

分类器组合

模式识别

成员分类器

16

58-61

多分类器组合及其应用

Classifier Combination and Its Application

韩宏 杨静宇

0235

TP391.4

(南京理工大学计算机系 南京210094)

Abstract Classifier combination has been successfully applied in many fields. From the point view of component classifier construction and combination, we review classifier combination in this paper. Four kinds of combination rules were discussed and compared theoretically. How to integrate component classifier construction and combination is the future work of classifier combination.

Keywords Classifier combination, Component classifier, Product rule, Sum rule

1 引言

传统的模式识别系统通常只使用样本的某种单一特征描述和特定的一个分类器来进行分类。这种系统对于类别数较大、输入样本带噪声的问题很难获得好的分类效果。近来发现不同的特征描述、不同的分类器在分类性能上存在着彼此互补的现象,因此同时使用多种特征描述和多个分类器可能提高分类精确性。目前,多分类器组合的研究吸引了学者们广泛的注意,并且已在许多实际应用上取得了很好的结果(如手写体识别、时间序列预测、蛋白质顺序辨识、卫星图像的地表辨识等)。但是多分类器组合中仍然存在许多尚未解决的问题,诸如能否对任意模型的分分类器组合产生一个高精度度的组合分类器,如何判断参与组合的分类器是否互补等等。

2 多分类器组合的框架

多分类器组合不仅可以提高分类的精确性和泛化能力,而且可以提高时间和空间上的效率。将一个大的特征向量空间划分为几个较小的特征空间后,在每个小特征空间上构造一个分类器,再将这些分类器组合成一个较大的分类器比在整个特征空间构造一个分类器在空间、时间上的效率高。同时,通过组合几个性能较差、结构较为简单的成员分类器还可以得到分类性能优于复杂结构分类器的组合分类器。图1即为分类器组合的框架图。

多分类器组合属于有指导学习的范畴,组合分类器首先在训练集上进行有指导的学习。这包括成员分

类器的学习和组合算法的学习(即对各成员分类器的输出结果进行学习),这两个学习过程有时是相互交织的。对未知的测试样本,首先获得参与组合的各成员分类器的分类结果,再由组合算法获得最终的结果。

为方便以下的讨论,本文采用如下记法。假设有 J 个类别 $\omega_1, \omega_2, \dots, \omega_J$, 训练数据集 T 为 $\{y_n, x_n\}, n=1, 2, \dots, N$, 其中 x_n 为特征向量, $y_n \in \{\omega_1, \omega_2, \dots, \omega_J\}$ 为 x_n 对应的类别标号, N 为训练样本数。参与组合的成员分类器为 $F = \{f_1, f_2, \dots, f_L\}$, L 为成员分类器的个数,而最终得到组合分类器为:

$$cf = \text{CombineAlgorithm}(F) = \text{CombineAlgorithm}(f_1, f_2, \dots, f_L)$$

分类器组合问题主要由两个部分构成:成员分类器的生成方法和成员分类器的组合方法。这两个部分有时是相辅相成的。参与组合的成员分类器应满足下述三个原则:

①成员分类器的精确度要高,否则组合出的分类器的精确度也不会很高。

②成员分类器的分类结果应具有多样性。若用对相同样本犯同样分类错误的成员分类器进行组合,则对组合分类器的精确度不能有任何提高。

③成员分类器的效率要高。这对于实际应用的分类任务尤其重要。

多分类器组合中最困难的问题是成员分类器很难同时满足精确性和多样性。如何平衡这两者的关系是选择成员分类器的主要工作,即如何评价成员分类器之间的相关程度(或互补程度)。

Ali 和 Pazzani^[1]提出了各成员分类器之间的总体

韩宏 博士生,研究方向为图像处理、模式识别,杨静宇 教授、博士生导师,研究方向为模式识别、计算机视觉。

错误相关度。对于训练数据集,假设存在最佳分类器 f ,即 f 对特征向量 x_i 的分类结果为 y_i 。成员分类器 f_i 与 f 犯同样错误的概率为:

$$p(f_i(x) \neq f(x))$$

各成员分类器之间的总体错误相关度为:

$$\Phi_c(F) = \frac{1}{L(L-1)} \sum_{i=1}^L \sum_{j=1}^L p(f_i(x) \neq f_j(x))$$

此总体错误相关度是所有参与组合的成员分类器两两之间犯同样错误的概率的平均值,适用于分类结果为类别编号的成员分类器。

即使确定了成员分类器间的互补程度的评价函数,如何构造(或选择)使此评价函数达到最大值(即成员分类器间的互补程度最大)的成员分类器仍然是个困难的问题,因为这包括分类器类型的选择(线性、二次或NN等等)、参数的选择、特征向量的选择等等。不过在实际应用中,通常只需考虑这些问题中的一部分。如分类器的模型通常是确定的,抽取的特征向量也是有限的几种,故面临的主要问题是有效地在这些可选项中得到一个最佳(或近似最佳)组合。

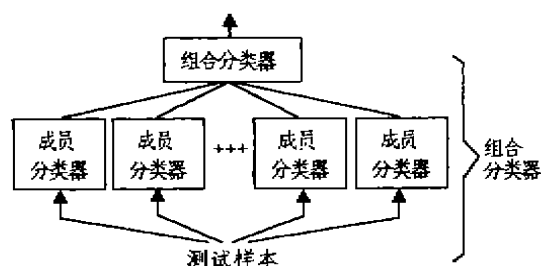


图1 分类器组合的框架

3 成员分类器的生成方法

参与组合的成员分类器的生成方法主要有以下几种:

1)对训练数据集进行处理。由 Breiman 提出的 Bagging (Bootstrap aggregating) 和 Boosting (Booststrapping) 算法是此种方法的典型。这两种算法均是对训练数据集进行抽样获得许多新的训练集,在这些抽样获得的训练集上构成相应的分类器,通常用投票法获得最终分类结果。实验表明此种算法成功地提高了一些分类器的分类精确度,但是 Breiman 指出这种方法仅当拟合算法是不稳定时才能得到较好的结果。

2)对输入特征(或预测变量)进行处理。在输入的特征向量中存在着许多高度相关的特征时,可以获得不同的特征向量子空间,在这些子空间上运行拟合算

法。例如,在文[2]中,为分析来自金星的图像,Cherkauer 运用了不同的预处理方法(包括主成分分析和傅立叶分析)获得了119个候选输入特征,基于8个不同的特征子集和4种不同的隐层单元大小,获得了32个神经网络分类器。由以上32个分类器构成的组合分类器在辨识火山的性能上可与人类专家相媲美。

3)对各分类器的输出的结果进行处理,Dietterich 和 Bakiri^[4]提出错误纠正输出码方法。此方法的大体思路如下。假设总体类别数 J 很大,则相应的分类问题能够通过随机地将 J 类划分为两个子集 A_i 和 B_i 。这样输入的数据可以按以下规则进行标号,在 A_i 中的类别被标号为1,在 B_i 中的类别被标号为0。对这些被重新标号的数据进行拟合算法,从而构成一分类器 f_i 。重复这一过程 L 次(产生不同的子集 A_i 和 B_i)。由此我们获得了参与组合的分类器 $f_1, f_2, \dots, f_L$ 。对于测试数据 x ,用所有 L 个分类器对其分类。如果 $f_i(x)=0$,所有在 A_i 的类别均得到一票;如果 $f_i(x)=1$,所有在 B_i 的类别均得到一票。当所有 L 个分类器均投过票后,获得最高票数的类别作为组合分类器的最终预测结果。

4)对拟合过程进行随机处理。在神经网络训练中,初始权值通常置为小的随机数值。通过设置不同的初始权值可以获得不同拟合效果的神经网络。文[10]中,Perone 和 Cooper 对用此方法得到的多个网络的结果进行加权投票,在回归问题上获得满意的结果,对树拟合算法进行随机处理已经被多次实验过。Ali 和 Paz-zani 应用类似 Boltzmann 分布为树节点选择每次最好的候选分裂。Kong 和 Dietterich 提出 C4.5 的变体,即在前20个最好的候选分裂中随机(同概率)选择一个进行分裂,此方法在不少大的分类任务上的性能可与 Bagging 方法相媲美(有时甚至更好)。

上述所提到的方法均是生成来自同一类型分类器的,但实际上方法1和2很容易应用到生成不同类型分类器,但在这方面还没有理论上的研究。至于如何选择来自不同类型的分类器来进行组合,如决策树、最近邻算法和线性判别函数等,现在仍然没有有效的方法进行组合或度量成员分类器之间的互补性。

4 成员分类器的组合方法及其比较

根据成员分类器的输出结果的不同,通常将组合方法分为两类。

1)所有成员分类器的输出结果具有相同的表示方法。当参与组合的成员分类器来自同一类型时,其结果自然具有相同的表示方法。如全部用神经网络分类器时,各成员分类器得到的结果均是对输入样本的后验类别概率的估计。故可以将这些成员分类器的输出结果用来估计最终的后验类别概率。

2) 所有成员分类器的输出结果具有不同的表示方法。此时各成员分类器的结果是在不同的测度的测度空间上获得的,故不能用来估计同一个函数值的后验概率。不过此种情况下,同样可以获得相同表示方法的结果,如将各分类器的结果都表示为按相似度排序的类别标号。

下面将分别对各种分类器组合方法进行研究。假设对测试样本获得 L 个测度向量 $x^1, x^2, \dots, x^L$, 它们分别是对应于 L 个成员分类器的输入向量, 对第 l 个成员分类器, 有 $f_l(\omega_j | x^l) = P(\omega_j | x^l) + e_l(\omega_j | x^l)$, $j = 1, 2, \dots, J$, 其中 $P(\omega_j | x^l)$ 是特征空间 l 上的真实后验类概率。分类器组合的最终目标是获得对 $P(\omega_j | x^1, x^2, \dots, x^L)$, $j = 1, 2, \dots, J$ 的最佳估计值。在这样的假设下, 存在两种极端情形:

1) 所有的测度空间是相同的, 即对应本节开始时的第一类分类器组合方法。此时所有的分类器使用相同的输入向量 x , 并且近似于相同的概率分布, 故有

$$P(x^1, x^2, \dots, x^L | \omega_j) = P(x | \omega_j) \quad (1)$$

$$P(x^1, x^2, \dots, x^L) = P(x)$$

其中 $x^l = x, l = 1, 2, \dots, L, j = 1, 2, \dots, J$

由 Bayes 原理,

$$P(\omega_j | x^1, x^2, \dots, x^L) = \frac{P(x^1, x^2, \dots, x^L | \omega_j) P(\omega_j)}{P(x^1, x^2, \dots, x^L)} \quad (2)$$

根据假设, 有:

$$P(\omega_j | x^1, x^2, \dots, x^L) = P(\omega_j | x^1) \quad (3)$$

其中 $x^l = x, l = 1, 2, \dots, L, j = 1, 2, \dots, J$

而 $P(\omega_j | x^1)$ 的估计值为 $f_1(\omega_j | x^1)$, 可以对这些估计值进行平均, 获得方差较小的估计值, 由于各成员分类器的估计值 $f_l(\omega_j | x^l)$ 的残差 $e_l(\omega_j | x^l)$ 不同, 故可以采取线性加权方法获取 x 后验类概率 $P(\omega_j | x^1, x^2, \dots, x^L)$ 的最佳估计值。这即为组合方法中的加法原则。可记为下式:

$$\hat{f}(\omega_j | x^1, x^2, \dots, x^L) = \sum_{l=1}^L a_l f_l(\omega_j | x^l) \quad (4)$$

其中 $\sum_{l=1}^L a_l = 1, j = 1, 2, \dots, J$

2) 所有的特征空间和测度是不同的, 且它们相互独立。此时有:

$$P(x^1, x^2, \dots, x^L | \omega_j) = P(x^1 | \omega_j) P(x^2 | \omega_j) \dots P(x^L | \omega_j), \quad j = 1, 2, \dots, J \quad (5)$$

由 Bayes 原理:

$$P(\omega_j | x^1, x^2, \dots, x^L) = \frac{P(\omega_j) \prod_{l=1}^L P(x^l | \omega_j)}{\sum_{j=1}^J P(\omega_j) \prod_{l=1}^L P(x^l | \omega_j)} \quad (6)$$

将 $P(\omega_j | x^l)$ 的估计值 $f_l(\omega_j | x^l)$ 代入 (6), 得到

$$P(\omega_j | x^1, x^2, \dots, x^L) = \frac{P(\omega_j) \prod_{l=1}^L f_l(x^l | \omega_j)}{\sum_{j=1}^J P(\omega_j) \prod_{l=1}^L f_l(x^l | \omega_j)} \quad (7)$$

上式即为组合方法中的乘法原则。

当成员分类器中有一个对类概率分布估计得很差并且包含许多零(或接近于零的)值时, 显然应用乘法原则会对最终的结果造成很大的影响, 因为这样获得的相应后验概率接近零, 尽管其他的成员分类器得到的后验概率可能很大。故乘法原则比加法原则对误差更敏感。

由上可见, 当成员分类器具有较小的估计误差并且是在几个独立的特征空间上进行分类的情况下, 使用乘法原则进行组合获得的结果较加法原则好。而在同一个特征空间上获得的成员分类器, 实验表明, 用乘法原则获得的组合结果与加法原则差别不大。乘法原则与加法原则构成了分类器组合的基本方法。

分类器组合的乘法原则与加法原则可以由关系式 (8) 中相应的上限、下限来逼近, 而且对后验概率 $P(\omega_j | x_l)$ 也可以由下式转化为二值函数:

$$\prod_{l=1}^L P(\omega_j | x_l) \leq \min_{l=1}^L P(\omega_j | x_l) \leq \frac{1}{R} \sum_{l=1}^L P(\omega_j | x_l) \leq \max_{l=1}^L P(\omega_j | x_l) \quad (8)$$

$$\Delta_j = \begin{cases} 0, & \text{如果 } P(\omega_j | x_l) = \max_{k=1}^J P(\omega_k | x_l) \\ 1, & \text{其他} \end{cases} \quad (9)$$

由式 (9), 可获得以下组合原则——投票原则:

如果 $\sum_{l=1}^L \Delta_j = \max_{k=1}^J \sum_{l=1}^L \Delta_k$, 则将此测试样本划分为类别 ω_j 。

上式中右边的求和是成员分类器对各类别的投票和, 获得最多票数的类别为最终的综合结果。

对应本节开始时的第二类分类器组合问题, 文 [6] 提出了相应的解决方案, 它将各成员分类器的输出结果都转化为按相似度排序的类别标号, 对排序过的类别标号集合, 文 [6] 中提出两种组合方法——类标号集合缩减和类标号集合重排。类标号集合缩减的目的是从类标号集合中抽取一个子集, 使其尽量小并同时仍然包含真正的类标号。类标号集合重排的目的是对给定的类别获得一致的排序, 并使正确的类标号尽可能地排在前面。这种组合方法的应用范围很广, 因为它不受成员分类器输出结果形式的影响。从统计意义上, 它是对各成员分类器结果的秩统计量进行处理, 故称此种组合方法为秩原则。尽管实际应用中存在着许多种组合方法, 但它们的基本出发点大多是基于以上四种原则。

目前多分类器组合的大部分研究工作集中于——

如何根据不同的多个分类器所给出的识别结果以及结果的置信度,选择最优的结果从而达到提高识别率之目的。在一些对识别速度敏感的应用场合,分类器的增加虽然能够有效地提高识别率,但却会较大地影响识别速度,造成性能的下降。文[11]中提出的方法就是针对这一问题提出的解决方案。此方案能够根据对输入样本影响分类器性能的某些特征——性能预测特征,动态地选择最优的分类器集合,使这样选择的分类器进行组合时能够在统计上使误差与识别消耗时间所造成的代价最小。

结束语 本文从成员分类器的生成方法和组合方法两个角度讨论了多分类器组合的问题,评述了近年来的研究发展,从理论上讨论了多分类器组合的四种原则,同时对它们作了一定的比较。我们认为今后多分类器组合研究的主要重点应当集中于从理论上提出更为完善的分析,如怎样更好地衡量各成员分类器之间相关程度,这是多分类器组合中的一个关键问题。目前对多分类器组合的研究通常是分别讨论成员分类器的生成和组合问题,如何将这两个问题有机地结合起来也应是今后研究的主要领域之一。

参考文献

- 1 Ali K M, Pazzani M J. Error Reduction through Learning Multiple Descriptions. *Machine Learning*, 24, 173~302
- 2 Cherkauer T G. Human Expert-Level Performance on A Scientific Image Analysis Task by A System Using Combined Artificial Neural Networks. In: Chan P, ed. Working Notes of the AAAI Workshop on Integrating Multiple Learned Models. AAAI Press, Menlo Park, CA
- 3 Tax D M J, et al. Comparison between Product and Mean Combination Rules
- 4 Dietterich T G, Bakiri G. Solving Multiclass Learning Problems via Error-Correcting Output Codes. *J. Artificial Intelligence Res.*, 1995, 2: 263~286
- 5 Dietterich T G, Kong E B. Machine Learning Bias, Statistical Bias, and Statistical Variance of Decision Tree Algorithms. [Technical Report]. Dept. Computer Science, Oregon State Univ., Corvallis, Oregon
- 6 Ho T K, et al. Decision Combination in Multiple Classifier Systems. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 1994, 16(1): 66~75
- 7 Ho T K. The Random Subspace Method for Constructing Decision Forests. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 1998, 20(8)
- 8 Ji C, Ma S. Combinations of Weak Classifiers. *IEEE Transactions on Neural Networks*, 1997, 8(1): 867~888
- 9 Kutler J, et al. On Combining Classifiers. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 1998, 20(3)
- 10 Kong E B, Dietterich T G. Error-Correcting Output Coding Corrects Bias and Variance. In: Priditis A, Russell S, eds. Twelfth International Conference on Machine Learning. Morgan Kaufmann, San Francisco, 1993
- 11 Perrone M P, Cooper L N. When Networks Disagree Ensemble Methods for Hybrid Neural Networks. In: Mammon R J, ed. *Neural Networks for Speech and Image Processing*. Chapman and Hall, London
- 12 征荆, 丁晓青, 吴佑青. 基于最小代价的多分类器动态集成. *计算机学报*, 1999, 22(2)

(上接第64页)

Givens 算法和2次 BP 算法的训练,最后确定隐层神经元个数为5,网络的实际输出见表7。

结束语 实验表明本文提出的正交神经网络建模方法是一种网络结构可优化设计的、高效的、具有自适应处理的、简单可行的动态神经网络建模方法。

参考文献

- 1 Hornik K, Stinchcombe M. Multilayer Feedforward Networks Are Universal Approximation. In: White H, ed. *Artificial Neural Networks: Approximation and Learning Theory*. Blackwell Publishers, 1992. 13~28
- 2 Narendra K S, Mukhopadhyay S. Adaptive Control of Nonlinear Multivariable Systems Using Neural Networks. *Neural Networks*, 1994, 7(5): 737~752
- 3 孙德保, 高超. 一种实用的克服极小的 BP 算法研究. *信息与控制*, 1995(5): 283~287
- 4 Yang S S, Tseng C S. An Orthogonal Neural Network for Function Approximation. *IEEE Trans. on System, Man, and Cybernetics, Part B*, 1996(5): 779~785
- 5 肖丹, 施渝萍. 正交投影神经网络. *计算机学报*, 1996(9): 673~678
- 6 肖少拥, 冯树椿, 胡上序. 基于正交变换的神经网络学习算法. *清华大学学报(自然科学版)*, 1998, 38(S2): 148~149
- 7 Golub G H, VanLoan C F. *Matrix Computation*. Johns Hopkins University Press, 1983