基于粒计算的多标签懒惰学习算法

赵海峰 余 强 曹俞旦

(安徽大学计算机科学与技术学院 合肥 230601) (安徽省工业图像处理与分析重点实验室 合肥 230039)

摘 要 多标签学习用于处理一个样本同时拥有多个标签的问题。已有的多标签懒惰学习算法IMLLA未充分考虑样本分布的特点,即在构建样本的近邻点集时,近邻点个数取固定值,这可能会将相似度高的点排除在近邻集之外,或者将相似度低的点包括在近邻集内,影响分类方法的性能。针对IMLLA的缺陷,将粒计算的思想加入近邻集的构建,提出一种基于粒计算的多标签懒惰学习算法(GMLLA)。该方法通过粒度控制,确定样本近邻点集,使得近邻集内的样本具有高相似度。实验结果表明,本算法的性能优于IMLLA。

关键词 K近邻,多标签学习,懒惰学习,IMLLA,粒计算

中图法分类号 TP181 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.12.034

Multi-label Learning Algorithm Based on Granular Computing

ZHAO Hai-feng YU Qiang CAO Yu-dan

(School of Computer Science and Technology, Anhui University, Hefei 230601, China)

(Key Lab of Industrial Image Processing & Analysis of Anhui Province, Hefei 230039, China)

Abstract Multi-label learning deals with the problem that each instance is associated with multiple labels. Existing multi-label learning algorithm IMLL based on lazy learning does not fully consider the distribution of instances. When building the nearest neighbor sets of the instances, the number of the neighbor for each instance is a constant value valued k. It may lead to such an outcome that the instances with higher similarity are ruled out of the nearest neighbor set or the instances with lower similarity are capsulated into the nearest neighbor set, which will affect the performance of the classification method. In this article, an improved multi-label lazy learning algorithm combined with the idea of granular computing was proposed. The nearest neighbor set of each instance is built by the controlling of the granularity. Then the instances in the nearest neighbor set of each instance behave high similarity. Experimental results show that the performance of our algorithm is superior to IMLLA.

Keywords K-nearest-neighbor, Multi-label learning, Lazy learning, IMLLA, Granular computing

1 引言

传统的二分类问题中每个样本只有一个标签,这类问题称为单标签学习问题。但是在现实生活中,每个样本可能同时属于多个类别,例如在文档分类[1,7,8]问题中,每篇文档可能同时属于多个主题,例如"电影"与"电视剧";在功能基因组学[2]问题中,每种基因可能具有多种功能,例如"新陈代谢"与"光合作用";在图像标注[3]问题中,每幅图片可能对应多个语义类别,例如"城市"与"高楼"。这类问题都称为多标签学习[4,5]问题。由于多标签数据标签间的相关性与共现性,传统的单标签分类方法不能直接应用到多标签分类问题中,因此多标签学习已成为机器学习领域研究的热点,并受到了广泛的关注[6-8]。

多标签学习的研究起源于文档分类[1,7,8]中遇到的歧义性问题,Schaoire和Singer提出了一种基于集成学习的方法BoosTexter[8],该方法其实是对AdaBoost[9]的扩展,它在训练过程中既改变训练样本的权重,也改变概念标签的权重。Zhang和Zhou提出了基于神经网络的多标签学习算法[10]。Elisseeff与Weston提出了基于支持向量机(SVM)的多标签分类方法[2],即将多标签分类问题转化为一系列的单标签SVM分类问题。Zhang与Zhou将传统的K近邻算法引入到多标签学习中,提出了K近邻多标签学习算法(ML-kNN)[11]。

ML-kNN算法是目前常用的一种多标签懒惰学习方法,具有较好的性能。然而它将多标签学习问题分解为多个独立的二分类问题,并且在预测样本所含标签时未考虑其他概念标签所蕴含的信息,即未充分考察样本标签之间的相关性,因此其泛化性能必然会受到一定程度的影响。Zhang针对ML-kNN算法存在的不足,提出了一种新型多标记懒惰学习算法(IMLLA)[12],对于给定的测试样本,该方法首先找出该样本与训练集中各个标签类对应的近邻样本;基于此,利用近邻样本的多标签信息生成一个标记计数向量,并提交给已训练的

到稿日期:2014-01-28 返修日期:2014-04-28 本文受国家自然科学基金项目(61272152,61202228),安徽省自然科学基金项目(1208085MF109),2013留学人员科技活动择优资助项目资助。

赵海峰(1972-),男,博士,副教授,主要研究方向为模式识别、医学图像处理与应用,E-mail:senith@163.com;余 强(1989-),男,硕士,主要研究方向为多标签数据分类、医学图像处理;曹俞旦(1988-),男,硕士,主要研究方向为医学图像处理。

线性分类器进行标签预测。然而 IMLLA 在构建样本近邻点集时没有充分考虑样本近邻点的具体分布特点，近邻数 K 是一个预先设定的值，可能将一些与样本相似度不高的点加入到样本近邻点集，也有可能未将与样本相似度高的点加入样本近邻点集，这势必会影响分类效果。

本文针对 IMLLA 中 K 是一个预先确定值的缺点，将粒计算[13]的思想引入到多标签学习方法中，提出了一种基于粒计算的多标签懒惰学习算法，在构建样本近邻集时通过粒度控制，使样本在不同的标签类集合中取不同个数的近邻点，从而构建具有较高相似性的近邻集。在公认数据集上的实验结果表明，本文的方法取得了较好的分类效果。本文接下来介绍 IMLLA，然后对该算法进行改进并给出改进的方法在多标签数据集上的实验结果，最后对本文的工作进行总结。

2 多标签学习

在多标签学习中，假设 $X=\{x_1,x_2,\cdots,x_n\}\in R^d$ 表示有 n 个样本，$Y=\{1,2,\cdots,Q\}$ 表示所有可能的概念标签构成的集合。$T=\{(x_1,Y_1),(x_2,Y_2),\cdots,(x_m,Y_m)\}(x_i\in X,Y_i\in Y)$，表示训练集，多标签学习系统的目标是输出一个多标签分类器 $h:X\to\{-1,1\}^Q$，并对特征属性已知而标签集未知的样本进行标签预测。不过在大多数情况下，学习系统的输出是某个实值函数 $f:X\times Y\to R$，对于给定属性值的样本 x_i，该实值函数将给出该样本在各个标签上的 f 值。一个性能良好的学习系统会在隶属于 Y_i 的概念标签上输出较大的值，而在不属于 Y_i 的概念标签上输出较小的值，再通过某种策略给定阈值 t，当样本在某个标签上的 f 值大于此阈值时，就认为该样本含有该标签，否则就认为该样本不含该标签。通常，函数 $f(\cdot,\cdot)$ 还可以转化为一个排序函数 $rank(\cdot,\cdot)$，该排序函数将所有的函数输出 $f(x_i,y)(y\in Y)$ 映射到集合 $Y=\{1,2,\cdots,Q\}$ 上，使得 $f(x_i,y_1)>f(x_i,y_2)$ 时 $rank(x_i,y_1)<rank(x_i,y_2)$ 成立。因此对于任一个给定特征属性值的测试样本，可以将标签集中的每个标签 y 代入到该算法所得到的排序函数中，再利用预先给定的阈值来得到该样本的预测标签集。

3 IMLLA 分析

ML-kNN 是一种经典的多标签分类算法，但它并未充分考察标记之间的相关性，因此有学者提出了 IMLLA[12] 算法来对其进行改进：对于给定的样本 $x\in X$ 及其概念标记集合 $y\subseteq Y$，设 y_x 是与样本 x 对应的 Q 维类别向量，其中，当 $l\in y_x$ 成立时，该向量的第 l 维的分量 $y_x(l)$ 的取值为 1；否则 $y_x(l)$ 取值为 -1。此外，设 $T_l=\{x_i\mid x_i\in T,l\in Y_i\}$ 为训练集 T 中具有标签 l 的样本构成的集合，此时，对于每一类 $l\in Y$，假设 IMLLA 考察的近邻样本个数为 k，样本 x 在 T_l 中的 k 近邻点构成的集合记为 N_x^l，即：

$$N_x^l=\{z\mid z\text{ 是样本 }x\text{ 在 }T_l\text{ 中的 }k\text{ 近邻点}\} \tag{1}$$

在构建样本近邻点集时，IMLLA 算法用曼哈顿距离度量两个样本 x_i 与 x_j 的相似度：

$$Dist(x_i,x_j)=\sum_{h=0}^{d}|x_i^h-x_j^h| \tag{2}$$

基于与每一个标签对应的 N_x^l，定义 Q 维的标签计数向量 C_x 如下：

$$C_x=\sum_{q\in Y}\sum_{z\in N_x^l}[|Y_z(l)==1|] \tag{3}$$

对于任意的 χ，当 χ 成立时，$[|\chi|]$ 取值 1，否则取值 0，C_x 的分量 $C_x(l)$ 给出了样本 x 与所有标签对应的近邻集中含有标签 l 的近邻个数。

对于每个测试样本 t，IMLLA 先确定该样本与各个概念标签类所对应的近邻样本集合 N_t^l，然后利用式(3)计算对应的标签计数向量 C_t，根据 C_t 中包含的信息，IMLLA 采用如下的线性分类器来确定样本 t 在标签 l 上的输出：

$$f(t,l)=w_l^{\mathrm{T}}\cdot C_t,l\in Y \tag{4}$$

其中，w_l 为与标签 l 对应的 Q 维列向量，T 为矩阵的转置操作，样本 t 在标签 l 上的输出 $f(t,l)$ 是列向量 w_l 与标记记数向量 C_t 的点积。若 $f(t,l)>0$，则测试样本 t 将拥有标签 l，否则，标签 l 将不属于样本 t。因此，与测试样本 t 对应的概念标签集合为 $h(t)=\{l\mid f(t,l)>0,l\in Y\}$。对于式(4)所需的列向量 $w_l(l\in Y)$，采用最小化式(5)所示的误差平方和函数的方式来求得：

$$E=\frac{1}{2}\sum_{i=1}^{m}\sum_{l\in Y}(\hat{y}_{x_i}(l)-y_{x_i}(l))^2 \tag{5}$$

其中，$\hat{y}_{x_i}(l)=w_l^{\mathrm{T}}\cdot C_{x_i}$ 为列向量 w_l 与训练样本 x_i 的标签计数向量之间的点积。式(5)对应于该分类器在整个训练集上关于概念标签的重构误差，IMLLA 通过最小化式(5)所在的训练集上关于概念标签的重构误差，来优化列向量 w_l 以预测未知样本的概念标签。

将式(5)右端对于向量 w_l 的第 j 维 $w_l(j)$ 求导并将导数设为 0，则最小化上述误差平方和函数等价于求解如下的正则方程组：

$$\sum_{i=1}^{m}\{(\sum_{q=1}^{Q}w_l(q)C_{x_i}(q)-y_{x_i}(l))C_{x_i}(j)\}=0$$
$$(1\leqslant l\leqslant Q,1\leqslant j\leqslant Q) \tag{6}$$

与式(6)所示的方程组等价的矩阵形式如下：

$$(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi})\cdot\boldsymbol{W}=\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{T} \tag{7}$$

其中，矩阵 $\boldsymbol{\Phi}=[\boldsymbol{\Phi}_{il}]_{m\times Q}$ 且含有元素 $\boldsymbol{\Phi}_{il}=\boldsymbol{C}_{x_i}(l)$，矩阵 $\boldsymbol{W}=[\boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_Q]$ 由列向量 w_l 组成，矩阵 $\boldsymbol{T}=[t_{il}]_{m\times Q}$ 且含有元素 $t_{il}=y_{x_i}(l)$，利用奇异值分解[14]对式(7)中的参数矩阵 $\boldsymbol{W}$ 进行计算。

在 IMLLA 中，样本最近邻点的个数是预先固定的，按这种方式构建的近邻集常常会出现相似度较小的点进入近邻集内或相似度较大点被排除在近邻集外的情况，因此，在使用固定近邻值构建近邻集时，难以反映出样本分布特点对分类结果的影响。本文将粒计算的思想引入到该方法中，在构建样本近邻集时动态地选择近邻点，使得近邻点的个数能够根据具体情况来选取。

4 GMLLA

粒计算包括粒化和粒的计算，粒化主要是构建不同粒度的空间，粒的计算主要是利用粒度的层次结构进行问题的求解。基于商空间理论的粒计算用一个三元组 $(\boldsymbol{X},f,\boldsymbol{T})$ 来描述待解决的问题，其中，$\boldsymbol{X}$ 代表论域，是研究对象组成的集合；f 代表属性函数，$\boldsymbol{T}$ 代表论域 $\boldsymbol{X}$ 上的拓扑结构，反映了元素的相互关系。给定等价关系 R，$x(x\in\boldsymbol{X})$ 关于 R 的等价类用 $[x]_R$ 表示。$[\boldsymbol{X}]=\{[x]_R\mid x\in\boldsymbol{X}\}$ 称为 $\boldsymbol{X}$ 关于 R 的商集，然后由 $[\boldsymbol{X}]$ 可构建相应的商空间 $([\boldsymbol{X}],[f],[\boldsymbol{T}])$，对于不同的等价关系，可以构建不同的商集，由 $\boldsymbol{X}$ 上的等价关系簇 $\overset{\vee}{R}$ 生成不同层次

的粒度空间，因此对问题的求解就转化为在其对应的商空间($[\boldsymbol{X}]$,$[f]$,$[\boldsymbol{T}]$)上进行求解。多标签学习中，所有的样本点形成论域 $\boldsymbol{X}$。我们用样本距离 $d(\cdot,\cdot)$来度量样本之间相关性，可以假设 $0=d_0<d_1<\cdots<d_i<\cdots$，那么$[d_{i-1},d_{i+j}](i=1,2\cdots;j=0,1,\cdots)$构成$[0,+\infty)$的一个划分，$x_0$ 是一样本点，定义：

$$R:xRy,\text{若 } d(x_0,x),d(x_0,y)\in[d_{i-1},d_{i+j}]\ x,y\in\boldsymbol{X} \quad (8)$$

其中，R 为 $\boldsymbol{X}$ 上的等价关系，且 j 的不同选取将形成不同的等价关系，从而形成 $\boldsymbol{X}$ 的不同层次的粒度空间。

在本文算法中，设 k 是样本点在每个标记类中选取的最小近邻点的个数，为使近邻集与其外的元素之间有较大差异，通过 *rate* 来控制粒度的选取，设样本为 x，它的近邻点按与其距离由近到远依次设为 $x_1,x_2,\cdots,x_k,\cdots$，距离记为 $dist(x,x_i)$，选取最后一个满足以下条件的样本点：

$$\frac{dist(x,x_p)}{dist(x,x_k)}\leqslant rate,p=k,k+1,\cdots \quad (9)$$

其下标记为 m，则点 $x_1,x_2,\cdots,x_k,\cdots x_m$ 是相互等价的，*rate* 的不同取值可以构建不同大小的等价类，即粒度的粗细，由此，样本的最近邻点个数及具体的近邻点就可以由 *rate* 确定。k 为事先设定的近邻点个数，对于 *rate* 的取值，可以采用人工控制 *rate* 的方式，亦可采用自适应法得到 *rate* 的值。

对于每个训练样本与测试样本，基于粒计算的多标签懒惰学习算法针对每个标记类按式(9)计算出训练样本或者测试样本在每个标记类中对应的近邻点个数 m，然后进行分类。即式(1)要改进为：

$$N_x^l=\{z|z\text{ 是样本 }x\text{ 在 }T_l\text{ 中的 }m\text{ 个近邻点}\} \quad (10)$$

算法 1 基于粒计算的多标签懒惰学习算法的伪代码表示：

$[y_t,r_t]$=IMLLA(**T**,k,t,rate)

Inputs:

T—训练样本集$\{(x_1,Y_1),(x_2,Y_2),\cdots,(x_m,Y_m)\}(x_i\in X,Y_i\subseteq Y)$。

k—事先设定的最小近邻数。

t—测试集。

y_t—测试集 t 的标记矩阵。

r_t—测试集 t 的 Q 维输出矩阵，$r_t(l)$按式(4)定义 f(t,l)的方式来计算得到。

Process:

1. for i∈{1,2,…,m} do
2. for l∈Y do
3. 按照式(10)定义计算出 $N_{x_i}^l$；
4. endfor
5. 按照式(3)定义计算出 x_i 的计数向量 C_{x_i}；
6. endfor
7. 按照式(7)使用 SVD 计算出线性转化矩阵 $\boldsymbol{W}$；
8. for l∈Y do
9. 按照式(10)定义计算出 N_t^l；
10. endfor
11. 按照式(3)定义计算出 t 的的计数向量 C_t；
12. for l∈Y do
13. $r_t(l)$按照式(4)定义将设为 f(t,l)的值；
14. 如果 $r_t(l)>0$，$y_t(l)=1$；
    否则 $y_t(l)=-1$；
15. endfor

算法 1 描述了 GMLLA 的详细处理步骤，基于多标签训练集，算法首先学习线性分类器所需的参数矩阵 $\boldsymbol{W}$(行 1－7)，然后计算测试样本 t 的标签计数向量(行 8－11)，最后将所得向量提交给学习得到的线性分类器，进而得到最终的输出结果(行 12－15)。

5 实验结果及分析

5.1 评价指标

对于给定的测试集 $S=\{(x_1,Y_1),(x_2,Y_2),\cdots,(x_p,Y_p)\}$，本文采用以下几种多标签学习领域常用的评价指标[12]来度量多标签学习方法的性能。

(1)汉明损失，该指标用来考察样本在单个标签上的误分类情况，即本属于该样本的标签未出现在预测标签集合中而本不属于该样本的标签出现在预测标签集合中：

$$hloss_s(h)=\frac{1}{p}\sum_{i=1}^{p}\frac{1}{Q}|h(x_i)\Delta Y_i| \quad (11)$$

其中，Δ 表示两个集合之间的对称差，$|\cdot|$表示集合的大小。

(2)1-错误率，该指标用于考察在样本的预测标签的排序序列中，序列最前端标签不属于样本标签集合的情况：

$$one\text{-}error_s(f)=\frac{1}{p}\sum_{i=1}^{p}[|[\operatorname{argmax}_{y\in Y} f(x_i,y)]\notin Y_i|] \quad (12)$$

(3)覆盖率，该指标用于考察在样本的预测标签的排序序列中，覆盖样本的所有标签平均所需搜索深度：

$$coverage_s(f)=\frac{1}{p}\sum_{i=1}^{p}\max_{y\in Y_i} rank_f(x_i,y)-1 \quad (13)$$

(4)排序损失，该指标用于考察在样本的预测标签的排序序列中，排序出现误排的情况：

$$rloss_s(f)=\frac{1}{p}\sum_{i=1}^{p}\frac{1}{|Y_i||\overline{Y_i}|}|\{(y_1,y_2)|f(x_i,y_1)\leqslant f(x_i,y_2),(y_1,y_2)\in Y_i\times\overline{Y}_i\}| \quad (14)$$

其中，$\overline{Y}_i$ 表示 Y_i 在集合 Y 中的补集。

(5)平均精确度，该指标用于考察在样本的预测标签的排序中，排在属于该样本的标签之前的标签仍然属于样本标签集的情况：

$$avgpre_s(f)=\frac{1}{p}\sum_{i=1}^{p}\frac{1}{|Y_i|}\sum_{y\in Y_i}\frac{|\{y'|rank_f(x_i,y')\leqslant rank_f(x_i,y),y'\in Y_i\}|}{rank_f(x_i,y)} \quad (15)$$

指标(1)是根据算法得到的预测标记集 $h(t)$来计算的，而指标(2)－(5)是根据算法得到的实值函数 $f(t,l)$来计算的。对于前 4 种评价指标而言，指标取值越小则表示算法性能越优；对于最后一种评价指标而言，指标取值越大则表示算法性能越优。

5.2 实验结果及分析

本文将 GMLLA 与 IMLLA[12]的实验结果进行比较，其中，IMMLA 采用原文献中对应的参数设置，用曼哈顿距离度量样本之间的相关度。

本文使用了 yeast 数据集[2]，将本文算法与 IMLLA 在该数据集上的性能进行比较。yeast 数据集共含有 2417 个对象，每个对象的特征属性维数为 103，总共有 14 个概念标记类别，每个对象平均对应 4.24 个概念标记。

实验采用 10-fold 交叉验证的方法进行测试，参与比较的原算法与本文改进算法采用了相同的数据集分划。

图 1 给出了在不同 K 取值下 IMLLA 算法在 yeast 数据集上实验得到的 Hamming Loss 评价指标值变化曲线。

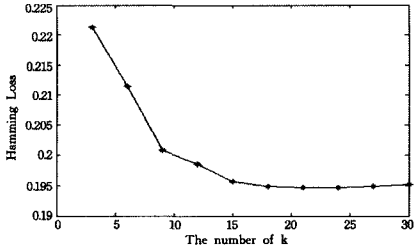

图 1　不同 K 取值下 Hamming Loss 值变化曲线

从图 1 可以看出，IMLLA 在近邻点个数不同情况下，随着近邻点个数的增加，各项指标均有所提升，但是随着近邻点值的继续增加，指标的提升趋于平缓，并且当近邻点取值超过一定值时，算法效率会趋于下降。

表 1 给出了 IMLLA 在 yeast 数据集上不同 K 值下的部分实验结果。

表 1　IMLLA 在 yeast 数据集上的评价指标

近邻点	汉明损失	1-错误率	覆盖率	排序损失	平均精确度
10	0.2044	0.2664	6.749	0.1884	0.7433
14	0.1976	0.2395	6.3386	0.1693	0.7547
15	0.1957	0.2344	6.3224	0.1673	0.7635
16	0.1935	0.2363	6.31	0.1665	0.7639

实验结果表明，当 K 值不断增加，算法在各个指标上的取值并非一直变好，综合考虑算法的效率与性能，对于本文算法在 yeast 上的效果考察均取近邻点最小个数为 15。另外，本文采用人工控制 *rate* 取值的方式。不同的 *rate* 取值会导致构建样本近邻集时动态选取不同的近邻点个数，这必然会影响实验结果，图 2 给出了在不同 *rate* 取值下，GMLLA 在 yeast 数据集上实验时评价指标 average precision 的值变化曲线。

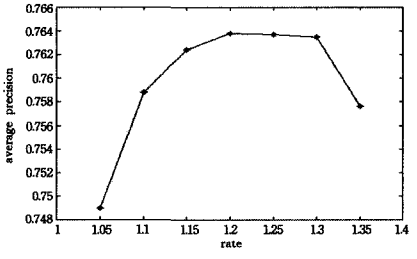

图 2　不同 *rate* 取值下 average precision 值变化曲线

表 2 给出了 GMLLA 在 yeast 数据集上不同 *rate* 取值下的部分实验结果。

表 2　本文算法在 yeast 数据集上的评价指标

rate	汉明损失	1-错误率	覆盖率	排序损失	平均精确度
1.25	0.1964	0.232	6.3071	0.1658	0.7637
1.200	0.1961	0.2315	6.3046	0.1657	0.7638
1.150	0.1963	0.2311	6.3079	0.1659	0.7628
1.050	0.1978	0.2398	6.3842	0.1711	0.7588

从图 2 可以看出，当 *rate* 取值偏大或者偏小时，算法在各评价指标上取得的效果均有所下降，主要原因是当 *rate* 取值偏小时，构造的样本的近邻点等价类中包含了相似度并不是很高的点；而当 *rate* 取值偏大时，构造的样本的近邻点等价类未能将相似度高的点尽可能多地包括进来，导致分类效果不好。另外从图 2 与表 2 中可以看出，在所取的各个 *rate* 值中，当 *rate* 为 1.20 时，本文算法在各评价指标上近似达到最优。表 3 给出了 IMLLA 算法取近邻值 15 与 GMLLA 算法 *rate* 取值为 1.20 时对应结果的对比。

表 3　实验结果对比

评价指标	算法	
	IMLLA	GMLLA
汉明损失	0.1957	0.1961
1-错误率	0.2344	0.2315
覆盖率	6.3224	6.3046
排序损伤	0.1673	0.1657
平均精确度	0.7635	0.7638

从表 3 可以看出，在评价指标 1-错误率、排序损失、覆盖率方面，本文算法优于 IMLLA 算法；在评价指标平均精确度上，本文算法与 IMLLA 算法旗鼓相当；仅在评价指标汉明损失上，本文算法稍逊于 IMLLA 算法。

结束语　本文针对现有的多标签懒惰学习算法 IMLLA 存在的不足进行改进，将粒计算的思想引入到多标签的学习中，提出一种改进的多标签懒惰学习算法，通过粒度控制，在确定与每一类对应近邻样本集时构建不同个数的近邻数，从而构建具有较高相似性的近邻集。在真实世界数据集上的实验结果表明，该算法表现出了较好的性能。今后将在自适应获得 *rate* 值上进行研究，以进一步提高算法性能。

参 考 文 献

[1] McCallum A. Multi-label text classification with a mixture model trained by EM[J]. AAAI'99 Workshop on Text Learning, 1999:1-7

[2] Elisseeff A, Weston J. A kernel method for multi-labelled classification[C]// Advances in Neural Information Processing System. Cambridge. MA: MIT Press, 2002:681-687

[3] Wang C, Yan S, Zhang L. Multi-label sparse coding for automatic image annotation [C]// Computer Vision and Pattern Recognition. 2009:1643-1650

[4] Tsoumakas G. Multi-label classification [J]. International Journal of Data Warehousing & Mining, 2007, 3(3):1-13

[5] Streich A, Buhmann J. Classification of multi-labelded data: a generative approach [C] // Proceeding of the 19th European Conferrence on Machine Learning and the 11th Principles and Practice of Knowledge Discovery in Data-bases. 2008:390-405

[6] 汤进，黄莉莉，赵海峰. 使用自适应线性回归的多标签分类算法[J]. 华南理工大学学报：自然科学版，2012，40(9)：69-74

[7] Schapire R E, Singer Y. Boostexter: A boosting-based system for text categorization[J]. Machine Learning, 2000, 39(2/3):135-168

[8] Hotho A, Maedche A, Staab S, Ontologies Improve Text Document Clustering [C] // Proc. of the IEEE International Conference on Data Mining. Melbourne, Australia, 2003:542-544

[9] Freund Y, Schapire R E. A decision-theoretic generalization of on-line learning and an application to boosting[C] // Lecture Notes in Computer Science 904. Berlin: Springer, 1995:23-37

[10] Zhang Min-ling, Zhou Zhi-hua. Multi-label neural networks with applications to functional genomics and text categorization[J]. IEEE Transon Knowledge and Data Engineering, 2006, 18(10): 1338-1351

[11] Zhang Min-ling, Zhou Zhi-hua. ML-Knn: A lazy learning approach to multi-label learning [J]. Pattern Recognition, 2007, 40(7):2038-2048

[12] 张敏灵. 一种新型多标签懒惰学习算法[J]. 计算机研究与发展，2012，49(11)：2271-2282

[13] 张铃，张钹. 问题求解理论及应用：商空间粒度计算理论及应用(第 2 版)[M]. 北京：清华大学出版社，2007