

文章编号: 0529-6579 (2000) 04-0027-05

基于决策树的神经网络规则抽取方法^{*}

侯广坤, 张劲峰

(中山大学计算机科学系, 广东 广州 510275)

摘 要: 将从神经网络中抽取一个可理解的模型视为一个归纳学习任务, 其中, 目标概念就是神经网络表达的功能, 所生成的可理解模型是一个能很好近似神经网络的决策树. 在这个过程中, 应用了决策树归纳学习的优化原则, 使得生成的决策树能最简洁、准确地描述神经网络学到的知识. 实验证明, 生成的决策树可以很好地近似神经网络, 且比用传统方法生成的决策树具有更好的分类精度, 同时 NNtoDT 算法也保持了具有较好的通用性和可扩充性的特性.

关键词: 神经网络规则抽取; 神经网络; 决策树; 聚类; 归纳学习

中图分类号: TP183 **文献标识码:** A

1 问题的提出

神经网络是目前公认的高精度分类器, 但它的分类过程却令人难于理解. 训练后的神经网络所表示的假设模型转换成一个更易于理解的语言(如符号规则), 如 if-then 规则、M-of-N 规则、模糊规则、决策树、有限状态自动机^[1]等, 一直为研究者关心的课题. 但是目前提出的一些方法都存在一些问题, 例如, KBANN 方法^[2]及 KT 方法^[3]等, 效率非常低, 因此, 没有很好的可扩充性; 在 KBANN 方法^[2]中, 神经网络与先验的领域规则有关, 没有很好的通用性; Craven 提出的 TREPAN 算法^[4]较好地解决了规则抽取算法的可扩充性和通用性问题. 但 TREPAN 算法所生成的决策树是基于 MofN 形式规则的, 所产生的单个规则通常都是不容易理解的^[5,6]. 一些研究表明, 大量的学习问题并不学习 MofN 类型的概念, 这一点大大限制了基于 MofN 形式规则的算法的应用性^[7]. 另外, Wnek^[7]还发现人们在判定实体类别时主观产生的概念往往是基于 DNF(析取范式)模式的. 在 TREPAN 算法中, 当生成的一个决策树具有多个基于 MofN 形式的内节点时, 树的可理解性也受到很大的影响, 转换为基于 DNF 形式的规则时, 要花费更多的时间.

在 TREPAN 算法思想的基础上, 本文提出的 NNtoDT 算法使生成的决策树具有(近似)最优的结构和很高的预测准确性, 同时也保持了具有较好的通用性和可扩充性的特性.

2 NNtoDT 算法

NNtoDT 是一个规则提取算法, 它把从训练后的神经网络提取可理解的假设模型的问题

^{*} 收稿日期: 1999-07-07; 作者简介: 侯广坤(1941~), 男, 教授.

题看作是一个归纳学习任务, 目标概念就是神经网络所表达的功能, 学习算法所产生的假设模型是一个能近似表达神经网络功能的决策树. 传统的决策树归纳算法是从问题领域的实例空间中归纳出蕴藏在实例空间中的人可理解的模型——决策树. 本文把神经网络当作一个“新”的实例空间, 然后用一个新的决策树算法将蕴藏在“新”的实例空间中的模型, 用人可理解的决策树模型表示出来, 从而实现神经网络规则抽取任务. 为了更好地模拟神经网络这个“新”的实例空间, NNtoDT 同样使用了神喻 (Oracle) 的概念^[4], 用它来回答学习过程中所提出的成员查询 (membership query). NNtoDT 与训练后的神经网络的相互作用是通过“成员查询” (membership query) 来实现的, 这一点也决定了 NNtoDT 具有很好的通用性. 一个“成员查询”就是一个向 Oracle 提出的由来自学习者的实例空间的一个实例组成的问题 (question). 给定一个查询实例, Oracle 返回这个实例的类标记. 这里试图学习的目标概念就是神经网络所表示的功能, 所以, 把训练后的神经网络当作 Oracle, 用它来回答一个成员查询仅仅是用训练后的神经网络来对一个 (未确定类属的) 实例进行分类. 成员查询在 NNtoDT 算法中有 2 种不同的使用方式. 最初它们被用来获得神经网络训练样本的类标记 (即把神经网络所用的训练样本都看作是未确定类属的样本), 成员查询返回的类标记仅仅是由神经网络所决定的类标记. 另一种方式就是为新的样本集确定其类标记. 进行带有神喻 (Oracle) 的学习的好处在于它使得学习者可以按照需要而作出多次查询, 而且它能为那些能为获取目标概念提供最大信息的实例进行查询^[4]. 由于在 NNtoDT 算法中所用的实例的类属都由训练后的神经网络 (Oracle) 来确定, 使得 NNtoDT 所建立的决策树与训练后的神经网络的所覆盖的实例空间有最大限度的一致, 从而使决策树和神经网络之间在功能上有很高的保真性 (fidelity).

NNtoDT 在构建决策树的过程中考虑了 3 个优化原则: ① 生成最少数目的叶子——即生成最少数目的公式 (合取式), 这相当于最优覆盖问题 (MCV); ② 生成的每个叶子的深度最小——即生成最少数目选择子及属性值的公式, 这相当于最简公式问题 (MCOMP); ③ 生成的决策树叶子最少且每个叶子的深度最小——即生成具有最简公式的最优覆盖, 这相当于最优示例学习问题 (OPL).

由于 (MCV, MCOMP, OPL) 这几个问题都已证明是 NP 难题, 只能寻找其近似或启发式算法.

假设 T 是由正例集 PE 和反例集 NE 所构造的决策树, 对于 T 中任意一个非叶结点 H , 假设 H 上的测试属性为 A_j , 由 H 所发出的一个树枝上的属性值为 v_j , 这样, 如果将 H 上的属性 A_j 及树枝上的值 v_j 用一个选择子 $[A_j = v_j]$ 来表示, 那么, 从根到任一叶结点的一条路径上的所有这样的选择子的合取就构成了一条公式, 而从根到所有的标识为正例的路径对应的公式的析取即为 PE 上的规则.

在 NNtoDT 中, 在扩展决策树的每个非叶节点时, 首先用信息熵的增益率作为属性选择标准, 选择一个增益率最大的属性作为扩展属性, 然后用值聚类方法进行分枝合并.

使用信息熵的增益率来选择扩展属性, 目的是保证该非叶节点到达各后代叶节点的平均路径最短, 即满足决策树优化的第 2 个条件, 这样使得在每个非叶节点进行测试时, 能获得关于被测试例子最大的类别信息.

合并分枝是为了保证在每个非叶节点上进行扩展时, 使它发出的分枝尽量少, 即叶子尽量少, 从而满足决策树优化的第 1 个条件. 假定在决策树的某个非叶节点上, 选择某个

属性 A 作为扩展属性, A 的取值范围是 $\{V_1, V_2, \dots, V_n\}$, 假设有 m 个类属 $\{C_1, C_2, \dots, C_m\}$, 计算出 A 的各个值分别在各个类属上 C_i 的条件概率 $\text{Prob}(A = V_i / |C_i|)$, 其中 $|C_i|$ 是该非叶节点所覆盖的实例中属于类属 C_i 的个数. 对应 A 的每一取值 V_i , 选择类别号记为 T_i , 使得: $T_i = C_i$, 如果 $\text{Max}(\text{Prob}(A = V_i / |C_i|))$. 条件概率 $\text{Prob}(A = V_i / |C_i|)$ 的意义是在训练样本类别 C_i 已知的条件下, 它的属性 A 取值为 V_i 的概率. 然后, 将 A 的各值按照如下规则进行聚类: 对于任意两值 V_i, V_j , 如果 $T_i = T_j$, 那么, V_j 就聚为一类, 这样就将 A 的各个值分为 m 组, $m < n$, 在进行树的扩展时, 每一组对应一个分枝, 这样在每个非叶节点上发出的树枝就减少了.

多数的决策树归纳算法都采用局部(Local) 停止准则. NNtoDT 采用了局部(Local) 和全局(Global) 停止准则. 一个全局的停止准则考虑的是整个树的状态, 而不仅是当前被扩展的结点的状态. NNtoDT 所用的局部停止准则是基于该结点所覆盖的样本集的“纯度”(Purity), 即假如该结点被单类(single class) 样本集覆盖的概率很高, 则该结点被扩展为叶结点. NNtoDT 的全局停止准则就是对 NNtoDT 返回的决策树的大小进行控制(如决策树的结点数、树的深度等). 这些全局停止准则(参数) 使得用户在某种程度上可以控制 NNtoDT 返回的决策树的可理解性.

NNtoDT 算法的计算复杂性与传统的决策树归纳算法(ID3, C4.5) 没有什么区别, 设定 m 为样本集的大小, n 为问题领域中特征的个数. 扩展结点在树中的深度为 d , 设 v 为离散值特征的值的最大个数, 所以扩展一个结点的时间复杂性为 $O(mnv)$. 由于 NNtoDT 返回的决策树的大小可视为一个常量 $\#nodes$, 所以 NNtoDT 算法的时间复杂性为 $O(mnv * \#nodes)$.

3 实验分析

为了与 TREPAN 方法作比较, 采用了文献中提到的 2 个数据库 Heart 和 Voting, 来自美国加州大学 Irvine 分校的机器学习数据库, 其中, Heart 数据库表示心脏病病人的症状的记录; Voting 数据库是众议院议员的投票记录. 并对于 Heart 数据库, 选出 303 个实验样本数, 8 个离散特征和 5 个实值特征, 类别数为 2 个; 对于 Voting 数据库, 选出 435 个实验样本数, 15 个离散特征, 类别数为 2 个.

实验所使用的神经网络算法和 C4.5 算法均来自机器学习标准程序库 MLC++, NNtoDT 算法的实现也部分采用了 MLC++ 中的内容. 实验结果对测试样本的预测准确性、保真性和决策树的“树复杂性”(即内点数) 3 个方面进行比较.

关于测试样本的预测准确性, 对于 Heart 数据库: NN 算法为 84.5%; TREPAN * 算法为 83.2%; C4.5 算法为 74.2%; NNtoDT 算法为 83.0%. 对于 Voting 数据库: NN 算法为 92.2%; TREPAN * 算法为 91.3%; C4.5 算法为 86.7%; NNtoDT 算法为 91.0%.

NNtoDT 方法抽取的决策树比 C4.5 算法生成的决策树有更好的预测准确性, 而与 TREPAN 方法基本相同.

保真性定义为决策树与对应的神经网络分类一致的百分比. NNtoDT 方法抽取的决策树的保真性水平都超过了 96%. 这些结果说明了 NNtoDT 方法抽取的决策树对它们各自对应的神经网络提供了很好的近似.

决策树的“树复杂性”的比较, 对于 Heart 数据库: C4.5 算法 16 个; TREPAN * 算法 10.7 个^[4]; NNtoDT 算法 11 个. 对于 Heart 数据库: C4.5 算法 12 个; TREPAN * 算法 11.5

个; NNtoDT 算法 12 个.

NNtoDT 算法优于 C4.5 算法, 而与 TREPAN 算法基本相同. 考虑到 TREPAN 算法的决策树的内节点是基于 MoN 形式的, 所以其单个规则的可理解性较差 (前提因子多), 且在转换为 DNF 形式规则时要比 NNtoDT 算法产生更多的规则, 其中有不少是冗余的.

本文提出的 NNtoDT 算法比用传统方法生成的决策树具有更好的分类精度, 具有较好的通用性和可扩充性.

参考文献:

- [1] ANDREWS R, DIEDERICH J, TICKLE A. A Survey and critique of techniques for extracting rules from trained neural networks [J]. Knowledge-based Syst, 1995, 8 (6): 1~33.
- [2] TOWELL G, SHAVLIK J. Extracting refined rules from knowledge-based neural networks [J]. Machine Learning, 1993, 13 (1): 71~101.
- [3] FU L. Rule generation from neural networks [J]. IEEE Trans on Syst Man and Cyber, 1994, 24 (8): 1114~1124.
- [4] CRAVEN M, SHAVLIK J. Using neural network for data mining [J]. Future Generation Computer Systems, 1997, 13: 211~229.
- [5] TOWELL G, SHAVLIK J. Interpretation of artificial neural networks: mapping knowledge-based neural networks into rules [M]. Advances in Neural Information Processing Systems IV, 1992, 7~26.
- [6] TOWELL G, SHAVLIK J. Refining Symbolic Knowledge Using Neural Networks [J]. Machine Learning, 1994 (4): 29~44.
- [7] WNEK J, MICHALSKI R. Comparing Symbolic and Subsymbolic Learning: Three Studies [J]. Machine Learning, 1994 (4): 23~39.

A Rule Extraction Method Based on Decision Tree

*
HOU Guang-kun, ZHANG Jin-feng

Abstract: An NNtoDT algorithm is presented. NNtoDT views the problem of extracting a comprehensible hypothesis from a trained neural network as inductive learning task. In this task, the target concept is the function represented by the neural network, and the hypothesis produced by the learning algorithm is a decision tree that approximates the network. In the process of constructing tree, three optimization principles are adopted to concisely and accurately describe the knowledge that the networks have learned. The experiments demonstrate that NNtoDT is able to extract decision tree that closely approximates the hypothesis learned by the networks, provides superior predictive accuracy to tree learned directly by conventional algorithm such as C4.5, and possesses the characteristics of generality and scalability.

Keywords: rules extraction from neural networks; neural network; decision tree; clustering; inductive learning

* Department of Computer Science, Zhongshan University, Guangzhou 510275, China