文章编号:1001 - 9081(2006)10 - 2509 - 04

一种新颖的 CRE 用户评论信息抽取技术

李 慧[1,2],张 舒[2],顾天竺[2],陈晓红[2],吴 颜[2]
(1. 淮海工学院 计算机科学系,江苏 连云港 222005; 2. 扬州大学 信息工程学院,江苏 扬州 225009)
(shufanzs@126.com)

摘 要:准确挖掘商务网站中的用户评论对于商家进行有效的推荐具有重要意义。提出了一种新颖的用户评论抽取(CRE)算法进行评论信息的抽取。该算法采用了页面分块与信息熵的迭代计算技术实现了评论块的自动发现与抽取。实验结果证明了该算法具有较高的查全率与查准率。

关键词:用户评论抽取;信息抽取;基于视觉的页面分块
中图分类号: TP391 **文献标识码**:A

Novel technology of customer review extraction

LI Hui[1,2], ZHANG Shu[2], GU Tian-zhu[2], CHEN Xiao-hong[2], WU Yan[2]
(1. *Department of Computer Science, Huaihai Institute of Technology, Lianyungang Jiangsu* 222005, *China*
2. *Department of Computer Science, YangZhou University, Yangzhou Jiangsu* 225009, *China*)

Abstract: Mining the customer reviews accurately in commercial websites has significant meaning in effective recommendation for trade company. A kind of novel algorithm—Customer Review Extraction (CRE) was put forward in this paper. CRE iteratively segments page and calculate the information entropy to automatically discover and extract the reviews. The experimental result has proved that the algorithm has higher recall and precision.

Key words: Customer Review Extraction(CRE); information extraction; Vision-based Page Segmentation(VIPS)

0 引言

近年来,众多研究者对评论抽取技术做出了深入研究,提出了许多具有重要价值的有效算法。然而,针对目前复杂多样的 Web 页面,进行评论抽取工作主要存在的问题是抽取方法的低效性及大量的人工干预。这就要求我们采用一种更为通用的信息抽取方法,能够对这些千差万别的网页进行统一处理,从而使算法适用于所有页面。

一般地,针对 Web 页面的信息抽取方法主要包括手工抽取和自动抽取。手工抽取是手工编写代码对目标信息进行抽取。这种方法面对现在数量惊人的 Web 页面,费时费力,是完全不可行的。自动抽取的方法主要利用 Wrapper[1,2],并利用了监督学习的思想来学习数据抽取规则。由于一个 Wrapper induction 系统需要大量人工来标记数据,所以仍然是浪费大量的人力和时间。此外,对于不同的网站甚至是同一网站内的不同页面,这种手工标记过程需要重复进行。Wrapper 的系统包括 WIEN[3],Softmealy[4]等。

在本文的研究中,我们提出了一种基于页面分块和信息熵的评论抽取算法(Customer Review Extraction,CRE)。CRE 算法充分考虑了信息抽取的自动性与通用性,可以实现对各种评论页面中评论信息的自动挖掘。

1 相关概念

1.1 关联规则挖掘

设 $I = \{i_1, i_2, \cdots, i_m\}$ 是项的集合。设任务相关的数据 D 是数据库事务的集合,其中每个事务 T 是项的集合,使得 $T \subseteq I$。每一个事务有一个标识符,称作 TID。设 A 是一个项集,事务 T 包含 $A \subseteq T$。

关联规则是形如 $A \Rightarrow B$ 的蕴涵式,其中 $A \subset I, B \subset I$,并且 $A \cap B = \varnothing$。规则 $A \Rightarrow B$ 在事务集 D 中成立,具有支持度 s,其中 s 是 D 中事务包含 $A \cup B$(即 A 和 B 二者)的百分比。它是概率 $P(A \cup B)$。规则 $A \Rightarrow B$ 在事务集 D 中具有置信度 c,如果 D 中包含 A 的事务同时也包含 B 的百分比是 c。这是条件概率 $P(B \mid A)$。即是:

$$SUPPORT(A \Rightarrow B) = P(A \cup B)$$

$$CONFIDENCE(A \Rightarrow B) = P(B \mid A)$$

同时满足最小支持度阈值(min_sup)和最小置信度阈值(min_conf)的规则称作强规则。关联规则挖掘的问题就是输出所有关联规则,这些关联规则的支持度与置信度均大于用户设定的最小支持度与最小置信度。

1.2 语义块划分

通常,一个 Web 页面都包含了不同的语义块,这些语义块都是不相关的。因此将整个页面看做一个不可再分的整体是不合适的。于是,国内外很多专家学者开始进行页面分块的研究,即将页面分割成多个语义块,使块内主题尽可能一致。对页面进行分割以后,一是可以直接去除一些无用信息块,比如导航信息和版权信息等。二是仅对语义更相关的块进行操作,而不是整个页面,这将提高信息检索的质量。

针对页面分块,使用较多的是基于 DOM(Document Object Model)树的方法[5~7]。然而,由于 HTML 语法的灵活性,很多页面并没有遵循 W3C 规范,因此在构建 DOM 树时就有可能产生错误。此外,DOM 树最初的引入是为了便于在浏

收稿日期:2006 - 04 - 28;**修订日期**:2006 - 07 - 04 **基金项目**:江苏省自然科学基金项目资助项目(BK2005046)

作者简介:李慧(1979 -),女,江苏连云港人,助教,硕士研究生,主要研究方向:数据挖掘、智能信息处理; 张舒(1979 -),男,江苏连云港人,硕士研究生,主要研究方向:数据挖掘、智能信息处理; 顾天竺(1981 -),男,江苏无锡人,硕士研究生,主要研究方向:数据挖掘; 吴颜(1980 -),女,陕西西安人,硕士研究生,主要研究方向:管理信息系统; 陈晓红(1981 -),女,江苏南通人,硕士研究生,主要研究方向:数据挖掘.

览器中显示,并不能描述页面的语义结构。基于视觉的页面分块(Vision-based Page Segmentation,VIPS)算法[8]的提出弥补了 DOM 分割的不足,在原有 DOM 方法的基础之上结合了视觉信息对页面进行语义块划分。本文利用了 VIPS 算法对页面进行分块处理。

VIPS 的工作主要分为三步:首先将 Web 页面解析成 DOM 树结构,之后从 DOM 树中抽取所有合适的块;其次,从抽取的块中找出分离因子,进行页面的划分;最后,构建整个页面的内容结构。

VIPS 算法的过程是这三步的一个循环过程。页面首选被分为几个大的语义块,并且记录下此层分割的层次结构。对每一个大的语义块,相同的分割算法循环调用,直到最终语义块的 DOC 值大于预先设定的 PDOC 值为止。DOC 与 PDOC 的定义如下:

定义 1 DOC(Degree of Coherence)用来测量每个可视块的相关程度。DOC 值越大说明块中内容与主题越相关。

定义 2 PDOC(Permitted Degree of Coherence)为分块程序预先设定的许可相关度,用于为不同应用设置不同的分块粒度。PDOC 值越小,页面内容结构越粗糙。PDOC 值越大,页面内容结构越精细。

1.3 信息熵理论

从信息源具有随机性不定度出发,为信源推出一个与统计力学的熵相似的函数,称为信息熵;这个熵就是信源的信息选择不定度的测度,从而可以认为信息表征信源的不定度。

先给出几个定义与性质。

定义 3 自信息量。任意随机事件的自信息量定义为该事件发生概率的对数的负值。

设该事件 x_i 的概率为 $p(x_i)$,那么,它的自信息定义式为:

$$I(x_i) = -\log p(x_i)$$

定义 4 平均自信息量。集 X 上,随机变量 $I(x_i)$ 的数学期望定义为平均自信息量。

$$H(X) = E[I(x_I)] = E[-\log p(x_i)] = -\sum_{i=1}^{q} P(x_i)\log P(x_i)$$

集 X 的平均自信息量又称作是集 X 的信息熵,简称作熵。

性质 1 Shannon 不等式。熵值具有极值性与非负性,即:

$$0 \leqslant H = -\sum_{i=1}^{n} p_i \log_2 P_i \leqslant \log_2 n$$

其中,P_i 表示事件 I 的概率。

2 评论抽取技术

本文提出了一种新颖的方法来抽取各类评论页面中的用户评论信息,称为 CRE 算法。该算法主要分为三个步骤:

第一步: 应用 VIPS 算法将页面分为若干个语义块。在每次分块结束之后,记录下各块的属性信息。

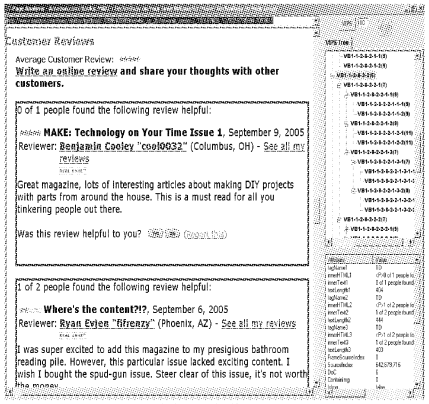
第二步: 对每一个块进行熵值计算,根据熵值大小来确定哪些块是评论块,哪些块是非评论块。

第三步: 对信息块迭代进行前两步的操作,直至抽取出页面中所有的评论信息。通过第二步的分析判断,评论块的坐标与属性信息已经被记录下来,故可以直接完成评论的抽取工作。

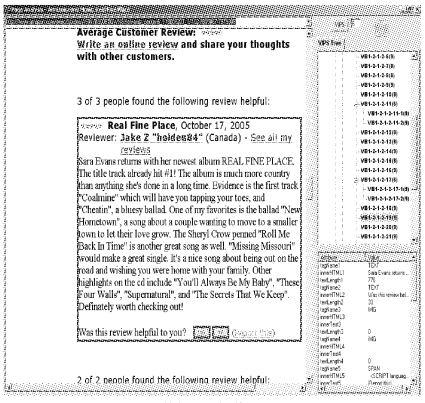
2.1 分割语义块

事实上,一个 Web 页面中通常包含了多个语义块,并且页面中不同位置的内容具有不同的重要度。因此,从语义角度上来讲,不应该将一个页面看成不可再分的最小单元。页面中不同的语义块通常是相关于不同的主题。很自然的,将语义块看作信息的最小单元会更合理。此外,页面中还会包含一些导航信息、链接信息、版权信息等“噪音”等无用信息。这些信息的存在会影响信息抽取的效果。通过对页面的分块处理,可以将不同主题的语义块区分开来、去除一些的噪音信息。同时各个块的坐标与大小等属性也均被记录,这将便于信息的抽取。

应用 VIPS 算法对页面进行分块,关键是要掌握分割的粒度。可以根据不同的应用,通过预设 PDOC 值来得到页面内容结构的不同粒度。PDOC 值设置的越小,分割后的内容结构越粗糙。由于不同的评论页面具有不同的内容构造,欲想得到合适的分割粒度,即正好将每个用户的评论信息单独分割出来,就需要相应地对 VIPS 调整 PDOC 值。如图 1(a)所示的是 PDOC = 10 时将各个用户评论分割出来的页面,图 1(b)所示的是 PDOC = 6 时将各个用户评论分割出来的页面。



(a)PDOC=10



(b)PDOC=6

图 1 应用 VIPS 对页面分块的结果

CRE 算法旨在适用于各种类型的评论页面,使算法处理不同架构的 Web 页面时,均能自动获得合适的分割粒度。而无需对不同页面手工更换 PDOC 值。在实验中,我们首先预设 PDOC 值为 5,然后循环调用 VIPS,并使 PDOC 值逐渐增大,直至取得合适的 PDOC 值。

2.2 计算各块的熵值

本算法中,通过计算各内容块的熵值来发现页面中的信息块,从而完成评论块的定位。通过最终所得熵值的趋向,可将分割出来的内容块分为评论块与非评论块。评论块就是我们所要抽取的目标。

2.2.1 抽取各块中的特征词

通过 VIPS 算法将页面分割成多个语义块,各块的内容可

由若干个关键词(称为特征词)概括。由于用户经常使用一些形容词(如“instructive”,“excellent,”)来表达对商品的评价,因此本文将各块内具有实际意义的形容词定义为表示用户观点评论的特征词。考虑到个别评论中出现的少许形容词具有狭隘性,因此不能选取所有的形容词作为特征词。应该从中选取高频形容词作为最终的特征词(称为频繁特征词)。因而将整个特征词抽取过程分为以下三个步骤:

第一步: 抽取出各块中特征词(本文仅抽取形容词/词组)。

第二步: 使用关联规则挖掘出频繁特征词。

第三步: 分别计算各频繁特征词的熵值,最后将块中包含的所有频繁特征词的熵值累加即可得到各块的熵值。

首先进行各块中特征词的抽取。本文采用 Part-Of-Speech(POS)标记(源于自然语言处理)的方法来识别各块中的形容词/词组。实验中,首先使用 NLProcessor 语言解析器对每个句子进行解析,并标记出每个单词的词性(单词是名词,动词还是形容词等),并识别出简单的名词/动词词组。

例 1　POS 实例

< P > < S > < NG > < WC = 'PRP' L = 'SL' T = 'w' S = 'Y' > I < /W > < /NG > < VG > < WC = 'VBD' > was < /W > < /VG > < WC = 'RB' > so < /W > < WC = 'JJ' > successful < /W > < WC = 'IN' > in < /W > < NG > < WC = 'VBG' > reading < /W > < /NG > < WC = 'CC' > and < /W > < WC = 'VBG' > understanding < /W > < NG > < WC = 'DT' > this < /W > < WC = 'NN' > book < /W > < /NG > < WC = '.' T = '.' > . < /W > < /S > < /P >

NLProcessor 系统的结果以 XML 格式输出。例如, < WC = 'NN' > 表示名词, < NG > 表示名词词组。由于用户多用形容词来表述自己对某产品的评价,所以我们仅抽取其中的形容词作为我们的特征词。被 POS 标记的每一句话都保存在我们的评论数据库中。

其次,进行频繁特征词的挖掘。这一步的目的是找到在评论信息出调频出现的特征词。我们采用关联规则挖掘来找到所有特征词中的频繁特征项。

在第一步抽取后得到的特征词都保存起来,作为关联规则挖掘的事务集。然后运行关联规则挖掘器:CBA[9] 系统是基于 Apriori 算法的。CBA 的输出结果是事务集中所有的频繁项。本文即将每一个频繁项作为最终的频繁特征词。

最后,通过计算每个频繁特征词的熵值,即可用来度量各特征词表示评论的信息量大小。此熵值越大,说明此频繁特征词为评论词的概率越大。同理,如果一个块的熵值越大,表示块内所载评论信息量越大,即成为评论块的概率越大。

2.2.2　内容块的熵值计算

在计算出各频繁特征词的熵值之后,即可进行各内容块的熵值计算。为了使信息熵理论能够很好地应用于评论抽取算法中,我们对信息熵的公式及传统的 TF、IDF 定义进行了变形与拓展。在我们的方法中,一个频繁特征词的熵值是由它在各块中的分布情况确定的,即定义熵值公式中的 P_i 为其权重,而不是出现的次数。这是因为考虑到某些频繁特征词出现的越多,其区分贡献反而越小,故用 TF * IDF 来代替 TF。此外,用块的概念代替传统的文档,即 TF 是指每个频繁特征词在各块中出现的次数,DF 是指页面中含有该频繁特征词的块数。数学公式如下:

定义 5　权重。任意特征词的权重等于此特征词的 TF 与 IDF 的乘积。

$$W_i = P_i = TF * IDF$$

为了方便熵值的比较,可以将熵值的上下限平滑到 0,1 之间,即:

性质 2　熵值的极值平滑。任意频繁特征词 F_i 的熵值可以通过将文档数 d 做为对数的底来实现将熵的极值平滑到 0,1 之间。

$$0 \leqslant H(F_i) = -\sum_{j=1}^{n} w_{ij} \log_d w_{ij} \leqslant 1$$

计算出各频繁特征词的熵值之后,即可计算出各内容块的熵值,计算公式如下:

定义 6　内容块熵值。内容块 CB_i 的熵值等于块内包含的所有频繁特征词的熵值相加。

$$H(CB_i) = \sum_{j=1}^{k} H(F_j)$$

其中 k 等于块 CB_i 内包含的频繁特征词个数。

由于不同块中包含的频繁特征词个数有所不同,为增加各块熵值的可比性,将上式标准化如下:

$$H(CB_i) = \frac{\sum_{j=1}^{k} H(F_j)}{k}$$

即内容块的熵值 $H(CB)$ 是块中所有频繁特征词熵值平均值。经过标准化熵值之后就可以认为同一页面中内容块的平均值是衡量。显然,计算每一个块熵值的时间复杂度为 $O(|CB|)$。

最后,根据块的熵值 $H(CB_i)$,可将块分为两类:评论块与非评论块。

1) 如果熵值 $H(CB)$ 大于预定阈值或趋向于 1,则可定义其为评论块。

2) 如果熵值 $H(CB)$ 小于预定阈值或趋向于 0,则可定义其为非评论块。

2.3　CRE 算法

评论抽取的关键在于评论信息的定位,本文提出的方法是对上述抽取语义块与熵值计算这两步进行迭代计算,直到抽取出所有的评论块。CRE 算法描述如下:

算法:CRE(Block,pDoC)

预设 pDoC 值为 5,运用 VIPS 对输入页面进行分块。(将页面看做一个大的 BLOCK)

```
IF (EnableDivide(BLOCK))
  { FOR (得到的每一个块)
     {ComputEntropy(Block);/*计算熵值*/
      IF (熵值 > 阈值)
      {标记为信息块;
       CRE(BLOCK, PDOC + 1);
      }
     }
  }
ELSE IF(熵值 > 阈值)
   OUTPUT(信息块);
BOOL EnableDivid (blcok)
{/*pos()表示 BLOCK 的 down - left 坐标*/
 IF (pos(PDOC(n + 1) < pos(POC(n))
    return 1;
 ELSE
    return 0;
}
FLOAT ComputEntropy(Block)
```

{ 计算各频繁特征词的熵值 $H(F_i)$;
计算各块的熵值 $H(CB_i)$;
Return $H(CB_i)$;
}

CRE 算法描述了利用 VIPS 分块结果与熵值信息进行评论抽取的完整步骤:程序从根节点出发(第 1 行),对输入的每个页面进行 VIPS 分块算法(预设 PDOC 为 5),对得到的每一个块进行熵值计算,标记出信息块,并存入信息队列当中。当 PDOC 值为 5 时所分得的所有块都计算完毕之后,PDOC 增 1,在信息块中循环进行 VIPS 以及熵值计算,直至每个信息块到达不可再分为止,从而抽取出页面中的所有评论。

3 实验与讨论

本文提出的评论抽取算法已经使用 Microsoft Visual C++6.0 编程实现。现在通过实验评估此自动抽取算法,以验证应用此算法进行评论抽取的有效性。

实验选择评论抽取的查全率与查准率作为评估 CRE 算法的指标,数学公式分别如下:

$$\text{查全率(recall)} = \frac{\text{判断正确的评论块}}{\text{判断的总块数}}$$

$$\text{查准率(precision)} = \frac{\text{判断正确的评论块}}{\text{实际页面的总块数}}$$

实验在 100 个评论页面上进行。这些包含评论信息的页面分别从大型商务网站(如 amazon. com , ebay. com, C|net. com, Epson. com 等)中下载。每一个用户评论都包括文本内容与标题。一些额外的信息,如发表日期,作者名等都是可以在今后的研究中加以利用。为了不失一般性,我们下载的网页包含了书刊、DVD、数码、服装、食品等 5 个类别的用户评论。对于每一个类别,分别下载 20 个页面。这评论页面首先经过去除 HTML 标记预处理,接着进行 NLProcessor 处理以生成 POS 标记。

为了便于比较,先人工阅读页面中的各个内容块信息。首先标记出各块中表示评论观点的特征词;其次,如果此块的内容为用户发表的评论,则将其标记为评论块。由于页面中的评论块信息非常明确,所以手工就能很容易的正确标记出评论块个数。之后再用 CRE 算法对每个页面进行评论抽取,最后统计抽取评论块的查准率、查全率。

表 1 实验结果

产品类别	评论数	CRE		查全率 (%)	查准率 (%)
		发现数	正确数		
书	112	111	110	98.2	99
DVD	96	96	94	97.9	97.9
数码相机	107	106	105	98.1	99
服饰	83	81	80	96.4	98.7
食品	79	76	75	94.9	98.6
总计	477	470	464	97.3	98.9

表 1 给出了详细的实验结果。表的第 1 列是实验选取的 5 个类别名。第 2 列是测试页面中手工标记出来的评论块数。第 3 列是运用 CRE 算法后所找到的评论块数,第 4 列是查找正确的评论块数。第 5 列和第 6 列分别是 CRE 算法的查全率和查准率。表格的最后一行分别统计了页面中包含的所有评论块数、CRE 算法所找到评论块总数和正确的评论块个数,以及算法的平均查全率与查准率。

从表 1 可以看出,Book、DVD 与 Digital camera 三个类别的抽取效果好于另外两类。通过对这些页面的仔细观察,发现商务网站的商品页面中除了包含用户的评论信息,一般都还包含本产品的介绍。前三类的商品介绍多为产品的性能描述,所以使用名词较多,与用户评论信息较容易区分。而后两类产品的介绍信息中则包含有大量的形容词,使得在熵值计算中将这些语义块误认为用户评论块而加以抽取,从而导致了这两类页面的查准率和查全率较低。

为了验证 CRE 算法的挖掘效率,还进行了另外一组测试。在同一测试集上与 MDR[10] 进行对比实验。CRE 算法仅输出页面中的用户评论信息,而 MDR 算法则将页面中的所有块均输出。然而输出块有很多是无用信息,所以 MDR 的查全率较低。具体的实验结果如表 2 所示。

表 2 与 MDR 对比实验结果

	CRE	MDR
评论数	477	477
发现数	469	701
正确数	464	431
查全率(%)	97.3	90.4
查准率(%)	98.9	61.5

4 结语

本文提出了一种新颖高效的技术来自动抽取页面中的评论信息。我们的算法主要基于页面分块和熵值计算,能够实现对评论信息的自动抽取,并且能适用于各种评论页面。对抽取出的评论信息,可进一步对其进行挖掘分析,可辅助商家做出更有效的智能推荐系统。实验表明,我们的算法对于评论信息的抽取具有较高的效率。

参考文献:

[1] CHAKRABARTI S. Mining the web: Discovring knowledge from hypertext data[M]. Morgan Kaufmann Publishers, 2002.

[2] HAN J, CHANG KCC. Data mining for web intelligence[J]. IEEE Computer, Nov. 2002.

[3] KUSHMERICK N, WELD D, DOORENBOS R. Wrapper induction for information extraction[J]. IJCAI-97, 1997.246-247.

[4] HSU C-N, DUNG M-T. Generating finite-state transducers for semi-structured data extraction from the Web[J]. Information Systems. 1998, 23(8):521-538.

[5] CHAKRABARTI S , PUNERA K . Accelerated focused crawling through online relevance feedback[A]. In Proceedings of the eleventh international conference on World Wide Web (WWW2002)[C]. 2002.148-159.

[6] CHEN J, ZHOU B, SHI J, *et al.* Function-Based Object Model Towards Website Adaptation[A]. In Proceedings of the 10th International World Wide Web Conference[C]. 2001.587-596.

[7] CHAKRABARTI S. Integrating the Document Object Model with hyperlinks for enhanced topic distillation and information extraction [A]. In the 10th International World Wide Web Conference[C]. 2001.211-220.

[8] CAI J-RWD, YU S, MA W-Y. Extracting content structure for web Pages based on visual representation[A]. In Proc. 5th Asia Pacific Web ConfP[C]. Xi'an , China, 2003.928-937.

[9] http://www.comp.nus.edu.sg/~dm2, 2006.

[10] LIU B, GROSSMAN R, ZHAI Y. Mining Data Records in Web Pages[Z]. ACM1-58113, 2000.