计算机研究与发展
Journal of Computer Research and Development
ISSN 1000-1239/CN 11-1777/TP
42(增刊):361~366,2005

基于最小最大模块化支持向量机的多标号文本分类

吕宝粮[1] 刘峰耀[1] 内山将夫[2] 井佐原均[2]
[1](上海交通大学计算机科学与工程系 上海 200030)
[2](日本国立信息与通信技术研究所 京都 619-0289)
(blu@cs.sjtu.edu.cn)

Multilabel Text Categorization Using a Min-Max Modular Support Vector Machine

Lü Baoliang[1], Liu Fengyao[1], Utiyama Masao[2], and Isahara Hitoshi[2]
[1](*Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai* 200030)
[2](*National Institute of Information and Communications Technology, Kyoto* 619-0289)

Abstract In this paper, a new method for dealing with multilabel text categorization problems is proposed. First, a K-class multilabel problem is divided into K two-class problems with the "one-versus-rest" strategy. Then the "part-versus-part" strategy is used to divide each two-class problem into a number of relatively balanced two-class subproblems which are as small as needed. The experimental results on Yomiuri News corpus and RCV1-v2 corpus indicate that the proposed method is superior to the traditional approach in the aspects of generalization performance and training time.

Key words multilabel text categorization; min-max modular support vector machine; parallel learning

摘 要 提出了一种新的解决多标号文本分类问题的方法.对于一个 K 类多标号问题,首先采用"一对其他"的问题分解方法将原问题分解为 K 个两类问题;然后按照最小最大模块化支持向量机(M^3-SVM)的"部分对部分"问题分解方法,再对这些两类问题进一步分解.这种方法的特点是能将大规模、训练样本极不平衡的两类问题分解成用户希望的任意大小的相对平衡的两类问题,并能容易地实现并列学习.对读卖新闻日文数据集和路透社英文数据集进行了文本分类实验,实验结果表明,该方法比传统的方法具有更好的泛化能力和更短的训练时间.

关键词 多标号文本分类;最小最大模块化支持向量机;并列学习

中图法分类号 TP391

1 引 言

文本分类是指根据文档的内容和属性,将大量的文档自动归到不同的文档类别的过程.在大多数情况下,一个文档可能同时属于多个文档类别.比如读卖新闻数据集,一个文本最多同时属于3个类别,即有3个标号,平均一个文本大约有1.75个标号.通常把这种多标号文本分类问题称之为多标号文本分类.

目前,对于多标号文本分类问题的处理,主要采用的是"一对其他"的问题分解方法[1],即将一个 K

收稿日期:2005-06-15
基金项目:国家自然科学基金项目(60375022,60473040)

类多标号问题分解为 K 个两类问题,每个两类问题可以采用传统的模式分类器进行处理,比如朴素贝叶斯、神经网络、最近邻法和支持向量机(SVM)[2]等. 由于 SVM 具有比其他模式分类器更好的学习和泛化能力,本文将采用 SVM 作为多标号文本分类系统的子模式分类器. 对于一个测试文本,K 个模式分类器可以回答出该文本到底属于哪几个文档类别.

但是,这种"一对其他"的问题分解方法主要存在两个问题:①每个子模式分类器的问题规模与原问题相同,而传统的 SVM 其训练时间和训练样本个数的平方成正比,因此,对于大规模多标号文本分类问题,训练时间将很长;②这种问题分解方法使每个两类问题的正负样本极不平衡,即正样本数远少于负样本数,因此,将降低训练后的 SVM 的泛化能力.

为了解决上述问题,本文提出了一种新的处理多标号文本分类问题的方法,即在进行"一对其他"任务分解之后,再利用"部分对部分"的问题分解策略. 也就是说,在将原问题分解为 K 个两类问题之后,采用最小最大模块化支持向量机[3]来解决每个两类问题. 这种方法的优点是不仅可以使子问题规模可控,提高并行性,而且可以使问题的平衡度得到提高,最终提高整个分类器的泛化能力.

本文内容按下列顺序组织:第 2 节描述多标号文本分类问题的分解策略;第 3 节简要介绍最小最大模块化支持向量机;第 4 节对路透社和读卖新闻两大数据集进行文本分类实验,并对实验结果进行分析、比较;第 5 节给出结论.

2 多标号文本分类问题的分解

解决多标号文本分类问题的关键是如何把原问题分解成一系列等价的、SVM 可以直接处理的子问题. 本文使用了两种问题分解策略,首先是"一对其他"策略,然后对两类问题使用"部分对部分"策略.

2.1 "一对其他"策略

"一对其他"问题分解方法是一种传统的解决多类支持向量机的方法[4]. 给定一个 K 类多标号问题 T,其训练样本集为

$$\chi = \bigcup \{(x_m, t_m)\}_{m=1}^{l}, t_m = \bigcup t_m^k, k = 1, \cdots, \tau_m, \tag{1}$$

其中,$x_m \in R^n$ 是输入样本;t_m^k 是样本 x_m 的第 k 个标号;t_m 是样本 x_m 的标号集;τ_m 是 x_m 标号集中标号的个数;l 是总训练样本个数.

"一对其他"方法把一个 K 类多标号问题 T 分解为 K 个两类问题 $T_i, i = 1, \cdots, K$,其训练样本集为

$$\chi_i = \{(x_m^{i+}, +1)\}_{m=1}^{l_i^+} \cup \{(x_m^{i-}, -1)\}_{m=1}^{l_i^-}, \tag{2}$$

其中,l_i^+ 是两类问题 T_i 的正样本个数;l_i^- 是负样本个数;$l_i^+ + l_i^- = l$,$x_m^{i+} \in \{x_m \mid (x_m, t_m) \in \chi, C_i \in t_m\}$ 是 T_i 的正类样本;$x_m^{i-} \in \{x_m \mid (x_m, t_m) \in \chi, C_i \notin t_m\}$ 是其负类样本,即 T_i 的正类训练样本是拥有类标号 C_i 的样本,而负类样本是不拥有类标号 C_i 的样本.

这样,就将一个 K 类多标号问题分解成 K 个两类问题,可以使用 K 个模式分类器进行并行学习.

2.2 "部分对部分"策略

"部分对部分"策略[3]是一种将两类问题细分的方法,它按照最小最大模块化(M^3)神经网络的问题分解策略[5],将一个多类问题分解为一系列规模较小的两类问题. 对于多标号文本分类问题,我们的主要任务是在完成"一对其他"的问题分解后再对两类问题进行分解. 因此,下面对两类问题的分解过程进行详细描述.

对于一个两类问题 T_i,χ_i^+ 表示其正的训练样本集,χ_i^- 表示其负的训练样本集:

$$\chi_i^+ = \{(x_m^{i+}, +1)\}_{m=1}^{l_i^+}, \chi_i^- = \{(x_m^{i-}, -1)\}_{m=1}^{l_i^-}. \tag{3}$$

根据文献[3],样本集 χ_i^+ 和 χ_i^- 又可以分别被分解为 N_i^+ 和 N_i^- 个子样本集:

$$\chi_{ij}^+ = \{(x_m^+, +1)\}_{m=1}^{l_{ij}^+}, \text{且 } j = 1, \cdots, N_i^+, \tag{4}$$

$$\chi_{ij}^- = \{(x_m^-, -1)\}_{m=1}^{l_{ij}^-}, \text{且 } j = 1, \cdots, N_i^-, \tag{5}$$

其中,l_{ij}^+ 和 l_{ij}^- 分别是样本集 χ_{ij}^+ 和 χ_{ij}^- 中的样本个数 $\bigcup_{j=1}^{N_i^+} \chi_{ij}^+ = \chi_i^+, 1 \leqslant N_i^+ \leqslant l_i^+$,和 $\bigcup_{j=1}^{N_i^-} \chi_{ij}^- = \chi_i^-, 1 \leqslant N_i^- \leqslant l_i^-$.

将正负样本集 χ_i^+ 和 χ_i^- 分别分解为 N_i^+ 和 N_i^- 个子样本集后,最初的两类问题 T_i 被分解为 $N_i^+ \times N_i^-$ 个相对较小、并且比较平衡的两类子问题 $T_i^{u,v}$:

$$T_i^{u,v} = \chi_{iu}^+ \cup \chi_{iv}^-, \tag{6}$$

且 $u=1,\cdots,N_i^+$, $v=1,\cdots,N_i^-$.

采用"部分对部分"任务分解策略将两类问题细分之后,所有两类子问题在学习阶段是相互独立的.因此可以采用 M^3-SVM 对子问题进行并行学习. 图 1 给出了用最小最大模块化支持向量机解决 K 类多标号文本分类问题的分类器结构.

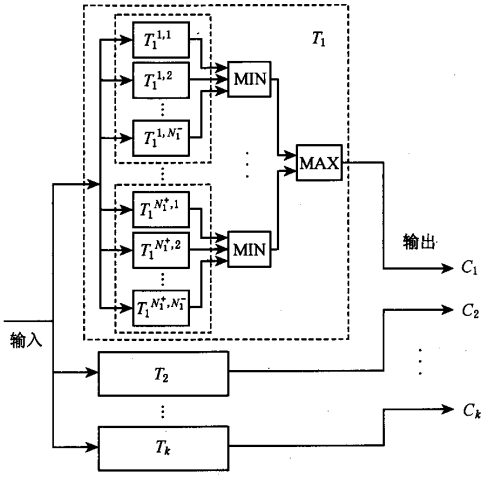


图 1 利用最小最大模块化支持向量机解决 K 类多标号文本分类问题的分类器结构

3 最小最大模块化支持向量机

最小最大模块化支持向量机将一个复杂的多类分类问题,分解成许多规模小的、各自独立的两类子问题,对于每一个两类子问题训练出一个支持向量机.然后根据两个模块组合规则,即最小化规则和最大化规则,将所有模块组合成一个 M^3-SVM.

对于两类问题 T_i,采用第 2.2 节的"部分对部分"策略将问题分解后,可使用并行方式对所有子问题进行学习,然后根据下面两个模块组合规则[3,5],利用 N_i^+ 个 MIN 单元和一个 MAX 单元,将 $N_i^+ \times N_i^-$ 个支持向量机组合成一个 M^3-SVM.

$$T_i^u(x)=\min_{v=1}^{N_i^-} T_i^{u,v}(x) \text{ 和 } T_i(x)=\max_{u=1}^{N_i^+} T_i^u(x), \tag{7}$$

其中,$T_i^{u,v}(x)$表示对应子问题 $T_i^{u,v}$ 训练的支持向量机的传递函数,$T_i^u(x)$表示将 N_i^- 个子支持向量机运用 MIN 单元集成后的支持向量机的传递函数,$T_i(x)$是对应两类问题 T_i 的 M^3-SVM 的传递函数.

M^3-SVM 泛化能力的高低与"部分对部分"问题分解策略密切相关. 这里我们考虑两种问题分解策略,一种是随机分解,另一种是超平面分解.

随机分解是一个非常简单、直接的问题分解方法. 它随机地从正负训练样本集中抽选出给定数量的样本,组合成一系列规模较小且正负样本相对平衡的子问题集作为 M^3-SVM 系统的任务集. 这种方法的特点是实现起来非常简单,且不需要样本集的先验知识. 但是,分解会改变子样本集的分布属性. 特别是当两类问题被分解成很多很小的两类问题时,系统的泛化能力会有所下降. 本文把使用这种问题分解方法的 M^3-SVM 称为 R-M^3-SVM.

一个合理的问题分解方法应该使分解后的问题尽可能不改变原训练集的样本分布属性. 基于这种考虑,我们提出了超平面分解方法[6]. 超平面分解的主要思想是引入一个特定的超平面 $\boldsymbol{H}:\boldsymbol{A}z=0$,这里,$\boldsymbol{A}=[a_1,\cdots,a_n]$是超平面 $\boldsymbol{H}$ 的法向量. 然后利用一系列和该超平面平行的超平面集将正负样本集按照需要分解成一系列子集,再两两组合成子问题集. 本文把使用这种问题分解方法的 M^3-SVM 称为 H-M^3-SVM. 为了实现上述超平面分解方法,我们采用的是样本排序法,且选用 $\boldsymbol{A}=[1,1,\cdots,1]$[7].

4 仿真实验

为了验证所提方法在多标号文本分类问题中的有效性,本节在下列两个数据集上进行了一系列仿真实验.

读卖新闻(Yomiuri News)数据集[8]:该数据集包含了 2190512 篇日文新闻文档(1987 年至 2001 年). 我们选取从 1996 年到 2000 年的 913118 篇文档作为训练集,选取 2001 年 7 月到 12 月的 181863 篇文档作为测试集. 采用 CHI(χ^2)特征提取方法将样本向量的维数降为 5000 维. 该数据集包含的类别数为 75 类,本文只选用其中的前 10 类进行仿真实验.

路透社(RCV-v2)数据集[9]:该数据集总共包含 804414 篇英文文档,我们同样按照年代顺序将该数据集分成训练集和测试集,训练集包含 23149 篇文档,测试集包含 781265 篇文档. 这里按照文章主题(topics)对文章进行分类,共包含 101 个主题类别,样本向量的维数为 47152.

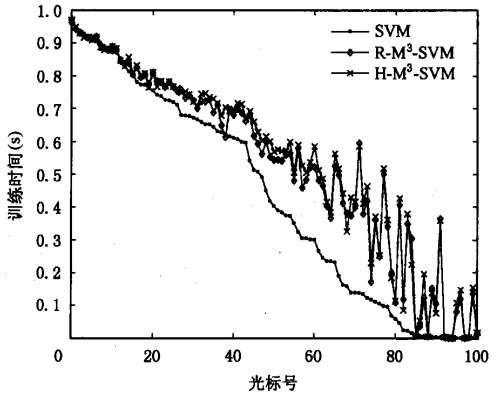
本文的实验分为两部分,对于读卖新闻数据集,模式分类器为传统的 SVM 和 R-M^3-SVM,两者均采用线性核函数. 我们根据不同的 C 值以及不同的模块数做了几组实验,这里只列出了最具代表性的一组,其泛化能力和训练时间的仿真结果见表 1.

表 1 读卖新闻前 10 类的仿真结果比较

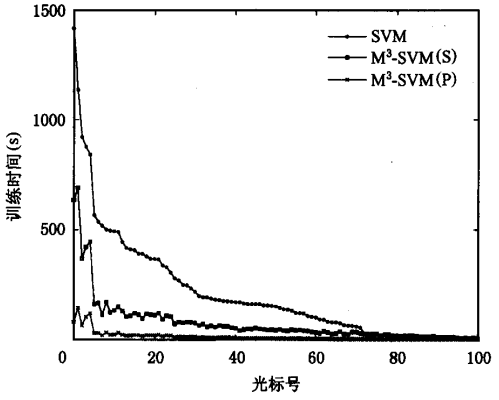
类别	分类器	#SVM	时间/h		加速比		R	P	F_1
			并行	串行	并行	串行			
1	SVM	1	162	162			78.85	92.32	85.06
	M^3-SVM	4	17	67	9.5	2.4	85.66	86.77	86.21
2	SVM	1	39	39			95.66	95.63	95.64
	M^3-SVM	2	17	32	2.3	1.2	96.45	94.70	95.57
3	SVM	1	65	65			58.91	84.42	69.40
	M^3-SVM	4	29	53	2.2	1.2	71.06	76.16	73.52
4	SVM	1	116	116			56.84	90.50	69.82
	M^3-SVM	4	16	49	7.3	2.4	68.23	83.63	75.15
5	SVM	1	83	83			54.04	88.75	67.18
	M^3-SVM	4	16	56	5.2	1.5	67.82	78.53	72.78
6	SVM	1	71	71			80.52	91.07	85.47
	M^3-SVM	3	25	62	2.8	1.1	86.39	85.89	86.14
7	SVM	1	128	128			31.59	82.00	45.61
	M^3-SVM	4	21	78	6.1	1.6	54.90	63.46	58.87
8	SVM	1	67	67			72.00	87.80	79.12
	M^3-SVM	3	42	60	1.6	1.1	78.82	80.58	79.69
9	SVM	1	286	286			54.48	82.67	65.68
	M^3-SVM	4	50	143	5.7	2.0	68.93	72.41	70.63
10	SVM	1	215	215			54.93	83.31	66.20
	M^3-SVM	4	19	65	11.3	3.3	72.35	72.57	72.46

注:"#SVM"表示模块个数,$C=2$

对于路透社数据集,子模式分类器为 R-M^3-SVM 和 H-M^3-SVM 时,我们根据 4 个不同的 C 值,做了 4 组实验,每组实验又依据不同的模块数量,做了 3 组实验,这里,我们详细给出了 $C=1$ 时,传统 SVM,R-M^3-SVM 和 H-M^3-SVM 的泛化能力和训练时间的比较,如图 2 所示. 图 2 中 M^3-SVM(s)表示串行时间,M^3-SVM(p)表示并行时间,类别标号根据 SVM 的训练时间由大到小重新排序,每个 M^3-SVM 分类器的模块数量约等于 7. 实验所用的 SVM 软件包为 libsvm2.4[10],M^3-SVM 是包含上述软件包的一个并行 MPI 程序. 所有的仿真实验在 IBM P690 大型机上完成,该机器拥有 32 个 Power 4 1.3GHz CPU 和 128GB 共享内存.



(a) 传统 SVM,R-M^3-SVM 和 H-M^3-SVM 泛化能力比较



(b) 传统 SVM 和 R-M^3-SVM 训练时间

图 2 路透社数据集实验结果

对单类别 C_i 的评价标准,我们采用查全率(recall)、查准率(precision)和 F_1 标准[11]. 查全率表示分类器对属于 C_i 类样本做出正确判断的样本个数与该类实际拥有的样本个数的比值;查准率表示分类器对属于 C_i 类的样本做出正确判断的样本个数与分类器实际判定为该类的样本个数的比值;F_i 表示对该类查全率和查准率取相同权重的一个综合评价:

$$F_1 = \frac{2RP}{R+P}, \tag{9}$$

其中,R 表示查全率;P 表示查准率.

对多类别的整体评价,通常采用宏观平均(macroaverage)和微观平均(microaverage)[12]. 宏观平均是指对所有类别的 F_i 值取平均后得到的值. 微观平均相当于将所有类别看作一个类别,按照单类别评价标准计算 F_i 值,该值即为微观平均.

为了综合评价传统 SVM,R-M^3-SVM 和 H-M^3-SVM 在多标号文本分类上的分类能力,我们在表 3 中列出了 3 者在 $C=1$ 情况下的泛化能力,同时在表 4 中给出了不同惩罚参数 C,对各个分类器的影响.

表 2 三种分类器的实验结果比较($C=1$)

分类器	#SVM	时间/s		加速比		泛化能力	
		并行	串行	并行	串行	宏观平均	微观平均
SVM	101	1417	20951			42.78	78.13
R-M^3-SVM	794	143	8566	9.91	2.45	54.91	80.50
	1089	37	7476	38.30	2.80	55.82	80.14
	1552	19	7458	74.58	2.81	56.47	79.55
H-M^3-SVM	794	159	7621	8.91	2.75	53.67	80.55
	1059	50	6897	28.34	3.04	54.79	80.57
	1552	30	6998	47.23	2.99	55.84	80.40

表 3 C 取不同值的结果比较

分类器	#SVM	泛化能力							
		$C=0.5$		$C=1$		$C=2$		$C=4$	
		Macro	Micro	Macro	Micro	Macro	Micro	Macro	Micro
SVM	101	34.63	75.99	42.78	78.13	49.28	78.42	49.95	76.57
R-M^3-SVM	794	47.81	80.09	54.91	80.50	57.10	79.49	56.00	77.91
	1089	48.35	78.85	55.82	80.14	57.68	79.22	56.65	77.92
	1552	50.17	79.05	56.47	79.55	57.99	78.84	57.07	77.79
H-M^3-SVM	794	46.50	80.19	53.67	80.55	56.02	79.51	54.99	77.84
	1059	47.92	80.26	54.79	80.57	56.72	79.59	55.77	78.16
	1552	49.31	79.97	55.84	80.40	57.46	79.56	56.54	78.38

注:"Macro"表示宏观平均,"Micro"表示微观平均

从图 2 以及表 1,2 和 3 我们可以得出如下结论:

(1) 对于多标号文本分类问题,M^3-SVM 包括 R-M^3-SVM 和 H-M^3-SVM 在大部分情况下都表现出了比传统 SVM 好的泛化能力,特别对于某些比较难分的类,M^3-SVM 的一般化能力有显著提高. 其原因可解释如下:采用"一对其他"分解策略所得到的两类问题,其正负样本个数极不平衡,通常正类的样本个数偏少. 因此,传统的 SVM 分类器的分界面较靠近于正样本区域,其查全率较低,由式(9)可知 F_i 值较小. 而采用"部分对部分"分解策略,得到的两类子问题的正负样本数量相对比较平衡. 因此,M^3-SVM 分界面较为合理,使得查全率明显升高,查准率有所下降,但 F_i 值增大. 从图 2(a)可以看出,由于类别标号是根据样本数由多到少排序,标号越靠后的类别,正样本数越少,即正负样本数越不平衡. 因此,对于大多数类,M^3-SVM 比传统的 SVM,具有更高的分类准确率.

(2) 训练时间方面,从图 2(b)可以看出,即使 M^3-SVM 所有模块以串行方式训练,其训练时间依然远小于传统 SVM 的训练时间. 若以并行方式训练所有模块,则训练时间可大大减少. 显然,问题的规模越大越能体现出 M^3-SVM 在训练时间上的优越性[5].

(3) 对于多类评价标准的微观平均值,H-M^3-SVM 比 R-M^3-SVM 具有更好的泛化能力,但是对于宏观平均值,前者却稍逊于后者.

(4) 随着模块数的增多,R-M^3-SVM 和 H-M^3-SVM 的宏观平均值都有上升的趋势. 但是 R-M^3-

SVM 的微观平均值却有逐渐下降的趋势,对于 H-M^3-SVM,微观平均值依然可以保持在一个比较稳定的水平上. 这是因为宏观平均值是由小类别决定的,即正样本较少的类别. 随着模块数增多,小类别的两类子问题的正负样本平衡度得到提高,F_1 值升高. 因此,宏观平均值有上升的趋势. 但是对于微观平均值,由于主要受大类别的影响. 因此,若采用随机分解方法,随着分解模块数的增多,大类别 F_1 的值有下降的趋势,微观平均值略微下降. 但是,若采用超平面分解方法,它保证了子问题的样本分布与原问题的样本分布的一致性,其 F_1 值比较稳定. 因此,微观平均值可以保持在一个较高水平上.

5 结束语

本文提出了一种新的解决多标号文本分类问题的方法,即在进行"一对其他"任务分解之后,再利用"部分对部分"的问题分解策略. 该策略利用最小最大模块化支持向量机作为子模式分类器,同时利用了两种不同的问题分解方法. 与传统的支持向量机相比,最小最大模块化支持向量机具有较高的并行性和可修改性. 实验结果表明,该方法比传统处理多标号文本分类问题的方法具有更好的泛化能力和更快的训练速度. 同时,超平面问题分解方法的 H-M^3-SVM 比随机分解的 R-M^3-SVM,具有更好的泛化能力,并且随着问题分解模块数的增多,H-M^3-SVM 的泛化能力可以一直保持在一个稳定的高值上. 从理论上分析 H-M^3-SVM 在多标号文本分类问题上的特性,以及宏观平均值和微观平均值的变化规律将是我们今后需要进一步探讨的问题.

参 考 文 献

1 T. Joachims. Learning to Classify Text Using Support Vector Machine: Method, Theory, and Algorithms. Netherland: Kluwer Academic Publishers, 2002

2 C. Cortes, V. N. Vapnik. Support-vector network, Machine Learning, 1995, 20(3): 273~297

3 B. L. Lü, K. A. Wang, M. Utiyama, *et al*. A part-versus-part method for massively parallel training of support vector machines. IJCNN'04, Budapast, 2004

4 L. Bottou, C. Cortes, J. S. Denker, *et al*. Comparison of classifier methods: A case study in handwritten digit recognition. The Int'l Conf. Pattern Recognition, Erusalem, 1994

5 B. L. Lu, M. Ito. Task decomposition and module combination based on class relations: A modular neural network for pattern classification. IEEE Trans. Neural Networks, 1999, 10(5): 1244~1256

6 K. A. Wang, H. Zhao, B. L. Lu. Task decomposition using geometric relation for min-max modular SVMs. LNCS3496, 2005. 887~892

7 F. Y. Liu, K. Wu, H. Zhao, *et al*. Fast text categorization with min-max modular support vector machines. Int'l Joint Conf. Neural Networks (IJCNN2005), Montréal, Québec, Canada, 2005

8 M. Utiyama, H. Isahara. Large-scale text categorization. The 9th Annunl Meeting of the Association for Natural Language Processing, Yokohama, Japan, 2003

9 D. D. Lewis, Y. M. Yang, T. G. Rose, *et al*. RCV1: A new benchmark collection for text categorization research. Journal of Machine Learning Research, 2004, 4(5): 361~397

10 C. C. Chang, C. J. Lin. LIBSVM: A library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001

11 D. D. Lewis. Evaluating and optimizing autonomous text classification systems. The 18th Annual Int'l ACM SIGIR Conf. Research and Development in Information Retrival (SIGIR 95), Washington, USA, 1995

12 D. D. Lewis. Evaluating text categorization. In: Proc. Speech and Natural Language Workshop. San Francisco: Morgan Kaufmann, 1991. 312~318

吕宝粮,1960 年生,博士,教授,博士生导师,主要研究方向为仿脑计算理论与模型、神经网络、并列机器学习、脑-计算机接口、人脸识别、生物信息学和自然语言处理.

刘峰耀,1979 年生,硕士研究生,主要研究方向为机器学习和文本分类.

内山将夫,1976 年生,博士,研究员,主要研究方向为统计学习、自然语言处理和机器翻译.

井佐原均,1956 年生,博士,研究员,主要研究方向为自然语言处理和机器翻译.

基于最小最大模块化支持向量机的多标号文本分类

作者： 吕宝粮， 刘峰耀， 内山将夫， 井佐原均
作者单位： 吕宝粮,刘峰耀(上海交通大学计算机科学与工程系 上海 200030)， 内山将夫,井佐原均(日本国立信息与通信技术研究所 京都 619-0289)