

采用 SVM 方法的文本情感极性分类研究

陈培文, 傅秀芬

(广东工业大学 计算机学院, 广东 广州 510006)

摘要: 文本情感极性分类是文本情感分析首先要解决的关键问题. 在分析影响文本情感分类的各类因素的基础上, 首先构建了情感词典, 并进行情感特征选取以及情感特征加权, 然后使用 SVM 分类的方法对文本进行情感识别及分类, 最后在语料数据集的基础上, 在单机平台上和 Spark 分布式计算平台上执行分类模型, 对比分析其分类精度和时间代价. 实验结果验证了本文构建的情感极性分类模型在单机和分布式云平台上中的有效性.

关键词: 情感分类; 支持向量机; Spark 分布式计算平台

中图分类号: TP309

文献标志码: A

文章编号: 1007-7162(2014)03-0095-07

Research on Sentiment Classification of Texts Based on SVM

Chen Pei-wen, Fu Xiu-fen

(School of Computers, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: The key problem to solve in a sentiment analysis of texts is the sentiment polarity classification. Based on the analysis of various factors affecting sentiment classification of texts, it built the sentiment lexicon, extracted affective characteristics, and weighted sentimental features. Then, it used support vector machine (SVM) classifier for emotion recognition and text classification. Finally, it performed the classification model with the corpus data sets on the single platform and the Spark distributed computing platform to analyze its classification accuracy and time cost. The experimental results verify the effectiveness of the text sentimental polarity categorization model on the single platform and on the spark distributed computing platform.

Key words: sentiment classification; support vector machine; Spark distributed computing platform

文本情感分析, 又称为倾向性分析、意见挖掘、情感分类等. 简言之, 是对带有情感色彩的主观性文本进行分析、处理、归纳和推理的过程^[1]. 开放性极强的互联网使得越来越多的用户愿意在各种社交网络分享评价平台上发表自己的真实想法, 分享个人的内心世界, 对某些人、事、物的主客观评价, 这也带来了各类情感信息的融合和汇聚. 这类情感信息的迅速膨胀使得文本情感分析成为当前自然语言处理领域的研究热点之一.

文本情感分析研究是多个领域交叉的结果, 涉及到的研究领域较为广泛, 包括语言学、信息检索、数据挖掘、机器学习和人工智能等多个领域. 因此, 正因为文本情感分析的多领域融合性以及情感信息的价值可挖掘性, 使得国内外学者对其做了许多的

研究, 并取得一定的成果, 大致分为两个方向.

1) 使用情感词典及规则

Turney 等^[2] 针对情感词的不足, 使用 PMI 方法扩展基准的情感词进行情感分析; Narayanan 等^[3] 基于时态信息并结合各种特征表示提出多种分类方式; 刘鸿宇^[4] 等通过归整相应的分析规则对抽取的评价对象进行情感分析; 李寿山等^[5] 利用标签传播算法(LP) 构建覆盖领域语境的中文情感词典用于文本情感分析; 吴江等^[6] 结合语义规则并利用提出的情感分析算法分析金融文本数据.

2) 使用监督和半监督的机器学习方法

Pang 等^[7] 采用对比实验比较 3 种机器学习分类方法: 朴素贝叶斯、最大熵、SVM 的在评论语料数

收稿日期: 2014-04-16

基金项目: 广东省自然科学基金资助项目(9151009001000007); 广东省科技计划项目(2012B091000173)

作者简介: 陈培文(1990-), 男, 硕士研究生, 主要研究方向为文本情感分析和数据挖掘.

据集上的情感倾向性分类精度. 唐慧丰等^[8]利用不同的特征选择方法组合多种机器学习方法做对比实验验证情感分类的精度; 杨经等^[9]通过提取分析情感词的相关特征, 使用 SVM 机器学习的分类方法对句子进行情感识别及分类; 谢丽星等^[10]运用 SVM 机器学习的方法进行多个层次和多种策略的实验分析比较其分类效果; 李素科等^[11]针对监督学习分类的不足之处, 对情感特征进行聚类, 并提出了一种半监督式的情感分析算法.

总的来看, 使用情感词典及规则进行文本情感分析, 其优点是粒度细, 对分析显式的、含有情感词的文本效果较佳, 但该方法易丢失数据集中隐藏的重要模式. 而基于机器学习方法的情感分析克服了情感词稀疏的影响, 但对训练语料的规模依赖性较高. 本文结合情感词典和关联信息以及机器学习的方法, 在分析影响情感分类的情感因素的基础上, 首先构建了自定义情感词典, 并使用卡方统计量选择情感特征, 利用改进的 TFIDF 和情感极值进行情感特征加权. 最后使用 SVM 的监督学习分类方法对文本进行情感分析, 在单机上对比验证不同的加权算法对分类精度的影响, 并在单机平台上和 Spark 分布式计算平台上分别执行情感分类模型, 对比分析其分类精度和时间代价.

1 文本情感特征

本文基于 SVM 的文本情感极性分类任务分为 3 个步骤, (1) 情感词典构建: 构建一个较完备的能识别情感特征词的情感词典; (2) 情感特征选择及特征向量加权: 从文本中提取情感特征组成特征向量, 计算特征向量中情感特征的文本情感权重; (3) 情感分类: 针对步骤 (2) 的加权后的情感特征向量, 用 SVM 的机器学习方法把文本分成正向情感极性和负向情感极性两类.

1.1 情感词典的构建

情感分析需要识别情感词特征, 但由于目前中文情感分析领域的研究并没有一个完备的情感词词典. 因此, 本文通过以下几个方面完成情感词典的构建:

(1) 利用《汉语褒贬义词语用法词典》, 该词典共收词条 1 015 个, 对其词语进行情感极性分类扩展情感词典.

(2) 利用知网发布的极性情感词词典, 台湾大学整理并发布的中文通用情感词典 (NTUSD) 以及

大连理工大学信息检索研究室的情感词汇本体词典^[12], 对其中的词语通过人工选择进行极性分类和整理, 并利用《同义词词林》对情感词进行扩展.

(3) 利用基准情感词, 通过大量的语料集对未知情感词采用 PMI 算法进行情感极性分类扩展情感词典, 实验采用人工选取的方式选择情感语义非常明显的基准情感词, 并利用式 (1) 设定阈值进行计算归类.

$$S(w) = \frac{\sum_{i=1}^m \log_2 \frac{p(w, x_i)}{p(w)p(x_i)}}{m} - \frac{\sum_{j=1}^n \log_2 \frac{p(w, y_j)}{p(w)p(y_j)}}{n}, \quad (1)$$

其中, m, n 是正向和负向基准情感词的个数, $p(w)$ 是待识别情感词出现的概率, $p(x_i), p(y_i)$ 分别是正向基准情感词和负向基准情感词出现的概率, $p(w, x_i)$ 是待识别情感词和正向基准情感词共同出现的概率, $p(w, y_i)$ 是待识别情感词和负向基准情感词共同出现的概率.

(4) 在情感词汇本体词典极值标注的基础上, 人工标记未赋予极值的情感词语, 这种做法带来了一定的误差.

1.2 情感特征选择

情感分析特征选取, 本文结合构建的情感词典利用卡方统计量选择与情感特征相关的词语, 剔除与情感特征无关的词语.

1.2.1 卡方统计量技术

本文选择卡方统计量 (chi-square statistic, CHI) 技术来建立分类模型的情感特征词向量, 计算公式如式 (2) 和式 (3) 所示.

$$\chi^2(w, s) = \frac{N [p(s, w)p(\bar{s}, \bar{w}) - p(\bar{s}, w)p(s, \bar{w})]^2}{f(s, w)}, \quad (2)$$

$$f(s, w) = [p(s, w) + p(s, \bar{w})][p(\bar{s}, w) + p(\bar{s}, \bar{w})] \times [p(s, w) + p(\bar{s}, w)][p(\bar{s}, w) + p(s, \bar{w})]. \quad (3)$$

其中, $\chi^2(w, s)$ 表示 s 情感类别中的词 w 的卡方计量, N 表示情感训练数据集的规模, $p(s, w)$ 表示在情感类别 s 中包含词 w 的文档规模, $p(\bar{s}, w)$ 表示排除情感类别 s , 其他情感类别包含词 w 的文档规模, $p(s, \bar{w})$ 表示在情感类别 s 中不包含词 w 的文档规模, $p(\bar{s}, \bar{w})$ 表示在情感类别 s 中, 不包含词 w 也不在该类别中的文档规模.

利用式 (2) 和式 (3), 本文模型可以统计出某一情感类别每个词的卡方统计量, 对每个类别设定一个情感特征量阈值, 使特征向量更具区分度, 并合并情感类别中的情感特征词组合作为本文选择的情感特征向量.

1.2.2 情感特征选择

情感特征的选择除了词本身的词特征之外,还需包括情感词的情感特征,单句的情感特征以及句间情感特征. 本文模型选取的情感特征如表 1 所示.

表 1 情感特征列表

Tab.1 List of sentiment features

特征类别	序号	特征描述
单句情感特征	1	情感词左词性
	2	情感词右词性
	3	关联词类型
	4	感叹号个数
	5	是否包含省略号
	6	是否包含问号
整句情感特征	7	程度级别词个数
	8	否定词个数
	9	动词个数
	10	形容词个数
	11	正向情感词个数
	12	负向情感词个数
	13	单句的数目
	14	单句情感正极值均值
	15	单句情感负极值均值
	16	首句情感极值
	17	尾句情感极值
	18	整句情感极值

2 文本情感极性分类

2.1 情感特征加权

2.1.1 单句的情感极值处理

本文以逗号为分隔对单句进行情感分析以及情感极值加权,分为以下几种情形:

(1) 若出现情感词,比如“开心”,“失望”,“搞笑”等,本文从情感极值表中找出相应的极值,对于正向情感词取正极值,负向情感词取负极值,而对于情感极性不明确的,则取其所有情感极值的期望值.

(2) 若出现否定词,比如“不”,“未”等,模型视它的出现是对后续情感极性的一种否定,对于否定词之后紧跟着情感词的情况,对情感极值取反并减少情感词对应的情感数目增量,增加反向的情感数目增量.

(3) 若出现反问副词,比如“凭什么”,“就算”,“怎么能”等,模型视它的出现是为了加强负向的情感极值,对于这样的情况,提高负向情感极值.

(4) 若出现一个感叹号或多个感叹号,模型视

它的出现是为了加强相应的情感极值,因此分析这样的句子,提高句子中的相应的情感极值,提高幅度视感叹号出现的连续程度而定.

(5) 若出现程度级别词,比如“极其”,“略为”“有些”等,模型视它的出现是为了增强或减弱情感极值,对于程度词修饰情感词的情况,视程度词级别增强或减弱情感极值.

(6) 若句子是首句或尾句,且出现人称代词或主张词语,比如“认为”“发现”等,模型视其是影响整体情感极值的重要因子,对于这种句子,加大其相应的情感极值.

因此,计算单句情感极值的公式为

$$\bar{E}(w) = \frac{f(p) \left(\sum_{i=1}^{n_p} \alpha E(w_i) \right) n_p + f(n) \left(\sum_{j=1}^{n_n} \beta E(w_j) \right) n_n}{n_p + n_n + 1}, \quad (4)$$

其中, $f(p)$ 是正向情感倾向值,本实验设置 $f(p) = 1$, $f(n)$ 是正向情感倾向值,设置 $f(n) = -1$, α 是正向影响因子, β 是负向影响因子,设置取值范围 $[0, 2]$, 两者的值再由试验统计分析得出, n_p 和 n_n 分别为正向和负向情感词的个数, $E(w_i)$, $E(w_j)$ 分别是正向和负向情感词极值的期望,计算公式为

$$E(w) = \sum_{k=1}^m w_k p_k, \quad (5)$$

其中, m 为情感词的极值个数, w_k 为情感词的极值, p_k 为其出现的概率,本实验中若出现一个词有多个极值的情况,取辅助情感值的 p_k 为 0.4.

2.1.2 整句的情感极值处理

本文以句号或感叹号为分割点对整句进行情感分析以及情感极值加权,分为以下几种情形:

(1) 若出现递进关系连词,比如“不但,而且”等,模型视第二复句比第一复句所表达的情感要更为强烈,因此在分析这种句式的复句时,提高第二复句中情感词的情感极值并在原来的基础上增加一个对应的情感数目增量.

(2) 若出现转折关系连词,比如“然而”,“但是”等,模型视转折连词前后所要表达的情感极性是相反的,连词之后的语句所表达的情感才是真正想要表达的情感. 因此在分析这种句式的复句时,提高连词之后的复句的情感极值并增加两个对应的情感数目增量,降低连词之前复句的情感极值并减少两个对应的情感数目增量.

(3) 若出现因果关系连词,比如“由于”,“于是”,“以致”“因此”等,模型视连词出现的复句才是重点要表达的情感极性,连词之前出现的情感只是

铺垫,因此在处理这类句式时,提高连词复句的情感极值并增加一个对应的情感数目增量。

(4) 若出现让步连词,比如“即使”,“尽管”,“就算”等,模型视让步连词出现的复句所表达的情感极性应该是为了增强后一个复句的情感极性,因此对于整体的情感极性来说,在分析这种句式的复句时,降低让步连词复句的情感极性并减少一个对应的情感数目增量。

计算文档整句情感极值的公式为

$$E(d_i, w) = \sum_{i=1}^C \delta E(w), \quad (6)$$

其中 $E(d_i, w)$ 为某文档的情感极值, $E(w)$ 为单句的情感极值, δ 为连词影响因子。

2.1.3 改进的情感 TFIDF 函数

TFIDF 函数常用于特征加权,它表征一个特征的重要程度。但是对于情感分类而言,情感词在某个类别中在多个文本中多次出现,本文视该情感词对于表征该类别越“重要”,而不是 TFIDF 函数所表征的越“不重要”,因此改进公式为

$$w(d, t) = \frac{tf(d, t) \times (p(t)/N)}{\sqrt{\sum_{i \in d} [tf(d, t) \times (p(t)/N)]^2}}, \quad (7)$$

其中, $w(d, t)$ 表示在文档 d 中特征项 t 的影响程度表征, $tf(d, t)$ 为特征项 t 在文档 d 中的频率, N 为情感训练数据集的规模, $p(t)$ 为情感训练数据集中出现特征项 t 的频率,分母为归一化因子。

在计算情感特征情感值时本文结合情感极值和改进的 TFIDF 公式进行计算,并对效果进行对比实验。

2.2 情感极性分类

支持向量机(Support Vector Machine, SVM)是一种常用的用于二分类的监督式学习的方法。在自然语言处理领域中, SVM 广泛应用于词义消歧、文本自动分类、信息过滤等方面。文献[7-10]的实验结果均表明,采用 SVM 的方法能够取得更好的分类效果。文献[7]进行对比实验表明支持向量机 SVM 的分类方法效果最佳,分类精确度最高达到 83%;文献[8]表明在训练集规模较大的情况下,使用支持向量机分类方法明显优于其他分类方法。因此论文采用支持向量机 SVM 融合情感特征向量对文本进行情感极性判别。

2.3 Spark 分布式计算平台

Spark^[3-14]是 UC Berkeley AMP lab 所开源的集群计算平台,与 Hadoop 有相似之处,但它是立足于内存计算的一个新的集群计算框架,具体见图 1。Hadoop 的数据共享相比 Spark 慢,因为前者需要额外的复制、序列化和磁盘 IO 带来的开销,而后者是基于内存计算且作业是基于构建的 Stage 有向无环图(DAG)执行的。由于 Spark 的集群架构设计和内核处理模式,使得其更适合进行大数据挖掘或者机器学习处理等需多次迭代的领域,因为 Spark 设计中融入了 RDD 这样的抽象概念。

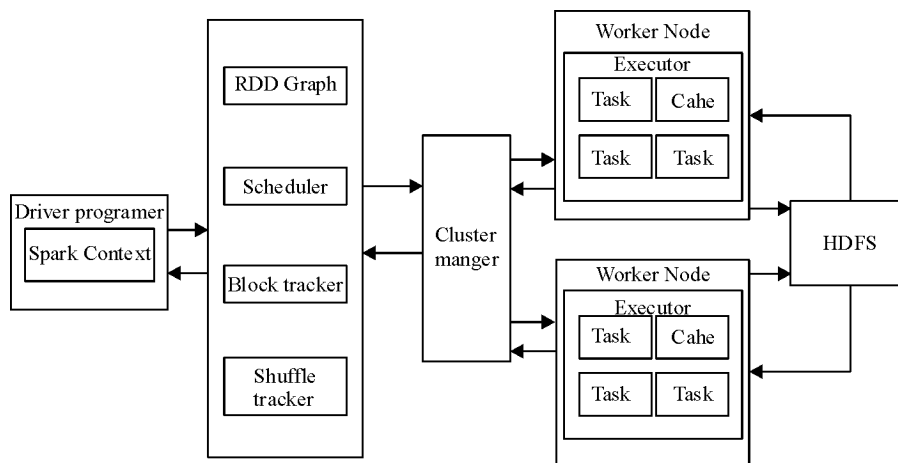


图1 Spark 分布式集群架构

Fig. 1 Diagram of Spark distributed cluster

RDD (Resilient Distributed Dataset) 是 Spark 的核心概念,指一个只读的、可分区的弹性分布式数据集。Spark 默认的缓存策略是将 RDD 以反序列化的方式缓存于内存中, RDD 的 Lineage 会记录每一个 RDD 的父依赖,也就是 Transformation 操作行为。当

这个 RDD 的部分分区数据丢失时,由于 Spark 的幂等特性以及函数式语义的设计,它可以通过 Lineage 获取依赖信息去重新运算并恢复丢失的数据分区。Spark 的这种粗粒度数据并行的计算范式,带来了性能的提升并保证了数据的鲁棒性。

本文中的实验采用 Spark 分布式计算平台执行情感分类模型并与单机平台进行对比实验。

3 实验结果与分析

3.1 情感训练语料准备

本文使用已经标注好的情感语料数据集^[3],数据集为 ChnSentiCorp_Htl_del_4000(酒店领域), ChnSentiCorp-Book-del-4000(书籍领域)和 ChnSentiCorp-NB-del-4000(笔记本领域)。

实验采用 ICTCLAS(<http://ictclas.nlpir.org/>)对文本内容进行分词并保留词性和位置。一般的文本数据预处理工作会过滤无意义的停用词,例如标点符号、代词、语气词、程度副词、连词等,但对于文本情感分析,这些词语恰恰是本文模型所需要的,所以模型会保留这些词语并自定义过滤器。

3.2 文本情感语料处理

在 Spark 分布式计算平台上进行语料预处理和生成特征向量文件的步骤大致如下:

从 HDFS 中读入原始文件,经过一系列 Transformation 操作,最后利用 Action 操作生成特征向量文件送入到 MLlib 库执行分类算法。

(1) 从 HDFS 中读入情感词典,经过 $\text{map}(f: T \Rightarrow U)$ 算子经过 $\text{reduceByKey}(f: (V, V) \Rightarrow V)$ 生成情感词典极值元组,构建 $\text{SentimentDictRDD}(RDD[K, V])$ 。

(2) 从 HDFS 中读入待分类的原始训练文件,经过 $\text{map}(f: T \Rightarrow U)$ 算子利用分词器对文件进行分词以及标记类别和文本 ID,利用 $\text{flatMap}(f: T \Rightarrow Seq(U))$ 算子和 $\text{map}(f: T \Rightarrow U)$ 算子生成文本分词元组并记录分词信息构建 $\text{WordRDD}(RDD[K, V])$,再经过 $\text{filter}()$ 算子利用自定义过滤器链过滤分词,通过 $\text{cache}()$ 算子缓存该 RDD。

(3) 在缓存 RDD 的基础上,一方面经过 $\text{reduceByKey}()$ 算子构建文本词表 $\text{WordsTermRDD}(RDD[K, Seq(V)])$,另一方面经过 $\text{map}(f: T \Rightarrow U)$ 算子和 $\text{groupByKey}()$ 算子构建倒排表 $\text{InvertedTableRDD}(RDD[K, Seq(V)])$ 。

(4) 在倒排表 RDD 的基础上,经过 $\text{map}(f: T \Rightarrow U)$ 算子计算卡方统计量以及 $\text{sortByKey}(c: Comparator[K])$ 算子并保留文本中所有的情感特征词,通过 $\text{union}()$ 算子合并所有的情感特征构建情感特征向量 $\text{SentimentFeatureRDD}(RDD[I])$ 。

(5) 利用词表 WordRDD 结合情感特征向量 $\text{SentimentFeatureRDD}$,根据单句情感极值和整句情感极值公式和 $\text{SentimentDictRDD}(RDD[K, V])$ 中的情感极值对特征向量进行加权或者通过改进的 TFIDF 对向量进行加权。

(6) 在上面的基础上,生成符合 Spark Machine Learning Library 线性支持向量机 SVMWithSGD 的训练特征向量格式的 $\text{OutputSVMVectorRDD}$,并存于 HDFS 中。

其中步骤(3)的 Scala 伪代码如下:

```
// Load graph as an RDD of ( doc, sentimentterms )
pairs
val terms = spark.textFile(...).map(...).persist()
// for ( i <= 1 to ITERATIONS ) {
// Build an RDD of ( term, ( doc, freq ) ) pairs
val terms = terms.flatMap{
  ( doc, sentimentterms ) =>
    sentimentterms.map( term => ( termdoc, 1 ) )
}.reduceByKey( _ + _ )
. map ( ( termdoc, freq ) => ( termdoc, split
( ... ), freq )
. map( x => ( x._1( ... ), x._1( ... ), freq ) )
// Build an RDD of ( term, docs ) pairs
val inverted = terms.flatMap {
  ( doc, sentimentterms ) =>
    sentimentterms.map( term => ( term, doc ) )
}.groupByKey()
}
```

3.3 预测与评估

实验环境具有 3 个节点,1 个 master 和 2 个 slave(worker)。集群中的单点配置如下: 2 核 Intel (R) core{ TM } i3 - 2350M cpu @ 2.30GHz CPU 处理器,4G 内存,操作系统为 Ubuntu 14.04, Hadoop 版本为 1.1.2, Spark 版本为 0.9.0,采用 Standalone 的集群架构模式。

单机上的 SVM 分类器,本文采用开源的工具 LibSVM, Spark 分布式计算平台上的 SVM 分类器,本文采用 Spark Machine Learning Library 的 SVMWithSGD 分类器。

对于已标注好的情感语料数据集,对于每一个情感类别,拆分数据为训练数据和测试数据,拆分比

例为 9:1,采用 10 折交叉验证进行实验,取 SVM 分类器最后的分类精度平均值作为评价标准,并采用不同的算法进行对比实验. 分类精度计算公式为

$$\text{Accuracy} = \frac{p(1|1) + p(2|2)}{N}, \quad (8)$$

其中, $p(1|1)$ 表示测试集中属于正向并被正确分类为正向的实例个数, $p(2|2)$ 表示测试集中属于负向并被正确分类为负向的实例个数, N 是测试集的规模.

实验结果如表 2 所示,表 2 的语料 1、2、3 分别表示酒店领域语料集、书籍领域语料集以及笔记本领域语料集,算法 A 未采用情感特征的卡方统计量和改进的 TFIDF,算法 B 采用了情感特征的卡方统计量的 CHI 值组合情感极值,算法 C 采用改进的 TFIDF 组合情感特征的情感极值,算法 D 组合了算法 B 和 C 进行实验. 从实验结果可以看出,算法 D 的分类精度较其他算法有了明显的提升,说明本实验通过改进的 TFIDF 和卡方统计量组合情感极值的算法能有效地提高分类精度,验证了本文情感分类算法的有效性.

表 2 单机平台情感分类算法实验对比结果

Tab. 2 Comparison of experimental results by the sentiment classification algorithm on the single platform %

语料	算法			
	A	B	C	D
1	76.5	77.0	78.89	80.53
2	88.33	89.20	89.22	91.25
3	73.5	80.89	79.0	81.56

表 3 是在单机上和 Spark 平台上进行情感分类实验,采用分类精度最佳的算法 D,对比不同平台上的分类精度和训练以及测试阶段的时间代价(ms).

从表 3 可以看出,采用 Spark 分布式计算平台进行分类预测的分类精度较单机上的分类预测精度有所下降;时间消耗上,在单机平台执行所消耗的时间比 Spark 分布式计算平台多,但由于目前使用的语料集规模太小,该特征并不明显. 对比文献 [6] 的实验结果,模型的分类精度略低于文献的分类精度,但模型的时间代价上却更少,由此说明本文的情感分类实验是有效的,而且本文建立的情感分类模型是适合在分布式云平台上运行并实现情感分类预测的.

表 3 单机和 Spark 平台上的情感分类实验对比结果

Tab. 3 Comparison of experimental results by the sentiment classification algorithm on the single platform and spark platform

语料	单机平台			Spark 分布式平台		
	P/%	$T_{\text{train}}/\text{ms}$	$T_{\text{test}}/\text{ms}$	P/%	$T_{\text{train}}/\text{ms}$	$T_{\text{test}}/\text{ms}$
1	80.53	28 143	5 033	79.75	21 357	4 719
2	91.25	30 099	4 815	89.14	22 873	4 296
3	81.56	15 281	3 912	80.53	11 274	3 483

4 结论

本文通过构建情感词典,利用情感特征及其关联信息,结合卡方统计量技术和改进的 TFIDF 算法及 SVM 机器学习的方法对文本进行情感极性分类. 在单机平台上的实验结果表明采用了情感特征的卡方统计量的 CHI 值及改进的 TFIDF 算法计算出的值组合情感极值的方法能有效地提升分类精度,分类精度最高能达到 91.25%. 在单机平台上和 Spark 分布式计算平台上执行分类模型,对比分析其准确度和时间消耗,实验结果验证了该模型在单机平台和分布式云平台下情感极性分类中的有效性. 下一步的工作是构建覆盖面更广的领域情感词典,研究更好的构建情感极值的算法,并通过采集更大规模的数据集运用半监督的机器学习方法进行情感分类.

参考文献:

- [1] 赵妍妍,秦兵,刘挺. 文本情感分析 [J]. 软件学报,2010,21(8):1834-1848.
- [2] Zhao Y Y, Qin B, Liu T. Sentiment analysis [J]. Journal of Software, 2010, 21(8):1834-1848.
- [3] Turney P D, Littman M L. Measuring praise and criticism in inference of semantic orientation from association [J]. ACM Transactions on Information Systems, 2003, 21(4):315-346.
- [4] Ramanathan N, Liu B, Alok C. Sentiment analysis of conditional sentences [C] // Proceedings of the 2009 Conference on EMNLP. USA Morristown: ACL, 2009, 180-189.
- [5] 刘鸿宇,赵妍妍,秦兵,等. 评价对象抽取及其倾向性分析 [J]. 中文信息学报,2010(1):84-88.
- [6] Liu H Y, Zhao Y Y, Qin B, et al. Comment target extraction and sentiment classification [J]. Journal of Chinese Information Processing, 2010(1):84-88.
- [7] 李寿山,李逸薇,黄居仁,等. 基于双语信息和标签传播算法的中文情感词典构建方法 [J]. 中文信息学报,

- 2013,27(6):75-80.
- Li S S, Lee Sophia Y M, Huang C R, et al. Construction of Chinese sentiment lexicon using bilingual information and label propagation algorithm [J]. Journal of Chinese Information Processing, 2013, 27(6): 75-80.
- [6] 吴江,唐常杰,李太勇,等. 基于语义规则的 Web 金融文本情感分析 [J]. 计算机应用, 2014, 34(2): 481-485, 495.
- Wu J, Tang C J, Li T Y, et al. Sentiment analysis on Web financial text based on semantic rules [J]. Journal of Computer Applications, 2014, 34(2): 481-485, 495.
- [7] Pang B, Lee L, Vaithyanathan S. Thumbs up? sentiment classification using machine learning techniques [C] // Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). USA Philadelphia: 2002, 79-86.
- [8] 唐慧丰,谭松波,程学旗. 基于监督学习的中文情感分类技术比较研究 [J]. 中文信息学报, 2007, 21(6): 88-94.
- Tang H F, Tan S B, Cheng X Q. Research on sentiment classification of chinese reviews based on supervised machine learning techniques [J]. Journal of Chinese Information Processing, 2007, 21(6): 88-94.
- [9] 杨经,林世平. 基于 SVM 的文本词句情感分析 [J]. 计算机应用与软件, 2011, 28(9): 225-228.
- Yang J, Lin S P. Emotion analysis on text words and sentences based on SVM [J]. Computer Applications and Software, 2011, 28(9): 225-228.
- [10] 谢丽星,周明,孙茂松. 基于层次结构的多策略中文微博情感分析和特征抽取 [J]. 中文信息学报, 2012, 26(1): 73-83.
- Xie L X, Zhou M, Sun M S. Hierarchical Structure Based Hybrid Approach to Sentiment Analysis of Chinese Micro Blog and Its Feature Extraction [J]. Journal of Chinese Information Processing, 2012, 26(1): 73-83.
- [11] 李素科,蒋严冰. 基于情感特征聚类的半监督情感分类 [J]. 计算机研究与发展, 2013, 50(12): 2070-2577.
- Li S K, Jiang Y B. Semi-supervised sentiment classification based on sentiment feature clustering [J]. Journal of Computer Research and Development, 2013, 50(12): 2070-2577.
- [12] 徐琳宏,林鸿飞,潘宇,等. 情感词汇本体的构造 [J]. 情报学报, 2008, 27(2): 180-185.
- Xu L H, Lin H F, Pan Y, et al. Constructing the affective lexicon ontology [J]. Journal of the China Society for Scientific and Technical Information, 2008, 27(2): 180-185.
- [13] Zaharia M, Chowdhury M, Das T, et al. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing [C] // Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation. CA San Jose: USENIX Association, 2012: 1-14.
- [14] Zaharia M, Borthakur D, Sen Sarma J, et al. Delay scheduling: a simple technique for achieving locality and fairness in cluster scheduling [C] // Proceedings of the 5th European Conference on Computer Systems. France Paris: ACM, 2010: 265-278.
- [15] 谭松波. 中文情感挖掘语料 ChenSentiCorp [EB/OL]. [2010-06-29]. <http://www.searchforum.org.cn/tan-songbo/corpus-senti.htm>.
- [16] 余永红,向小军,商琳. 并行化的情感分类算法的研究 [J]. 计算机科学, 2013, 40(6): 206-210.
- Yu Y H, Xiang X J, Shang L. Research on parallelized sentiment classification algorithms [J]. Computer Science, 2013, 40(6): 206-210.