基金项目论文

基于标签聚类的多标签分类算法

申超波，王志海，孙艳歌

(北京交通大学 计算机与信息技术学院 北京 100044)

摘 要: 多标签分类的实质就是为给定实例预测一个与其关联的标签集合。典型方法可以分为两类：问题转换型和算法适应型。本文主要研究基于标签幂集的问题转换型算法。由于已有的标签幂集算法很难发现甚至可能忽略隐藏在训练集中的重要标签集合，因此，本文提出了一种基于标签聚类的标签幂集方法，通过改进平衡 *k*-means 聚类来发现训练集中潜在的重要标签集合，并用于形成新的训练集进行多标签分类。经实验验证，该算法在多个评价指标上较原有的标签幂集方法具有更好的分类性能。

关键词: 多标签；分类器；标签聚类；标签集合

中图分类号: TP301.6 **文献标识码:** A **DOI:** 10.3969/j.issn.1003-6970.2014.08.004

本文著录格式: [1]申超波，王志海，孙艳歌. 基于标签聚类的多标签分类算法[J]. 软件，2014，35(8)：16–21

A Multi-Label Classification Algorithm Based on Label Clustering

SHEN Chao-bo, Wang Zhi-hai, SUN Yan-ge

(*School of Compute and Information Technology*, *Beijing Jiaotong University*, *Beijing*, 100044, *China*)

【Abstract】: The essence of a multi-label classifier is to assign a set of labels to a given instance. There are the two classical methods: problem transformation and algorithm adaptation. This paper mainly explores the problem transformation of label powerset. By analyzing existing label powerset methods, we find out that it is easy for them to underutilize multi label information. Therefore, this paper proposed a novel label powerset method based on label clustering. Firstly, it identifies unseen multilabels by improving balanced k-means clustering. Then based on that unseen multilabels, it forms new training data for multi-label classification. The experimental results show that the new method has competitive performance with respect to multiple evaluation metrics.

【Key words】: Multi Label; Classifier; Label Clustering; Label Set

0 引言

传统的单标签分类问题实际上就是通过对实例集的学习，然后建立一个分类模型来解决分类任务。单标签分类问题的每条实例都只有一个单独的类标 y_i，且这个标签来自于互不相交的有限标签集合 $L=\{y_1,y_2,\cdots,y_Q\}$。然而，在现实的许多问题中，一条实例往往可能同时属于多个标签。比如，一首歌可能包含了多种情感的标签，一则新闻可能同时涉及政治类和宗教两个领域。这种情况可以广泛扩展现实中的许多应用，如图像和视频的语义标注(新闻剪辑，电影剪辑)，功能基因组学(基因与蛋白质功能)，文本分类(电子邮件，书签)以及其他应用。正是由于新的多标签应用问题的不断出现，多标签的学习吸引了越来越多研究者的关注，因此它成为了数据挖掘领域新的研究热点之一。

通过学习来将一条实例 $x\epsilon X$(X 表示实例集合)映射到一个标签集合 $y\subseteq L$ 的任务就被称之为多标签分类[1]。相比于单标签分类，多标签分类并不假定标签之间是互斥的，也就是说，多个标签可能与一个实例相关联，或者每个实例不只属于一个类。

经过最近这些年的不断研究，已经形成了许多不同的方法来解决多标签分类的问题。Tsoumakast 和 Katakis

基金项目: 北京市自然科学基金资助(4142042)

作者简介: 申超波(1988–)，男，湖南邵阳人，北京交通大学在读研究生

通信联系人: 王志海(1963–)，男，河南安阳人，北京交通大学教授，博士生导师．主要研究领域为机器学习，数据挖掘等

将这些方法分为两个大类：(a) 算法适应的方法；(b) 问题转换的方法[1]。算法适应的方法就是将原来的单标签学习算法进行扩展以能够直接处理多标签数据的算法，例如 ML-C4.5 则修改了原来计算熵的公式和 ML-*k*NN 扩展于原来的 *k*NN[2]。问题转化方法是将多标签分类问题转化为一个或多个单标签分类问题。尽管这种方式简化了问题，但它同样要求谨慎应用标签之间的关联信息以用于更好的预测。由于问题转换方法的简化性以及在大多数数据集上应用的良好性，本文将主要研究问题转换的方法。

本文接下来的组织如下：第 1 节总结了多标签学习研究的相关工作；第 2 节详细描述了提出的算法；第 3 节给出了实验的设计和实验结果的分析；最后第 4 节总结本文的工作以及结论。

1 多标签学习研究的相关工作

在问题转化方法当中，目前最常用的两类转化方法为二值相关算法(Binary Relevance，BR)，标签幂集算法(Label PowerSet，LP)。其中，BR 方法是问题转换方法中转换策略最简单的多标签方法之一，即将一个多标签问题转换为多个二值分类问题来进行处理。但 BR 方法的这种转化策略是建立在标签彼此独立的这种假设上，而这种假设在现实许多领域中是不成立的，这也是 BR 方法的局限性。为此，Read 等人提出了 CC(Classifier Chain，分类器链)算法[3]，它将这些基分类器(C_j, j=1…q)串联起来形成一条链，第 C_j 分类器总是依赖于前 j-1 分类器的结果，这就考虑到标签之间的关联性。但是由于 CC 算法中分类器链中分类器顺序是随机的，就可能出现当前面分类器分类效果不好，会将这种误差效果不断传递到后面的分类器。为此，Sucar 等人提出了基于贝叶斯网络的 CC 算法[4]，通过建立贝叶斯网络来寻找到分类器链的适当顺序，从而达到优化的目的。同样，Goncalves 等人也提出了基于遗传算法的 CC 算法[5]来优化基分类器的最佳顺序。

本文重点研究是另一种问题转换策略的方法，即 LP 方法。在 LP 方法中，它将每一种标签之间的组合都看作一个新的类标，即形成一个新的标签，这样就隐式考虑了标签之间关联信息。但是，这种方法也存在三个弊端。第一，若实例集中有 N 个标签，则标签的可能组合就为 2N，如果这个 N 很大，那个标签组合的规模将是非常巨大的。第二，其中许多标签的组合出现是不频繁的，这就可能引起分类器的标签不平衡。第三，在分类的过程中许多不可见的标签组合不能直接被发现。为此，Read 等人提出了 PS(Pruned Sets)的算法和 PPT(Pruned LP)算法[6]，即在使用 LP 算法之前，先除掉训练实例(x, y)中标签 y 出现的次数不频繁的所有实例。然后重新产生一条新的实例(x, y')来代替(x, y)，但要求 $y' \subset y$ 且 y'是频繁的。这样增加了多标签的数量，也消除了标签不平衡，改善了朴素 LP 算法。但是这也常常扔掉了一些重要的标签信息，错过了许多重要的标签组合，因此，第三个弊端仍然没有得到解决。Tsoumakas 等人提出了 RA*k*EL(Random *k*-labElsets)算法[7]，它通过创建 *m* 个 LP 分类器，并综合它们的预测结果，以达到检测不可见的标签组合的目的。但是，RA*k*EL 方法的训练时间过长。与此同时，Read 等人也提出了 EPS(Ensemble of Pruned Sets)算法[8]，它通过随机抽取训练集形成 *m* 个子集来创建 *m* 个 PS 分类器，然后再组合这 m 个分类器的预测结果，这与 RA*k*EL 算法类似。同样的，EPS 算法发现那些不可见的标签组合也需要很长的训练时间，并且容易丢失大量标签信息。

通过回顾目前用得比较广泛的几种基于 LP 的算法，讨论它们各自的优缺点，针对它们的不足，并结合聚类算法的基本思想以及它们在实际中的一些应用[9-13]，我们提出了一种新的算法 LCMLC(Multi-Label Classification Based on Label Clustering)，它采用层次结构的平衡 *k*-means 聚类方法将相关度高的标签聚合在一起来形成新的标签组合，以此来发现那些重要但不可见的标签组合。

2 基于标签聚类的多标签分类算法(LCMLC)的设计

LCMLC 算法是通过聚类的方法将相关度高的标签聚合在一起来形成新的标签组合，以此来发现那些重要但不可见的标签组合。该算法首先是基于这么一个假设，即彼此相关度高的标签具有更大可能形成一个标签组合。它们是基于训练集聚合得到这些标签组合，并把它们表示为聚类簇。这里的每个标签都用一个 *N*-维的布尔向量表示，即 0 和 1，如果第 *i* 维是 1 时就表示它在训练集中第 *i* 条训练实例出现过，否则就为 0。然后对这些标签向量进行聚类形成各个聚类簇，而这里采用的聚类算法是层次结构的平衡 *k*-means 聚类方法。得到这些聚类簇之后，就可以通过这些聚类簇在训练集中找到那些不可见的重要标签，并将其加入到原来的训练集形成新的训练集，最后用新的训练集来建立 PS 分类器进行多标签分类。

2.1 层次结构的平衡 *k*-means 聚类方法

在这里，我们提出了一种新的聚类方法，叫做层次结构的平衡 *k*-means 聚类方法，它是对传统 *k*-means 聚类方法进行了扩展，对其聚类的每个聚类簇的大小进行了明确的限制，并在聚类的过程按自顶向下的方式在各层都采用平衡聚类。由于我们聚类的对象只是标签，所以我们只考虑训练集 (x_i, y_i) 的标签部分 y_i，因此最终我们会得到一棵类似于树结构的标签聚类树，而这个树中的结点即为各个标签子集。图 1 详细介绍平衡 *k*-means 聚类算法的实现过程，其中输入为标签集合 $L_n \subseteq L$，标签数据集 D_i，聚类簇的数量 k 以及迭代的次数 T。

2.2 训练集的修改过程

利用平衡 *k*-means 聚类方法得到这些聚类簇之后，我们就可以通过这些聚类簇在训练集中找到那些不可见的重要标签，然后将它们加入到原来的训练集形成新的训练集。

其具体过程如下：

(1)对于每一个聚类簇 c(实际是一个标签组合)，然后我们将这个聚类簇 c 的所有子标签组合 $y \subseteq c$ 都作为新的标签组合加入新标签组合集中，当然这里的子标签组合也是有限制的，即 $|y| \geqslant t$，它是一个可变的参数。在图 2 中，$t = 2$；

(2)找到所有的新标签组合后，我们要遍历整个训练集，对任意实例(x, y)，如果 y 包含某个新的标签组合 y'，我们就将实例的 x 部分和 y'组合形成一条新的实例(x, y')添加到训练集中形成新的训练集，其详细过程见图 4(算法描述)和图 3(图 3 的数据是相对图 2 中的数据进行表示的)；

(3)最后，我们在新的训练集上进行学习，形成最终的 PS 分类器。

3 实验

为了验证 LCMLC 算法的有效性，我们将在多个多标签数据集上分别进行相实验，并将它与其他基于 LP 的多标签分类算法进行比较以及分析。本节主要介绍以下几个方面：实验所选取的数据集；算法的评价指标；实验的结果以及分析。

3.1 数据集

本次实验采用的数据集有 enron，gebase，medical，yeast，tmc2007[①]。其中 enrorn，medical 和 tmc2007 分别是邮件信息，医学方面信息以及航空安全方面信息的文本类数据集；而 gebase 以及 yeast 则分别是用于蛋白质分类和基因功能分类的生物学类数据集。下面表 1 是对这五个数据集的统计信息描述。

[①] 上述五个数据集均源自 http://mulan.sourceforge.net/datasets.html

```
输入：聚类簇数 k，标签集合 Ln，标签数据集 Di，迭代次数 T
输出：k 个平衡的标签聚类簇
具体过程：
For i ← 1 to k do
    Ci ←空集 ci ← Ln随机分配集合；
    ci← Ln随机分配集合；
End For
While T > 0 do
  Foreach y∈Ln
    For i ← 1 to k do
      dyi ← distance(y,ci,Di);
    End For
    Finish ← false;
    v← y;
    while not finished do
      j ← argmin(dvi);
      Insert sort (v,dv) to sorted list Cj ;
      If |Cj| > (|Ln|/k) then
          v ← 移除 Cj 中最后一个元素;
        dvi ← 无穷;
      End If
      Else
          Finished ← true;
      End Else
    End While
  End Foreach
  Recalculate centers;
  T ← T-1
End While
Return C1,C2,…,Ck
```

图 1 层次结构平衡 ***k*-mean** 聚类算法

Fig. 1 Hierarchical balanced *k*-Means Algorithm

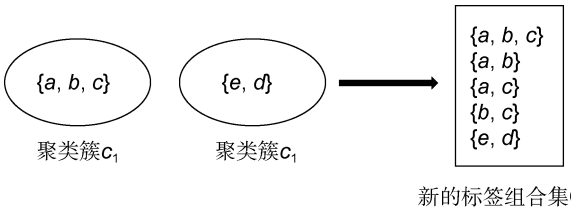


图 2 由聚类簇 ***c*** 形成新的标签组合

Fig. 2 New Label-set Come from Cluster *c*

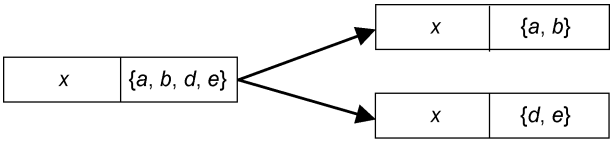


图 3 **LCMLC**：训练实例的修改

Fig. 3 LCMLC：Modification of Training Data

```
输入：训练实例集 D={(x1, Y1),(x2, Y2),…,(xn, Yn)};
新的标签组合集 C
输出：新的训练实例集 D

修改过程：
For (x,y)∈D do
   For l⊂y do
     If l∈C then
        T ← T∪{(x,l)}
     End If
   End For
End For
```

图 4 LCMLC：训练实例的修改过程

Fig. 4 LCMLC：The Modification Process of Training Data

表 1　多标签数据集及统计信息
Tab. 1　Multi-label datasets and their statics

名称	领域	训练实例	测试实例	属性个数	标签个数	L_{CARD}
enron	文本	1123	579	1001	53	3.378
genbase	生物	463	199	1186	27	1.252
medical	文本	645	333	1449	45	1.245
tmc2007	文本	21519	7077	500	22	2.158
yeast	生物	1500	917	103	14	4.237

3.2　评价指标

在本次实验中，我们选取了三种评价指标，分别是汉明损失，子集准确率以及 F-measure。在这里，我们以 y_i 表示实例第 i 个标签的预测值，以 c_i 表示实例第 i 个标签的真实值，N 表示测试实例的个数，m 表示标签的个数。以上三种评价指标的具体意义以及定义如下：

Hamming loss 表示的是实例中被错误分类的比例，它包括以下两种情况：预测的标签不属于该实例和属于该实例的标签没有被预测。Hamming loss 的值越小，则表示该分类算法的性能越好。当 hamming loss=0 时，则性能是完美的。这个指标的定义如下：

$$\text{Hamming-Loss}(h,D)=\frac{1}{N}\sum_{i=1}^{N}\frac{Y_i\oplus C_i}{m} \tag{1}$$

子集精确率表示的是分类正确率。子集精确率认为当预测标签集合和真实标签集合完全相同时才是分类正确，否则就是错误，它统计的是测试集中被完全正确分类的实例的比例。所以当子集精确率值越大时，则表示该分类算法性能越好。这个指标的定义如下：

$$\text{SubsetAccuracy}(h,D)=\frac{1}{N}\sum_{i=1}^{N}I(C_i=Y_i) \tag{2}$$

本次实验采用的是基于实例的 F-measure 值评价指标。F-measure 值也称为综合分类率，它是结合精确率和召回率得到的评价指标，其中精确率统计的是被预测标签集中有多少标签被预测正确的，而召回率则是指在真实标签中有多少标签被正确预测。所以其值为 1 时，F-measure 达到最好；反之为 0 时最差。这个指标的定义如下：

$$\text{F}=\frac{1}{N}\sum_{i=1}^{N}\frac{2|Y_i\cap C_i|}{|C_i|+|Y_i|} \tag{3}$$

3.3　评价指标

针对提出的算法 LCMLC，本次实验采用 5 重交叉验证的方式来评价其性能。为了验证算法 LCMLC 的有效性，我们还将其与 PS 算法、PPT 算法、EPS 算法分别进行了比较。所有实验都是在 Mulan[②]平台[14]下进行，采用的基分类器则为 WEKA[③]平台下的决策树算法 J48。在评价各个算法时，我们采用的是将原始数据集分割为训练集以及测试集两部分来进行实验。

其次我们需要对以上比较的算法进行一些简单的参数设置。其中对于算法 LCMLC，经多次重复实验以及考虑聚类算法的时间效率，我们最终将聚类划分的 k 值设置为 3 以及聚类迭代的次数设置为 20 次，分类时被预测为相关的标签的阈值是 0.5，标签集合的个数为标签个数的 2 倍或实例个数。对 EPS 算法，同样也将其模型数设置为 20。

3.4　实验结果及分析

本小节主要介绍的是 LCMLC 算法与其他四种同类型算法的实验结果比较。表 2 到表 4 分别给出了 PS 算法，PPT 算法，EPS 算法，LCMLC 算法在 enron，genbase，medical，yeast 和 tmc2007 这 5 个数据集上三种评价指标(即汉明损失，子集精确率以及 F-measure)的值。另外，表 6 中给出了 EPS 算法和 LCMCL 算法在分类器建立时间上的比较。其中每行中用黑色加粗的数据为 4 个算法中在该数据集上表现最好的那个算法。

从表 2 至表 4 中可以看到，LCMLC 算法在 5 个数据集的 15 个评价指标上有 10 个评价指标都是最好的，这足以说明 LCMLC 算法相对于其他同类型算法的优越性。在 3 种评价指标中，LCMLC 算法在子集精确率上的表现最好，对 5 个数据集中的 4 个它都是最佳的，对剩下的一个数据集也是仅次于最佳，这说明 LCMLC 算

② Mulan 可从 http://mlkd.csd.auth.gr/multilabel.html 获取
③ WEKA 可从 http://www.cs.waikato.ac.nz/ml/weka 获取

表 2　各算法在 5 个数据集上的汉明损失

Tab. 2　Hamming-Loss of the above algorithms on all 5 datasets

	PS	PPT	EPS	LCMLC
enron	0.0624	0.0691	0.0517	**0.0514**
genbase	**0.0039**	0.0051	**0.0039**	0.0046
medical	0.0130	0.0141	0.0116	**0.0109**
tmc2007-500	0.0709	0.0786	**0.0549**	0.0558
yeast	0.2740	0.3006	0.2720	**0.2645**

表 3　各算法在 5 个数据集上的子集精确率

Tab. 3　Subset Accuracy of the above algorithms on all 5 datasets

	PS	**PPT**	**EPS**	**LCMLC**
enron	0.1322	0.1210	0.1310	**0.1357**
genbase	0.9380	0.9350	0.9365	**0.9399**
medical	0.6758	0.6533	0.6830	**0.6891**
tmc2007-500	0.3587	0.3511	**0.3852**	0.3659
yeast	0.1407	0.1499	0.1680	**0.1693**

表 4　各算法在 5 个数据集上的 F-measure 值

Tab. 4　F-measure of the above algorithms on all 5 datasets

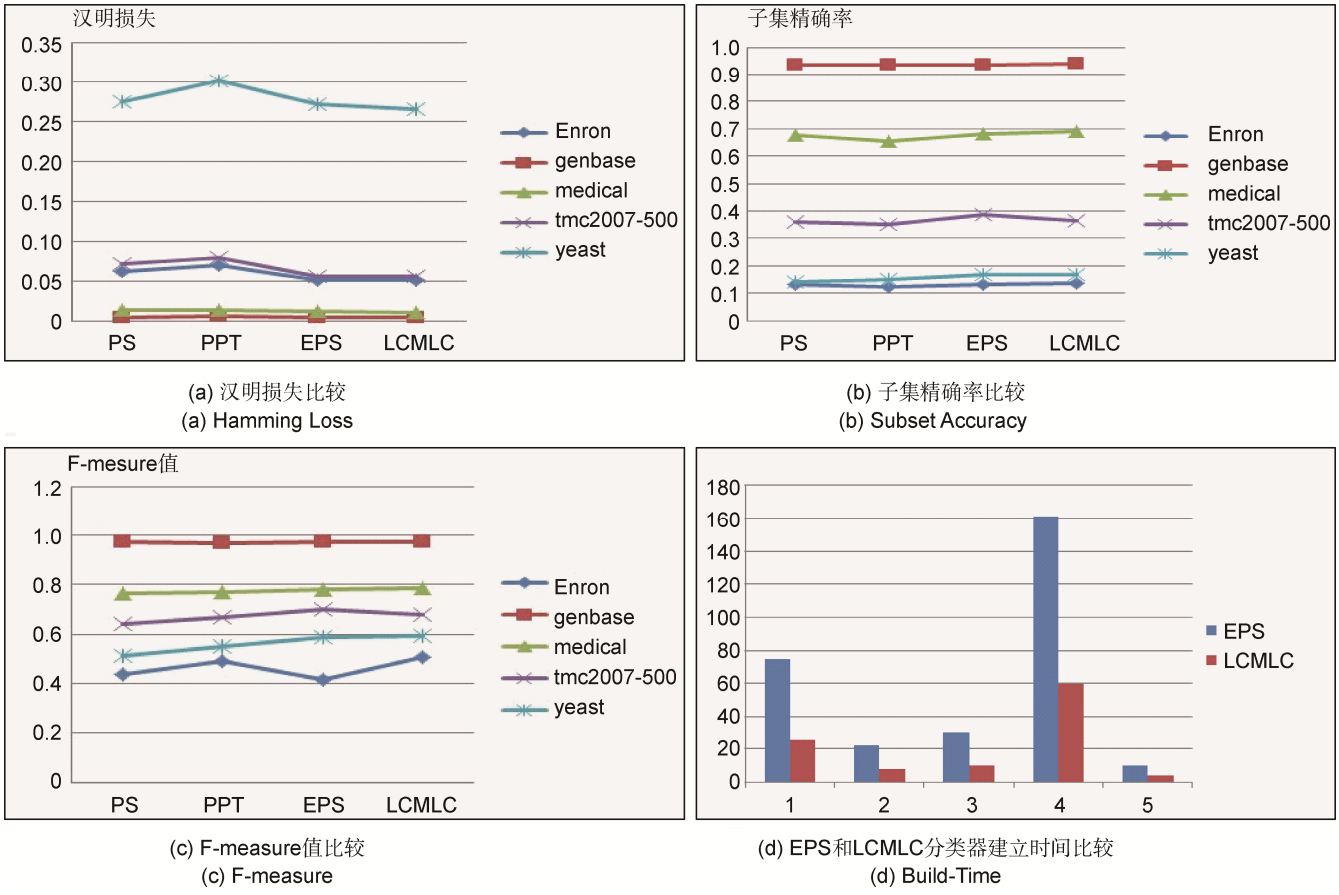
	PS	**PPT**	**EPS**	**LCMLC**
enron	0.4378	0.4928	0.4178	**0.5048**
genbase	0.9729	0.9662	**0.9730**	0.9737
medical	0.7620	0.7703	0.7780	**0.7842**
tmc2007-500	0.6399	0.6672	**0.6995**	0.6774
yeast	0.5126	0.5502	0.5896	**0.5940**

表 5　比较 EPS 算法和 LCMLC 算法分类器的建立时间

Tab. 5　Build Time of EPS *vs* Build Time LCMLC

	EPS	**LCMLC**
enron	74.62	**25.85**
genbase	22.38	**8.18**
medical	30.44	**10.56**
tmc2007-500	160.78	**60.09**
yeast	10.27	**3.91**

法在优化子集精确率上的有效性。同时，LCMLC 算法在 enron 和 yeast 上的表现较好，根据表 1 中对数据集的描述，这两个数据集中与每个实例相关的标签个数分别为 3.378 和 4.237，是 5 个数据集中值最大的两个数据集，这说明 LCMLC 算法适用那些和实例相关标签数较多的数据集，而与实例相关标签数较多表明标签间的依赖关系比较强，因此 LCMLC 算法能够有效的寻找到那些隐藏的但是依赖关系又较强的标签集。同时，可以看出，LCMLC 算法总的分类准确率是明显优于 PS 算法和 PPT 算法，说明 LCMLC 算法确实在某种程度上克服了 PS 算法存在的缺点，提高了分类结果的准确率。结合图 5-a 到 5-c 可以看到，虽然 EPS 算法和 LCMLC 算法两种



(a) 汉明损失比较
(a) Hamming Loss

(b) 子集精确率比较
(b) Subset Accuracy

(c) F-measure值比较
(c) F-measure

(d) EPS和LCMLC分类器建立时间比较
(d) Build-Time

图 5　4 个算法的比较

Fig. 5 Comparisons of 4 algorithms

算法在各个指标上的结果都是优于其他两种算法。但从表 5 以及图 5-d，我们可以看出，就分类器的建立时间而言，LCMLC 算法是更具有优势的。

上述实验结果表明 LCMLC 算法确实能通过聚类的方法来找到训练集中潜在的重要标签，并将这些重要标签结合原来的训练集形成更加完备的新的训练集，从而建立更加优化的分类器模型，提高分类的预测准确度。

4 总结

本文主要研究了如何从训练集中挖掘出那些隐藏的但又具有较强依赖关系的重要标签集合，从而形成更加完备的新的训练集来提高多标签分类器的性能。为此，本文提出了基于标签聚类的分类算法，它通过层次平衡聚类的方法形成聚类簇来挖掘隐藏的重要标签集合，以此得到新的训练集来进行多标签分类。

本文将 LCMLC 算法与已有的几个 LP 多标签学习算法在 5 个多标记数据集上进行了比较，实验结果说明了 LCMLC 算法确实是有效的。该算法能够在多个评价指标上都取得较好的结果，尤其在 Hamming Loss 和 F-Mesusre 评价指标上相较 PS 算法以及 PPT 算法具有明显的优势，而就建立时间而言，LCMLC 算法相较 EPS 算法更具效率。但是，本算法在研究标签之间的相关性时，采用的只是简单的 *k*-mean 聚类方法，如何使用更合适的聚类方法以更好的发现隐藏的标签关系是将来值得研究的问题。

参考文献

[1] Madjarov G, Kocev D, et al. An extensive experimental comparison of methods for multi-label learning[J]. In:Mario Hernandez, Jordi Vitria and Joao MiguelSanches(eds.). Pattern Recognition, 2012, 45(9): 3084–3104.

[2] Zhang M, and Zhou Z. A k-nearest neighbor based algorithm for multi-label classification[C]. In: Proceedings of the 1st IEEE International Conference on Granular Computing. Beijing, China, 2005: 718–721.

[3] Read J, Pfahringer B, and Holmes G, et al. Classifier chains for multi-label classification[C]. In: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part II. Bled, Slovenia, 2009: 254–269.

[4] Sucar J E, Bielza C, Moral E F, et al. Multi-label Classification with Bayesian network-based chain classifiers[J]. Pattern Recognition Letters, 2013, 22(1): 14–22.

[5] Goncalves E C, Freitas A A. A Genetic Algorithm for Optimizing the Label Ordering in Multi-Label Classifier Chains[C]. In: IEEE International Conference on Tools with Artificial Intelligence(ICTAI). Washington DC, USA, 2013: 469–476.

[6] Read J. A pruned problem transformation method for multi-label classification[C]. In: Proceedings of the NZ Computer Science Research Student Conference, Christchurch, New Zealand, 2008: 143–150.

[7] Tsoumakas G, Katakis I, and Vlahavas I. Random k-labelsets for multilabel classification[J]. IEEE Transactions On Knowledge and Data Engineering, 2011, 23(7): 1079–1089.

[8] Read J, Pfahringer B, Holmes G, et al. Multi-label classification using ensembles of pruned sets[C]. In: Proceeding of the IEEE International Conference on Data Mining. Vancouver, Canada, 2011: 995–1000.

[9] Banerjee A, Ghosh J. Scalable Clustering Algorithms with Balancing Constraints[J]. Data Mining and Knowledge Discovery, 2006, 13(3): 365–395.

[10] 郑文超, 徐鹏. 利用word2vec 对中文词进行聚类的研究[J]. 软件, 2013, 34(12): 160–162.

[11] HE Kun, YUAN Ling, LI Zhuming. Distributed Clustering and Greedy Scheduling Algorithm based on Task Duplication[J]. The Journal of New Industrialization, 2012, 2(11): 1–11.

[12] Cao Ge, Cheng Yuhu. K-means Clustering Algorithm Based on Initial Clustering Centre Selection and Points Division[J]. The Journal of New Industrialization, 2011, 1(5): 90–94.

[13] Zhang Xiaohua, Wang Le. Change detection based on multiscale and clustering[J]. The Journal of New Industrialization, 2011, 1(3): 72–79.

[14] Tsoumakas G, Spyromitros-Xioufis E, Vilcek J, et al. Mulan: A java library for multi-label learning[J]. Journal of Machine Learning Research, 2011, 12: 2411–2414.