一种针对交互式学习系统日志数据的轻型化挖掘方法

刘锟，邢延，蔡延光

(广东工业大学 自动化学院，广东 广州，510006)

摘要：提出一种针对交互式学习系统产生的日志数据的轻型化挖掘方法。该方法以选择性集成学习为框架，采用 C4.5 为基本分类器。其轻型化是通过在数据预处理阶段，引入新的基于 *K* 均值的属性取值归约算法对部分取值水平丰富的类别属性进行归约，并在模型集成阶段，采用贪心算法对基本分类器进行选择，使最终集成模型得到大幅度精简。上述 2 项措施在保证模型具有较好预测表现的前提下，大幅度降低了学习代价，提升了系统泛化能力。为了检验方法的有效性，以直接源于教育领域的现实数据——KDD Cup 2010 挑战数据集进行检验。结果表明，该方案即便实践于单核 PC 机(CPU：2.0G；RAM：2.0G)亦具有较高的模型训练效率和较好的泛化能力。

关键词：分类；非均衡数据；属性值归约；数据集成；集成学习
中图分类号：TP181　　**文献标志码**：A　　**文章编号**：1672−7207(2011)S1−0755−05

A lightweight solution to educational log data mining

LIU Kun, XING Yan, CAI Yan-guang

(Faculty of Automation, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: A lightweight framework was presented for educational data mining based on selective ensemble, using C4.5 as the basic learning algorithm. The solution introduces a novel algorithm, based on *K*-mean, to aggregate the categorical attributes having too much value levels. Finally, some experimental results and discussions are provided to validate the proposed approach, using the challenge data set of educational KDD Cup 2010. The results show that the approach has an efficient model training ability and good model generalization even if the algorithm is applied to a single-core PC with a 2.0G CPU and 2.0G RAM.

Key words: classification; imbalanced data; attribute value aggregation; data reduction; ensemble

近几年教育领域的数据挖掘逐渐兴起并开始应用于实践。例如，通过挖掘在线式学习系统中学生与系统交互而产生的日志数据来获得其学习新知识的能力；然后，利用这些信息来安排学生的学习内容与进度，达到因材施教的目标[1]。若能有效挖掘这方面的信息，对于提高学生的学习效率十分显著[2]。预测学生的学习能力，主要是预测学生在学习过程中能否适应新的学习任务，如某个学生能够正确解答方程 −18+*x*=15，预测其能否求解 5+*x*=−39。与其他领域的数据挖掘相比，教育领域的日志数据有 2 个显著特点：其一，数据规模巨大，动辄上千万条记录，同时，类别属性的取值水平也非常丰富，即单个类别属性取值种类很多；其二，数据标签正、负比例呈现非均衡，即对于某个问题，答对与答错的比例通常是不对等的。数据的这 2 个特点带来了 2 个问题：数据规模巨大，意味着训练代价比较大；类别属性取值种类过于丰富，不仅导致训练过程中运算量急剧上升，同时因其包含了过多的细枝末叶信息，容易导致模型过拟合

收稿日期：2011−04−15；**修回日期**：2011−06−15
基金项目：国家自然科学基金资助项目(61074147，60374062)；中国−爱尔兰科技合作研究基金资助项目
通信作者：刘锟(1983−)，男，湖南邵阳人，硕士研究生，从事机器学习与数据挖掘研究；电话：15899961637；E-mail：kunliu0213@163.com

(Overfitting)，致使模型泛化能力变差；而数据非均衡通常会导致分类模型的预测精度下降。因此，能否成功挖掘此类数据，取决于预处理及算法设计过程中能否解决上述的 2 个问题。这方面的研究已经有一些成果。例如，Cen 等[2]通过逻辑回归建立认知模型来预测学生的学习能力；Thai-Nghe 等[3]比较了决策树与贝叶斯网络在教育数据上的表现。此外，还有 Thai-Nghe 等[4]利用支持向量机对教育领域数据非均衡(Imbalanced classification)难题进行了尝试。但还未见从上述问题的 2 个方面同时进行改进的研究成果，本文提出一种双管齐下、基于轻型架构的教育领域数据挖掘方法。该方法以集成学习为框架，C4.5 为基本分类器，通过在数据预处理阶段，引入基于 K 均值的属性取值归约算法对部分取值水平丰富的类别属性进行取值归约；并在模型集成阶段采用贪心算法对基本分类器进行选择，使最终集成模型精简化。上述 2 项措施在保证模型具有较好的预测表现的前提下，大幅度降低了训练代价，提升了系统的预测性能。为了检验模型的有效性，直接采用源于 KDD Cup 2010 (URL：https://pslcdatashop.web. cmu.edu/KDDCup)的数据。KDD Cup 是由 SIGKDD(ACM Special Interest Group on Knowledge Discovery and Data Mining)组织的数据挖掘与知识发现挑战赛。每年举办一次，与 SIGKDD 国际会议同期举行，同时面向学术界与业界。

1 数据

本文提出的数据挖掘方法针对的是交互式学习系统日志数据。它直接源于系统，记录的是学生利用学习向导求解几何问题过程中的每个交互细节。KDD Cup 2010 数据的形式如图 1 所示，为了便于说明问题，这里省略了部分属性。这些交互日志数据包含如下几个关键概念：问题、步骤、知识点、机会数。通常学生求解一个问题需要若干步骤，每个步骤对应 1 个或多个所需的知识点。此外，在问题求解过程中同一知识点可能会被反复使用，而机会数记录的是某个知识点在问题求解过程中被使用的次数。数据表中的前几列与学习过程中的任务有关，在图中已经圈出。最后一例为标签属性，它的含义是学生求解某一步骤时，初次尝试的表现(即做对了，还是做错了)。挑战赛的任务就是利用已打上标签的训练数据训练各类机器学习方法来建立模型以预测学生应对新问题的表现。

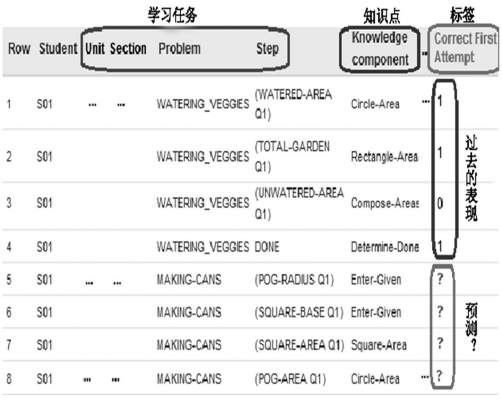

图 1 KDD Cup 2010 数据形式

Fig.1 Data format of KDD Cup 2010

这份 KDD Cup 数据与其他源自真实世界的数据一样：混有噪声、含有空缺、取值不一致。其中，属性“KC(KTracedSkills) ”中 50.4%的取值为空，有些属性的取值大、小写不一致。除此之外，它还有如下 4 个特性：(1) 数据规模巨大，挑战数据包中较小的那个就含 8 918 054 条记录(我们所采用的)。对于分类任务，数据的海量总比不足要好很多。但数据过于巨大将不得不考虑算法在时间、空间上的消耗与匮乏的计算资源之间的矛盾。 (2) 部分类别属性的取值水平过于丰富。类别属性取值水平是指该属性所有取值中不同取值的种类数，在数值上等于该属性其取值所组成集合的元素个数。从表 1 可以看出，属性“Problem Name”, “Step Name”取值水平数超过 10 万。若直接使用这些属性进行学习，将会带来如下 3 个问题：其一，属性选择及后续分类算法的训练都有可能被取值水平过于丰富的属性而主导(dominate)，致使其他属性的效用被忽略。其二，过于丰富的属性取值意味着存在冗余的细节信息，这容易导致模型过拟合。因为在训练集中出现的细节信息，在测试集中未必出现。其三，噪声也通常隐藏在这些细节信息里。(3) 数据呈现非均衡特性。数据非均衡是指数据集中不同类的样本数

表 1 类别属性及其取值水平数

Table 1 Categorical attributes and their value levels

属性名称	属性取值水平数	
	原始	归约后
Problem Name	188 368	83
KC(KTracedSkills)	921	45
KC(Rules)	2 978	94
Problem hierarchy	165	未归约
Problem view	18	未归约
Step Name	695 674	未归约
KC(SubSkills)	1 824	未归约

目所占比例严重不等。对于第二分类问题，就是其中一类的样例数远超过另一类的。传统机器学习通常假设类之间是均衡的或近似均衡的，并以最大化总体分类精度为目标，导致算法只顾着提高多数类(即样本数占绝对优势的那一类)的分类精度，而忽略样本中少数类的预测精度。(4) 属性间存在强相关关系，属性间的相关系数矩阵见表 2 所示。传统算法大都假定预测属性(除标签属性以外)间是相互独立的，如朴素贝叶斯，因此难以将这些属性直接应用于模型训练。

表 2　属性相关性分析结果

Table 2　Correlation matrix of massive categorical

Attributes	Problem Name	Step Name	KC (subSkills)	KC (KTracedSkills)	KC (Rules)
Problem Name	1	0.641	0.048	0.021	0.142
Step Name	0.641	1	−0.026	−0.073	−0.042
KC (subSkills)	0.048	−0.026	1	0.845	0.590
KC (KTracedSkills)	0.021	−0.073	0.845	1	0.538
KC (Rules)	0.142	−0.042	0.590	0.538	1

2　属性值归约

前面已经提到，过于丰富的属性取值水平不仅使学习代价增大，同时易导致模型过拟合。因此，有必要对过于丰富的属性取值水平进行归约。属性取值归约即使用较高的概念水平将多个意义相同、相近的取值替换为一个新的取值。

2.1　归约属性的选择

在具体的挖掘项目中，针对属性值进行归约并不容易，尤其是在缺乏相关领域专家指导的情况下。这包括 2 方面的问题：其一，选择哪些类别属性进行归约；其二，每个类别属性具体归约到哪个概念水平比较合适。由于缺乏领域专家，采用信息增益率对几个取值非常丰富的类别属性进行衡量。各类别属性信息增益率值如表 3 所列。从表 3 可以看出：几个取值丰富的类别属性其信息增益率相差并不大，也就是，这几个属性的分类贡献差不多。因此，单从信息增益率方面还无法对这些类别属性进行选择。于是，利用相关性分析来进一步探索这些属性间可能存在的关系，其结果如表 2 所列。相关分析表明：“Problem Name”与“Step Name”，“KC(SubSkills)”与“KC (KTracedSkills)”分别存在强相关性(一般认为相关系数 ρ 大于 0.6，为强相关)。2 个属性强相关说明其信息上存在一定重叠，因此，在构建训练集的属性子集时可以选择其中一个作为二者的代表。从数据特性分析可知：一个问题含有多个步骤，同时一个子知识点包含在父知识点中。因此我们在“Problem Name”与“Step Name”中选取前者作为代表，因为前者包含了后者；在“KC(SubSkills) ”与“KC(KTracedSkills)”中选取后者作为代表，理由同上。最后选出“Problem Name”，“KC(KTracedSkills)”与“KC(Rules)”3 个取值水平非常丰富的类别属性进行属性取值归约。

表 3　属性信息增益率

Table 3　Information gain ratio of massive attributes

Problem Name	Step Name	KC (SubSkills)	KC (KTracedSkills)	KC (Rules)
0.08	0.10	0.09	0.08	0.12

2.2　属性值归约算法

当待归约的属性集选定后，对每个属性的归约按照表 4 所示算法进行。此算法基于经典的聚类算法 *K*-均值。算法第 2 步中的 *K* 值，即 *K* 均值中的聚类数，是此属性值归约算法的关键。它是通过多次实验并通过人工分析聚类结果来确定的。一分好的聚类结果意味着在常识层面同类型的概念被聚为同一类。例如，对于属性“Problem Name”里的代数表达式−18+*x*=15 与 5+*x*=−39 应被归为一类。

表 4　属性值归约算法

Table 4　Algorithm of attribute-value-aggregation

属性值归约算法
对于每一个欲进行归约的类别属性:
1: 获得属性取值的集合(即去除重复值)
2: 利用经典的 *K*-均值进行聚类
3: 分析聚类结果
4: if 聚类结果不合理
5: 返回第 2 步，修改 *K* 值，再次进行聚类
6: else
7: 赋给每一个类一个较高层次的抽象概念值，对应关系记入字典中
8: end if
9: 按照上述所得的字典对原始数据进行替换操作

类别属性值中长度过长的字符串会导致计算量急剧增加，因此不易直接应用于上述算法。KDD Cup 2010 数据中就含有一些特别的类别属性如“KC(KTracedSkills)”与“KC(Rules)”，其值的字符串非

常长，一些长度超过了 8 000 个字符。对于经典的聚类算法，处理这类数据都将是比较困难的，不仅训练速度会明显变慢，聚类效果也会变差。因此，有必要对这一类别的属性先进行一次文本挖掘，将其中频度最高的 N 个关键词提取出来。N 依据具体情况而定，这里取 N=1。这一过程会丢失一部分信息，但对于降低计算复杂度却是非常有益的。最后结果表明，这种做法也是可取的。

从表 1 可知，经过此处理后属性取值种类由原来的数以万计降低到百以内，这对实现整个数据挖掘架构的轻型化很有意义。用较高层次的概念代替低层次的概念，这在方法论上与粒度计算是一致的。

3 模型训练框架

集成学习是提升模型预测精度的一种有效策略。实验表明，当模型中的基本分类器间存在较显著的差别时，集成模型将有可能得到更好的预测精度和更佳的泛化能力[5−6]。由于弱分类器对数据样本比较敏感，因此，欲使训练出的模型间存在差异，只需抽样使训练样本间存在一定差异即可。集成学习还是应对数据海量规模的有效手段。它通过抽样将数据分为多份，使每份数据的规模削减到实际计算资源允许的范围内。此外，集成学习对于数据非均衡难题也有一定的改善作用[7]。因此，对于 KDD Cup 的这份数据选用集成学习作为框架是很合适的，这不仅有效应对了海量数据规模，还在一定程度上应对了数据的非均衡特性。

集成学习大体分为 2 类，即 Bagging 与 Boosting。这里采用 Bagging 算法，因为它相对比较简单，而且训练过程可以并行化，有利于在计算资源不足的情况下，分批次进行基本分类器的训练。训练过程如图 2 所示。

经典的 Bagging 在学习中，其抽样采用的是 Bootstrap 方法，一种有放回全抽样，即抽样规模与原数据规模相等。由于此份数据的规模过大，直接采用 Bootstrap 抽样方法显然行不通。因此，我们对集成框架中的抽样进行重新设计。新设计的抽样方案秉承抽样数据尽可能体现总体特性的原则，并使抽样结果间存在适度差异化。具体从以下 5 个方面来进行设计：其一，考虑到机器的计算能力，这里只抽取 50 份样本，每份样本的规模为 100 000。其二，每份数据中标签正、负比例与原数据相同，保持 85:15。其三，争取每份数据中，每个学生至少包含一个问题。其四，采用不放回、随机抽样方式。其五，从剩余的数据中随机抽取 15 份样本，每份抽样亦为 100 000，作为校验数据集。

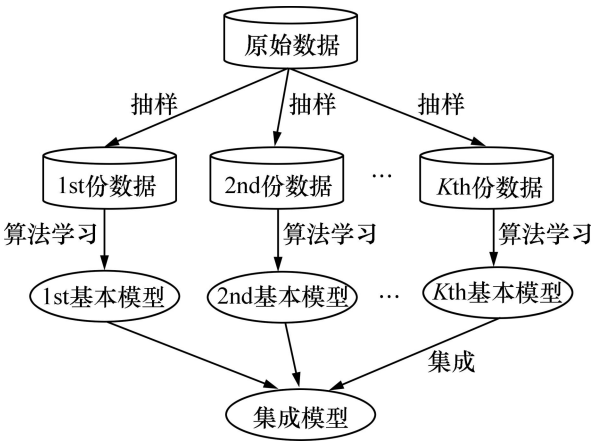

图 2 集成学习框架

Fig.2 Idea of ensemble learning

为了防止学习过拟合，对每份数据的训练都采用 10 规模的交叉验证方法。当基本分类器训练好后，将进入模型集成阶段。最近几年集成学习领域的研究表明，在集成前对基本分类器进行选择将有可能在精简模型的同时获得更佳的效果[8]。为了进一步实现学习框架的轻型化，对训练所得的 50 个基本分类器进行选择，最终得到 13 个基本分类器，以简单投票机制组合为一个集成学习模型。分类器的选择由贪心算法逐步试探性删除来完成。该过程从 50 个分类器组成的集合开始，每一步随机删除集合中的 1 个元素(即一个分类器)，然后测试剩余子集组成的模型性能，若分类表现没有显著下降，则说明删除成功；反之，则说明此分类器不应删除。算法伪代码如表 5 所示。

表 5 基本分类器选择算法

Table 5 Algorithm for base classifiers selection

基本分类器选择算法
1: 初始化计算器 counter = 0；
2: for k=1 to N (N 为训练所得分类器总数)
3: 从分类器集合中随机删除一个分类器
4: 评估集合中剩余分类器所组成模型的分类表现
5: if (表现优于先前或跟先前相差不大) {
跳转到第 2 步；
--counter； }
6: else {
将删除的分类器重新添回分类器集合中；
if (++counter >= 5)，则停机；}
7: end for

最终所得的集成学习模型的表现，是以均方根误差(Root Mean Square Error)来衡量的。均方根误差(RMSE)常用于衡量模型预测值与真实值之差的差异程度。其计算公式如下，这里假定真实标签的向量为 $\boldsymbol{\theta}_1$，预测标签向量为 $\boldsymbol{\theta}_2$，其中 $\boldsymbol{\theta}_1=(a_{11}, a_{12}, \cdots, a_{1n})$，$\boldsymbol{\theta}_2=(a_{21}, a_{22}, \cdots, a_{2n})$，则，

$$\mathrm{RMSE}(\boldsymbol{\theta}_1-\boldsymbol{\theta}_2)=\sqrt{\sum_{i=1}^{n}(a_{1i}-a_{2i})^2}\Big/\sqrt{n}$$

将模型最终的预测表现，列在表 6 中。结果源自 KDD Cup 2010 官方网站中提供的评测系统。第 1 行是我们方法的结果，第 2 行是挑战赛第一名的成绩。对比二者可知，本文提出的针对交互式学习系统日志数据的轻型化挖掘方法，以只损失约 0.06 RMSE 的代价换来了计算资源方面的巨大优势。

表 6　最终模型表现

Table 6　Prediction performance of solution

方　案	RMSE	计算资源
我们的轻型化方法	0.332 801	单核 PC 机(CUP：2.0G；RAM：2.0G)
挑战获胜冠军方法	0.274 568	多核并行计算平台(其中 RAM：≥32G)

4　结论

提出一轻型化方法从教育领域交互式学习系统日志数据中挖掘有价值的信息。它通过在数据预处理阶段对属性取值水平过于丰富的属性进行概念层归约，并在集成学习过程中引入分类器选择机制，从而用较少的计算资源达到较好的挖掘效果。该方法具有很强的伸缩性与可操作性。例如，在类别属性归约过程中的目标属性选择、归约时针对具体问题应归约到哪个概念层次等方面，若能配合相关领域的专家，将会获得更佳的效果。此外，在训练基本分类器时，可根据实际计算资源对分类器的个数进行调整。在资源允许的前提下，尽可能多地训练一些基本分类器并将其纳入候选分类器集中，对于后续分类器选择过程是有益的。

参考文献：

[1] Feng M, Heffernan N, Koedinger K. Addressing the assessment challenge with an online system that tutors as it assesses[J]. User Modeling and User-Adapted Interaction, 2009, 19(3): 243−266.

[2] Cen H, Koedinger K, Junker B. Learning factors analysis—A general method for cognitive model evaluation and improvement[C]//Intelligent Tutoring Systems. Berlin, Heidelberg: Springer, 2006, 4053: 164−175.

[3] Thai-Nghe N, Busche A, Schmidt-Thieme L. Improving academic performance prediction by dealing with class imbalance[C]//Proceeding of 9th IEEE International Conference on Intelligent Systems Design and Applications (ISDA'09). Pisa: IEEE Computer Society, 2009: 878−883.

[4] Thai-Nghe N, Janecek P, Haddawy P. A comparative analysis of techniques for predicting academic performance[C]//Proceeding of 37th IEEE Frontiers in Education Conference (FIE'07). Milwaukee: IEEE Xplore, 2007: T2G7−T2G12.

[5] Kuncheva L, Whitaker C. Measures of diversity in classifier ensembles[J]. Machine Learning, 2003, 51: 181−207.

[6] Sollich P, Krogh A. Learning with ensembles: How overfittingcan be useful[J]. Advances in Neural Information Processing Systems, 1996, 8: 190−196.

[7] Chen J J, Tsai C A, Young J F, et al. Classification ensembles for unbalanced class sizes in predictive toxicology[J]. SAR and QSAR in Environmental Research, 2005, 6: 517−529.

[8] ZHOU Zhi-hua, WU Jian-xin, TANG Wei. Ensembling neural networks：Many could be better than all[J]. Artificial Intelligence, 2002, 137(1/2): 239−263.

(编辑　杨华)