DOI:10.13656/j.cnki.gxkx.20180727.001
时雷,段其国,张娟娟,等.基于粗糙集的决策树集成学习算法[J].广西科学,2018,25(4):423-427.
SHI L,DUAN Q G,ZHANG J J,et al.Decision tree ensemble learning algorithm based on rough set[J].Guangxi Sciences,2018,25(4):423-427.

基于粗糙集的决策树集成学习算法*

Decision Tree Ensemble Learning Algorithm Based on Rough Set

时　雷[1],段其国[2],张娟娟[1],熊明阳[1],席　磊[1],马新明[1**]
SHI Lei[1],DUAN Qiguo[2],ZHANG Juanjuan[1],XIONG Mingyang[1],XI Lei[1],MA Xinming[1]

(1.河南农业大学信息与管理科学学院,河南粮食作物协同创新中心,河南郑州　450002;2.郑州商品交易所,河南郑州　450008)
(1.College of Information and Management Science,Henan Agricultural University/Collaborative Innovation Center of Henan Grain Crops,Zhengzhou,Henan,450002,China;2.Zhengzhou Commodity Exchange,Zhengzhou,Henan,450008,China)

摘要:【目的】为提高决策树集成的泛化能力和效率,解决集成全部决策树的情况下有时并不显著提高精度、反而导致额外存储和计算开销的问题,提出一种基于粗糙集的决策树集成学习算法。**【方法】**该算法基于粗糙集理论,从训练的全部决策树中选择一部分进行集成。**【结果】**与目前流行的集成学习算法 Bagging 和 Boosting 相比,本文提出的算法有效地减小了集成规模,并获得更好的泛化能力。**【结论】**该算法提高了决策树集成的泛化能力和效率。

关键词:集成学习　粗糙集　决策树　Bagging　Boosting
中图分类号:TP391　　**文献标识码:**A　　**文章编号:**1005-9164(2018)04-0423-05

Abstract:【Objective】The research of the paper focuses on the improvement of the generalization ability and efficiency of ensemble, and resolves the problems that aggregating all decision trees in ensemble usually improves the accuracy of classification slightly, but leads to extra memory costs and computational times. A decision tree ensemble learning algorithm based on rough set is proposed in this paper.**【Methods】**The algorithm is based on the rough set theory and selects a part from all the decision trees of the training for integration.**【Results】**The experiment results show that compared with the current popular ensemble learning algorithm Bagging and Boosting, the proposed algorithm not only effectively reduces the scale of ensemble but also obtains stronger generalization ability.**【Conclusion】**The algorithm improves the generalization ability and efficiency of decision tree integration.

Key words: ensemble learning, rough set, decision tree, Bagging, Boosting

0　引言

【研究意义】集成学习是把若干个学习器集成起

收稿日期:2018-06-24
修回日期:2018-07-26

作者简介:时　雷(1979－),女,博士,副教授,主要从事数据挖掘、计算机农业应用研究。

*国家自然科学基金(31501225),河南省高等学校重点科研项目(16A520055),河南省现代农业产业技术体系(S2010-01-G04),国家重点研发计划(2016YFD0300609),粮食丰产增效科技创新专项(SQ2017YFNC050081),国家留学基金资助(201709160005)和河南省科技攻关项目(162102110120)资助。

**通信作者:马新明(1962－),男,博士,教授,主要从事计算机农业应用研究,E-mail:xinmingma@126.com。

来对新的实例进行分类，通过对多个学习器的分类结果进行组合并决定最终的分类结果，以取得比单个学习器更好的性能。集成学习方法可以有效地提高学习系统的泛化能力，因此它已经成为机器学习领域的研究热点，被国际机器学习界的权威 Dietterich 称为机器学习四大研究方向之首[1]。要使得 Bootstrap AGGregatING(Bagging)等集成学习算法有效，基分类器的学习算法必须是不稳定的，也就是说要对训练数据敏感。决策树就是一种不稳定的学习算法，训练集的轻微扰动会导致其学习结果发生显著的变化。因此，很多集成学习算法都将决策树作为基学习器进行集成。**【前人研究进展】**目前的决策树集成学习算法在训练大量的决策树之后，通常是对所有的决策树都进行集成。但是集成全部的决策树会导致高昂的存储和计算开销，使得算法在很多实际问题中难以应用。而且，当集成的决策树数目增加之后，很难得到决策树之间的差异性。因此随着集成中决策树的大量增加，决策树集成学习算法的分类性能增长很缓慢，有时反而还会降低。粗糙集是由 Pawlak 于 20 世纪 80 年代提出的一种数学工具，它可以用于处理不确定或不精确的知识[2]。近年来，粗糙集理论逐渐成为信息科学领域中的一个研究热点，在特征选择[3]、半监督学习[4]和数据分类[5-7]等很多方面都取得成功的应用。**【本研究切入点】**针对集成全部分类器时既增加内存和计算时间，有时又不能有效提高分类性能的问题，本文基于粗糙集理论提出一种新的决策树集成学习算法 DTELARS。**【拟解决的关键问题】**基于粗糙集理论在训练的全部决策树中进行选择，只使用一部分决策树进行集成，提高决策树集成的泛化能力和效率。

1 集成学习算法

1.1 Bagging

Bagging（Bootstrap AGGregatING）是由 Breiman 提出的一种著名的集成学习算法[8]。它的基本思想是对训练集有放回地抽取训练样例，为每一个基分类器构造出一个跟训练集同样大小但各不相同的训练集，从而训练出不同的基分类器。在分类时使用投票法将这些基学习器的分类结果结合起来，并得到最终的分类结果。对于 Bagging 而言，基分类器的学习算法对训练数据越敏感，其效果越好。

1.2 Boosting

Boosting 是另一类非常有效的集成学习算法[9]。它的基本思想是对那些容易分错的训练实例加强学习。其步骤如下：首先，给每一个训练样例赋予相同的权重，然后训练第一个基学习器并用它来对训练集进行测试，对于那些分类错误的测试样例则提高其权重；其次，用调整后的带权训练集训练第二个基学习器；然后，重复这个过程直到最后得到一个足够好的学习器。

2 粗糙集

粗糙集理论是有效地分析和处理不精确、不一致等各种不完备信息的数学工具。本节简要地介绍粗糙集理论的有关概念。

定义 1 信息系统。四元组 $T=<U,A,V,f>$ 为一个信息系统或知识表达系统，其中 U 为所讨论对象的集合即论域；A 为属性的集合；$V=\bigcup_{a\in A} V_a$，V_a 是属性 a 的值域；$f:U\times(A)\rightarrow V$ 是一个信息函数，即对 $a\in A$，有 $f(x,a)\in V_a$。如果 $A=C\cup D$，$C\cap D=\varnothing$，C、D 分别为关于 U 的条件属性集和决策属性集，则具有条件属性和决策属性的信息系统被称为决策表。

定义 2 不可分辨关系。设有一个信息系统 $T=<U,A,V,f>$，$B\subseteq A$，定义 B 在 U 上的不可分辨关系 IND(B) 为

$$\mathrm{IND}(B)=\{(x,y)\in U\times U:\forall a\in B\ f(x,a)=f(y,a)\},\quad(1)$$

如果 $(x,y)\in \mathrm{IND}(B)$，则 x 和 y 称为 B 不可分辨，不可分辨关系是二元关系，满足自反性、对称性和传递性。显然，不可分辨关系是 T 的一个等价关系。IND(B) 的所有等价类族，即由 B 决定的划分，用 $U/\mathrm{IND}(B)$ 表示或简记为 U/B，包含 x 的等价类用 $[x]_B$ 表示。

定义 3 近似集合。设有一个信息系统 $T=<U,A,V,f>$，X 为 U 的非空子集，$B\subseteq A$，且 $B\neq\varnothing$，集合 X 的 B 下近似和上近似分别定义为

$$\underline{B}X=\{x\in U:[x]_B\subseteq X\},\quad(2)$$

$$\overline{B}X=\{x\in U:[x]_B\cap X\neq\varnothing\},\quad(3)$$

$\underline{B}X$ 是那些根据已有知识判断肯定属于 X 的对象所组成的最大集合，$\overline{B}X$ 是那些根据已有知识判断可能属于 X 的对象所组成的最小集合。

定义 4 正域。$\mathrm{POS}_B(X)=\underline{B}(X)$ 称为 X 的 B 正域。对于 U 上的两个等价关系 C 和 D，D 的 C 正域定义为

$$\mathrm{POS}_C(D)=\bigcup_{X\in U/D}\underline{C}X。\quad(4)$$

定义 5 约简。令 R 为一族等价关系，$A\in R$，如果 $\mathrm{IND}(R)=\mathrm{IND}(R-\{A\})$，则称 A 为 R 中不必

要的；否则称 A 为 R 中必要的。如果每一个 $A \in R$ 都为 R 中必要的，则称 R 为独立的；否则称 R 为依赖的。设 $P \in Q$，如果 P 是独立的，且 $\mathrm{IND}(Q) = \mathrm{IND}(P)$，则称 P 为 Q 的一个约简，记为 $\mathrm{RED}(Q)$。

本文采用 QuickReduct 算法[10]计算约简，该算法使用属性依赖度 $r_p(D)$ 作为属性选择准则。属性依赖度 $r_P(D)$ 的定义如下：

$$r_P = \frac{\| \mathrm{POS}_P(D) \|}{\| U \|}。 \tag{5}$$

基于属性依赖度的 QuickReduct 约简算法如算法 1 所示。

算法 1：QUICKREDUCT Reduction Algorithm

Input：C，the set of all conditional attributes；

D，the set of decision attributes.

Output：the reduct R of C ($R \subseteq C$).

Step 1：R ← {}

Step 2：do

Step 3：　T ← R

Step 4：　$\forall x \in (C-R)$

Step 5：　if $\gamma_{R\cup\{x\}}(D) > \gamma_T(D)$

Step 6：　$T \leftarrow R \cup \{x\}$

Step 7：　R ← T

Step 8：until $\gamma_R(D) == \gamma_C(D)$

Step 9：return R

3　基于粗糙集理论的决策树集成学习算法

本文提出的集成学习算法(DTELARS)主要包括 4 个步骤。

(1)将原始训练集 S 随机地分为两个集合，即 S_1 和 S_2。

(2)采用可重复取样技术从集合 S_1 中重采样出一系列新的训练集，随后从每个新的训练集中训练出一个决策树，从而得到一组决策树。

(3)使用全部的决策树对集合 S_2 中的样本进行分类，通过对集合 S_2 中样本的预测标签和样本的真实标签构造出决策表。例如，假设集合 S_2 中有 M 个样本，由第二步中训练得到 N 个决策树。每个决策树对集合 S_2 中的每个样本都进行预测，则对集合中 M 个样本的预测可以表示为一个 $M \times N$ 矩阵 $W = [p_{ij}]$，p_{ij} 表示第 j 个决策树对第 i 个样本给出的预测标签。将样本的真实标签作为决策属性，这个矩阵就构成一个决策表。在这个决策表中，每个决策树都被看作为一个条件属性，而决策树对样本的预测标签则作为条件属性的值。使用上节中介绍的 QuickReduct 约简算法对决策表进行约简，从而选择出全部决策树中的一个子集用于集成。

(4)当对一个原始训练集 S 之外的新样本进行分类时，只使用选择的决策树对该样本进行预测，然后采用多数投票方法给出样本的最终分类结果。

详细的 DTELARS 如算法 2 所示。

Algorithm 2：Decision Tree Ensemble Learning Algorithm

Inputs：training set S，decision tree，trials T

Outputs：final classifier

Step 1：Partition the S into two sets $S = S_1 \cup S_2$

Step 2：Create a pool including of T decision trees

2.1. For t = 1 to T {

2.2. St = bootstrap sample from S_1

2.3. Train the decision tree dtt on St

2.4. }

Step 3：Select the decision trees from the pool by using QuickReduct algorithm

3.1. Apply T decision trees in the pool to classify the examples in the S_2

3.2. Construct a decision table based on the predicted class labels and real class labels of examples in the S_2

3.3. Apply the QuickReduct algorithm to obtain a reduct of the decision table

Step 4：Construct the final classifier for classification

4.1. Create a combining pool including of the decision trees in the reduction

4.2. When classifying new example，only use the decision trees in the combing pool to predict example and then use major voting method to aggregate the corresponding predictions of decision trees to yield the final decision.

4　算法评价

4.1　数据集

为对本文提出的集成学习算法的性能进行评价，在 UCI 机器学习数据库[11]中的 7 个数据集上进行实验。这 7 个数据集的具体信息如表 1 所示。因为两类分类问题相对于多类分类问题更具有一般性，一个多类分类问题可以转化为一组两类分类问题来解决。因此本文中只考虑两类分类问题，即对于一个样本，或者将它分为正例，或者将它分为负例。

4.2　评价指标

采用精度来衡量分类算法的性能。分类器对样本的分类结果有 4 种情况[12]：TP，被正确地分类为属于此类别的样本数量；TN，被正确地分类为不属于此类别的样本数量；FP，被错误地分类为属于此类别

的样本数量;FN,被错误地分类为不属于此类别的样本数量。

表 1　实验所用数据集

Table 1　The datasets in the experiment

数据集 Dataset	样本数 Number of samples	属性数 Number of attributes	类别数 Number of classes
breast	699	9	2
diabetes	768	8	2
heart	270	13	2
hepatitis	155	19	2
ionosphere	351	34	2
sonar	208	60	2
vote	435	16	2

根据以上 4 种情况,分类性能可以按照精度来评价,精度的定义如下:

$$\text{Accuracy}=\frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}。\tag{6}$$

4.3　评价结果

实验中以两个流行的集成学习算法 Bagging 和 Boosting 作为基准,对本文提出的算法 DTELARS 性能进行评价。实验中,对于 Boosting 采用的是 AdaBoost. M1 算法[9];对于决策树采用的是 C4. 5 算法[13]。同时为比较性能,实验中也包括了使用单个决策树进行分类得到的结果。分类性能的评价方法采用的是十折交叉验证法。在实验中分别设置 Bagging 和 Boosting 的集成规模(集成中决策树的数目)从 10 个到 50 个。以此来比较集成规模的变化对 DTELARS、Bagging 和 Boosting 性能产生的影响。

当集成的规模设置为 10 个,即 DTELARS 是从 10 个训练决策树中进行选择和集成,Bagging 和 Boosting 是分别训练和使用 10 个决策树进行集成,不同算法得到的分类精度如表 2 所示。表 2 最后一列为 DTELARS 集成中决策树的平均数目,表中的 avg 表示在全部数据集上分类精度的平均值。如表 2 所示,DTELARS 得到的平均分类精度为 87. 43%,比仅使用一个决策树进行分类的平均精度高出 3. 48%,比 Bagging 高出 1. 92%,比 Boosting 高出 1. 40%,DTELARS 在所有数据集上的平均分类精度均超过其它的算法。由 DTELARS 创建的集成中所使用的决策树数目仅是 Bagging 和 Boosting 使用数目的 54%(5. 4/10. 0)。

当集成的规模设置为 40 个,即 DTELARS 是从 40 个训练的决策树中进行选择和集成, Bagging 和 Boosting 是分别地训练和使用 40 个决策树进行集成时,DTELARS 在所有数据集上的平均分类精度也高于其它算法(表 3)。由 DTELARS 创建的集成中所使用决策树的数目仅是 Bagging 和 Boosting 使用数目的 17. 8% (7. 1/40. 0)。

图 1 展示了各种算法(包括单个决策树、DTELARS、Bagging 和 Boosting)在全部数据集上的平均分类精度随着集成规模的变化情况。注意在图中 Bagging 和 Boosting 是“全部集成”,因为它们对于横轴上标注的每一个数目而言都集成了全部的决策树。而单个决策树仅使用一个决策树进行分类,本文提出的算法 DTELARS 对于每个数目都是仅选择和使用一部分的决策树进行集成。

表 2　不同算法在各数据集上的精度比较(集成规模=10)

Table 2　Accuracy values of the different algorithms on datasets (ensemble size=10)

Dataset	Single C4. 5	Bagging	Boosting	DTELARS	Number
breast	95. 13	95. 73	95. 70	96. 13	6. 4
diabetes	74. 08	74. 34	73. 17	74. 46	4. 2
heart	78. 14	81. 48	78. 88	83. 60	7. 1
hepatitis	82. 58	83. 22	84. 51	87. 79	5. 4
ionosphere	91. 16	93. 44	93. 11	92. 82	4. 9
sonar	70. 19	74. 03	80. 76	79. 98	4. 6
vote	96. 36	96. 33	96. 09	97. 23	5. 2
avg	83. 95	85. 51	86. 03	87. 43	5. 4

表 3　不同算法在各数据集上的精度比较(集成规模=40)

Table 3　Accuracy values of the different algorithms on datasets (ensemble size=40)

Dataset	Single C4. 5	Bagging	Boosting	DTELARS	Number
breast	95. 13	95. 99	96. 56	96. 37	7. 9
diabetes	74. 08	74. 86	71. 74	75. 23	6. 5
heart	78. 14	82. 22	79. 25	84. 26	7. 6
hepatitis	82. 58	81. 93	87. 74	87. 90	7. 3
ionosphere	91. 16	93. 16	94. 01	93. 41	5. 7
sonar	70. 19	78. 84	86. 05	84. 65	6. 8
vote	96. 36	96. 09	95. 86	97. 43	8. 2
avg	83. 95	86. 15	87. 31	88. 46	7. 1

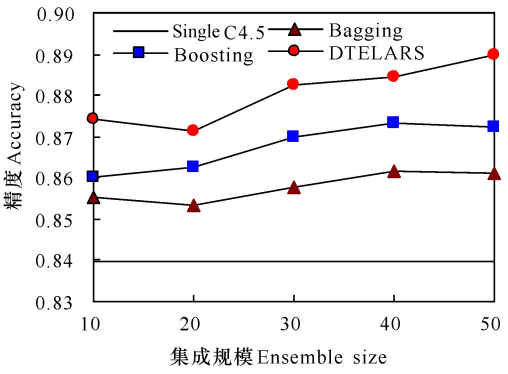

图 1　在全部数据集上不同算法的平均性能曲线

Fig. 1　The average performance curves of different algorithms on datasets

从图 1 中可以得出以下结论：DTELARS 在全部数据集上的平均分类精度始终高于单个决策树、Bagging 和 Boosting。

随着“全部集成”中决策树数目的增加，DTELARS 所选择和集成的决策树只占全部决策树的一小部分；并且当“全部集成”中决策树的数目从 10 增加到 50 时，DTELARS 所选择的决策树的数目只是缓慢地增加(图 2)。因此，DTELARS 和“全部集成”的 Bagging、Boosting 相比，在存储开销和计算时间上都有显著的改善。

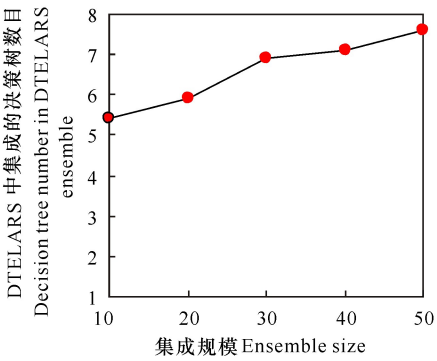

图 2　在全部数据集上 DTELARS 集成的平均决策树数目的变化曲线

Fig. 2　The varying curve of average number of decision trees in the proposed algorithm on datasets

5　结论

为提高集成算法的泛化能力和效率，本文提出一种新的集成学习算法 DTELARS。DTELARS 基于粗糙集理论从训练的全部决策树中进行选择，只使用一部分决策树进行集成，并与 Bagging、Boosting、决策树算法在 UCI 数据集上进行性能比较。实验表明 DTELARS 不但减小集成规模，而且还获得更好的泛化能力。

参考文献：

[1] DIETTERICH T G. Machine learning research：Four current directions[J]. AI Magazine，1997，18(4)：97-136.

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

[3] 顾翔元，郭继昌，田煜衡，等. 基于条件互信息的空域隐写检测特征选择算法[J]. 天津大学学报：自然科学与工程技术版，2017，50(9)：961-966.
GU X Y，GUO J C，TIAN Y H，et al. Spatial-domain steganalytic feature selection algorithm based on conditional mutual information[J]. Journal of Tianjin University：Science and Technology，2017，50(9)：961-966.

[4] SHI L，MA X，XI L，et al. Rough set and ensemble learning based semi-supervised algorithm for text classification[J]. Expert Systems with Applications，2011；38(5)：6300-6306.

[5] SHI L，DUAN Q，SI H，et al. Approach of hybrid soft computing for agricultural data classification[J]. International Journal of Agricultural and Biological Engineering，2015，8(6)：54-61.

[6] HONG Y H，XU S，LIANG J R. A model of classifier based on rough set and genetic neural network[J]. Guangxi Sciences，2013，20(2)：128-131，136.

[7] 樊艳英，徐章艳，张伟，等. 一种基于粗糙集理论的值约简算法[J]. 广西科学院学报，2013，29(1)：4-6，10.
FAN Y Y，XU Z Y，ZHANG W，et al. A complete value reduction algorithm based on rough set theory[J]. Journal of Guangxi Academy of Sciences，2013，29(1)：4-6，10.

[8] BREIMAN L. Bagging predicators[J]. Machine Learning，1996，24(2)：123-140.

[9] FREUND Y，SHAPIRE R E. A decision-theoretic generalization of on-line learning and an application to boosting[J]. Journal of Computer and System Sciences，1997，55(1)：119-139.

[10] CHOUCHOULAS A，SHEN Q. Rough set-aided keyword reduction for text categorization[J]. Applied Artificial Intell- igence，2001，15(9)：843-873.

[11] DUA D，KARRA T E. UCI machine learning repository[Z]. Irvine，CA：University of California，School of Information and Computer Science，2017. http://archive. ics. uci. edu/ml.

[12] YANG Y. An evaluation of statistical approaches to text categorization[J]. Journal of Information Retrieval (1)，1999：69-90.

[13] QUINLAN J R. C4. 5：Programs for machine learning[M]. San Francisco：Morgan Kaufmann Publishers，1993.

（责任编辑：米慧芝）