支持向量机加权类增量学习算法研究

秦玉平[1,2],李祥纳[2],王秀坤[1],王春立[1]
QIN Yu-ping[1,2],LI Xiang-na[2],WANG Xiu-kun[1],WANG Chun-Li[1]

1.大连理工大学 电子与信息工程学院,辽宁 大连 116024
2.渤海大学 信息科学与工程学院,辽宁 锦州 121000
1.School of Electronic and Information Engineering,Dalian University of Technology,Dalian,Liaoning 116024,China
2.College of Information Science and Technology,Bohai University,Jinzhou,Liaoning 121000,China

QIN Yu-ping,LI Xiang-na,WANG Xiu-kun,et al.Study on weighted class-incremental learning algorithm for support vector machines.Computer Engineering and Applications,2007,43(34):177-179.

Abstract: In order to solve the misclassification problem resulted from the imbalance of the number of training samples of different classes in the process of class-incremental learning for support vector machine,presents a weighted class-incremental learning algorithm.It uses one against rest training method to construct a new binary classifier takes all the samples from known classes as negative and that of the new class as positive,also introduces weight factors for classes according to the proportion of the training samples,which can effectively improve the classification accuracy of class that has fewer samples.The experiment shows that the result of this method is effective.

Key words: Support Vector Machines(SVM);class-incremental learning;classification algorithm;weight

摘 要: 针对支持向量机类增量学习过程中参与训练的两类样本数量不平衡而导致的错分问题，给出了一种加权类增量学习算法,将新增类作为正类,原有类作为负类,利用一对多方法训练子分类器,训练时根据训练样本所占的比例对类加权值,提高了小类别样本的分类精度。实验证明了该方法的有效性。

关键词: 支持向量机;类增量学习;分类算法;加权
文章编号: 1002-8331(2007)34-0177-03 **文献标识码:** A **中图分类号:** TP311

1 引言

支持向量机增量学习通过保留历史训练结果,对新增数据进行再学习,有效地解决了因数据增大或新样本出现而导致时间和空间复杂度增加问题。支持向量机增量学习可以归结为两种,一种是在类别不变情况下增加原有类别样本学习,另一种是类增量学习。

对于第一种情况,有很多增量学习算法,如增量训练精确解[1]、Divisional Training SVM algorithm[2]、快速增量学习算法[3]、α-ISVM 算法[4]和基于中心距离比值的增量支持向量机(CDR-ISVM)[5]等,这些算法都是针对二分类问题提出的,对于多分类问题,现有的方法多数都是通过组合多个二值分类器实现对多类分类器的构造[6-9],如一对多方法(1-v-r),对于 k 类,将其中的每个类和其它所有类分开,共得到 k 个分类器,此方法简单、有效,训练时间较短,适合大规模数据。但是,当类别数较大时,某一类的训练样本将大大少于其它类训练样本的总和,这种训练样本间的不均衡将影响分类精度。一对一方法(1-v-1)是将 k 类中的任意两个类分开,共得 $k(k-1)/2$ 个分类器,其优点是单个 SVM 容易训练，缺点是分类器数量会随类别数量增加而急剧增加,导致在决策时速度较慢。此外还有有向无环图多类 SVM 分类法(DAG-SVMs)、层次支持向量机(H-SVMS)、纠错编码支持向量机(ECC-SVM)等。

对于类增量学习问题,文献[10]提出了 CIL 算法,该算法的思想是将新增类样本作为正类,原有类样本作为负类,训练得到的二值分类器作为二叉树的根结点。使用该算法进行增量学习,能大大减少训练时间,但随着新类别的不断增加,新增类别的训练样本数将会远远少于原有类训练样本总和,致使两类训练样本数量严重不均衡,导致分类精度降低。为此,本文对 CIL 算法作了改进,给出了加权类增量学习算法(加权 CIL 算法),依据每类样本在总样本中所占的比例,对参加训练的两类样本分别加类权值,有效解决了由于两类样本数量的不平衡而造成分类精度较低的问题。

本文第 2 章介绍了类加权支持向量机数学模型,第 3 章详细阐述了加权类增量学习算法，第 4 章给出了在 Reuters 21578 标准语料库上的实验结果,第 5 章给出结论。

基金项目: 国家自然科学基金(the National Natural Science Foundation of China under Grant No.60603023);国家重点基础研究发展规划(973)(the National Grand Fundamental Research 973 Program of China under Grant No.2001CCA00700)。

作者简介: 秦玉平(1965-),男,博士研究生,教授,主要研究领域为机器学习;李祥纳,女,硕士研究生,主要研究领域为机器学习;王秀坤(1945-),女,博导,教授,主要研究领域为数据库系统;王春立(1972-),女,博士,教授,主要研究领域为模式识别。

2 类加权支持向量机

设给定训练样本集$\{x_i,y_i\}_{i=1}^{l}$和核函数$K(x_i,x_j)$,其中,$x_i\in R^n$,$y_i\in\{-1,1\}$,K对应某特征空间Z中的内积,即$K(x_i,x_j)=\langle g(x_i),g(x_j)\rangle$,变换$g:X\mapsto Z$将样本从输入空间映射到特征空间,类加权支持向量机的数学模型为:

$$\text{Min}\ \frac{1}{2}\|\boldsymbol{w}\|^2+c\lambda_{y_i}\sum_{i=1}^{l}\xi_i \tag{1}$$

$$\text{s.t.}\ y_i(\boldsymbol{w}x_i+\boldsymbol{b})\geqslant 1-\xi_i,x_i\in R^n,y_i\in\{\pm 1\},i=1,\cdots,l \tag{2}$$

其中,$\boldsymbol{w}$为超平面的法向量,$\boldsymbol{b}$为超平面的偏置,ξ_i是松弛变量,C为指定的惩罚因子,λ_{y_i}为类别权重($y_i=+1$代表正类,$y_i=-1$代表负类)。

其对偶问题为:

$$\text{Max}\ \sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{l}y_iy_j\alpha_i\alpha_jK(x_i\cdot x_j) \tag{3}$$

$$\text{s.t.}\quad 0\leqslant\alpha_i\leqslant c\lambda_{y_i} \tag{4}$$

$$\sum_{i=1}^{l}\alpha_iy_i=0,i=1,\cdots,l \tag{5}$$

其中α为Lagrange乘子。对偶问题的最优解$\alpha=\{\alpha_1,\alpha_2,\cdots,\alpha_l\}$使得每一个样本$x_i$都要满足优化问题的KKT条件[11]:

$$\alpha_i=0\Rightarrow y_if(x_i)\geqslant 1 \tag{6}$$

$$0<\alpha_i<c\lambda_{y_i}\Rightarrow y_if(x_i)=1 \tag{7}$$

$$\alpha_i=c\lambda_{y_i}\Rightarrow y_if(x_i)\leqslant 1 \tag{8}$$

当$\alpha_i=0$时,对应的样本在分类器的间隔以外;

当$0<\alpha_i<c\lambda_{y_i}$时对应的样本位于分类间隔之上,称为边界向量。

当$\alpha_i=c\lambda_{y_i}$时,位于分类器间隔之内。

3 加权类增量学习算法

设有k类样本,第m类样本集$A^m=\{x_i^m,y_i^m\}_{i=1}^{l^m}$,前$m$类样本集$S^m=A^1\cup A^2\cup\cdots\cup A^m$,$l^m$是第$m$类样本的数量,$U^m=\sum_{i=1}^{m}l^m$是前$m$类的样本总数。$K$对应某特征空间$Z$中的内积,即$K(x_i^m,x_j^m)=\langle g(x_i^m),g(x_j^m)\rangle$,变换$g:X\mapsto Z$是将样本从输入空间映射到特征空间。$SC_m=\{f_m,f_{m-1},\cdots,f_i,\cdots,f_3,f_2\}$是前$m$类样本的训练结果,其中$f_i$是以$A^i$作为正类、$S^{i-1}$作为负类训练后得到的二值分类器。在训练分类器$f_i$时,根据正类样本和负类样本所占的比例对两类样本加类权值,权值为:

$$\lambda_{y_i=1}=\frac{U^{i-1}}{l^i+U^{i-1}} \tag{9}$$

$$\lambda_{y_i=-1}=\frac{l^i}{l^i+U^{i-1}} \tag{10}$$

初始加权类增量学习算法如下:

步骤 1 $m=1,SC_1=\Phi$;

步骤 2 $m=m+1$,若$m>k$,转步骤4,否则转步骤3;

步骤 3 以A^m作为正类,以S^{m-1}作为负类,根据公式(9)、(10)计算类权值,训练二值分类器f_m,$SC_m=SC_{m-1}\cup f_m$,转步骤2;

步骤 4 结束。

类增量学习算法的主要目的是实现新类的随时加入。若有新增类样本集A^{k+1}加入,首先对特征集进行扩充。设样本集A^{k+1}的特征集为T^{k+1},样本集S^k的特征集为Γ^k。将两个样本集的特征集都扩充为$T^{k+1}\cup\Gamma^k$,扩充部分的特征值取0。然后以A^{k+1}作为正类,以S^k作为负类,根据公式(9)、(10)计算类权值,训练二值分类器f_{k+1},$SC_{k+1}=SC_k\cup f_{k+1}$。

对于待分类样本x,根据SC_{k+1}进行分类,其具体过程如下:

步骤 1 $m=k+1$;

步骤 2 用分类器f_m对x进行分类,若x属于正类,则x属于第m类,转步骤4,否则转步骤3;

步骤 3 $m=m-1$,若$m\geqslant 2$,转步骤2,否则转步骤4;

步骤 4 分类结束。

4 实验结果及分析

本文使用标准数据集Reuters 21578,从中选取4类共821篇文本进行实验分析。使用其中的548篇作为训练样本,其余的273篇作为测试样本。将文本数据经过预处理后形成高维词空间向量,采用信息增益的方法来进行特征降维,向量中每个词的权重根据*tf-idf*公式计算。实验环境为Pentium 1.6 G,512 M内存,采用线性核函数,系统参数$C=10$。算法实现参考了Chang和Lin所开发的libsvm[12],并在此基础上进行了相应修改,见表1。

表1 训练语料和测试语料

类别	wheat	corn	coffee	soybean
训练集规模	204	168	97	79
测试集规模	101	84	48	40

实验中采用通用的准确率、召回率和F_1值作为评价指标。

$$\text{准确率}(P)=\frac{\text{分类正确的文本数}}{\text{实际分类的文本数}}$$

$$\text{召回率}(R)=\frac{\text{分类正确的文本数}}{\text{应有的文本数}}$$

$$F_1=\frac{\text{准确率}\times\text{召回率}\times 2}{\text{准确率}+\text{召回率}}$$

其中,每一类的准确率、召回率和F_1值称为微平均;所有类的准确率、召回率和F_1值称为宏平均。

实验中初始样本集含有两类(第一类为wheat,第二类为corn,$k=2$),表2给出了增加第三类(coffee)后CIL算法和加权CIL算法的F_1值比较,表3给出了增加第四类(soybean)后CIL算法和加权CIL算法的F_1值比较。

表2 加入第三类后CIL算法和加权CIL算法的F1值比较

F_1	第一类(微平均F_1)	第二类(微平均F_1)	第三类(微平均F_1)	宏平均F_1
CIL算法	79.28%	70.67%	95.74%	81.90%
加权CIL算法	81.69%	76.73%	95.74%	84.72%

表3 加入第四类后CIL算法和加权CIL算法的F1值比较

F_1	第一类(微平均F_1)	第二类(微平均F_1)	第三类(微平均F_1)	第四类(微平均F_1)	宏平均F_1
CIL算法	75.77%	58.11%	95.74%	54.74%	71.34%
加权CIL算法	78.00%	65.75%	95.74%	56.60%	74.30%

由于类权值主要由样本数比例决定,所以新增类样本数所占的比例越小,赋予该类的权值就越大,判别函数在分类时就越倾向该类,该类的分类精度就会越高。从实验结果可以看出,本文给出的加权 CIL 算法与 CIL 算法相比, 平均 F_1 值有明显提高,由于每次训练都要加类权值,所以训练时间要比 CIL 算法略高(见表 4),但对分类速度没有影响。

表 4 CIL 算法和加权 CIL 算法训练时间比较

新增类别	3	4
CIL 算法	32	312
加权 CIL 算法	47	488

5 结论

本文对 CIL 算法进行了改进,提出了加权 CIL 算法,有效地解决了多类别增量学习问题,尤其适合新类样本和原有类样本数目差别较大,且对新类样本分类精度要求较高的情况。此外,当新类别进入时,该算法只需再训练一个二分类器,训练时间较低。但随着新类的不断加入,原有类的样本数量将越来越大,如何有效地减少原有样本参加训练的数量是本文进一步研究的内容。(收稿日期:2007 年 8 月)

参考文献:

[1] Cauwenberghs G,Poggio T.Incremental and decremental support vector machine[J].Machine Learning,2001,44(13):409-415.

[2] Zhang Jin-pei,Li Zhong-wei,Yang Jing.A divisional incremental training algorithm of support vector machine[C]//Proceeding of the IEEE International Conference on Mechatronics and Automation,Canada,2005,8:853-855.

[3] 孔锐,张冰.一种快速支持向量机增量学习算法[J].控制与决策,2005,20(10):1129-1132.

[4] 萧嵘,王继成,孙正兴.一种 SVM 增量学习算法 α-ISVM[J].软件学报,2001,12(12):1818-1824.

[5] 孔波,刘小茂,张钧.基于中心距离比值的增量支持向量机[J].计算机应用,2006,26(6):1434-1436.

[6] 刘志刚,李德仁,秦前清,等.支持向量机在多类分类问题中的推广[J].计算机工程与应用,2004,40(7):10-13.

[7] Hsu C W,Lin C J.A comparison of methods for multiclass support vector machines[J].IEEE Transactions on Neural Networks,2002,13(2):415-425.

[8] Krebel U G.Pairwise classification and support vector machines[C]//Advances in Kernel Methods:Support Vector Learning.Cambridge,MA:MIT Press,1999:255-268.

[9] Platt J,Cristianini N,Shawe-Taylor J.Large margin DAGs for multiclass classification[C]//Advances in Neural Information Processing Systems.Cambridge,MA:MIT Press,2000:547-553.

[10] Zhang Bo-feng,Su Jin-shu,Xu Xin.A class-incremental learning method for multiclass support vector machines in text classification[C]//Proceedings of the Fifth International Conference on Machine Learning and Cybernetics,Dalian,2006:13-16.

[11] 曾文华,马健.支持向量机增量学习的算法与应用[J].计算机集成制造系统-CIMS,2003,9(21):144-148.

[12] Chang Chinchang,Lin Chihjen.LIBSVM:a library for support vector machines[EB/OL].[2005].http://www.csie.ntu.tw/~cjlin/libsvm.

(上接 158 页)

5 结束语

多重密钥共享是现代密码学的一个重要工具,在现实系统中使用多重密钥共享,有利于保护系统密钥,减小密钥持有者的责任,降低敌手破译密钥的成功率。Harn 方案[5]、Chen 方案[9]、Shi 方案[10]给出了有效的多重密钥共享方案,但是都存在一些不足。本文提出的动态安全的多重密钥门限共享方案可以动态的更新成员子密钥,攻击者无法根据当前时段的子密钥计算前驱或者后继时段的成员子密钥,在动态更新时执行 Feldman 的 VSS 方案,能够有效地防止成员欺骗和管理者欺骗,方案中各阶段执行多项式时间算法, 其实现性能优于目前已有的方案。(收稿日期:2007 年 7 月)

参考文献:

[1] Shamir A.How to share a secret[J].Communications of the ACM,1979,22(11):612-613.

[2] Blakley G R.Safeguarding cryptographic keys[C/OL]//Proceedings of AFIPS 1979 National Computer Conference,1979,48:313 -317.http://citeseer.nj.nec.com/contest.

[3] He J,Dawson E.Multistage secret sharing based on one-way function[J].Electronic Letters,1994,30(19):1591-1592.

[4] He J,Dawson E.Multisecret-sharing scheme based on one-way function[J].Electronic Letters,1995,31(2):93-95.

[5] Harn L.Efficient sharing(broadcasting)of multiple secrets[J].IEE Computersand Digital Techniques,1995,142(3):237-240.

[6] Chang C C,Hwang R J.Efficient cheater identification method for threshold schemes[J].IEE Computers and Digital Techniques,1997,144(1):23-27.

[7] Karnin E D,Greene J W,Hellman M E.On secret sharing systems[J].IEEE Transactions on Informational Theory,1983,IT-29(1):35-41.

[8] Mceliece R J,Sarwate D V.On sharing secrets and Reed-Soloman code[J].Communications of the ACM,1981,24(3):583-584.

[9] Chen L,Gollmann D,Mitchell C J,et al.Secret sharing withreusable polynomials[C]//Proceeding of the 2nd Australasian Conference on Information Security and Privacy,Canberra,Australia,1997:183-193.

[10] Shi R H.A multisecret sharing authenticating scheme[J].Chinese Journal of Computers,2003,26(5):552-556.

[11] Wang Gui-lin.Analysis and improvement of a multisecret sharing authenticating shceme[J].Journal of Software,2006,17(7):1627-1623.

[12] Pedersen T P.Non-interactive and information-theoretic secure verifiable secret sharing[C]//LNCS 576:Advances in Cryptography-CRYPTO,1991:129-140.

[13] Feldman P.A practical scheme for non-interactive verifiable secret sharing[C]//Proc of IEEE Fund of Comp Sci,1987:427-437.

[14] Frankel Y,Gemmell P,MacKenzie P D,et al.Optimal-resilience proactive public-key cryptosystems[C]//IEEE Symposium on Foundations of Computer Science,1997:384-393.