结合 SVM 和 KNN 的 Web 日志挖掘技术研究方法*

曾 俊
(长江师范学院 数学与计算机学院, 重庆 408100)

摘 要: 将 SVM 和 KNN 算法结合在一起,组成一种新的 Web 文本分类算法——SVM-KNN 算法。当 Web 文本和 SVM 最优超平面的距离大于预选设定的阈值,则采用 SVM 进行分类,反之采用 SVM 作为代表点的 KNN 算法对样本分类。实证结果表明,SVM-KNN 分类算法的分类精度比单纯 SVM 或 KNN 分类算法有不同程度的提高,为 Web 数据挖掘提供了一种有效的分类方法。

关键词: 支持向量机; 数据挖掘; 网页分类; 特征提取
中图分类号: TP181 **文献标志码**: A **文章编号**: 1001-3695(2012)05-1926-03
doi:10.3969/j.issn.1001-3695.2012.05.087

Research method of Web log mining technology with combination of SVM and KNN

ZENG Jun
(*College of Mathematics & Computer Science*, *Yangtze Normal University*, *Chongqing* 408100, *China*)

Abstract: This paper used SVM and KNN algorithm together to form a new classification algorithm for Web text—SVM-KNN algorithm. When optimal super plane distance of Web text and SVM was greater than the preselected threshold, used SVM to classify, otherwise it adopted KNN algorithm to classify the samples of SVM as the representative point. The experimental results show that the accuracy of SVM-KNN classification algorithm are better than pure SVM or KNN classification algorithm, and the Web text classification provides an effective classification method.

Key words: support vector machines(SVM); data mining(DM); Web classification; feature selection

随着 Internet 的发展,网络信息规模呈指数增长。网络信息以文本形式提供给用户,如何快速、有效地从 Web 文本中挖掘用户所需信息越来越重要,因此 Web 文本自动分类技术已经成为 Web 数据挖掘领域的研究热点[1]。

Web 文本自动分类是指将一个 Web 文本归到一个或几个预定义的 Web 文本类别中的过程。传统上,Web 文本分类是通过人工来完成的,需要大量的人力资源,效率极低,分类结果具有很大的主观性[2]。随着数据挖掘和机器学习技术的发展,出现了基于支持向量机、K-近邻(K-nearest neighbors,KNN)、神经网络、贝叶斯算法、决策树[3~5]等 Web 文本自动分类方法。贝叶斯算法是一种概率分类模型,Web 文本类别与长度无关,且要求所有词在 Web 文本中出现的概率是相互独立的,分类结果稳定性比较差。神经网络的 Web 文本分类精度比较高,但是神经网络存在许多自身难以克服的缺陷,如网络结构复杂、参数难以确定等[6]。SVM 是基于统计学习理论的一种新的机器学习方法,是当前 Web 文本自分类的主流算法,只需较少的训练样本就可以获得具有相对较高精度的文本分类器,具有出色的学习性能。但 SVM 也有其缺点,如:核函数参数的选择较难;对于复杂分类问题精度不高;当规模比较大时,训练时间长,实现的代价高[7]。KNN 算法对于特征维数相当高的 Web 文本,易引起维度灾难,同时其只考虑文本样本数目而忽略样本之间的差异性,有时分类效果比较差,因此对 Web 文本进行分类时,如何选择一个好的分类器算法至关重要。

1 Web 文本自动分类原理

Web 信息是以文本形式存在的,Web 文本自动分类是 Web 数据挖掘的重要研究内容之一。Web 文本自动分类原理是首先利用网页抓取器抓取不同的类型网页作为语料库;然后对网页进行预处理并转换为文本文档,将文本文档进行分词,得到能够较好地反映文本内容的特征,即文本特征向量;再采取一定方法进行特征抽取,形成文本特征库;最后利用分类器对文本进行训练,建立 Web 文本分类器,对系统分类结果进行评估。其具体原理如图 1 所示。从图 1 可知,文本分类器直接决定 Web 文本分类精度,但是由于 Web 文本是一种半结构化数据,特征维数十分高,通常为几千、几万甚至几十万维,采用 KNN 算法进行分类容易出现维数灾难。SVM 算法对大规模的 Web 文本训练时间长,因此单一的 SVM 和 KNN 算法均难以适合大规模、高维的 Web 文本分类。

2 Web 文本自动分类系统

2.1 Web 文本的表示

对于 Web 文本自动分类系统,首先需要将 Web 文本表示成计算机能够识别的形式,一般采用向量空间模型(VSM)表示文本特征。在 VSM 模型中,Web 文本被描述成为一个有 N 维向量空间的向量:

$$D=[w_1,w_2,\cdots,w_n] \tag{1}$$

收稿日期: 2011-09-16; **修回日期**: 2011-11-10 **基金项目**: 教育部 2009 年"春晖计划"资助项目(Z2009-1-63004)
作者简介: 曾俊(1979-),女,四川资中人,讲师,硕士,主要研究方向为数据挖掘与智能信息处理、计算机应用(piao_yi_xue@sohu.com).

其中:D 表示 Web 文本向量;w_i 表示第 i 项特征权值。

2.2 Web 文本特征权值计算

在 Web 文本自动分类系统中,文本特征权值表示每一个词在描述文本内容时所起作用的程度。本文采用统计的方法 TF-IDF 计算权值:

$$w_{ik}=\frac{t_{ik}\log(\frac{N}{n_k}+0.01)}{\sqrt{\sum_{k=1}^{n}(t_{ik})^2\log^2(\frac{N}{n_k}+0.01)}} \tag{2}$$

其中:t_{ik} 表示特征 t_k 在文档出现的次数;n_k 为出现特征项 t_{ik} 的文档数;N 表示训练集中所有 Web 文本总数。

因为 Web 文本与普通的文本不一样,Web 文本的标记信息对 Web 文本分类十分重要,所以将 Web 文本的标记信息进行分析处理,根据其重要程度来赋予不同的权值[8]。这样改进加权后的权值计算式为

$$t'_{ik}=\frac{t_{ik}'\log(\frac{N}{n_k}+0.01)}{\sqrt{\sum_{k=1}^{n}(t'_{ik})\log^2(\frac{N}{n_k}+0.01)}} \tag{3}$$

其中,$t'_{ik}=\mathrm{set}(t_{ik})\times t_{ik}$,$\mathrm{set}(t_{ik})$ 为标记信息的级别。

2.3 Web 文本特征抽取

采用 VSM 模型获得的特征向量维数非常多,通常为几千、几万甚至几十万维,且存在大量冗余、带噪声的特征,若将这些特征全部输入 KNN 或 SVM 算法进行分类,计算量十分巨大,时间复杂高,分类精度也难以保证,因此必须对 Web 文本进行特征抽取。

Web 文本特征抽取从特征中选出最有代表性的特征,降低特征维数,去除噪声特征,简化计算,防止分类结果过拟合出现,提高分类的准确率。当前文本特征抽取方法有信息增容法、文档频率法、互信息法和 χ^2 统计法等。本文采用互信息对特征进行评估,选取权值高于预定阈值的特征,抛弃其余特征。

词条 T_i 和文本类别 C_i 的互信息为

$$MI(W,C_j)=\log[\frac{P(W/C_j)}{P(W)}] \tag{4}$$

其中:$P(W/C_j)$ 表示 W 在 C_j 出现的频率;$P(W)$ 表示 W 在整个训练集中出现的概率。$MI(W,C_j)$ 值越大,表示词条 T_i 和文本类别 C_i 相关程度越高。

2.4 Web 文本自动分类器设计

2.4.1 KNN 分类器

KNN 算法是一种非参数模式识别方法,首先对训练样本进行学习,对待分类 Web 文本进行分类时,从训练样本中找出与待分类样本最相似的 K 个样本,然后按照 K 个样本所属的类别信息来判断待分类样本的类别属性,算法具体如下:

设 $f'(x_r)$ 表示 $f(x_r)$ 的估计值,其计算式为

$$f'(x_r)\leftarrow\underset{i=1}{\operatorname{argmax}}\sum_{i=1}^{k}\delta(v,f(x_i)) \tag{5}$$

其中:x_i 表示样本 x 中第 i 个特征的值;$f(x_i)$ 表示 x_i 所属类别;$\delta(a,b)$ 是 Web 文本类别的判别函数。

$$\delta(a,b)=\begin{cases}1 & \text{if } a=b\\ 0 & \text{otherwise}\end{cases} \tag{6}$$

2.4.2 SVM 分类器

SVM 是当前数据挖掘领域中比较流行的机器学习方法,其结构风险最小,泛化能力优异,能解决高维度、非线性等问题。对于 Web 文本分类问题,SVM 算法可描述为将输入空间中的样本通过一种非线性函数关系映射到一个高维特征空间中,使样本在该高维特征空间中线性可分,并找到样本在该维特征空间中的最优线性分类超平面。

$$f(x)=\mathrm{sgn}(w\cdot x+b)=\mathrm{sgn}\{\sum_{i=1}^{k}\alpha_i y_i k(x,x_i)+b\} \tag{7}$$

其中,$k(x,x_i)$ 表示核函数。

在 SVM 中,常用的核函数有线性核函数、多项式核函数、径向基核函数和 Sigmoid 核函数,其中基于径向基核函数的 SVM 性能要优于其他核函数,因此本文采用径向基核函数作为 SVM 核函数,即

$$k(x,x_i)=\exp(-\frac{\|x_i,x_j\|}{2\times\sigma^2}) \tag{8}$$

因此,基于 SVM 的 Web 文本分类器为

$$f(x)=\mathrm{sgn}\{\sum_{i=1}^{k}\alpha_i y_i\exp(-\frac{\|x_i,x_j\|}{2\times\sigma^2})+b\} \tag{9}$$

其中:σ 表示核函数宽度;α_i 表示拉格朗日乘子。

2.4.3 KNN-SVM 组合的 Web 文本分类器

SVM 算法在 Web 文本分类应用中,对位于两个类别的边界区域或重叠区域的样本点,SVM 算法无法正确分类,存在一定的分类错误,因此处理类别边界区域的样本点时,需要对 SVM 算法进行改进。KNN 在进行 Web 文本分类时,将所有 Web 文本样本存入计算机中,这样对于高维的 Web 文本来说,每次分类判断都要计算待分类样本与全部样本的相似度,因此需要较长的计算时间和较大的存储空间,分类效率比较低,不利于 Web 文本实时在线分类。因此本文利用 SVM 和 KNN 算法优点,将其组合起来用于 Web 文本分类。具体思想如下:

对于一个待分类的 Web 文本 x,首先在特征空间计算其与两类 SVMs 代表点 $\varphi(x)^+$ 和 $\varphi(x)^-$ 的距离差 $g(x)$。如果 $g(x)>R$,R 表示阈值,那么就说明 x 离分类超平面较远,落在区域Ⅱ中,采用 SVM 分类就可以进行正确分类;如果 $g(x)<R$,那么就说明 x 离分类超平面较近,落在区域Ⅰ中,采用 SVM 分类容易出错,此时就采用 KNN 对待识别 Web 文本进行分类,将每个 SVMs 作为代表点,计算待识别样本和每个 SVMs 的距离对其得出判断。分类示意图如图 2 所示。

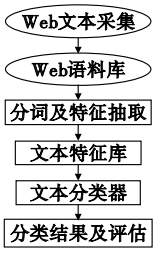

图1　Web文本自动分类的基本原理

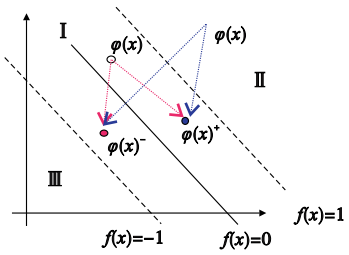

图2　SVM-SVM分类示意

综上可知,SVM-KNN 组合算法分类原理为:当 Web 文本和 SVM 最优超平面的距离大于预选设定的阈值,则采用 SVM 进行分类,反之采用 SVM 作为代表点的 KNN 算法对样本分类。

2.5 SVM-KNN 的 Web 文本分类实现流程

输入:Web 文本训练集 train,测试集 test。

输出:待识别的 Web 文本类别。

a)采用 SVM 算法对 Web 文本 train 的样本进行训练,获得相应的支持向量集、Lagrange 乘子和常数 b。

b)从 test 中取出一个待测试的 Web 文本,如果 test 为空,就输出 Web 文本类别,停止学习。

c)在高维特征空间计算待识别 Web 文本与支持向量代表

点之间的距离,具体计算式为

$$g(x) = \sum_{i=1}^{n} \alpha_i y_i k(x, x_i) + b \tag{10}$$

d)比较 $g(x)$ 与阈值 R 大小,如果 $g(x) > R$,则调用 SVM 分类过程,否则调用 KNN 分类过程。KNN 分类的具体步骤为:首先通过距离公式计算待分类文本与支持向量机之间的距离,并按照距离从小到大进行排序,选择最小的 k 个;然后分别统计这 k 个距离所属的类别;最后待测 Web 文本类别与数目多的类别相同。

e)test←test − $\{x\}$,返回步骤 b)。

基于 SVM-KNN 组合算法的 Web 文本分类流程如图 3 所示。

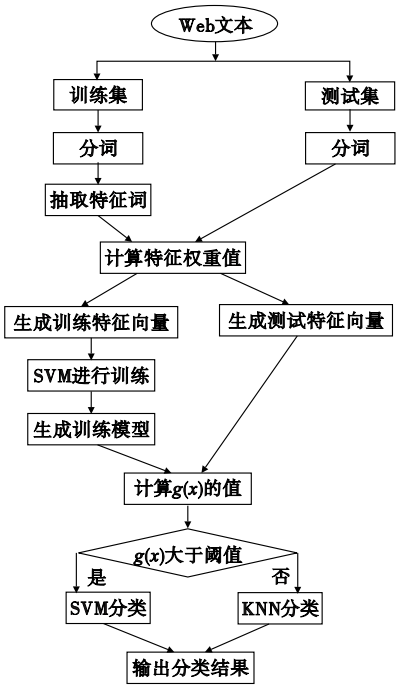

图3 SVM-KNN的Web文本分类流程

3 Web 文本分类的实证分析

3.1 Web 文本数据与参比模型

对 Internet 上的实际 Web 文本数据进行实证分析。首先从 Internet 下载 6 500 个中文网页,采用人工分类方式分为 10 类,分别为计算机、环境、教育、交通、经济、医药、军事、艺术、体育、政治等,其中,每一类别 4/5 的样本作为训练集用于训练建模,1/5 用做测试集,对建立模型的性能进行检验。首先对文本进行预处理,再采用互信息法提取 Web 文本特征,然后采用 SVM-KNN 进行分类,对比模型为单一的 SVM 和 KNN 算法,采用查全率和准确率作为各模型的 Web 文本分类结果评价指标。

3.2 实验结果与分析

3.2.1 SVM-KNN 的分类性能分析

将训练样本输入到 SVM-KNN 进行学习和训练,建立 Web 文本分类器,然后将测试集输入到已构造的 Web 文本分类器进行分类,得到相应的分类结果,其结果如表 1 所示。从表 1 可知,SVM-KNN 的平均查全率和准确率为 93.77% 和 90.74%,具有较高的分类能力,达到了 Web 文本自动分类的要求。

3.2.2 分类性能比较

采用相同的数据和特征提取方法,使用 SVM、KNN 进行分类,并将结果与 SVM-KNN 结果进行比较,对比结果如图 4 所示。从图 4 可知,SVM 比 KNN 算法的 Web 文本分类结果好得多,但是 KNN 算法的分类速度要优于 SVM。SVM-KNN 算法要优于 SVM、KNN 算法的分类结果,主要是由于采用 KNN 对 SVM 进行改进,不仅提高了 Web 文本分类精度,同时提高了分类速度,性能更加稳定。对比结果表明,SVM-KNN 更适合于 Web 数据挖掘中的文本分类。

表 1 SVM-KNN 的 Web 文本分类结果

种类	训练样本	测试样本	查全率/%	准确率/%
计算机	800	200	90.1	87.5
环境	400	100	90.4	88.1
教育	800	200	92.5	89.7
交通	400	100	95.7	92.3
经济	400	100	92.4	90.4
医药	400	100	93.8	92.5
军事	400	100	96.7	92.4
艺术	400	100	95.5	92.6
体育	400	100	94.5	90.4
政治	800	200	96.1	91.5

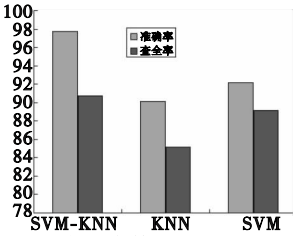

图4 各分类算法的性能对比

4 结束语

本文将 SVM 和 KNN 算法结合在一起,组成一种新的分类算法,将该分类算法应用于 Web 文本分类中,实证结果表明,SVM-KNN 分类算法的分类性能比单纯 SVM 或 KNN 分类算法有不同程度的提高,且 SVM-KNN 分类算法稳健性更好,为 Web 文本分类提供了一种有效的分类方法。

参考文献:

[1] DEBOLE F, SCBASTIANI F. An analysis of the relative hardness of reuters-21578 subsets[J]. Journal of the American Society for Information Science and Technology,2010,56(6):584-596.

[2] AHN B S, CHO S S, KIM C Y. The integrated methodology of rough set theory and artificial neural network for business failure prediction [J]. Expert Systems with Applications,2000,18(2):65-74.

[3] 邹涛,王继成. WWW 上的信息挖掘技术及实现[J]. 计算机研究与发展,1999,36(8):1019-1024.

[4] 涂承胜,鲁明羽,陆玉昌. Web 内容挖掘技术研究[J]. 计算机应用研究,2003,20(11):5-9,15.

[5] 牛强,王志晓,陈岱,等. 基于 KNN 的 Web 文本分类方法的研究[J]. 计算机应用与软件,2007,24(10):210-211.

[6] 徐家树,覃征,杨盾. 基于 BP 神经网络的 Web 页面分类算法[J]. 微电子学与计算机,2006,23(5):83-85.

[7] 钟将,温罗生,冯永,等. 基于近似支持向量机的 Web 文本分类研究[J]. 计算机科学,2008,35(3):167-169,202.

[8] 张义忠,赵明生,梁久祯. 基于自组织特征映射的网页分类研究[J]. 信息与控制,2003,32(2):108-112,117.

[9] 丁一,卢正鼎. 基于 Web 挖掘的用户服务研究[J]. 计算机仿真,2004,21(6):83-85,93,94.

[10] 张志强,郑家恒. 基于加权类轴的 Web 文本分类方法研究[J]. 计算机应用,2004,24(2):148-150.