m}\left(\frac{|X_p\cap C\_Y_j|^2}{|X_p||U|}\right)$$

$$-\sum_{j=0}^{m}\left(\frac{|X_q\cap C\_Y_j|^2}{|X_q||U|}\right)+\sum_{j=0}^{m}\left(\frac{|(X_p\cup X_q)\cap C\_Y_j|^2}{|X_p\cup X_q||U|}\right)，$$

$$S_\Delta=S(RD;P)-S(RD;P-\{a\})$$

$$=\sum_{j=0}^{m}\left(\frac{|X_p\cap C\_Y_j|^2}{|X_p||U|}\right)+\sum_{j=0}^{m}\left(\frac{|X_q\cap C\_Y_j|^2}{|X_q||U|}\right)$$

$$-\sum_{j=0}^{m}\left(\frac{|(X_p\cap C\_Y_j)\cup(X_q\cap C\_Y_j)|^2}{|X_p\cup X_q||U|}\right)。$$

令 $|X_p|=x$，$|X_q|=y$，$|X_p\cap C\_Y_j|=ax$，$|X_q\cap C\_Y_j|=by$。

显然有 $x>0$，$y>0$，$0\leq a\leq 1$，$0\leq b\leq 1$，则

$$S_\Delta=\sum_{j=0}^{m}\frac{(ax)^2}{x|U|}+\sum_{j=0}^{m}\frac{(by)^2}{y|U|}-\sum_{j=0}^{m}\frac{(ax+by)^2}{(x+y)|U|}$$

$$=\sum_{j=0}^{m}\frac{xy(a-b)^2}{(x+y)|U|}=\frac{1}{|U|}\sum_{j=0}^{m}f_j=\frac{1}{|U|}\sum_{j=0}^{m}\frac{xy(a-b)^2}{x+y}。$$

对于任意 j（$j=0,1,\ldots,m$），有 $f_j=\dfrac{xy(a-b)^2}{x+y}$。显然，当 $a=b$ 时，函数 f_j 取最小值为 0。因此，当决策表删除某一条件属性 a 后，有 $S_\Delta\geq 0$，即 $S(RD;P)\geq S(RD;P-\{a\})$。

由定理 3.5 知，当 P 删除任意属性 a 后合并的划分块可能不仅仅是 X_p 与 X_q，由于选择 X_p 与 X_q 的任意性，对多个划分块的合并可以分解为两两划分块合并的过程。因此，删除条件属性 a 后，若有多个划分块合并，则等价关系 $P-\{a\}$ 在论域 U 上形成的新划分比划分 U/P 的粒度更大[21]。所以，知识的决策强度是随着信息粒度变小（通过更精细的划分）而非单调递减。

定理 3.6 设 U 是一个论域，P 为 U 上的一个条件属性子集，对任意属性 $a\in P$ 在 P 中是不必要的充要条件 $S(RD;P)=S(RD;P-\{a\})$。

由定理 3.5 可知该定理显然成立，证明略。

定义 3.17 在决策表 $S=(U,C,D,V,f)$ 中，条件属性子集 $P\subseteq C$，任意属性 $a\in C-P$ 的属性重要性定义为：

$$SGF(a,P,D)=S(RD;P\cup\{a\})-S(RD;P)。\tag{3-20}$$

特别当 $P=\varnothing$ 时，$SGF(a,\varnothing,D)=S(RD;\{a\})$。

$SGF(a,P,D)$ 的值越大，说明在已知 P 的条件下，属性 $a\in C-P$ 关于知识 P 就越重要。

利用定理 3.5 与定理 3.6，可以得到下面的属性约简判定定理。

定理 3.7 在决策表 $S=(U,C,D,V,f)$ 中，条件属性子集 $P\subseteq C$，若 $S(RD;P)=S(RD;C)$，且 P 的任何真子集 P^* 均满足 $S(RD;P^*)<S(RD;P)$，则 P 是 C 相对于决策 D 的一个属性约简。

3.5.3 基于决策强度的决策表约简设计

由上述理论分析可知，若把 $SGF(a,P,D)$ 作为搜索最优或次优知识约简的启发式信息时，每次循环计算条件属性子集 P 的 $S(RD;P)$ 均不变，那么求 $SGF(a,P,D)$ 最大的属性 a 就是求 $S(RD;P\cup\{a\})$ 最大的属性。所以，在计算 $SGF(a,P,D)$ 的过程中，就只需计算 $S(RD;P\cup\{a\})$，避免计算 $S(RD;P)$，不仅减少了计算量，而且也减小了搜索时间与空间，提高了算法的运行效率。具体的算法步骤：首先将决策等价类进一步细化与分离；其次以属性核为起点，自底向上逐步选择最重要的属性添加到核中，直到获取最小属性约简为止。

算法 3.8（基于决策强度的决策表约简算法）

输入：一个决策表 $S=(U,C,D,V,f)$；

输出：决策表 S 的一个相对约简 $P\subseteq C$。

步骤 1：计算决策表 S 的正区域 $POS_C(D)$与不一致对象集 $U-POS_C(D)$，并由此得到等价关系 RD；

步骤 2：计算条件属性集 C 相对于决策 D 的属性核 $CORE_D(C)$与决策强度 $S(RD;C)$；

步骤 3：初始化 $P=CORE_D(C)$，如果 $P=\varnothing$，那么转步骤 5；

步骤 4：如果 $S(RD;P)=S(RD;C)$，那么转步骤 8；

步骤 5：对任意属性 $a\in C-P$，计算 $S(RD;P\cup\{a\})$；

步骤 6：选择使 $S(RD;P\cup\{a\})$最大的属性 a，

如果有多个属性同时使 $S(RD;P\cup\{a\})$达到最大值，那么从中选取一个属性 a，使其与 P 的等价类基数$|U/(P\cup\{a\})|$最大，且 $P=P\cup\{a\}$；

步骤 7：如果 $S(RD;P)\neq S(RD;C)$，那么转步骤 5，

否则{ $P^*=P-CORE_D(C)$；

$r=|P^*|$；

$for(i=1;i\le r;i++)$

{ $a_i\in P^*$；

$P^*=P^*-\{a_i\}$；

如果 $S(RD;P^*\cup CORE_D(C))<S(RD;P)$，那么 $P^*=P^*\cup\{a_i\}$；

}

$P=P^*\cup CORE_D(C)$；

}

步骤 8：输出 P 为一个最小相对属性约简；

步骤 9：结束。

算法 3.8 中步骤 7 保证该属性约简算法是完备的[41]，即 P 不能再约简。使用文献[42]计算划分与文献[41]计算正区域及核的方法，经分析得到算法 3.8 总的最坏时间复杂度为 $O(|C|^2|U|)$，低于文献[67,71]约简算法的时间复杂度。

3.5.4 实验比较与分析

表 3-6 给出不一致决策表 $S=(U,C,D,V,f)$，其中 $U=\{x_1,x_2,\ldots,x_{10}\}$，$C=\{a_1,a_2,\ldots,a_5\}$，

$D=\{d\}$。

表 3-6　不一致决策表 S

U	a_1	a_2	a_3	a_4	a_5	d
x_1	0	0	0	1	0	0
x_2	0	0	1	0	1	0
x_3	1	0	0	0	0	0
x_4	0	0	1	0	1	1
x_5	1	0	0	0	1	0
x_6	1	1	1	1	0	1
x_7	0	0	1	0	0	0
x_8	0	1	1	1	0	1
x_9	1	0	0	0	0	1
x_{10}	0	0	1	0	0	1

对表3-6所示的不一致决策表，表3-7依次给出了文献[41]的约简算法(简称算法A)、文献[67]的CEBRKCC算法（简称算法B）和算法3.8求得的条件属性相对核的重要性和属性约简结果。

表 3-7　应用实例的约简算法比较（m 为条件属性集基数，n 为对象集基数）

约简算法	$C-\{a_5\}$相对核$\{a_5\}$的重要性				约简结果	算法的时间复杂度
	a_1	a_2	a_3	a_4		
算法 A	0.100	0.200	0.100	0	$\{a_1,a_2,a_3,a_5\}$	$O(m^2n\log(n))$
算法 B	0.089	0.204	0.165	0.014	$\{a_1,a_2,a_3,a_5\}$	$O(mn^2)+O(n^3)$
算法 3.8	0.200	0.240	0.200	0.267	$\{a_3,a_4,a_5\}$	$O(m^2n)$

由表3-7分析可知，算法A与算法B分别求属性a_4相对核的重要性较小，均获取约简结果为$\{a_1,a_2,a_3,a_5\}$，而不是最小约简结果$\{a_3,a_4,a_5\}$；算法3.8对于表3-6求得$SGF(a_4,\{a_5\},D)$最大，搜索结果是最小约简$\{a_3,a_4,a_5\}$。所以，基于决策强度的属性重要性更能准确地描述属性a_4的重要性，以其为启发式信息的约简方法更有可能获取最优或次优约简。与算法A和算法B相比，算法3.8不需要大量的数学计算，尤其是对数计算；文献[67]中采用试探法比较属性间的条件熵，而每个条件熵的计算复杂度是$O(n^2)$，当决策表有m个属性时，就需要计算2^m次条件熵，如果n与m值较大，算法B实现起来就显得比较困难了。这说明在不一致决策表中，本节算法3.8能够弥补基于正区域与信息论的决策表约简算法的不足。

3.5.5 UCI 离散数据集的属性约简比较

为了进一步验证，从文献[43]与文献[69]用到的UCI机器学习数据库中选择6个离散数据集进行约简比较，上述3种算法都使用文献[42]计算划分与文献[41]计算正区域

及核的方法，在 PC 机（P4 2.6G，256M RAM，WINXP）上使用 Java 语言来实现，其运行结果如表 3-8（m 与 n 分别为约简前后条件属性集基数，t 为执行时间/s）所示。

表 3-8　UCI 离散数据集的约简结果与执行时间比较

数据集	是否	对象	m	算法 A		算法 B		算法 3.8	
Database	一致	基数		n	t	n	t	n	t
Balloon(1)	是	20	4	2	0.04	2	0.05	2	0.03
Zoo	否	101	17	10	0.13	11	0.31	10	0.12
Voting-records	是	435	16	9	0.25	9	0.52	9	0.27
Tic-tac-toe	是	958	9	8	0.46	8	1.40	8	0.62
Chess end-game	是	3196	36	29	4.61	29	24.10	29	5.56
Mushroom	是	8124	22	5	6.40	4	17.56	4	6.58

3.5.6 小结

本节在决策表中，证明了知识的决策强度随着信息粒度变小而非单调递减的规律，设计了一种基于决策强度的启发式决策表约简方法，并用 UCI 属性离散数据集进行比较验证。该方法弥补了文献[41]中基于正区域的约简算法反映决策表“决策能力”的局限性，使搜索最优或次优知识约简的过程简单直观，且时间复杂度较低。实验分析的结果表明，该方法在效率上较现有知识约简算法有一定的提高，并且该研究在一定程度上扩展粗糙集理论及其应用领域。

第四章 基于粗糙集的决策树规则提取研究

粗糙集理论具有从决策表中抽取分类规则的能力，并且这个过程就是对决策表进行值约简的过程。决策表属性简化、决策规则简化是粗糙集理论及其实际应用的主要研究方向之一。属性约简只是信息系统数据简化的一部分，经过属性约简的信息系统仍然存在数据冗余。而决策规则简化就是值约简，值约简的任务是去掉多余的属性及其属性值，进一步简化信息系统。决策表中的每个实例都可以看作一条规则，其中可能包含冗余属性值，因此，对实例属性值的约简就是对决策规则约简，即值约简就是指对决策规则的提取。决策规则约简是指分别消去每条规则的不必要条件，不是整体上约简属性，而是针对每条决策规则去掉表达该规则的冗余属性值，以便使规则最小化。对于决策表而言，使其形式更简单，又尽可能地保留原决策表的信息。

4.1 现有值约简算法分析

在决策表中抽取规则的一般方法[6]：

（1）在决策表中将信息相同（即具有相同描述）的对象及其信息删除，只留其中一个，得到压缩后的信息表，即删除重复的实例；

（2）删除多余的属性；

（3）对每个实例删除多余的属性值；

（4）求出最小约简；

（5）根据最小约简，求出逻辑规则。

在粗糙集理论中，值约简（规则生成）算法的主要任务是根据条件划分与决策划分从信息系统中寻找反映特定概念的特征规则[91]。规则获取算法通常关注以下两个方面：

（1）生成的规则是否为某一概念的最小判别描述，即规则的条件部分在能够反映概念特征的前提下，应保持最简，不应有无关的冗余条件属性；

（2）生成的规则集所表达的知识是否完备。

这两个方面通常是相互关联的，但是，在采用某种方法剔除无关属性的过程中，可能导致一些重要属性被删除，从而影响规则的决策能力。

近年来，在属性值约简方面，有关文献报道相对较少。对于决策表知识约简，确定

性粗糙集模型是以一致规则为研究对象的，并没有涉及不一致规则的处理。文献[92]提出一种基于分类一致性的规则获取算法。文献[93]提出 RITIO 算法采用熵测度来度量决策表中条件属性与决策属性的相关性，逐步删除最不相关属性，从一致对象中提取规则，该方法抗噪声能力强，但规则前件较复杂，有冗余且可能有些规则会出现不一致[92]。文献[94]提出的属性值约简算法是从核值开始，对信息系统中的条件属性进行逐个考察，根据属性值对信息系统的不同影响作出不同标记，针对不同标记作不同处理，而且需要考虑某些属性值的恢复问题，实际操作很不方便，计算复杂度也较大。文献[95]指出利用文献[94]提出的值约简算法得到的规则，仍存在属性冗余与规则冗余，并给出了反例。文献[96]提出了知识库中面向规则的约简算法，先找到各条规则的核值，对核值规则进行检验，若该核值规则协调一致，则其为约简规则，否则对核值外的条件属性依次加入，直到找到所有约简规则。文献[96]整个算法的时间复杂度为 $O(|U|^2(|C|+2^{|C-CORE(\alpha\to\beta)|}))$，当条件属性集基数$|C|$的取值较大时，计算量也很大。文献[97]也指出现有的值约简算法仍存在一定的不足。为了解决以上这些问题，本章采用归纳学习中最有效的决策树[98]分类规则学习方法，但构造最优决策树已被证明是 NP-Hard 问题[56]。因此，在属性的选择过程中，采用更优的启发式函数来构造决策树，提取决策规则的方法显然具有重要的实用意义。

4.2 决策树技术

目前，对分类问题的研究主要集中在知识的模型表示上，如决策树、贝叶斯网络、概念格和粗糙集等。其中决策树是一种常用的分类模型，与其它分类模型相比，在构造决策树过程中，不需要除训练数据集之外的任何额外信息，训练速度比较快，分类精度比较高，易于理解。因此，决策树在实践中得到了广泛应用。

决策树是一种倒立的树型结构，是从一组无次序、无规则的实例中推理出树状形式表示的分类规则。它采用自顶向下的递归方式，根据一定标准选取属性作为决策树的内部节点，每个内部节点代表对一个属性取值的测试，并根据该属性的不同取值构造不同的分支，每个分支代表测试结果，进而在树的叶结点得到结论。因此，从根节点到叶结点所形成的每一条路径就对应一条分类规则，沿着该路径用对应的属性值对代替，便构成了分类规则的前件部分，叶子结点所标记的类别就构成了规则的结论部分。

总的来说，决策树的构造是一种自上而下、分而治之的归纳过程，本质上是一种贪

心算法[98-99]。决策树上的各个分支是在对数据不断分组的过程中逐渐生长出来的。首先，选择一个属性作为根节点，然后把该属性每一个可能值作为子节点，这样就把整个训练集分成了几个子集，根节点属性的每个取值都对应着一个子集，然后递归应用到每个子树上进行进一步划分，直到对所有数据的继续分组不再有意义时，决策树的生长过程宣告结束，此时便生成了一棵完整的决策树。其中，测试属性的选择是构造决策树的关键环节，不同决策树算法在此使用的技术都不尽相同。

1966 年，Hunt 等人提出了第一个可用于构造决策树的概念学习系统 CLS（Concept Learning System）；Quinlan 于 1983 年和 1993 年分别研制了 ID3 与 C4.5 算法；Richard 和 Charles 于 1984 年研制了 CART（Classification and Regression Tree）算法。

在实际应用中，数据中不可避免的会存在噪声和异常，决策树在建模的过程中既对正确数据建模，又对噪声和异常数据建模。为消除噪声和异常数据对决策树的影响，需要对生成的决策树进行剪枝。剪枝按照实施时间不同分为预剪枝和后剪枝。预剪枝是在决策树的构建过程中对每一个节点进行判断，如果符合某种预剪枝的标准，就停止树的构造，生成叶节点。后剪枝则是待决策树完全生成后，运用特定的剪枝算法对整棵决策树进行修剪。

在构造决策树过程中，选取一个属性作为当前节点，使它提供比其他属性多的系统相关信息，从根节点构建到各个叶结点，使系统变得越来越清晰。基于这种想法，本章结合粗糙集理论的优势，构造决策树，进而设计规则约简过程，简化获取的决策规则。

与 ID3 算法相比，基于粗糙集理论的决策树生成方法主要是通过划分对决策的近似质量来选择属性，即属性的选取应使系统中未完全分类的实例数下降最多，而 ID3 算法是基于信息论进行的，通过系统中信息量的增加来衡量属性的选取，并且它们生成的决策树在结构上也是不同的。在实践过程中，对许多实例而言，基于粗糙集理论构造的决策树要比基于信息熵的 ID3 算法生成的更简洁，并且具有较高的分类精度，这也是粗糙集理论的一个优势。

但是，在构造决策树的过程中，粗糙集理论不包含处理不精确或不确定原始数据的机制。因此，单纯地使用该理论不一定能有效地描述不确定或不精确的实际问题。这样生成的决策树虽然能精确地表达原数据集的信息，但不能保证对未知数据具有较好的泛化能力。由于数据受到噪声的影响，如果生成的决策树细化到连噪声都能精确匹配，那么就不能有效地表达原始数据的特性，无法起到有效预测未知数据类别的目的。为了更

好的处理实际数据中含有的各种不确定信息，在属性的选择过程中，采用更优的启发式函数来构造决策树，让模型具有一定的抗噪声能力，增强规则的泛化能力。

在粗糙集理论中，若决策表 S 是一致决策表，则决策树的各叶子结点只对应相同决策类的对象，即每个叶子结点对应的是确定性决策规则，其可信度等于 1；否则决策树的某些叶子结点对应不同决策类的对象，这样的叶子结点对应的是不确定性决策规则，其可信度小于 1。

4.3 基于新的条件熵的决策树规则提取方法[60]

由 3.3 节分析可知，经典的知识信息熵并没有完全客观地反映决策表决策能力的真实变化情况。在此基础上本节定义了一种新的条件熵概念；然后从优化决策树方面考虑，对传统启发式方法中选择属性的标准进行改进，由此定义新的属性重要性，并以新的属性重要性为启发式信息构造决策树；最后设计一个规则约简过程，简化所提取的决策规则。该方法的优点也是在构造决策树与提取决策规则前不进行属性约简，计算简单直观，时间复杂度较低。实验分析结果表明，该方法能提取更为简洁有效的决策规则。

4.3.1 现有信息熵的局限性

定义 4.1[71] 设 P 为一个条件属性集合，d 为决策属性，则 $Q \subseteq P$ 是 P 相对于决策属性 d 的一个约简的充分必要条件为：

（1）$H(\{d\}|Q) = H(\{d\}|P)$；

（2）对于 Q 中任意一个属性 r 都有 $H(\{d\}|Q) \neq H(\{d\}|Q-\{r\})$成立。

由文献[67]的基于条件信息熵的决策表约简算法分析可知，一个条件属性是否可以约简，取决于删除该条件属性后决策表 S 产生的条件熵是否改变。由于决策表 S 中一致对象集 $POS_C(D)$产生的条件熵为 0，所以决策表 S 的条件熵改变是由不一致对象集 $U - POS_C(D)$产生的，而决策表 S 删除某一条件属性后，产生新的不一致对象集属于各决策属性分类的概率分布改变，就会引起条件熵发生变化。因而，现有基于条件信息熵的约简算法不仅考虑到产生的确定性决策规则数目不变，又考虑到产生的不确定性决策规则的可信度不变。我们知道若决策表 S 产生确定性决策规则的数目不变，就意味着这些决策规则的可信度不变（可信度仍为 1）。但是，在决策应用中，决策规则除了其可信度外，规则对样本（对象）的覆盖度也是衡量其“决策能力”的重要指标。因此，文献[67]的约简算法存在局限性，未能真实客观地反映决策能力变化的实质。

4.3.2 新的条件熵

定义 4.2[98] 设 U 是一个论域，属性集合 R（$U/R=\{R_1,R_2,\ldots,R_m\}$）的信息熵定义为：

$$E(R)=\sum_{i=1}^{m}\frac{|R_i|}{|U|}(1-\frac{|R_i|}{|U|}), \tag{4-1}$$

其中$|R_i|/|U|$表示 R_i 在论域 U 上的概率。

为了研究能够体现对象覆盖度的知识信息熵，我们可以引入 3.3 节中的引理 3.1。这样，在决策表 S 中，两个属性集合 $P\cup D$（$P\subseteq C$）的信息熵可有如下定义。

定义 4.3 在决策表 $S=(U,C,D,V,f)$中，条件属性集合 $P\subseteq C$，$U/P=\{X_1,X_2,\ldots,X_n\}$，决策属性 $D=\{d\}$，$U/D=\{Y_1,Y_2,\ldots,Y_m\}$，则属性集合 $P\cup D$ 的信息熵定义为：

$$E(P\cup D)=\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{|X_i\cap Y_j|}{|U|}(1-\frac{|X_i\cap Y_j|}{|U|})。 \tag{4-2}$$

在现有条件信息熵的定义中，$p(Y_j|X_i)=|X_i\cap Y_j|/|X_i|$可以代表决策表产生某一规则的可信度，而在 $P\cup D$ 的信息熵定义中，$|X_i\cap Y_j|/|U|$可以代表该决策规则的对象覆盖程度。这样我们可以把两种信息熵的定义结合起来，使其更能客观地反映决策表 S “决策能力”的实质。在此基础上，我们提出了一种信息论定义形式——新的条件熵。

定义 4.4（新的条件熵） 在决策表 $S=(U,C,D,V,f)$中，$P\subseteq C$ 是 U 上的一个条件属性集合，决策属性 $D=\{d\}$，则 P 关于决策属性 d 的新的条件熵记为 $H(D;P)$，定义为：

$$H(D;P)=H(D|P)-E(P\cup D)。 \tag{4-3}$$

4.3.3 基于新的条件熵的决策树规则提取

在各种构造决策树方法中，比较有影响的是 Quinlan 提出的 ID3 算法[98]。它用信息增益作为在各级非叶节点上选择属性的标准，来获得对象集最大的类别信息。但这种方法并不是最优的，即决策树的节点不是最少的。这种启发式方法往往偏向于选择属性取值较多的属性，而属性值较多的属性却不总是最优的属性，并且 ID3 学习简单的逻辑表达式能力较差[56]。本节针对这些问题提出如下改进方案：在构造决策树的过程中，改进各级非叶节点属性的选择标准，重点考虑决策树中不同分支节点上属性重要性的计算，避免 ID3 算法中子树重复和某些属性被多次选择的缺点，便于得到更优的决策树。

定义 4.5（新的属性重要性） 设 U^*为论域 U 上与决策树某分支节点相关的对象集（若 U^*为决策树根节点相关的对象集，那么 $U^*=U$。），C 为条件属性集，$D=\{d\}$为决

策属性集，$B\subseteq C$，则任意属性 $a\in C-B$ 关于 U^*的属性重要性定义为：

$$SGF(a,B,U^*,D)=H(D;B)-H(D;B\cup\{a\})。\tag{4-4}$$

特别当 $B=\varnothing$ 时，$SGF(a,\varnothing,U^*,D)=-H(D;\{a\})$。

$SGF(a,B,U^*,D)$的值越大，说明在已知 B 的条件下，属性 $a\in C-B$ 关于知识 B 就越重要。在计算 $SGF(a,B,U^*,D)$的过程中，每次循环时条件属性子集 B 的 $H(D;B)$均不变，这使得$SGF(a,B,U^*,D)$最大的属性a就是$H(D;B\cup\{a\})$最小的属性。因此，把$SGF(a,B,U^*,D)$作为搜索最小或次优知识约简的启发式信息时，只需计算 $H(D;B\cup\{a\})$，就可以省去计算 $H(D;B)$，减少了计算量，进而减小了搜索空间与时间。由此可见，以 $SGF(a,B,U^*,D)$为启发式信息的约简算法，必须计算 $H(D;B\cup\{a\})$。为降低该方法的时间复杂度，就需要研究计算 $H(D;B\cup\{a\})$的高效算法，由文献[67]中的定理 1 可得到算法 4.1 如下：

算法 4.1（计算 $H(D;B\cup\{a\})$的算法）

输入：决策表 $S=(U^*,C,D,V,f)$和 $B\subseteq C$；

输出：划分 $U^*/(D\cup B\cup\{a\})$和 $H(D;B\cup\{a\})$。

步骤 1：计算划分 $U^*/(B\cup\{a\})$和 $U^*/(D\cup B\cup\{a\})$；

步骤 2：计算 $H(B\cup\{a\})$，$H(D\cup B\cup\{a\})$和 $E(D\cup B\cup\{a\})$；

步骤 3：计算 $H(D;B\cup\{a\})=H(D\cup B\cup\{a\})-H(B\cup\{a\})-E(D\cup B\cup\{a\})$；

步骤 4：输出 $H(D;B\cup\{a\})$和划分 $U^*/(D\cup B\cup\{a\})$；

步骤 5：结束。

用文献[42]中计算划分的方法，步骤 1 的时间复杂度为 $O((|B|+2)|U|)$，步骤 2 的时间复杂度为 $O(|U|)$，因而算法 4.1 总的最坏时间复杂度为 $O(|C||U|)$。

在算法 4.1 的基础上，从空树 T 开始，以 $SGF(a,B,U^*,D)$最大的属性 a 为分支节点（包括根节点），也就是选择 $H(D;B\cup\{a\})$最小的属性 a 为分支节点，自顶向下递归构造决策树。我们在选择一个新的属性时，不仅要考虑它基于所有对象集是最重要的，而且要考虑它在相关对象集上也是最重要的，这样就能有效改进选择新属性的启发式函数，达到更好的分类效果，弥补 ID3 算法容易导致决策树中子树重复与某些属性在同一决策树中被多次选择的不足，得到更优的决策树。具体操作步骤如下：

算法 4.2（构造决策树的算法）

输入：对象集 U^*，条件属性集 C，决策属性集 $D=\{d\}$；

输出：最简决策树 T。

步骤 1：合并 U^*中的相同对象；

步骤 2：初始化 $B=\varnothing$，T 为空树；

步骤 3：对任意属性 $a\in C-B$，计算 $H(D; B\cup\{a\})$；

步骤 4：选择使 $H(D; B\cup\{a\})$最小的属性 a 为决策树 T 的根节点(或分支节点)，

（1）如果有多个属性同时使 $H(D; B\cup\{a\})$达到最小值，那么从中选取使 $U^*/(D\cup B\cup\{a\})$构成等价类最少的属性 a，即 $D\cup\{a\}$构成的等价类能够覆盖更多的对象；

（2）如果仍有多个属性使$|U^*/(D\cup B\cup\{a\})|$达到最小值，

那么选择顺序靠前的属性；

步骤 5：用选择的属性 a 对 U^*进行分类，即计算 $U^*/\{a\}=\{U_1^*,U_2^*,\ldots,U_t^*\}$，开始建立子决策表（即决策树的分支）$S_i=(U_i^*,C,D,V,f)$，其中 $i=1,2,\ldots,t$；

步骤 6：如果分支 S_i（$i=1,2,\ldots,t$）中 U_i^*的所有对象具有相同的决策属性值，那么在分支 S_i下生成一个叶子结点，标识其决策属性值，否则 $B=B\cup\{a\}$；

步骤 7：如果 $B=C$ 或 U^*被决策树分支完全分类，那么输出决策树 T，算法结束，否则转步骤 3；

步骤 8：结束。

用算法 4.1 的方法，可得步骤 3 到步骤 7 总的最坏时间复杂度为 $O(|C||U|)+O((|C|-1)|U|)+O((|C|-2)|U|)+\ldots+O(|U|)=O(|C|^2|U|)$，因而算法 4.2 最坏的时间复杂度为 $O(|C|^2|U|)$。

在决策树中，每个叶子结点就是一个分类，从根到叶子结点对应一条分类规则。下面我们在算法 4.2 的基础上，设计一个规则约简过程，用来简化所提取的决策规则。

算法 4.3（规则提取算法）

步骤 1：遍历决策树每个分支中根到叶子结点的所有路径，生成决策规则集；

步骤 2：简化决策树分支中的每一条决策规则，

如果该决策规则中的任一非叶节点去掉后，在所属分支中仍能唯一表示，那么继续去掉第 2,3,...个非叶节点，直到不能在所属分支中唯一表示；

步骤 3：输出最小决策规则集；

步骤 4：结束。

分析可得算法 4.3 最坏的时间复杂度为 $O(|C||U|)$。

算法 4.3 中步骤 2 对循环提取的原始决策规则进行简化，删除所有不影响规则表达的冗余条件属性及其属性值，简化决策规则，这就能保证所提取的决策规则最小，即规则所含的属性及其属性值最少，且在约简表中唯一表示。

4.3.4 实验结果

表 4-1 给出了一致决策表 $S=(U,C,D,V,f)$，其中 $U=\{1,2,...,14\}$，$C=\{a_1,a_2,a_3,a_4\}$，$D=\{d\}$。

表 4-1　一致决策表 S

论域 U		1	2	3	4	5	6	7	8	9	10	11	12	13	14
条件属性集 C	a_1	1	1	2	3	3	3	2	1	1	3	1	2	2	3
	a_2	1	1	1	2	3	3	3	2	3	2	2	2	1	2
	a_3	1	1	1	1	0	0	0	1	0	0	0	1	0	1
	a_4	0	1	0	0	0	1	1	0	0	0	1	1	0	1
决策属性 d		0	0	1	1	1	0	1	0	1	1	1	1	1	0

用一致决策表 4-1 来验证上述算法的有效性，可以得到一棵与最小确定性决策规则集（见表 4-2）对应的最小决策树（见图 4-1）。

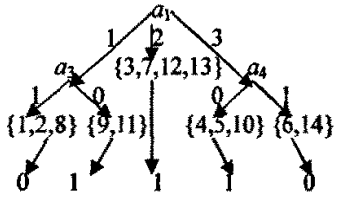

图 4-1　决策树

表 4-2　最小确定性决策规则集

序号	决策规则
1	$(a_1,1)\wedge(a_3,1)\rightarrow(d,0)$
2	$(a_1,1)\wedge(a_3,0)\rightarrow(d,1)$
3	$(a_1,2)\rightarrow(d,1)$
4	$(a_1,3)\wedge(a_4,0)\rightarrow(d,1)$
5	$(a_1,3)\wedge(a_4,1)\rightarrow(d,0)$

由图 4-1 可知，算法 4.2 可得到与文献[98]相同的单变量决策树。对于表 4-1 所示的一致决策表，文献[93]中 RITIO 算法得到的规则集共有 7 条规则，其中有一条规则是不一致的，它与表 4-1 的第 6 个对象矛盾，文献[100]中 LEM2 算法对于表 4-1 得到的规则集也是 7 条规则，以上两种算法得到的规则集均比算法 4.3 得到的规则集数目多；对于不一致决策表，在算法 4.2 得到的决策树中，不一致对象对应的决策属性值为两个，算法 4.3 得到的不一致对象对应简化后的不确定性决策规则的可信度均小于 1。

4.3.5 小结

本节提出一种决策树规则提取方法，它以新的条件熵来度量属性重要性。该方法具有如下特点：

（1）弥补了现有信息熵反映决策表“决策能力”的局限性；

（2）改进了传统启发式方法中选择属性的标准；

（3）不需要在构造决策树阶段前进行属性约简；

（4）设计了一个规则约简过程来简化决策规则，增强规则的泛化能力。

实例分析的结果表明，该方法不仅有助于进一步加深对粗糙集理论中规则提取算法的认识，也有助于时效性更优算法的推广使用。

需要指出的是，该方法没有考虑到在大型数据集分析中，数据测量的误差、数据获取能力的不足、噪声干扰等原因，在一定程度上可能制约其处理复杂应用问题的有效性。

4.4 基于决策熵的决策树规则提取方法

由 4.3 节分析可知，在决策应用中，规则可信度与对象覆盖度都是衡量决策能力的重要指标，但是粗糙集理论中知识粗糙熵并没有完全客观地反映决策表决策能力的变化情况。在此基础上，本节针对现有规则获取方法中存在的问题，分析了知识粗糙熵的局限性，提出一种新的粗糙熵定义——决策熵，并定义其属性重要性；然后以条件属性子集的决策熵来度量其对决策分类的重要性，选择决策熵最小且涵盖最多决策分类对象的属性为分枝节点，自顶向下递归构造决策树；最后遍历决策树，简化所获得的决策规则。该方法计算直观，时间复杂度较低。理论分析和实例比较结果表明，该方法是有效的。

4.4.1 现有粗糙熵的局限性与决策熵概念

定义 4.6[88] 设 U 是一个论域，属性集合 R 在 U 上导出的划分 $U/R=\{R_1,R_2,\ldots,R_m\}$，则 R 在 U 上导出划分 U/R 的粗糙熵定义为：

$$E(R)=\sum_{i=1}^{m}\frac{|R_i|}{|U|}\log|R_i|, \tag{4-5}$$

其中 $|R_i|/|U|$ 表示 R_i 在论域 U 上的概率。

为了研究能够体现对象覆盖度的知识粗糙熵，我们可以引入 3.3 节中的引理 3.1。这样，在决策表 S 中，属性集合 $P\cup D$（$P\subseteq C$）的粗糙熵可有如下定义。

定义 4.7 设 U 是一个论域，条件属性集合 P 在 U 上导出的划分 $U/P=\{X_1,X_2,\ldots,X_n\}$，决策概念集 $D=\{d\}$，$U/D=\{D_1,D_2,\ldots,D_t\}$，则属性集合 $P\cup D$ 的粗糙熵定义为：

$$E(P\cup D)=\sum_{i=1}^{n}\sum_{j=1}^{t}\frac{|X_i\cap D_j|}{|U|}\log|X_i\cap D_j|。 \tag{4-6}$$

在 $P\cup D$ 的粗糙熵定义中，$|X_i\cap D_j|/|U|$ 代表了某一决策规则的对象覆盖度，所以该粗

糙熵定义就反映了决策表“决策能力”变化的一个重要指标。

定义 4.8[90] 设 U 是一个论域，P（$U/P = \{X_1,X_2,\ldots,X_n\}$）为一个条件属性集合，决策概念集 $D=\{d\}$，$U/D=\{D_1,D_2,\ldots,D_t\}$，则决策概念集 D 的粗糙熵定义为：

$$E(D_P)=-\sum_{i=1}^{n}\sum_{j=1}^{t}\frac{|X_i|}{|U|}\log\frac{|X_i\cap D_j|}{|X_i|}。\tag{4-7}$$

由定义 4.8 知，在条件属性集合的划分 $U/P = \{X_1,X_2,\ldots,X_n\}$ 中，存在两种情况：$x\in X_i\subseteq D_j$ 和 $x\in X_i\propto D_j$。其中 $i=1,2,\ldots,n$，$j=1,2,\ldots,t$。在前一种情况下，D_j 中的 x 是完全可确定的。因此，我们只需考虑 $x\in X_i\propto D_j$ 的情况，决策概念集 D 关于知识 P 的粗糙熵可简化为：

$$E(D_P)=-\sum_{i=1}^{n}\sum_{j=1}^{k}\frac{|X_i|}{|U|}\log\frac{|X_{ij}|}{|X_i|},\tag{4-8}$$

其中，$X_{i1},X_{i2},\ldots,X_{ik}$（$k\le t$）是 X_i 与 $D_1,D_2,\ldots,D_t$ 的非空交集。

在决策概念集的粗糙熵定义中，$|X_i\cap D_j|/|X_i|$代表了决策表所产生某一决策规则的可信度。这样我们可以把两种粗糙熵的定义结合起来，使其完全客观地反映决策表决策能力的真实变化情况。由此我们提出了一种新的粗糙熵信息论定义形式——决策熵。

定义 4.9 设 U 是一个论域，P 是 U 上的一个条件属性集合，$D=\{d\}$为决策概念集，则 P 关于决策 D 的决策熵记为 $E(D|P)$，定义为：

$$E(D|P)=E(D_P)+E(P\cup D)。\tag{4-9}$$

定义 4.10 在决策表 $S=(U,C,D,V,f)$中，条件属性子集 $B\subseteq C$，任意属性 $a\in C-B$ 的属性重要性定义为：

$$SGF(a,B,D)=E(D|B)-E(D|B\cup\{a\})。\tag{4-10}$$

特别当 $B=\varnothing$ 时，$SGF(a,\varnothing,D)=-E(D|\{a\})$。

$SGF(a,B,D)$的值越大，说明在已知 B 的条件下，属性 $a\in C-B$ 关于知识 B 就越重要。在计算 $SGF(a,B,D)$的过程中，每次循环时 B 的 $E(D|B)$均不变，那么求 $SGF(a,B,D)$最大的属性 a 就是求 $E(D|B\cup\{a\})$最小的属性 a。所以，若把 $SGF(a,B,D)$作为搜索最小或次优知识约简的启发信息时，就只需计算 $E(D|B\cup\{a\})$，减少计算量，减小搜索空间。

4.4.2 基于决策熵的决策树规则提取

决策树是用树形结构来表示决策集合，这些决策集合通过对数据集的分类产生决策规则。下面以知识决策熵的属性重要性为启发式信息来设计值约简方法。首先从空树 T

开始，逐步加入条件属性，选择最小的知识粗糙熵，以涵盖最多决策分类对象的属性为分枝节点，自顶向下递归构造决策树；然后根据分块处理的思想，尽量以少的属性提取隐含在决策表中有用的决策规则；最后删除所有不影响规则表达的冗余条件属性及其属性值，简化决策规则。该方法的具体操作步骤描述如下：

算法 4.4（基于决策熵的决策树规则提取算法）

输入：决策表 $S=(U,C,D,V,f)$；

输出：最简决策树 T 和决策规则集。

步骤 1：合并决策表 S 中的相同对象；

步骤 2：初始化 $B=\varnothing$，T 为空树；

步骤 3：对任意属性 $a\in C-B$，计算 $E(D|B\cup\{a\})$；

步骤 4：选择使 $E(D|B\cup\{a\})$最小的属性 a 为决策树 T 的根节点（或分支节点），

（1）如果有多个属性同时使 $E(D|B\cup\{a\})$达到最小值，

那么从中选取一个属性 a 使其与 B 的依赖性 $\gamma_{B\cup\{a\}}(D)$最大；

（2）如果仍有多个属性使 $\gamma_{B\cup\{a\}}(D)$达到最大值，那么选顺序靠前的属性；

步骤 5：用选择的属性 a 对 U 进行分类，即计算 $U/\{a\}=\{U_1,U_2,\ldots,U_t\}$，开始建立子决策表（即决策树的分支）$S_i=(U_i,C,D,V,f)$，其中 $i=1,2,\ldots,t$；

步骤 6：如果分支 S_i（$i=1,2,\ldots,t$）中的所有对象 U_i 具有相同的决策属性值，那么在分支 S_i 下生成一个叶子结点，标识其决策属性值，遍历根到该叶子结点的一条路径，产生相应的决策规则，如果该决策规则中的任一非叶节点去掉后，在 S_i 中仍能唯一表示，那么继续去掉第 2,3,...个非叶节点，直到不能在 S_i 中唯一表示，否则 $B=B\cup\{a\}$；

步骤 7：如果 $B=C$ 或 U 被决策树分支完全分类，

那么输出决策树 T 与决策规则集，算法结束，否则转步骤 3；

步骤 8：结束。

用文献[42]中计算划分与正区域的方法，分析可得步骤 3 到步骤 7 最坏的时间复杂度为 $O(|C|^2|U|)$，因而算法 4.4 总的最坏时间复杂度为 $O(|C|^2|U|)$。

与文献[98]多变量决策树构造方法相比，算法 4.4 得到的是单变量决策树。其中步骤 4 考虑了属性之间的依赖关系，易于消除冗余属性；步骤 5 采用分块处理的方法，弥补了 ID3 算法容易导致决策树中子树重复与某些属性在同一决策树中被多次选择的不

足；步骤 6 对循环提取的原始决策规则进行化简，删除所有不影响规则表达的冗余条件属性及其属性值，这就保证了所提取的决策规则最小，即包含条件属性及其属性值最少，且在约简表中唯一表示。

4.4.3 应用实例分析与比较

表 4-3 给出了一致决策表 $S=(U,C,D,V,f)$，其中 $U=\{1,2,...,14\}$，$C=\{a_1,a_2,a_3,a_4\}$，$D=\{d\}$。

表 4-3　一致决策表 S

U	1	2	3	4	5	6	7	8	9	10	11	12	13	14
a_1	1	1	2	0	0	0	2	0	1	1	0	2	2	1
a_2	1	1	1	2	0	0	0	2	0	2	2	2	1	2
a_3	1	1	1	1	0	0	0	0	0	1	1	1	0	0
a_4	0	1	0	0	0	1	1	0	0	0	1	1	0	1
d	0	0	1	1	1	0	1	1	1	0	0	1	1	1

用一致决策表 4-3 来验证算法 4.4 的有效性，可得到一棵与最小确定性决策规则集（见表 4-4）对应的最小决策树（见图 4-2）。

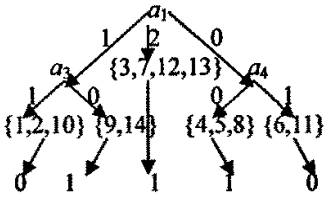

图 4-2　决策树

表 4-4　最小确定性决策规则集

序号	决策规则集	可信度	覆盖度
1	$(a_1,1)\wedge(a_3,1)\rightarrow(d,0)$	1	3/14
2	$(a_1,1)\wedge(a_3,0)\rightarrow(d,1)$	1	2/14
3	$(a_1,2)\rightarrow(d,1)$	1	4/14
4	$(a_1,0)\wedge(a_4,0)\rightarrow(d,1)$	1	3/14
5	$(a_1,0)\wedge(a_4,1)\rightarrow(d,0)$	1	2/14

对于表 4-3 所示的一致决策表，文献[93]中 RITIO 算法共得到 7 条规则，其中有一条规则是不一致的，它与表 4-3 的第 6 个对象矛盾，文献[100]中 LEM2 算法也可得到 7 条规则；但以上两种算法得到规则集数目均比表 4-4 所示规则集多。对于不一致决策表而言，由算法 4.4 得到的决策树，不一致对象对应的决策属性值有两个。

4.4.4 小结

在决策表中，本节以知识决策熵的属性重要性为启发信息，自顶向下递归构造决策树，然后遍历决策树，并简化所获得的决策规则。实例分析的结果表明，该算法为从决策表中搜索最小决策规则提供了一种有效的方法。

第五章 结 论

5.1 粗糙集理论的前景

粗糙集理论特别适合数据约简、数据的近似分类、数据相关性发现等，而且在实践中已经被证实是非常有效的。这一理论对 AI 和认知科学尤为重要。其已经构成 KDD 的一个完备基础，说明它也是分布式和多 Agent 系统中数据挖掘的新方法。为智能 Agent 在分布式环境下对个体进行综合与分析的 Rough Mereology 已被用作研究信息 Granule 计算的基础，正向词计算的公式化方向发展。在数据挖掘与知识发现中，可利用各种知识源与各种知识结构的有利条件探讨混合系统中的新方法，也就是说，粗糙集方法与模糊集、神经网络、进化计算、统计推理、证据理论、置信网络等方法的结合。同时如何将粗糙集理论、模糊集理论、证据理论与概率论等不确定理论用一个统一的逻辑模型来解释也很值得关注。目前，粗糙集理论中值得重视的课题主要包括基于粗糙集的粗糙逻辑研究、粗糙函数理论与实践研究、粗糙控制理论研究、基于粗糙集的神经网络与遗传算法研究等。从数据库知识发现的角度来说，高效约简方法、大数据集问题、多方法融合、增量算法、信息不一致问题等也是很有价值的研究方向。在数据挖掘与软计算方面，对海量数据挖掘的粗糙集方法已经提出，特别是处理大型数据库与复杂问题等方面将具有广阔的发展及应用前景。由此可见，以上这些方面的研究与开发将是粗糙集理论发展的关键，也是 KDD 技术研究的中心问题。

5.2 不足之处与今后研究设想

本论文分析了在知识约简过程中经典粗糙集理论知识约简方法的不足：

（1）由于不一致对象的存在，基于正区域和信息论方法无法等价地表示知识约简；

（2）决策表的不一致性导致差别矩阵的多种定义；

（3）没有完全客观地反映决策表决策能力的变化情况；

（4）约简算法的时间复杂度比较高。

针对上述不足，3.2 节为克服区分矩阵方法时间复杂度随系统大小增加而呈指数增长的缺陷，定义一种新的属性重要性，给出分布约简的数学判定定理；3.3 节和 3.4 节提

出了新的条件熵概念和决策熵定义，给出以不等式为条件的约简判定定理，以弥补现有信息论约简算法的局限性；3.5 节给出决策强度的代数定义，以弥补正区域方法的不足，证明知识的决策强度随着信息粒度变小而非单调递减的规律，并用 UCI 离散数据集进行实验比较。针对文献[95,97]指出现有值约简算法存在的不足，采用归纳学习中效率高且实用性强的决策树分类规则学习方法，在属性选择中，采用更优的启发式函数来构造决策树，提取决策规则；4.3 节为弥补现有信息熵的不足，定义一种新的条件熵概念，对传统启发式方法中选择属性的标准进行改进，在新的属性重要性基础上设计决策树规则提取方法；4.4 节分析了现有知识粗糙熵的局限性，提出了知识决策熵的概念，并以条件属性子集的决策熵来度量其对决策分类的重要性，选择决策熵最小且涵盖最多决策分类对象的属性为分枝节点，自顶向下递归构造决策树，遍历决策树，简化所获取的决策规则。实验比较与分析的结果表明，与现有属性约简和规则提取算法相比，本文提出的方法在一定程度上提高了运行效率，节省了搜索空间与时间。

本文不足之处是，提出的约简算法还没有完全在计算机上实现，正努力设计仿真实验平台；另外对扩展的粗糙集研究不够，只针对完备决策系统约简进行了一些研究，未进行其它扩展方向的研究，如与模糊集、神经网络、多 Agent 的结合；对人工智能与数据挖掘中其他研究方法的了解也有待进一步探索。

下面对今后工作提出几点研究设想：

（1）粗糙集理论的精髓在于知识约简，能否建立新的模型将其拓展到连续性数据。

（2）新的属性重要性度量是否已充分利用了决策系统中蕴涵的信息，能否进一步挖掘其中的信息，以获取更高效的属性约简算法。

（3）针对目前差别矩阵多种定义的情况，能否寻找差别矩阵的统一模型，高效获取决策系统的所有约简。

（4）我们只是针对静态系统展开讨论和研究，对决策系统的增量式学习将是今后进一步研究的方向。

（5）将多 Agent 的思想应用于粗糙集的分类研究。

（6）将粒计算的思想应用于决策系统规则的高效提取研究。

（7）将 Fuzzy 集、Vague 集与 Rough 集相结合，建立相似性度量方法的统一模型。

参考文献

[1] Zadeh L A. Fuzzy sets[J]. Information and control, 1965, 8: 338-359.

[2] Pawlak Z. Rough sets[J]. International Journal of Computer and Information Science, 1982, 11(5): 341-356.

[3] Pawlak Z. Rough Sets-Theoretical Aspects of Reasoning about Data[M]. Boston: Kluwer Academic Publishers, 1991: 72-80.

[4] 王志海, 胡可云, 胡学纲, 等. 基于粗糙集合理论的知识发现综述[J]. 模式识别与人工智能, 1998, 11(2): 176-183.

[5] 张文修, 吴伟志. 粗糙集理论介绍和研究综述[J]. 模糊系统与数学, 2000, 14(4): 1-12.

[6] 王珏, 苗夺谦, 周育健. 关于 Rough Set 理论与应用的综述[J]. 模式识别与人工智能, 1996, 9(4): 337-344.

[7] 刘清, 黄兆华, 姚力文. Rough 集理论:现状与前景[J]. 计算机科学, 1997, 24: 1-5.

[8] 韩祯祥, 张琦, 文福拴. 粗糙集理论及其应用综述[J]. 控制理论与应用, 1999, 16(2): 153-157.

[9] Slowinski R. Intelligent Decision Support-Handbook of Applications and Advances of the Rough Set Theory[M]. Dordrecht: Kluwer Academic Publishers, 1992: 49-60.

[10] Pawlak Z, Grzymala-Busse J, Slowinski R, et al. Rough sets[J]. Communications of the ACM, 1995, 38(11): 89-95.

[11] 曾黄麟. 粗糙集理论及其应用-关于数据推理的新方法[M]. 重庆: 重庆大学出版社, 1996: 8-28.

[12] 张文修, 吴伟志, 梁吉业, 等. 粗糙集理论与方法[M]. 北京: 科学出版社, 2001: 1-25, 57-97.

[13] 王国胤. Rough 集理论与知识获取[M]. 西安: 西安交通大学出版社, 2001: 1-55, 168- 217.

[14] 刘清. Rough 集及 Rough 推理[M]. 北京: 科学出版社, 2001: 1-75, 194-241.

[15] 张文修, 梁怡, 吴伟志. 信息系统与知识发现[M]. 北京: 科学出版社, 2003: 42-55, 90-95.

[16] 张文修, 姚一豫, 梁怡. 粗糙集与概念格[M]. 西安: 西安交通大学出版社, 2006: 23-60.

[17] 梁吉业, 李德玉. 信息系统中的不确定性与知识获取[M]. 北京: 科学出版社, 2005.

[18] 张文修, 仇国芳. 基于粗糙集的不确定决策[M]. 北京: 清华大学出版社, 2005.

[19] Ziarko W. Rough sets: Trends, challenges, and prospects[C]. In: Ziarko W, Yao Y Y (eds). Rough Sets and Current Trends in Computing (RSCTC 2000). Berlin: Springer-Verlag, 2001: 1-7.

[20] Skowron A. Rough sets and boolean reasoning[C]. In: Pedrycz w (ed). Granular Computing: An Emerging Paradigm. New York: Physicar-Verlag, 2001: 95-124.

[21] Xu Jiu Cheng, Shen Jun Yi, Wang Guo Yin. Rough set theory analysis on decision subdivision[J].

Fourth International Conference on Rough Sets and Current Trends in Computing 2004(RSCTC'2004), Uppsala, Sweden, 2004, 340-345.

[22] 梁吉业. 关于粗糙集度量与粗糙计算方法的研究[D]. 西安: 西安交通大学, 2001.

[23] 安秋生. 粗糙集与信息颗粒原理在数据库理论中的应用[D]. 西安: 西安交通大学, 2003.

[24] Banerjee M, Pal S K. Roughness of a fuzzy set[J]. Joural of Information Sciences, 1996, 93(3/4): 235-246.

[25] Yao Y Y. A comparative study of fuzzy sets and rough sets[J]. International Journal of Information Sciences, 1998, 109: 227-242.

[26] 徐久成. 粗糙集理论与不确定信息的处理及度量研究[D]. 西安: 西安交通大学, 2003.

[27] Wu Wei Zhi, Mi Ju Sheng, Zhang Wen Xiu. Generalized fuzzy rough sets[J]. International Journal of Information Sciences, 2003, 151: 263-282.

[28] Lin T Y. Granular computing on binary relations Ⅱ: Rough set representations and belief functions[C]. In: Skowron A, Polkowski L (eds). Rough Sets In Knowledge Discovery. Heidelberg(Germany): Springer-Verlag, 1998: 121-140.

[29] Skowron A, Stepaniuk J. Information granules: Towards foundations of granular computing[J]. International Journal of Intelligent Systems, 2001, 16: 57-85.

[30] Liu Qing, Sun Hui. Theoretical study of granular computing[J]. RSKT2006, 2006, 93-102.

[31] Pawlak Z. Rough sets and intelligent data analysis[J]. International Journal of Information Sciences, 2002, 147: 1-12.

[32] Cattaneo G, Ciucci D. Algebraic structures for rough sets[J]. Lecture Notes in Computer Science, 2004, 3135: 208-252.

[33] Skowron A, Polkowski L. Analytical morphology: Mathematical morphology of decision tables[J]. Fundamenta Informaticae, 1996, 27(2/3): 255-271.

[34] Liu Qing. The OI-resolutions of operator rough logic[J]. In: Proc 1st International Conference, RSCTC'98, Warsaw, Poland, 1998, 432-436.

[35] Polkowski L. On Convergence of Rough Sets[M]. Intelligent Decision Support Handbook of Applications and Advances of Rough Sets Theory, Dordrecht, Kluwer Academic, 1992: 305-311.

[36] Yao Y Y, Lingras P J. Interpretations of belief functions in the theory of rough sets[J]. Joural of Information Sciences, 1998, 104(1/2): 81-106.

[37] Beynon M. Reducts within the variable precision rough sets model: A further investigation[J]. European Journal of Operational Research, 2001, 134(3): 592-605.

[38] Slowinski R, Vanderpooten D. A generalized definition of rough approximations based on similarity[J]. IEEE Transactions on Knowledge and Data Engineering, 2000, 12(2): 331-336.

[39] Chmielewski M R, Grzymala-Busse J W. Global discretization of continuous attributes as

preprocessing for machine learning[J]. In Third International Workshop on Rough Sets and Soft Computing, 1994, 294-301.

[40] 蒋思宇. 新的决策表约简模型下的一种增量算法[J]. 计算机工程与应用, 2005, 28: 21-25.

[41] 刘少辉, 盛秋戬, 吴斌, 等. Rough 集高效算法的研究[J]. 计算机学报, 2003, 26(5): 524-529.

[42] 徐章艳, 刘作鹏, 杨炳儒, 等. 一个复杂度为 max(O(|C||U|), O(|C|^2|U/C|))的快速属性约简算法[J]. 计算机学报, 2006, 29(3): 391-399.

[43] 刘启和, 李凡, 闵帆, 等. 一种基于新的条件信息熵的高效知识约简算法[J]. 控制与决策, 2005, 20(8): 878-882.

[44] Muraszkieqicz M, Rybinski H. Towards a parallel rough sets computer[J]. Rough Sets, Fuzzy Sets and Knowledge Discovery, Edited by Ziarko W, Springer-Verlag, 1994, 434-443.

[45] 杨明. 一种基于改进差别矩阵的核增量式更新算法[J]. 计算机学报, 2006, 29(3): 407-413.

[46] Skowron A, Grgymala-Busse J W. From rough set theory to evidence theory[C]. In: Yager R R, Fedijji M, Kacprjyk J (eds). Advances in the Dempster Shafer Theory of Evidence. New York: John Wiley and Sons Inc, 1994: 193-236.

[47] Alina L, Sethil K. Decision rule extraction from trained neural networks using rough sets[J]. Intelligent Engineering Systems through Artificial Neural Networks, 1999, 10: 493-498.

[48] Wong S K M, Ziarko W, Li Ye R. Comparison of rough set and statistical methods in inductive learning[J]. International Journal of Man-Machine Studies, 1986, 24: 53-72.

[49] 刘清. 多 Agent 系统中基于 Rough 集的推理[J]. 计算机研究与发展, 2000, 39(9): 1076- 1081.

[50] Shan Ning, Ziarko W. An incremental learning algorithm for constructing decision rules[C]. In: Kluwer R S (ed). Rough Sets, Fuzzy Sets and Knowledge Discovery. Berlin: Springer Verlag, 1994: 326-334.

[51] 苗夺谦. Rough Set 理论及其在机器学习中的应用研究[D]. 北京: 中国科学院自动化研究所, 1997.

[52] Planka L, Mrozek A. Rule-based stabilization of the inverted pendulum[J]. International Journal of Computational Intelligence, 1995, 11(2): 348-356.

[53] Mrozek A. Rough sets and dependency analysis among attributes in computer implementations of expert's inference models[J]. International Journal of Man-Machine Studies, 1989, 30(4): 457-473.

[54] Hong Yu, Guoyin Wang, Dachun Yang. Knowledge reduction algorithms based on rough set and conditional information entropy[J]. In: Data Mining and Knowledge Discovery: Theory, Tools, and Technology IV, Proceedings of SPIE, 2002, 422-431.

[55] 胡丹. 基于 Rough Set 的规则提取与粗—模糊神经网络研究[D]. 四川: 四川师范大学, 2002.

[56] Tu P L, Chung J Y. A new decision-tree classification algorithm for machine learning[J]. In: Proceedings of the 1992 IEEE International Conference on Tools for Artificial Inelligence Arlington, Virginia, USA: IEEE Computer Society, 1992, 370-377.

[57] Quinlan J R. Induction of decision trees[J]. Machine Learning, 1986, 1: 81-106.

[58] 刘小虎, 李生. 决策树的优化算法[J]. 软件学报, 1998, 9(10): 797-800.

[59] 孙林, 徐久成, 马媛媛. 一种新的基于决策熵的决策表约简方法[C]. 河南省计算机学会. 计算机研究新进展. 北京: 电子工业出版社, 2006: 105-109.

[60] 孙林, 徐久成, 马媛媛. 基于新的条件熵的决策树规则提取方法[J]. 计算机应用, 2007, 27(4): 884-887.

[61] Hu Xiao Hua. Knowledge discovery in databases: An attribute-oriented rough set approach[D]. Canada: University of Regina, 1995.

[62] Jelonek J, Krawiec K, Slowinski R. Rough set reduction of attributes and their domains for neural networks[J]. International Journal of Computational Intelligence, 1995, 11(2): 339-347.

[63] Guan J W, Bell D A. Rough computational methods for information systems[J]. Artificial Intelligence, 1998, 105: 77-103.

[64] 苗夺谦, 胡桂荣. 知识约简的一种启发式算法[J]. 计算研究与发展, 1999, 36(6): 681-684.

[65] 苗夺谦, 王珏. 粗糙集理论中概念与运算的信息表示[J]. 软件学报, 1999, 10(2): 113-116.

[66] 于洪, 杨大春, 吴中福, 等. 基于信息熵的一种属性约简算法[J]. 计算机工程与应用, 2001, 37(17): 22-23.

[67] 王国胤, 于洪, 杨大春. 基于条件信息熵的决策表约简[J]. 计算机学报, 2002, 25(7): 759-766.

[68] Wang Jue, Wang Ju. Reduction algorithms based on discernibility matrix: The ordered attributes method[J]. Journal of Computer Science and Technology, 2001, 16(6): 489-504.

[69] 李佩, 刘玉树, 王蕾. 一种粗糙集属性约简算法[J]. 计算机工程与应用, 2002, 38(5): 15-16.

[70] 杜金莲, 迟忠先, 翟巍. 基于属性重要性的逐步约简算法[J]. 小型微型计算机系统, 2003, 24(6): 976-978.

[71] 蒋思宇, 卢炎生. 两种新的决策表属性约简概念[J]. 小型微型计算机系统, 2006, 27(3): 512-515.

[72] Hu Xiao Hua, Cercone N. Learning in relational databases: A rough set approach[J]. International Journal of Computational Intelligence, 1995, 11(2): 323-338.

[73] 叶东毅, 陈昭炯. 一个新的差别矩阵及其求核方法[J]. 电子学报, 2002, 30(7): 1086-1088.

[74] 胡可云. 基于概念格和粗糙集的数据挖掘方法和研究[D]. 北京: 清华大学, 2001.

[75] 代建华, 李元香. 一种基于粗糙集的决策系统属性约简算法[J]. 小型微型计算机系统, 2003, 24(3): 523-526.

[76] 王国胤. 决策表核属性的计算方法[J]. 计算机学报, 2003, 26(5): 611-615.

[77] Bjorvand A T. 'Rough Enough'—A system supporting the Rough Sets Approach[EB/OL]. http://home. sn. no/~torvill, 1998.

[78] Kryszkiewicz M, Rybinski H. Finding reducts in composed information systems[C]. In: Ziarko W P (eds). Proceedings of RSKD'93. London: Springer-Verlag, 1994: 261-273.

[79] Starzyk J, Nelson D E, Sturtz K. Reduct generation in information system[J]. Bulletin of International Rough Set Society, 1999, 3(1/2): 19-22.

[80] 王加阳, 陈松乔, 罗安. 粗集动态约简研究[J]. 小型微型计算机系统, 2006, 27(11): 2056-2060.

[81] Bazan J G, Skowron A, Synak P. Dynamic reducts as a tool for extracting laws from decisions tables[C]. In: Ras Z W, Zemankiva M (eds). Methodologies for Intelligent Systems. Berlin: Springer-Verlag, 1994: 346-355.

[82] 徐燕, 怀进鹏, 王兆其. 基于区分能力大小的启发式约简算法及应用[J]. 计算机学报, 2001, 26(1): 97-103.

[83] 韩斌, 吴铁军, 杨明晖. 结合粗糙集理论的动态属性约简研究[J]. 系统工程理论与实践, 2000, 22(6): 67-73.

[84] Qin Ke Yun, Pei Zheng, Du Wei Feng. The relationship among several knowledge reduction approaches[J]. Lecture Notes in Computer Science, Berlin: Springer-Verlag, 2005, 3613: 1232-1241.

[85] Guan J W, Bell D A, Guan Z. Matrix computation for information systems[J]. International Journal of Information Sciences, 2001, 131: 129-156.

[86] 唐彬, 李龙澍. 关于基于分明矩阵的属性约简算法的探讨[J]. 计算机工程与应用, 2004, 40(14): 184-186.

[87] Kryszkiewicz M. Comparative studies of alternative type of knowledge reduction in inconsistent systems[J]. International Journal of Intelligent Systems, 2001, 16(1): 105-120.

[88] Liang Ji Ye, Shi Zhong Zhi. The information entropy, rough entropy and knowledge granulation in rough set theory[J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2004, 12(1): 37-46.

[89] Blake C L, Merz C J. UCI repository of machine learning databases[DB/OL]. http://www. ics. uci. edu/~mlearn/MLRepository. htm, 2003.

[90] 郑芳, 吴云志, 杭小树. 粗糙集理论中知识的粗糙性研究[J]. 计算机工程与应用, 2002, 38(4): 98-101.

[91] Wang Guo Yin, Wu Yu, Fisher P S. Rule generation based on rough set theory[J]. In Data Mining

and Knowledge Discovery: Theory, Tools, and Technology II, Belur V. Dasarathy, Editor, Proceedings of SPIE, 2000, 4057: 181-189.

[92] 代建华,潘云鹤. 一种基于分类一致性的决策规则获取算法[J]. 控制与决策, 2004, 19(10): 1086-1090.

[93] Wu Xin Dong, Urpani D. Induction by attribute elimination[J]. IEEE Trans on Knowledge and Data Engineering, 1999, 11(5): 803-812.

[94] 常犁云,王国胤,吴渝. 一种基于 Rongh Set 理论的属性约简及规则提取方法[J]. 软件学报, 1999, 11(10): 1206-1211.

[95] 林嘉宜,彭宏,郑启伦. 一种新的基于粗糙集的值约简算法[J]. 计算机工程, 2003, 29(4): 70-71.

[96] 王清毅,范嵌,蔡庆生. 知识的约简研究[J]. 小型微型计算机系统, 2000, 6(21): 623-627.

[97] 黄兵,周献中. 不一致决策表中规则提取的矩阵算法[J]. 系统工程与电子技术, 2005, 27(3): 441-445.

[98] 苗夺谦,王珏. 基于粗糙集的多变量决策树构造方法[J]. 软件学报, 1997, 8(6): 425-431.

[99] 王威. 基于决策树的数据挖掘算法优化研究[D]. 西安: 西安交通大学, 2002.

[100] Grzymala-Bausse D M, Grzymala-Busse J W. The usefulness of a machine learning approach to knowledge acquisition[J]. International Journal of Computational Intelligence, 1995, 11(2): 268-279.

致 谢

本文的研究工作得到了河南省自然科学基金项目（No.0511011500）和河南省高校新世纪优秀人才支持计划项目（No.2006HANCET-19）的资助，在此表示诚挚的感谢！

我衷心感谢我的导师徐久成教授！在将近三年的硕士研究生学习生活中，我得到了徐老师无微不至的关怀，感谢他在学业上对我的严格要求；感谢他给我提供了很多相关资料；也感谢他在百忙之中对我的悉心指导，使我纠正了错误的观点，引发了创新的思想，顺利完成了学业及论文的撰写，而且使我掌握了扎实的专业知识、具备了一定的科研能力，这都将对我以后的人生道路有所裨益。徐老师学识渊博、治学严谨、学术思想活跃，待人平易谦和、诚朴敦厚，他的教学思想和作风都是我学习的楷模。再次由衷地感谢徐老师无私的支持、培养和教诲。

同时也要感谢刘清教授、苗夺谦教授、王晓东教授、薛占熬教授、冯乃勤教授、闫林副教授、张聪品副教授！他们不但交给了我知识，而且给了我思想和生活上的关心帮助，我也从各位教授身上学到了踏实做人、勤恳做事、谦和待人的人生态度。感谢河南师范大学计算机与信息技术学院其他各位老师的关心与厚爱。研究生处的老师们也给了我很多关心和帮助，在此一并表示感谢！

在这里还要感谢我的各位同学和师弟师妹们，很荣幸能和你们一起走过这一段人生道路。

最后，向所有关心和帮助过我的老师和同学表示衷心的感谢！

攻读学位期间的科研成果

发表论文：

1. 孙林,徐久成,马媛媛. 基于包含度的不一致决策表约简新方法. 计算机工程与应用, 2007,(已录用).
2. 孙林,徐久成,马媛媛. 基于新的条件熵的决策树规则提取方法. 计算机应用, 2007, 27(4): 884-887.
3. 孙林,徐久成,马媛媛. 一种新的基于决策熵的决策表约简方法. 计算机研究新进展, 河南省计算机学会2006年学术年会论文集, 2006, 105-109.
4. 孙林,徐久成,马媛媛. 基于决策熵的决策树规则提取方法. 计算机技术与发展（微机发展）, 2007,(已录用).
5. 徐久成,孙林,马媛媛. 基于新的条件熵的决策表约简方法. 计算机工程与设计,2007,(已录用).
6. 徐久成,孙林,马媛媛. 决策强度的决策表约简设计与比较. 微计算机应用, 2007,(已录用).
7. 马媛媛,徐久成,孙林. 基于粒计算的贴近度理论研究. 第六届中国 Rough 集与软计算学术研讨会(CRSSC2006), 计算机科学(专刊), 2006, 33(11A): 114-115.
8. 马媛媛,徐久成,孙林. 基于粒计算的格贴近度理论研究. 河南师范大学学报, 2007, 1: 48-50.

参加研究项目：

1. 河南省教育厅自然科学基金项目（No.2003520273，粗糙集智能数据分析模型系统理论及应用研究）。
2. 河南省教育厅自然科学基金项目（No.0511011500，智能数据分析中不确定信息处理的研究）。
3. 河南省教育厅重点教改项目（师范类本科院校计算机专业素质教育与创新能力培养的研究与实践）。

4. 河南省高校新世纪优秀人才支持计划项目（No. 2006HANCET-19，基于粗糙集理论的不确定性信息处理及其度量方法的研究）。

获奖：

1. 论文（基于粒计算的贴近度理论研究）获第六届中国 Rough 集与软计算学术研讨会（CRSSC2006）、中国人工智能学会粗糙集与软计算专业委员会、中国计算机学会人工智能与模式识别专业委员会颁发的优秀学生论文奖。