基于 MapReduce 的多级特征选择机制

宋哲理[1] 王 超[2] 王振飞[3]

(郑州财税金融职业学院 郑州 450048)[1] (中国船舶重工集团公司第七一三研究所 郑州 450015)[2]

(郑州大学信息工程学院 郑州 450001)[3]

摘 要 特征选择是文本分类的关键步骤,分类结果的准确度主要取决于选择得到的特征词的优劣。文中提出一种基于 MapReduce 的多级特征选择机制,一方面利用改进的 CHI 特征选择算法进行初次筛选,再通过互信息方法对初选结果进行噪声词过滤、优质特征词前置等操作;另一方面将本机制载入 MapReduce 模型中,以减少多级特征选择作用于海量数据的时间消耗。实验结果表明,该机制能在较短的时间内处理大规模数据,同时也提升了文本分类的精度。

关键词 文本分类,特征选择,CHI,互信息,MapReduce

中图法分类号 TP301 **文献标识码** A

Multi-level Feature Selection Mechanism Based on MapReduce

SONG Zhe-li[1] WANG Chao[2] WANG Zhen-fei[3]

(Zhengzhou Vocational College of Finance and Taxation, Zhengzhou 450048, China)[1]

(The 713th Research Institute of China Shipbuilding Industry Corporation, Zhengzhou 450015, China)[2]

(School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China)[3]

Abstract Feature selection is a committed step of text classification. The classification accuracy mainly depends on the merits and demerits of the selected feature words. This paper proposed a multi-level feature selection mechanism based on MapReduce. On the one hand, the mechanism screens the original dataset by an improved CHI feature selection algorithm, then uses the mutual information method to filter the noise words and to put the high quality feature words forward for the primaries. On the other hand, the time consumption of multi-level feature selection is reduced by introducing the mechanism into MapReduce model. Experimental results show that the mechanism improves both the classification accuracy and its runtime when dealing with big data problems.

Keywords Text classification, Feature selection, CHI, Mutual information, MapReduce

1 引言

文本分类是数据挖掘的研究热点,指将一系列文本划分到预先设定的类目中。特征选择是文本分类的关键步骤,指从文本中挑选出最能表示此文本信息的一类词充当分类参照,因此特征选择算法的好坏直接影响文本分类的精度[1]。

MapReduce 是 Google 公司处理为大规模数据而提出的基于分布式并行计算的编程模型,其核心策略是分而治之,即将庞大的作业集均衡分配到多个节点进行处理[2]。通过对数据的拆分与组合,不仅提高了并行处理数据的能力,也极大地提升了系统性能。

传统的特征选择方法有 χ^2 统计量(Chi Square Statistic, CHI)、互信息(Mutual Information, MI)、文档词频(Document Frequency, DF)等。传统的特征选择算法考虑的制约条件较少,分类精度偏低;多级特征选择模型能有效解决分类精度偏低的问题。文献[3]提出了基于特征贡献度(Feature Contribution Degree, FCD)和隐式语义索引(Latent Semantic Indexing, LSI)的多级特征模型。文献[4]则利用类似 CHI 的提取器进行初级选择。然后再使用基于 LSI 的遗传算法(Genetic Algorithm, GA)进行二级选择。文献[5]提出了以信息增益序列(Information Gain Sequences, IGS)为初级选择,GA 方法为二级选择的多级特征选择模型。上述方法虽然能提高分类精度,但也存在明显不足,在完成二级特征选择后,特征词集合会丢失一部分原始语义,原因是多级特征选择的过滤性强且很少考虑上下文。文献[6]首先用 IGS 对文本进行初级选择,然后用马尔科夫链过滤(Markov Blanket Filtering, MBF)算法进行二级选择。此方法在一定程度上保留了较完整的原始语义,但具有较高的时耗。类似地,文献[7]用一种改进的 TF-IDF 方法做初步筛选,再用最大相关最小冗余(MRMR)进行第二次筛选,二次筛选时使用增量式搜索来减少时耗。此方法在无损原始语意和消除冗余的同时,还在时耗上做出优化,但宏观上时耗优化仍有改进空间。

为在改进特征选择算法的同时尽可能地保留原始语义,并且保证模型具有较低的时间消耗,本文提出一种基于 Ma-

本文受国家自然科学基金项目(61379079)资助。

宋哲理(1983—),女,硕士,讲师,主要研究方向为计算机应用;**王 超**(1988—),男,硕士,工程师,主要研究方向为机器学习,E-mail:854909839@qq.com(通信作者);**王振飞**(1973—),男,博士,副教授,CCF 会员,主要研究方向为社交网络、大数据分析等。

pReduce 的多级特征选择机制，利用改进的 CHI 方法进行初级选择，然后用互信息方法过滤 CHI 方法产生的噪声词，并将合适的特征词前置。考虑到互信息方法需要消耗大量时间，将本模型载入到 MapReduce 框架中，充分发挥其处理海量数据的优势，提升模型的执行效率。

2　相关理论

2.1　CHI 方法

CHI 方法是基于统计学的分类方法，通过比较实际观测值和理论值的偏移量，来判断理论值的正确与否[8]。应用时，经常假设待评判的两个变量相互独立，比较实际观测值和理论值的偏移量，若偏移量浮动在偏移阈值之内，则判定两个变量相互独立，可理解为造成此次偏移是因为测量误差或小概率事件的发生；若偏移量浮动在偏移阈值之外，则认为两个变量存在相关性。若偏移量记为 D，理论值记为 E，一组实际观测值表示成 $(x_1,\cdots,x_i,\cdots,x_n)$，则 CHI 方法的计算公式如式(1)所示：

$$DI=\sum_{i=1}^{n}\frac{(x_i-E)^2}{E} \tag{1}$$

将 CHI 方法用于特征选择时，有如下过程：提出假设，假设特征词(记作 t_i)与类别(记作 C_j)相互独立，即两者不相关；计算偏移量，计算每个特征词 t_i 与类别 C_j 的偏移值；选择特征词，偏移量越大说明特征词 t_i 与类别 C_j 的相关性越大，所以将计算的偏移量降序排列，取阈值之外的前 k 个。对于一组文本集(记作 Ds)，篇数记作 N，与类别 C_j 相关的文档篇数记作 M，引入情形分析表来更直观地表述字母含义，如表 1 所列。

表 1　情形分析表

特征选择	属于类别 C_j	不属于类别 C_j	总计
包含词 t_i	A	B	$A+B$
不包含 t_i	C	D	$C+D$
总数	$A+C\ (M)$	$B+D\ (N-M)$	N

考查特征词 t_i 与类别 C_j 之间相关性的 CHI 方法可用式(2)所示的等价公式来表示。

$$\chi^2(t_i,C_j)=\frac{(AD-BC)^2}{(A+B)(C+D)} \tag{2}$$

其中，$\chi^2(t_i,C_j)$ 表示偏移值。由式(2)可知，尽管 CHI 方法有较好的分类效果，但也存在明显缺陷：

1)CHI 方法只考虑文档频数，忽略了特征词频数，致使 CHI 方法更多地选择那些在多数类别的多数文档中低频出现的特征词，反而过滤掉那些在某类文档中高频出现的特征词。

2)产生新的噪声特征词。当 AD 值很小、BC 值很大时，$(AD-BC)^2$ 很大，这类特征词也会被挑选出来作为噪声特征词。

2.2　互信息法

互信息是基于信息论的分类方法。与 CHI 方法相似，互信息同样用于描述两个变量之间的相关性[9]。互信息值越大，表明两个变量的相关程度越大。将互信息方法用于特征选择时，特征项 t_i 与类别 C_j 的互信息记作 $MI(t_i,C_j)$，其计算公式如式(3)所示：

$$MI(t_i,C_j)=\log\frac{p(t_i,C_j)}{p(t)\times p(C_j)} \tag{3}$$

其中，分类类别数记为 m，$p(t_i,C_j)$ 指特征词 t_i 在类别 C_j 中出现的概率，$p(t)$ 指特征词 t_i 在整个训练文本集中出现的概率，$p(C_j)$ 指类别为 C_j 的文档在整个文本集中出现的概率。根据上文提到的文本集 Ds 与表 1，互信息公式也可表示成：

$$MI(t_i,C_j)=\log\frac{AN^2}{(A+B)M^2} \tag{4}$$

2.3　MapReduce

MapReduce 是一个大数据集的分布式处理平台，具有高可用性、高可扩展性和高容错性能[10]。MapReduce 模型集成多个计算机节点对海量数据进行并行处理[11]。

MapReduce 主要包括 Map 阶段(映射)和 Reduce 阶段(归约)[12-13]，图 1 描述了 MapReduce 的工作过程。原始数据经过预处理并按块划分输入到 Map 函数中，经过中间过程的缓存、排序等处理，最后由 Reduce 函数把具有相同标签的值合并在一起。数据将以键值对〈Key,$Value$〉的形式保存或流动。Map 函数和 Reduce 函数都可以由用户预先编写设定[14]。

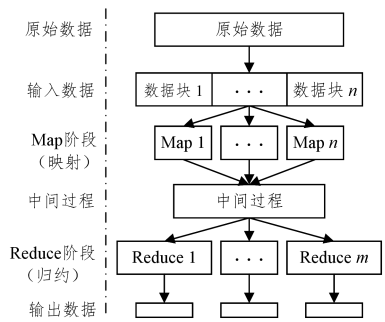

图 1　MapReduce 的工作流程

3　基于 MapReduce 的多级特征选择机制

3.1　多级特征选择模型

针对上文描述的 CHI 方法的不足，文中提出一种基于改进 CHI 方法以及互信息法的两级特征选择模型。改进的 CHI 方法(记为 NCHI)作为初级特征选择，可以消除传统 CHI 方法只考虑文档频数而忽略特征词频数带来的影响，将利用初级选择得到的特征词集再经过互信息法进行二级选择，过滤掉在初级选择时生成的噪声词，如图 2 所示。

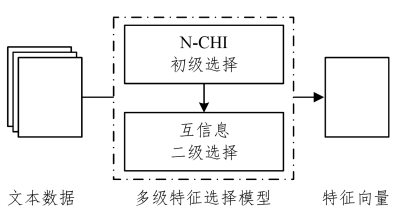

图 2　多级特征选择模型

不同于传统 CHI 方法，NCHI 中引入类内频数作为新的参数，记作 CIF_{ij}。类内频数表示特征词 t_i 在类别 C_j 的所有文档中出现次数的最大值，计算公式如式(5)所示：

$$CIF_{ij}=\max_{u=1}^{M}\{tf_{iju}\} \tag{5}$$

其中，tf_{iju} 为特征词 t_i 在类别 C_j 下的文档中出现的次数，M 为类别 C_j 的文档数(同表 1 中的 M)。由式(5)可知，tf_{iju} 越

大，说明特征词 t_i 在类别 C_j 下的文档 d_u 中出现的频数越大，则将此特征词作为该类别的候选特征词的可能性就越大。

综上所述，NCHI 方法的计算公式如式(6)所示，其中 $\chi^2_{new}(t_i,C_j)$ 为 NCHI 方法的值。

$$\chi^2_{new}(t_i,C_j)=CIF_{ij}\times\chi^2(t_i,C_j) \tag{6}$$

NCHI 中引入类内频数可以解决传统 CHI 方法忽略特征词频数的问题，但没有解决 CHI 方法产生额外噪声词的问题，因此在二级选择时，使用互信息法可以有效过滤在 NCHI 初级选择过程选中的噪声词。由式(2)可知，噪声词有如下特点：1)包含特征词 t_i 但不属于类别 C_j；2)属于类别 C_j 但没有包含特征词 t_i。这类噪声词中特征词 t_i 和类别 C_j 具有很弱的相关性；而互信息方法正是一种考查变量相关性的分类方法，所以将其作为二级选择可以充分发挥优势，有效过滤初级选择中生成的噪声词；另外，优秀的特征词因具有较高的相关度而被前置，不会被取词值域截断而遗漏。实际上，相比 IGS-GA 等算法，NCHI 是一种弱特征选择方法，选择后不会有损原始语意，但会保留部分噪声词；而互信息的优势在于根据相关性大小进行选择，这样可以有效过滤噪声词，最终选择最佳特征词。

3.2 基于 MapReduce 的多级特征选择机制

本文提出的多级选择模型能有效提升分类精度，但 NCHI 方法中类内频数的计算和互信息方法的使用会造成大量的时间开销，降低模型的执行效率。因此，将模型载入 MapReduce 框架中，利用其处理海量数据的高效性优势来缩短多级特征选择模型的执行时间。基于 MapReduce 的多级特征选择机制的流程如图 3 所示。

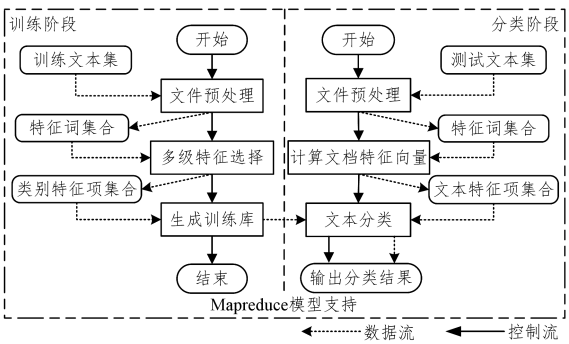

图 3 基于 MapReduce 的多级特征选择机制的流程

从图 3 中可以看出，基于 MapReduce 的多级特征选择模型分为两部分：文本训练阶段和文本分类阶段。

3.2.1 文本训练阶段

图 4 给出基于 MapReduce 的多级特征选择文本训练阶段模型。

对于给定的训练文本集，首先进行预处理，包括分词断句、去停用词等，将处理后的文本集标记为一阶训练集输入训练模型。如图 4 所示，文本训练阶段包括 3 个 MapReduce 过程。第(1)个 MapReduce 过程的作用是计算特征词的类内频数。一阶训练集分块输入到不同的节点执行 Map 函数，主要执行式(5)的计算方法，得到基于〈特征词，〈类别，tf_{iju}〉〉的键值对，记作〈Tid，〈Cid，tf_{iju}〉〉；经过中间过程处理，最后经过 Reduce 函数，以 Tid 为分类主键，查找 tf_{iju} 的最大值 CIF_{ij}，从而得到了键值对〈特征词，〈类别，类内频数〉〉，记作〈Tid，〈Cid，CIF_{ij}〉〉。将键值对作为二阶训练集输入到第(2)个 MapReduce 过程，其中 Map 阶段的核心计算公式为式(6)，将得到的特征词 t_i 和类别 C_j 的 NCHI 值存储为键值对，记作〈Tid，〈Cid，χ^2_{new}〉〉；经过中间过程处理后，以 Cid 为分类主键，执行 Reduce 函数，降序排列得到的 χ^2_{new} 值，取预先规定的前 f 个，从而得到类别 C_j 的初级特征向量，最后将其归入初级训练库。

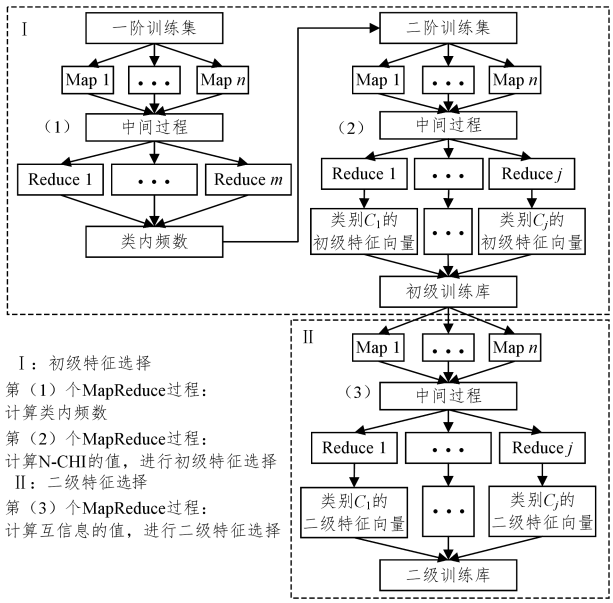

图 4 基于 MapReduce 的多级特征选择训练模型

初级训练库数据收集完成后，可转而进行二级特征的选择。将不同类别的初级特征向量以类为整体按块划分，输入到不同节点上执行 Map 函数，主要执行式(4)的计算方法，得到键值对〈〈类别，特征词〉，MI 值〉，记作〈〈Cid，Tid〉，MI〉；经过中间过程处理后，以 Cid 为分类主键，执行 Reduce 函数，同样降序排列得到的 MI 值，取预先规定的前 s 个，得到类别 C_j 的二级特征向量，最后将其归入二级训练库。根据上述流程，算法 1 给出基于 MapReduce 的多级特征选择训练算法描述。

算法 1 基于 MapReduce 的多级特征选择训练算法

输入：一阶训练集，类别 C，文档 d，特征词 t

输出：类别的二级特征向量

```
1.  Map1
2.  {
3.      //计算一阶训练集中特征词的类内频数
4.      for each d_u ∈ C_j do
5.          for each t_i ∈ d_u do
6.              计算 tf_iju;
7.              输出中间键值对〈Tid,〈Cid,tf_iju〉〉;
8.          end for
9.      end for
10. }
11. Reduce1
12. {
13.     输入中间键值对〈Tid,〈Cid,tf_iju〉〉;
14.     for each Tid do
15.         计算类内频数;
16.         输出键值对〈Tid,〈Cid,Itf_iju〉〉;
17.     end for
```

```
18.  }
19.  //获取类别的初级特征向量
20.  Map2
21.  {
22.    输入键值对〈Tid,〈Cid,Itf_iju〉〉;
23.    for each Tid do
24.      计算 NCHI 的值 χ²_new;
25.      输出键值对〈Tid,〈Cid,χ²_new〉〉;
26.    end for
27.  }
28.  Reduce2
29.  {
30.    输入中间键值对〈Tid,〈Cid,χ²_new〉〉;
31.    for each Cid do
32.      χ²_new降序排列,并取前 f 个;
33.      输出类别的初级特征向量;
34.    end for
35.  }
36.  //获取类别的二级特征向量
37.  Map3
38.  {
39.    输入类别的初级特征向量;
40.    for each Cid do
41.      for each Tid do
42.        计算 MI 的值;
43.        输出键值对〈〈Cid,Tid〉,MI〉;
44.      end for
45.    end for
46.  }
47.  Reduce3
48.  {
49.    输入键值对〈〈Cid,Tid〉,MI〉;
50.    for each Cid do
51.        MI 降序排列,并取前 s 个;
52.        输出类别的二级特征向量;
53.    end for
54.  }
```

3.2.2　文本分类阶段

图 5 是基于 MapReduce 的多级特征选择文本分类阶段的模型。

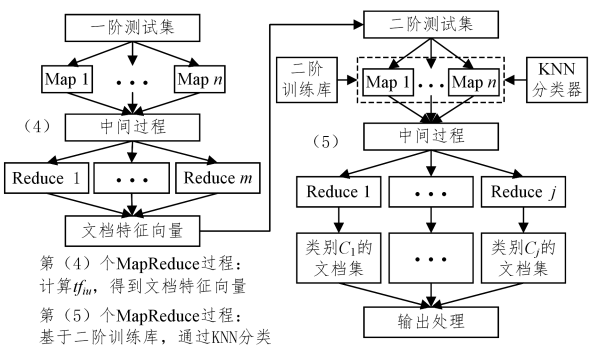

图 5　基于 MapReduce 的多级特征选择测试模型

分类模型的数据来源于经过预处理的测试文本集，记作一阶测试集。第(4)个 MapReduce 过程的作用是统计特征词 t_i 在测试文档中出现的次数，进而生成各个文档特征向量。需强调的是，此过程只关心特征词 t_i 在文档 d_u 中出现的频数（记作 tf_{iu}），不关心文档类别。一阶测试集以文本为整体，按块划分归入不同节点，执行 Map 函数，以计算特征词的 tf_{iu}，之后得到键值对〈特征词，文档，tf_{iu}〉，记作〈Tid，Did〉，tf_{iu}〉；经过中间过程处理，最后经过 Reduce 函数，将同 Did 的特征词按 tf_{iu} 值降序排列，取预先规定的前 k 个值组成文档 d_u 的特征向量。

第(4)个 MapReduce 过程的结果归档后记为二阶测试集，输入到第(5)个 MapReduce 过程。第(5)个 MapReduce 过程的作用是基于二阶训练库，用 KNN 分类器分类二阶测试集。经过 KNN 分类后得到基于〈类别，文档〉的键值对〈Cid，Did〉，同样是经过中间过程处理后，最后执行 Reduce 函数，以类别为分类依据，将同 Cid 的文档归类，即得到最终的分类结果，输出处理。分类算法的描述如算法 2 所示。

算法 2　基于 MapReduce 的多级特征选择分类算法

输入：一阶测试集，文档 d，特征词 t

输出：各个类别的文档集

```
1. Map4{
2.    for each t_i ∈ d_u do
3.      计算特征词的 tf_iu;
4.      输出中间键值对〈Tid,Did〉,tf_iu〉;
5.    end for
6. }
7. Reduce4{
8.    for each Did do
9.      tf_iu倒序排列,取前 k 个;
10.     输出文档特征向量;
11.   end for
12. }
13. Map5{
14.   输入二阶训练库;
15.   输入二阶测试集;
16.   for each d_u do
17.     KNN 分类器分类;
18.     输出中间键值对〈Cid,Did〉;
19.   end for
20. }
21. Reduce5{
22.   for each Cid do
23.     相同 Cid 索引号的文档归并;
24.     输出类别文档集;
25.   end for
26. }
```

在算法 1 和算法 2 中，每个 Mapreduce 过程均可在给定节点上分布式运行，而且对于每个 Map 阶段和 Reduce 阶段，也可以在多个节点上并行执行。相对于前文所提及的文献，该方式可大幅度消减时间上的复杂度，提升系统效率。

4　实验与分析

4.1　实验数据

本文使用复旦大学（复旦大学计算机信息与技术系国际数据库中心自然语言处理小组）提供的语料库[1)]，选取其中 6 类文档作训练集和测试集，本文最终所用语料集为非平衡语

1) http://www.nlpir.org/?action-viewnews-itemid-103.

料集，选取情况如表 2 所列。

表 2 测试集和训练集文档的选取情况

语料类	运动	农业	航空	政治	医药	经济
训练集	1000	1000	500	500	50	50
测试集	1000	100	500	50	50	500

4.2 实验设置

本文实验从以下几个方面考虑：

1)干扰性检验实验：检验 MapReduce 机制对多级特征选择模型的特征选择向量维度是否有干扰或者不良影响。具体实验为：将多级特征选择模型分别运行在普通 PC 机和 Hadoop 单节点上，对比初级特征选择和二级特征选择文本向量的维度。

2)性能对比实验：检验多级特征选择模型的性能。具体实验为：对比文献[5]提出的 IGS-GA 方法、文献[7]提出的 TFIDF-MRMR 方法以及本文提出的多级特征选择方法(NCHI-MI)的分类效果。

3)效率检验实验：检验本文提出的多级特征选择模型在 Hadoop 平台上的执行效率。具体实验为：对比 IGS-GA 方法、TFIDF-MRMR 方法以及本文提出的 NCHI-MI 方法的执行时间，另外还将观察 NCHI-MI 方法在 Hadoop 平台上不同节点数时的执行加速比。

实验所用机器均按以下配置搭建环境：CPU 为 Intel Core i5-65003.20 GHz，8 GB 内存，2 TB 硬盘，操作系统为 Ubuntu 14.04，Hadoop 版本为 1.2.1，Java 版本为 1.7.0。实验中 KNN 分类器中的 K 取值为 10。

4.3 评价指标

1)准确率，用来反映分类结果的准确性，记作 P。计算公式如式(7)所示：

$$P=\frac{\text{类别中正确分类的文本数}}{\text{分为该类的文档总数}} \tag{7}$$

2)召回率，用来反映能正确分类的能力，记作 R。计算公式如式(8)所示：

$$R=\frac{\text{类别中正确分类的文本数}}{\text{本应分到该类的文本总数}} \tag{8}$$

3)$F1$ 值，是基于查准率和召回率的综合评价指标，反映了系统的综合性能。计算公式如式(9)所示：

$$F1=\frac{2PR}{P+R} \tag{9}$$

4)加速比，用来衡量任务并行处理的性能[15]，记作 T_s。计算公式如式(10)所示：

$$T_s=\frac{\text{单处理器下的执行时间}}{\text{多节点并行执行时间}} \tag{10}$$

4.4 结果分析

1)干扰性检验实验的结果如表 3 所列。

表 3 PC 单机和 Hadoop 单节点在各阶段取词的数量

序号	类别	PC 单机		Hadoop 单节点	
		初级特征选择文本向量维度	二级特征选择文本向量维度	初级特征选择文本向量维度	二级特征选择文本向量维度
1	运动	863	698	851	676
2	农业	719	565	739	591
3	航空	724	464	716	450
4	政治	688	571	680	563
5	医药	535	460	522	462
6	经济	618	463	626	475

由表 3 中的实验数据可看出：初级特征选择向量维度和二级特征选择向量维度在 PC 单机和 Hadoop 单节点上的维数差距不大，在误差范围内和执行差异可接受范围内。综上所述，MapReduce 技术的引入对多级特征选择模型没有产生明显干扰或不良影响。

2)表 4－表 6 分别记录了 3 种方法下，各个类别的准确率、召回率、$F1$ 值数据。为了更直观地显示此实验结果，图 6－图 8 以直方图形式显示了表 4－表 6 的实验结果。

表 4 准确率对比

(单位：%)

序号	类别	IGS-GA	TFIDF-MRMR	NCHI-MI
1	运动	82.73	90.92	91.07
2	农业	81.65	90.88	90.51
3	航空	79.31	86.61	87.23
4	政治	78.68	85.95	86.73
5	医药	50.26	58.30	58.62
6	经济	40.20	43.71	45.28

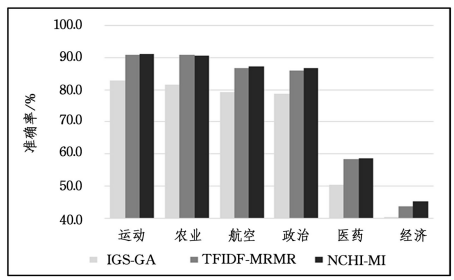

图 6 准确率对比直方图

表 5 召回率对比

(单位：%)

序号	类别	IGS-GA	TFIDF-MRMR	NCHI-MI
1	运动	86.20	93.20	93.80
2	农业	81.00	91.00	91.00
3	航空	82.20	88.60	89.20
4	政治	80.00	86.00	86.00
5	医药	54.00	68.00	70.00
6	经济	42.80	46.60	48.80

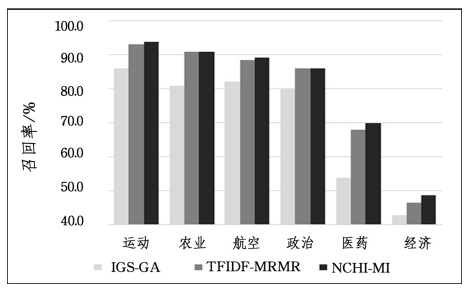

图 7 召回率对比直方图

表 6 $F1$ 对比

(单位：%)

序号	类别	IGS-GA	TFIDF-MRMR	NCHI-MI
1	运动	84.00	92.04	92.41
2	农业	81.32	90.93	90.75
3	航空	80.72	87.59	88.20
4	政治	79.33	85.97	86.36
5	医药	52.06	62.77	63.80
6	经济	41.45	45.10	46.97

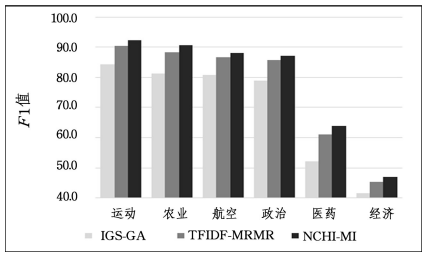

图 8　$F1$ 值对比直方图

由表 4－表 6、图 6－图 8 可以得出以下结论：

①总体上看，对于 3 种方法，训练文本集越多，分类准确度越高。由于语料集的不平衡性，“经济”和“医药”的训练文本集较少，因此 3 种方法均无法较好地提取足够多的特征向量作为训练库，导致分类精度较差；当语料集相对充足时，3 种方法都可以选择到足够的特征词，各项指标都有大幅度提升；当语料集足够多后，3 种方法的各个指标也有一定提升，但是提升幅度不大，趋于平稳，这说明增加语料集的规模已经不是决定指标提升的主要因素，需改进算法或提升硬件水平。

②3 种方法的平均召回率略高于平均准确率。正确分类的能力是算法分类的前提，也是提升分类精确度的前提，3 种方法都能较好地区分正负文本集，但是分类精度略微偏低，表明 3 种方法正确分类的能力要优于分类的精确度。

③NCHI-MI 在准确率、召回率和 $F1$ 值上比 TFIDF-MRMR 略有提升，但相差不大，且都高于 IGS-GA。IGS-GA 的分类效果已经相当出色，但其强过滤性导致会丢失一部分原始语意，因此各个指标相对 NCHI-MI 和 TFIDF-MRMR 较低，这也是 IGS-GA 的瓶颈。NCHI-MI 和 TFIDF-MRMR 针对 IGS-GA 丢失原始语意做出改进，使用相对平和的选择方法，然后再进行冗余过滤和去噪等优化处理。实验数据表明各个指标均有明显提升，但 NCHI-MI 有更好的优化效果。

3）表 7 记录了 NCHI-MI 和 TFIDF-MRMR 在一台 PC 机上的运行时间。实验数据显示，在普通 PC 机上，NCHI-MI 的执行时间比 TFIDF-MRMR 多了大约 18s，TFIDF-MRMR 考虑到 MRMR 计算比较耗时，利用增量式搜索获取合适的特征词，这样在一定程度上减少了时间消耗，但是加速效果并不乐观。正是如此，本文将 NCHI-MI 置于 Hadoop 平台上进行分布式并行处理，从而大幅度减少了耗时。

表 7　运行时间的对比

序号	类别	运行时间/ms
1	TFIDF-MRMR	179236
2	NCHI-MI	197054

表 8 记录了 NCHI-MI 工作在不同节点数的 Hadoop 集群上的运行时间，节点数分别为 1，3，6，9，12，15。数据显示，随着 Hadoop 节点数的增加，系统运行所时间不断减少，执行速度提升显著。Hadoop 集群节点为 15 台时，用时约27 s，在 PC 机上用时约 197 s，执行效率提升 6 倍左右。另外，结合表 7 可以发现，NCHI-MI 在一台 PC 机上的执行时间比在 Hadoop 单节点上的执行时间短。分析原因发现，虽然 Hadoop 节点只有一个，但系统仍需要对输入的语料集分块，同时要经过中间过程处理，这些操作都将增加运行时间。

表 8　不同节点数的运行时间

序号	机器数/台	运行时间/ms
1	1	201969
2	3	114755
3	6	66877
4	9	40967
5	12	30235
6	15	27110

为了更直观地显示加速比趋势，图 9 是根据表 8 中运行时间计算得到的加速比折线图。数据显示，随着 Hadoop 节点数的增加，系统加速比呈攀升趋势。节点数为 3，6，9 时，加速比增长速度较快；节点数达到 15 时，加速比增长趋于缓慢，此时节点数即将趋于饱和。当节点数过多时，系统处理任务的时间很短，主要耗时在数据分配、均衡负载等方面。

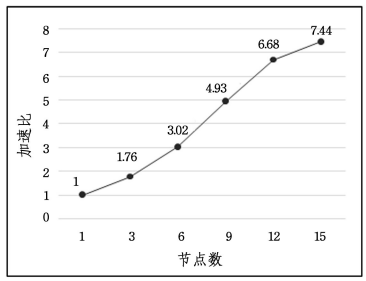

图 9　不同节点数的加速比

结束语　本文提出一种基于 MapReduce 的多级特征选择模型，使用改进的 CHI 方法解决忽略特征词频数的问题，将其作为初级特征选择；再利用互信息方法过滤初级选择产生的噪声词，并将合适的特征词前置作为二级特征选择；最后利用 MapReduce 技术处理大数据的性能，将此两级模型载入 Hadoop 平台上。实验结果表明，本机制能显著提高文本分类精度，同时提升分类处理效率。

值得注意的是，本文提出的机制仍有很多有待改进的细节。样本较少时，分类准确率过低，二级选择互信息方法仍可进行优化，这些都将是未来的重点研究方向。另外，本文提出的“多级特征选择模型＋MapReduce 框架”是一类可套用的基本机制，推广到其他算法是否有更好的表现，也值得深入研究。

参 考 文 献

[1] DASH M，LIU H. Feature selection for classification[J]. Intelligent Data Analysis，1997，1(3)：131-156.

[2] DEAN J，GHEMAWAT S. MapReduce：simplified data processing on large clusters[C]// Conference on Symposium on Opearting Systems Design & Implementation. USENIX Association，2004：10.

[3] MENG J N，LIN H F，YU Y H. A two-stage feature selection method for text categorization [C]// 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD). IEEE，2010：1492-1496.

[4] KURSAT U A，SERKAN G. A novel probabilistic feature selection method for text classification [J]. Knowledge-Based Systems，2012，36(6)：226-235.

[5] HARUN U. A two-stage feature selection method for text categorization by using information gain，principal component analysis and genetic algorithm [J]. Knowledge-Based Systems，2011，24(7)：1024-1032.

（下转第 479 页）

[6] JAINA K. Data Clustering：50 Years Beyond K-means[J]. Pattern Recognition Letters，2010，31(8)：651-666.

[7] BOCK H H. Clustering Methods：A History of K-means Algorithms[M]// Selected Contributions in Data Analysis and Classification. Springer Berlin Heidelberg，2007：161-172.

[8] 王千，王成，冯振元，等. K-means 聚类算法研究综述[J]. 电子设计工程，2012，20(7)：21-24.

[9] PERONAP，FREEMAN W. A Factorization Approach to Grouping [C]// Proceedings of European Conference on Computer Vision (ECCV). 1998：655-670.

[10] NGA Y，JORDANM I，WEISS Y. On Spectral Clustering：Analysis and an Algorithm[C]// Proceedings of the 14th Advances in Neural Information Processing System. 2002：849-856.

[11] LI C G，YOU C，VIDAL R. Structured Sparse Subspace Clustering：A Joint Affinity Learning and Subspace Clustering Framework[J]. IEEE Transactions on Image Processing，2017，26(6)：2988-3001.

[12] LI C G，YOU C，VIDAL R. On Geometric Analysis of Affine Sparse Subspace Clustering[J]. IEEE Journal on Selected Topics in Signal Processing，2018，12(6).

[13] LI C G，ZHANG J J，GUO J. Constrained Sparse Subspace Clustering with Side Information[C]// Proceedings of the 24th International Conference on Pattern Recognition (ICPR). 2018：2093-2099.

[14] LIANG J Q，HAN Y H，HU Q H. Semi-Supervised Image Clustering with Multimodal Information[J]. Multimedia Systems，2016，22：149-160.

[15] LUXBURGU V. A Tutorial on Spectral Clustering[J]. Statistics & Computing，2007，17(4)：395-416.

[16] 金建国. 聚类方法综述[J]. 计算机科学，2014，41(S2)：288-293.

[17] ZELNIK-MANORL，PERONA P. Self-Tuning Spectral Clustering[C]// Proceedings of the 16th Advances in Neural Information Processing System. 2004：1601-1608.

[18] WANG F，ZHANG C S. Robust Self-Tuning Semi-Supervised Learning[J]. Neurocomputing，2007，70(16)：2931-2939.

[19] YANG C，ZHANG X，JIAO L，et al. Self-Tuning Semi-Supervised Spectral Clustering [C] // International Conference on Computational Intelligence & Security. 2008：1-5.

[20] KUMARV，HAHNJ，ZOUBIR A M. Band Selection for Hyperspectral Images Based on Self-Tuning Spectral Clustering[C]// Proceedings of the 21st EuropeanSignal Processing Conference (EUSIPCO). 2013.

[21] POLITOM，PERONAP. Grouping and Dimensionality Reduction by Locally Linear Embedding[C]// Proceedings of International Conference on Neural Information Processing Systems：Natural & Synthetic. 2001：1255-1262.

[22] ZENGC P，ZHOUA M，ZHANGG X. Self-adaptive Spectral Cluster Number Detecting with Particle Swarm Optimization Algorithm[C]// Evolutionary Computation. IEEE，2016：4607-4611.

[23] CHAN P，SCHLAG M，ZIEN S. Spectral K-Way Ratio-Cut Partitioning and Clustering[C]// Proceedings of Conference on Design Automation. 1993：749-754.

[24] TENENBAUM J B，SILVAV D，LANGFORD J C. A Global Geometric Framework for Nonlinear Dimensionality Reduction [J]. Science，2000，290(5500)：2319-2323.

[25] RODRIGUEZ A，LAIO A. Clustering by Fast Search and Find of Density Peaks[J]. Science，2014，344(6191)：1492-1496.

[26] 李金泽，徐喜荣，潘子琦，等. 改进的自适应谱聚类 NJW 算法[J]. 计算机科学，2017，44(S1)：424-427.

[27] 周海松，黄德才. 密度自适应的半监督谱聚类算法[J]. 计算机科学，2016，43(12)：209-212.

[28] 杨虎，付宇，范丹. 噪音特征对聚类内部有效性的影响[J]. 计算机科学，2018，45(7)：22-30.

(上接第 473 页)

[6] KASHIF J，SAMMEN M，BABRI HAROON A. A two-stage Markov blanket based feature selection algorithm for text classification [J]. Neurocomputing，2015，157：91-104.

[7] 李军怀，付静飞，蒋文杰，等. 基于 MRMR 的文本分类特征选择方法[J]. 计算机科学，2016，43(10)：225-228.

[8] 黄源，李茂，吕建成，等. 一种基于开方检验的特征选择方法[J]. 计算机科学，2015，42(5)：54-56.

[9] ZHENG Z，LEI W，HUAN L，et al. On Similarity Preserving Feature Selection[J]. IEEE Transactions on Knowledge & Data Engineering，2013，25(99)：1.

[10] DEAN J，GHEMAWAT S. MapReduce：simplified data processing on large clusters[J]. Communications of the ACM，2008，51(1)：107-113.

[11] MASHAYEKHY L，NEJAD M，GROSU D，et al. EnergyAware-Scheduling of MapReduce Jobs[J]. IEEE International Congress on Big Data，2014，26(10)：32-39.

[12] HAN L，SUN X Z，WU Z C，et al. Optimization Study on Sample Based Partition on MapReduce[J]. Journal of Computer Research and Development，2013，50(Suppl.)：77-84.

[13] GUNTHER N，PUGLIA P，TOMASETTE K. Hadoop Superlinear Scalability[J]. Communications of the ACM，2015，58(4)：1542-7730.

[14] FEI X，LI X F，SHEN C. Parallelized text classification algorithm for processing large scale TCM clinical data with MapReduce[C]// IEEE International Conference on Information and Automation. IEEE，2015：1983-1986.

[15] LIU J，ZHU A，QIN C. Estimation of theoretical maximum speedup ratio for parallel computing of grid-based distributed hydrological models [J]. Computers & Geosciences，2013，60(10)：58-62.