

一个用于汉语汽车评论的意见挖掘系统

姚天昉, 聂青阳, 李建超, 李林琳

娄德成, 陈珂, 付宇

上海交通大学计算机科学与工程系 上海 200030

E-mail: yao-tf@cs.sjtu.edu.cn tonywhitewhite@qq.com flyinghigher@sjtu.edu.cn ridingwind@sjtu.edu.cn

lou-dc@163.com equalchen@sjtu.edu.cn ralf_kk@hotmail.com

摘要: 近年来, 对描述非事实的主观性文本处理方面的研究十分活跃, 主要的特点是对基于断言或评论的文本进行处理, 此类文本内容包含有个人、群体、组织等的意见、情感和态度等。基于主观性文本的意见挖掘技术是一种新颖的语言技术, 它不仅可以运用于自然语言接口、文本分类、文本过滤、自动摘要、自然语言生成、问答系统等方面, 还可以应用于现实生活中的许多方面, 如电子商务、电子学习、商业智能、出版编辑、企业管理、信息监控、民意调查等。我们所开发的用于汉语汽车论坛的意见挖掘系统的目的是在电子公告板、门户网站的各大论坛上挖掘并且概括顾客们对各种汽车品牌的不同性能指标的评论和意见, 并且判断这些意见的褒贬性以及强度。通过对文本处理的综合统计, 给出可视化的结果。这项研究是对汉语主观性文本处理的一次探索, 特别是文本各个层次处理所涉及到的词汇、句子和文本极性和强度判断以及它们之间关系的分析。意见挖掘系统的测试结果显示了系统总平均召回率达到了80%, 总平均精确率达到了60%。因此, 这种用于汉语汽车评论的意见挖掘系统是合理和有效的。

关键词: 意见挖掘; 意见抽取; 观点分析; 语义倾向; 汉语汽车评论

An Opinion Mining System for Chinese Automobile Reviews

Yao Tianfang , Nie Qingyang , Li Jianchao , Li Linlin

Lou Decheng , Chen Ke , Fu Yu

Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200030

Abstract: In recent years, the investigation on subjective texts concerning non-factual description is extremely active. Its main characteristic is to process texts based on the allegations or the arguments. This kind of text content contains opinions, feelings and attitudes from a person, a community, or an organization and so on. The subjective texts based opinion mining technology is an emerging language technology, which can be applied to not only natural language interface, text classification, text clustering, automatic summarization, natural language generation, question-answering system etc., but also real-world application, such as electronic commerce, electronic learning, business intelligence, publication edition, business management, information monitoring, canvassing etc. The goal we develop this system is to extract and summarize the opinions and reviews which appear in the forums on BBS or homepages and are given by customers towards different features of different brands of cars, and determine whether these reviews and opinions are positive, negative or neutral and how strong they are.

After that, the system can exhibit visible results by means of synthetical statistics from text processing. This research is an exploration for Chinese subjective text processing, specifically, the polarity and strength discrimination for different linguistic layers in texts – word, sentence and paragraph and the relationship analysis among these layers. The testing result for the system has shown that the total average recall and precision are 80% and 60% respectively. Therefore, this opinion mining system used for Chinese automobile reviews is reasonable and effective.

Key words: Opinion mining; Opinion extraction; Sentiment analysis; Semantic orientation; Chinese automobile reviews

1 概述

文本挖掘是抽取新颖、有价值、可理解、可利用和分布在文本文件中的知识，并利用这些知识更好地组织信息的过程。文本挖掘利用智能算法，如神经网络、基于案例的推理、可能性推理等，并结合语言处理技术，分析大量的非结构化文本源（如文档、电子表格、电子邮件、问题查询、网页等），抽取或标记关键字概念、文字间的关系，并按照文本内容对文档进行分类，获取有用的信息和知识[1, 2]。

近年来，对描述非事实的主观性文本处理方面的研究十分活跃[3, 4, 5, 6]，主要的特点是对基于断言或评论的文本进行处理，此类文本内容包含有个人、群体、组织等的意见、情感和态度等。其中对意见型文本进行研究是文本挖掘研究领域十分重要的方面之一。

意见挖掘技术作为一种新颖的语言技术不仅可以运用于自然语言接口、文本分类、文本过滤、自动摘要、自然语言生成、问答系统等方面[7, 8, 9, 10]，还可以应用于现实生活中的许多方面，如电子商务、电子学习、商业智能、报刊编辑、企业管理、信息监控、民意调查等[6, 11, 12, 13, 14]。

日前，网上交易已逐渐成为一种潮流，对产品的各种评价也以惊人的速度增长。许多评论都出现在各大论坛、电子公告板以及门户网站上。于是，当厂商需要了解顾客使用其产品的反馈意见时，当潜在的购买者需要做出是否购买某个产品决定时，他们很难以人工方式在这浩如烟海的评论中仔细阅读，做出全局统计或做出合适的决定。我们开发用于汉语汽车论坛的意见挖掘系统的目的是在电子公告板、门户网站的各大论坛上挖掘并且概括顾客们对各种汽车品牌（例如马自达、本田）的各种不同性能指标（例如质量、价格、安全性等）或重要部件的评论和意见，并且判断这些意见的褒贬性以及强度。在研究中，我们采用了统计和自然语言处理的技术，对汽车本体、极性词词典、句子极性和强度分析、段落和文本极性分析等方面开展了专题研究，并利用这些研究成果开发了本系统。本系统在Windows XP操作系统下采用Java语言开发（JDK版本为1.5.6）。整个体系结构包括预处理、分句、建立元素、指代消解和省略恢复、重构元素、语法关系分析、文本权值分析、可视化结果显示等模块。经测试，系统的召回率达到了80%，精确率达到了60%。

2 有关的工作

对词汇的语义倾向或极性的研究是意见挖掘方法和技术研究的基础。加拿大国家研究委员会信息技术研究所的Turney和美国罗格斯大学的Littman使用了PMI (Pointwise Mutual Information) 和LSA (Latent Semantic Analysis) 统计关联方法对形容词、副词、名词和动词的语义倾向进行了研究,在完全测试集上的精确率达到了82.8%[15]。香港城市大学Yuan等人在Turney研究工作的基础上,对汉语极性词的自动获取进行了研究[16]。研究发现,采用一个字符的汉语极性词素比用汉语极性词作为极性基准词素(词)(比较对象)的效果要好。从实验结果得知:在三千四百万词的汉语语料中,分别采用20和40个汉语极性词作为基准词,然后用统计关联方法自动获取汉语其它极性词。其查全率是79.96%到81.05%,查准率为45.56%到59.57%。而改用20个汉语极性词素作为基准词,用同样的方法在同一语料库中自动获取汉语其它极性词。其查全率是85.03%,查准率为80.23%。与Turney的研究结果相比,在汉语语料库的规模大大小于Turney所需要的语料库规模,但性能却比Turney的研究结果要好。复旦大学朱嫣岚等人提出了基于HowNet的两种词汇语义倾向计算方法[17],即基于语义相似度的方法和基于语义相关场的方法。实验表明,在同一测试集上,基于HowNet语义相似度的方法比基于语义相关场的方法精确率高。在使用40对基准词条件下,在常用词集中自动获取极性词的精确率可达80%以上。

句子的语义倾向处理,如判别极性和强度,是架设于词汇语义倾向处理和文本语义倾向处理之间的一座“桥梁”,它的处理质量将直接影响到文本语义倾向处理的质量。意大利CELI(语言与信息技术公司)的Dini和Mazzini建议采用信息抽取技术解决句子中重要成分的抽取,如expression(贬褒表达),entity(贬褒实体)和content(贬褒内容),结合句型匹配模板,从而决定句子的极性[18]。此外,为了处理更复杂的句子结构,他们还建议采用HPSG深层句法分析与信息抽取中所使用的浅层分析器相结合来处理句子的语义倾向。这样,既能覆盖更多的语句结构,又能提高句法分析的效率。美国匹兹堡大学Wilson等人采用语言学特征(结构特征、句子特征和文本特征)方法对短语的语义倾向进行分类,并解决了词汇的预先语义倾向(静态语义)和上下文语义倾向(动态语义)的歧义问题[19]。这样,也就为判别句子的语义倾向奠定了基础。在实验中,他们采用了两个步骤:第一步是检查句中待判别成分在上下文的情况下是否有极性还是中性的;第二步针对第一步所得到的有极性的待判别成分来进一步判别它的极性。上述两步的分类器采用的是BoosTexter分类器,它是采用自扩展机器学习的方法,对10个结构、句子和文本特征为对象进行了实验,共进行了5000次循环扩展。分类精确率达到65.7%,其中褒义性成分的F度量为65.1%,贬义性成分的F度量为77.2%,中性成分的F度量为46.2%。

微软美国研究院Gamon等人所开发的Pulse系统可以自动挖掘网上用户所上传的自由文本中有关汽车评价中的贬褒信息和强弱程度[13];美国伊利诺斯大学Liu等人所开发的原型系统Opinion Observer可以处理网上在线顾客产品评价,对涉及产品(如电子照相机)各种特征的优缺点进行统计,并采用可视化方式对若干种产品的特征的综合质量进行比较[20];IBM Almaden研究中心的Yi和Niblack介绍了该研究中心开发的WebFountain系统中的意见挖

掘器[9]。WebFountain系统是一个意见挖掘的研究和开发平台。主要有三个核心单元：(1)平台。它集成了挖掘器、爬行器以及面向特定或全局任务、具有开放且可以升级的标准的各种应用，以便为客户提供完整的解决方案；(2)多类型数据集合。它可以存取几千兆字节的非结构化和半结构化数据，数据源包括因特网、网络日志、电子公告板、企业、遗产、许可证、报刊、期刊和行业杂志等；(3)多学科文本分析方法集合。它提供了自然语言处理、统计、概率、机器学习、模式识别和人工智能的文本分析方法。该作者采用了自然语言处理技术将该系统应用于两个领域：电子照相机和音乐评价。其性能为查全率56%、查准率87%。香港城市大学Tsou等人在上述研究工作基础上对中国四地（北京、香港、上海、台北）报刊上有关四位政治人物（克里、布什、小泉纯一郎、陈水扁）贬褒性的新闻报道进行了分类研究[6]。在研究中，首先通过标记语料库获得文本中的极性元素，然后主要采用了三个衡量指标，即极性元素的分布、极性元素的密度和极性元素的语义强度来对每个文本进行统计，得出文本贬褒分类和强度大小的结果。其中对确定极性元素之间的关系有所提及，但没有深入研究。

3 系统设计

在意见挖掘系统设计中，我们定义了一些基本的资源和处理算法的数据结构，如汽车本体、极性词词典、基本元素、语法关系树、匹配规则。在这一节中，我们先对这些基本资源和数据结构作一个简单的介绍，然后再介绍系统的总体结构设计。

3.1 汽车本体

我们的意见挖掘系统是针对汽车领域的文本信息进行挖掘的，在这个领域中会存在大量的特定专用名词来表示一些概念，例如：制动器、动力系、防风挡板等等。这些概念通常不存在于普通词典里。所以，在系统设计中定义了本体的概念。在此需要说明的是：在Web挖掘中的本体其实是一个很广泛而复杂的概念，但是无论从效率还是可靠性的角度来考虑，本系统无需OWL那样的庞杂的概念体系。我们主要关心汽车本体概念中的两个部分：实体(Car)和特征(Feature)，因为意见挖掘的最终目标便是得到指定实体的某个特征的评价信息。

实体指被评价的主体对象。实体分类模型(Car Hierarchical Model)图1 实体分类模型指的是同类实体间通过它们之间的相互关系而构成的树型结构的模型。CHM中的每一个实体结点都会其父结点（当然，顶层结点的父结点为null）和子结点（当然，叶子结点没有任何子结点）。在实际环境中，CHM被表示为一棵多义树。每个CHM结点又会有若干同义词(Synonym)，用来描述表示同一本体概念的专用词汇，如图1所示。注意：图1中并没有将同义词表示出来。另外，实体的根结点命名为“汽车”。虽然图中如此表示，但是这个根结点在实际操作中是不存在的，也就是说CHM并不是一棵单根结点的树，而是一个森林。图1中的一些“??”结点，这些结点称之为泛型实体（介绍见后）。

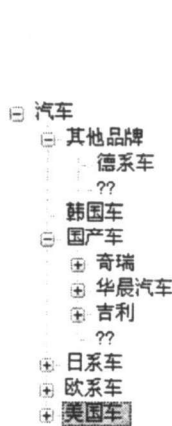


图1 实体分类模型

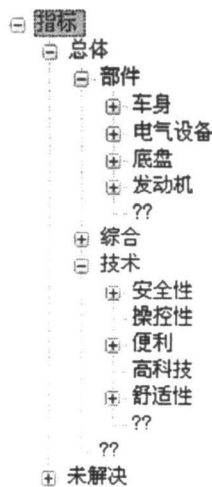


图2 特征分类模型

特征是被评价的实体的某一种属性。特征分类模型（Feature Hierarchical Model）指的是同类特征通过描述它们之间的相互关系而构成的树型结构的模型。FHM中的每一个特征结点都有其父结点（当然，顶层结点的父结点为null）和子结点（当然，叶子结点没有任何子结点）。在实际情况下，FHM被表示为一棵多叉树。每个FHM结点又会有若干同义词，用来描述表示同一本体概念的专用词汇，如图2所示。

FHM虽然非常类似CHM，也含有泛型特征（图2中“??”结点）。但是，FHM和CHM不同之处在于：FHM是一棵单根树，而不是森林，它的根结点为“总体”。为什么会有这样的差别呢？因为评价可以针对某一实体而不涉及其具体属性，但却无法只针对某一属性却不知道此属性所属的实体（至少是无意义的）。举例来说，我们可以说“桑塔纳不错”，却不会在上下文都没有提到实体的情况下说“刹车不错”。

在对实际语料的研究过程中，我们发现一个有趣的问题，这就是意见挖掘系统的输出到底应该是什么样的？一般来说，我们把结果定义为“对特定实体的特定特征的评价”。但是，由于采用了本体来表述实体和特征的概念，本体中的结点都有了上下位的语义关系，那么这句话中的“实体”和“特征”到底是指CHM和FHM中哪些结点呢？是全部的结点还是只是叶子结点？

最初我们考虑将结果限定在叶子结点上，这样便于形式化的表述和处理。但是，在实际处理中我们便遇到了两个问题：

- 如果用户不是特别关心某一个实体的最末端的叶子结点特征的情况，而是只想大致了解一下实体的整体情况，该怎么办？例如，用户并不是很懂汽车，搞不懂许多专业术语，他（她）只想知道宝马和奔驰在整体性能上谁更好一些。很明显，这个用户想知道的不是有关叶子结点上的信息，而是高层结点的综合信息。

- 在实际语料中确实会有专家能够回答像刚才那位用户所关心的问题。用户的问题确实得到了解答，但是这位专家所给出的宝贵意见却无法落实到任何一个叶子结点上。

面对上述问题，我们的解决方案是：

对于第一个问题，加入“整体（General）”的概念。我们对每个本体结点都赋予了权值。

对于第二个问题，加入“泛型（Generic）”的概念。每一个非叶子结点都会有一个名字为“??”的结点，这个结点表示其直接父结点的“泛型结点”。即如果有哪一个评论只是对某一个结点表示的概念泛泛而谈，无法归入任何一个叶子结点时，就将其归入该非叶结点的泛型结点中。注意：泛型结点必然是叶子结点，这样做的好处是仍然保证了只有叶子结点才有实际的评价，使得最终的结果得以很好的形式化。

```

<Ontology>
<Tree name="Car">
  <Node name="其他品牌" weight="1.0">
    <!--the end of 其他-->
  </Node>
  <Node name="韩国车" weight="1.0"/>
  <Node name="国产车" weight="1.0">
    <Node name="奇瑞" weight="1.0">
      <Node name="华晨汽车" weight="1.0">
        <Node name="金杯" weight="1.0">
          <Node name="中华" weight="1.0">
            <Node name="骏捷" weight="1.0"/>
            <Node name="尊驰" weight="1.0"/>
          </Node>
        </Node>
      </Node>
    </Node>
  </Node>
  <Node name="吉利" weight="1.0">
  </Node>
  <Node name="日系车" weight="1.0">
  </Node>
  <Node name="欧系车" weight="1.0">
  </Node>
  <Node name="美国车" weight="1.0">
  </Node>
</Tree>
<Tree name="Feature">
  <Node name="总体" weight="1.0">
    <Node name="部件" weight="1.0">
      <!--the end of 部件-->
    </Node>
    <Node name="综合" weight="1.0">
      <!--the end of 综合-->
    </Node>
    <Node name="技术" weight="1.0">
      <Node name="安全性" weight="1.0">
      </Node>
      <Node name="操控性" weight="1.0">
        <Synonym 操控 /Synonym>
      </Node>
      <Node name="便利" weight="1.0">
      </Node>
      <Node name="定速巡航" weight="1.0"/>
      <Node name="雨量传感器" weight="1.0"/>
    </Node>
    <Node name="高科技" weight="1.0"/>
    <Node name="舒适性" weight="1.0">
    </Node>
    <Node name="防眩后视镜" weight="1.0"/>
  </Node>
  <!--the end of 技术-->
</Node>
<!--the end of 总体-->
</Node>
<Node name="未解决" weight="1.0">
  <Node name="市场保有量" weight="1.0"/>
  <Node name="生产部门" weight="1.0"/>
  <Node name="服务部门" weight="1.0"/>
</Node>
</Tree>
</Ontology>

```

3.2 极性词词典

在人类语言中有一类词，人们直接使用它们来表达自己的感情，或喜或厌，或褒或贬。尤其是当人们对某一事物的好坏作出评价时候，往往通过使用这类词语来表达自己鲜明的观点。我们把这一类带有情感倾向的词语称之为极性词（Polarity Word）。显而易见，对极性

词的研究是意见挖掘研究的基础。因为意见挖掘就是要挖掘人们的对事物所表达的观点,而极性词则是人们用来表达自己观点的直接描述工具。

典型的极性词通常有三类情感倾向:褒义(Positive)、中性(Neutral)、贬义(Negative)。例如:“美丽”、“成功”、“英雄”为褒义词,“普通”、“人”、“吃饭”为中性词,“愚笨”、“摧残”、“笨蛋”为贬义词。而现在的许多意见挖掘研究也只是针对这三类极性词展开的。但其实在同一类中的词语中,还有不同的情感强烈程度。例如同要是褒义词:“不错”、“良好”、“优秀”、“专家级”四个词都是表示“好”这个概念,是褒义词,但是很明显情感的强烈程度从“不错”到“专家级”递增。所以在我们所设计的极性词词典中,极性词不使用情感倾向类别,而是直接使用一个整数 n ($n \in \{-3, -2, -1, 0, 1, 2, 3\}$)来表示极性,其中负数表示贬义、0表示中性、正数表示褒义,绝对值越大表示情感越强烈。

如果只考虑极性词的话,那还是有很多问题难以处理,甚至处理错误。最简单的情况就是否定的场合,例如:“马自达一点都不便宜”。便宜本来是褒义词,但是由于前面加了一个“不”,整句句子就是对马自达的贬义评价了。另外一种情况就是感情的强化和弱化,例如:“东风轮胎绝对经久耐用!!!”。这句话中“经久耐用”是对东风轮胎的赞美,结果说话人还加上了“绝对”二字进行强调,则感情色彩更强烈了。又如:“其实法拉利只是样子好看点而已”。这句话中“好看”是对法拉利的“样子”的褒义评价,但是句中出现了“只是”、“而已”等弱化感情色彩的词语,可以看出评论人对法拉利的外观不是特别欣赏,虽然还是认为它的好的。像这一类用来修饰极性词,使得极性词表示的感情色彩发生变化的词语(多数是副词)我们称之为修饰词(Modifier Word)。对于每一个修饰词,我们定义了一个属性:强化度(Enhancement),其中 $-\infty < \text{Enhancement} < +\infty$, $\text{Enhancement} \in \text{整数}$ 。强化度若为负数表示起到否定作用,强化度的绝对值表示对情感的放大程度,大于1表示强化情感,小于1表示弱化情感,理论上强化度可以为0,但是没有实际意义。

虽然有了极性词和修饰词之后就能比较好地定义词语对感情的描述作用了,但是还有一类问题无法解决。例如:在“奇瑞的性价比虽然高,但油耗也高!”这句话里,同样是一个“高”,前者评价“性价比”是褒义,而评价“油耗”则是贬义了。这种情况还有很多例子可举。这说明极性词的褒贬并不是静态的,而是有时会根据其作用的特征发生变化的,于是极性词极性的动态化便成了需求。根据上述情况,我们定义了异常特征(Exceptional Feature)和动态极性词(Dynamic Polarity Word)两个概念:

- 如果某个极性词在修饰某个特征时极性会发生正负变化,那么这个特征称为这个极性词的一个异常特征。
- 如果某个词语通常不表现为极性词,只有在修饰某个特征时候才含有褒贬意义,那么这个词语成为动态极性词,这个特征称为这个词语的动态特征(Dynamic Feature)。

当然,如果要对极性词的动态性做更好的解决方案,那么势必要进行上下文环境的严格定义,然后将极性词的极性表示为一个二元组{Context, Strength},这项工作在本文将不进行讨论。

在对实际语料的研究中还发现了一个有趣的现象。例如:“便宜”、“省油”、“漂亮”等等,

它们每个词都既包含了情感色彩，又同时隐含了它们所修饰的特征：

便宜：特征=价格 极性=褒义

省油：特征=油耗 极性=褒义

漂亮：特征=外观 极性=褒义

这种既包含情感倾向，又隐含修饰特征的词语，称为特征极性词（Feature-Polarity Word）。在对这类词语的处理时我们使用了词标（Word Flag）技术（介绍见后）。

我们和定义汽车本体一样，同样使用XML Schema来定义极性词词典。但为简略起见，我们在此给出了极性词词典形式化定义的可视化结构，如图4所示。

形容词“高”是一个动态极性词，这里给出它在极性词词典中的例子：

```
<word text="高" pos="a" strength="2" definition="a Value|属性值,degree|程度,very|很">
```

```
<dynamic-polarity id=noun-relation>
  <basic-polarity strength="2" />
  <exceptional-words>
    <feature-word text="价格" id-ontology="10"/>
    <feature-word text="油耗" id-ontology="5"/>
  </exceptional-words>
</dynamic-polarity>
</word>
```

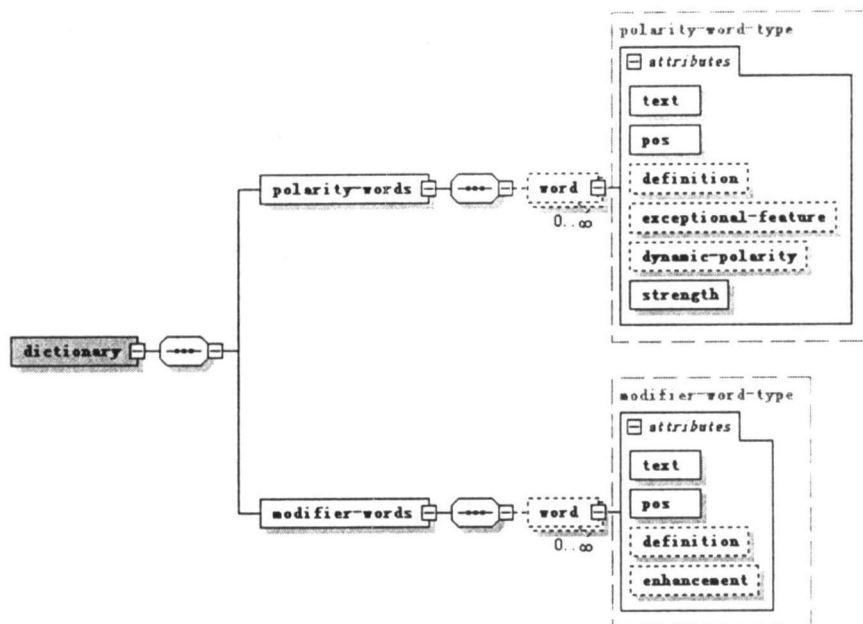


图4 极性词词典形式化定义的可视化图

3.3 基本元素

在此，我们将简要介绍系统中使用的各种语法基本元素的构造特点。词语（Word）是

自然语言处理的最小单位，在本系统中一个词包含下列属性：

- 词语文本 (Text)
- 词性 (POS)
- 词标 (Word Flag)
- 内隐对象 (Hidden Object)，可以是实体、特征、极性强度、强化度中的一个或多个，视具体的词标而定

词标指的是词语在意见挖掘分析中所处的角色。在本系统中一共定义了以下词标：

Normal(0),	无特殊含义的词语（如“云”、“水”）
Car(1),	表示实体的词语（如“马自达”、“三菱”）
Feature(2),	表示特征的词语（如“油耗”、“价格”、“性能”）
CarFeature(3),	既表示词语又表示特征的词语
Polarity(4),	极性词（如“差”、“强”、“牛”）
CarPolarity(5),	实体极性词
FeaturePolarity(6),	特征极性词（如“便宜”、“省油”）
PolarityCarFeature(7),	实体特征极性词
Modifier(8),	修饰词（如“非常”、“很”、“有点”）
CarModifier(9),	实体修饰词
FeatureModifier(10),	特征修饰词
CarFeatureModifier(11),	实体特征修饰词
PolarityModifier(12),	修饰极性词
CarPolarityModifier(13),	实体修饰极性词
FeaturePolarityModifier(14),	特征修饰极性词
PolarityCarFeatureModifier(15),	实体特征修饰极性词

括号内为例词。粗体表示的是基本词标，其本值 (Ordinal) 都为2的N次幂，其它词标均为这些基本词标的逻辑或合成得到。当一个词语被赋予不同的词标的时候，便有了可以取到不同类型的内隐对象，其内隐对象和基本词标的对应关系如下：Car – WordFlag.Car；Feature – WordFlag.Feature；Strength – WordFlag.Polarity；Enhancement – WordFlag.Modifier。

当词标为合成词标（非基本词标）时，取到合成它的所有基本词标可以取到的内隐对象。

主题 (Topic) 是系统能处理的数据的最小闭包，即系统最多能够在一个主题的范围内容联系上下文对文本进行分析。一个主题通常是由一篇报道、一个论坛主题系列的帖子集合、一个产品的一天内的评论集合构造所得。主题包含的属性有：

- 源文本 (Source Text)
- 默认主题实体 (Topic Car)
- 一系列评论 (Comment)
- 额外信息（在本系统这个版本中并未使用到）

评论指的是一个主题中相对具有独立性的一段文本。例如报道中的一个段落、论坛中的一个帖子、主题系列中的一个回帖都是一个评论。有时候评论还会包含子评论 (Sub

Comment), 这在实际处理时会遇到。

元素指经过语法分析的, 含有分析结果和原始信息的一个结构。通常元素是由评论经过语法分析得到的。一个元素包含以下属性:

- 分词列表 (Word List)
- 语法关系树 (Relation Tree)
- 原始文本 (Source Text)
- 默认主题实体 (Topic Car)

3.4 语法关系树

语法关系树刻画了句子中词语互相之间的语法关系。由于本系统使用了哈尔滨工业大学信息检索研究室的汉语句法分析器Deparser (<http://www.ir-lab.org/>)。所以, 关系类型其实就是依存对的关系类型。

语法关系树结点指的是在语法树中的一个以词为中心的结点, 一个语法关系树结点包含以下属性:

- 中心词 (Word)
- 子结点列表 (Child Node List)
- 父结点关系弧 (Parent Arc) ——指向父结点的语法弧

如果词W和词V之间存在依存关系, 其关系类型为R, 那么二元组{W, R}称为W指向V的语法弧。

需要说明的是: 为了算法的效率, 在构建语法关系树的时候同时将其中所有特殊节点(即中心词的词标不是Normal的结点)都缓存到了哈希表中, 这个哈希表成为特殊节点映射表。

图5是评论、元素、语法关系树的一个例子。

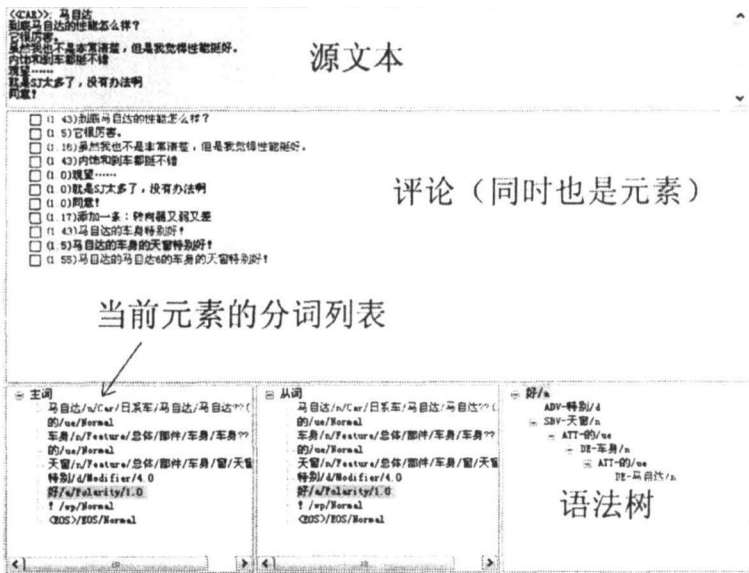


图5 评论、元素、语法关系树的一个例子

3.5 匹配规则

这里所介绍的匹配规则特指本系统用来挖掘意见信息的规则。匹配规则由规则优化器（Pattern Simplifier）分析人工标注的语料得到。系统的规则集一共由6组匹配规则组成，每组匹配规则分别负责一类意见信息的抽取工作。

在定义路径（Route）之前，需要先定义最年轻公共父结点的概念：对于同属于一棵语法关系树T的词W和词V，若结点P既是W的父结点也是V的父结点，并且任取结点P'既是W的父结点也是V的父结点，P'是P的父结点或就是P本身，那么称P是W和V的最年轻的公共父结点。然后再定义结点序列的概念：对于同属于一棵语法树T的词W和词V，若P是W和V的最年轻公共父结点，则按照如下方法构造的序列UpRoute：

- 1) 令U为W；
- 2) 将U的词性（POS）放入UpRoute末尾；
- 3) 若U就是P，结束；
- 4) 将U的Parent Arc的Relation Type放入UpRoute末尾；
- 5) 令U为U的父结点；
- 6) 转 2)。

按照如下方法构造的序列DownRoute：

- 1) 令U为V；
- 2) 将U的词性（POS）放入DownRoute头部；
- 3) 若U就是P，结束；
- 4) 将U的Parent Arc的Relation Type放入DownRoute头部；
- 5) 令U为U的父结点；
- 6) 转 2)。

UpRoute和DownRoute合称在T中以W为主词，以V为从词的路径，UpRoute称为上行路径，DownRoute称为下行路径。可以注意到，路径其实是以词性开始和结尾，每两个词性之间就有一个关系类型的序列。

一个匹配规则包含一个上行路径正规式（UpRoute Regular Expression）和一个下行路径正规式（DownRoute Regular Expression），可以用来匹配语法树中的两个词。当然，在实际的系统中，为了性能考虑，路径正规式使用了原生字符串（Raw String）。

另外，为了使匹配规则能够具有一定程度的通用性，可以使匹配规则在匹配词的时候使用泛型词性，即将所有同种类的语素都看作同一词性。这样做的目的可以提高召回率。

与汽车本体、极性词词典一样，匹配规则集在系统中同样使用XML Schema来定义。为描述简洁起见，我们在这里采用了可视化结构来定义匹配规则集，如图6所示。注意：feature-polarity-patterns以下的五个结点均为展开，它们的结构和car-feature-patterns相同。

匹配规则集中car-feature-pattern的例子如下：

```
<?xml version="1.0" encoding="UTF-8" ?>
<patterns version="1.0">
  <car-feature-patterns>
```

```
<pattern up-route="%v%%DE%%u%%ATT%(%ws% | %n%)%SBV% (%i% | %a%) " down-route="%v%%IC% (%ws% | %n%) %SBV% (((%ws% | %n%) %COO% (%i% | %a%)) | ((%i% | %a%))) " />
```

...

```
</car-feature-patterns>
```

...

在上述匹配规则中，a, i, n, u, v, ws分别表示词性形容词、习用语、名词、助词、动词和字符串。ATT, COO, DE, IC, SBV分别表示语法关系定中、并列、“的”字结构、独立分句和主谓关系。

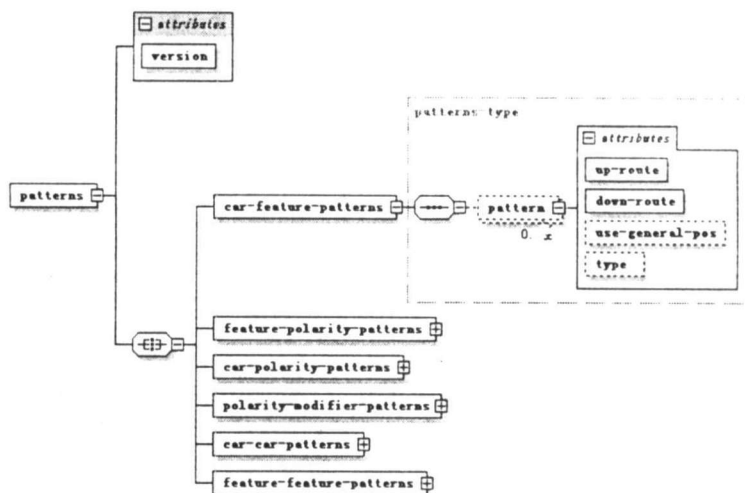


图6 匹配规则集形式化定义的可视化图

3.6 总体结构

在设计意见挖掘系统的总体结构时，我们采用了流水线体系结构。在这个结构中，集成了预处理、分句、建立元素、指代消解和省略恢复、重构元素、语法关系分析、文本权值分析、可视化结果显示模块。体系结构如图7所示。

下面描述一下意见挖掘系统的工作流程：

- 1) 系统的输入是从太平洋汽车论坛上 (<http://bbs.pcauto.com.cn/>) 抽取的大量关于汽车评论的语料。首先对它们进行预处理，即通过噪音去除、数据整合以及数据选取等一系列操作，把每个从论坛上抽取的评论转换成便于我们处理的主题结构，每个主题结构中含有主题、评论等信息。
- 2) 对每个主题结构进行处理，利用分句模块把每个主题结构中的评论信息切分成一个个的句子，并把它装入评论结构。
- 3) 利用DeParser句法分析器提供的词性标注、分词功能以及依存关系树，以及汽车本体和极性词词典所提供的信息。可以把每个评论构建成一个元素结构。元素结构中包含原始文本、分词列表、语法关系树和默认主题实体。

- 4) 对每个元素结构进行指代消解和省略恢复操作。经过上面的指代消解和省略补充处理后，我们又获得了新的信息，因此进行元素结构的重构。
- 5) 重构完成后，利用之前标注语料提取的匹配规则进行关系抽取，并进行语法分析得到关系分析结果。
- 6) 利用上述步骤得到的结果，并对整个主题结构中的每个评论进行权重计算，选取关键句，从而得到整个主题结构的语义倾向和强度。
- 7) 将上述主题结构的语义倾向和强度转换为可视化结果并以直方图形式显示。

由于论文的篇幅限制，我们在此只介绍两个模块的算法，即指代消解和省略恢复模块以及语法关系分析模块的算法。

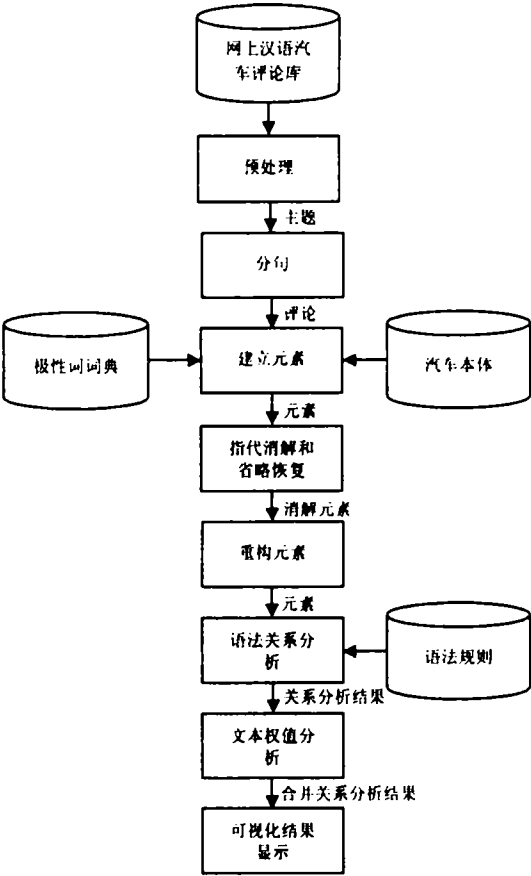


图7 意见挖掘系统的体系结构

指代消解和省略恢复模块的算法：

指代消解算法主要分为两个部分：首先判断接受到的元素结构是否需要指代消解，对于待消解的元素结构，利用一些约束原则和上下文语境形成先行语候选集合。接着由前面模块语法分析得到的信息，以及一些规则进行先行语权值计算，最终确定先行语候选。

在确定先行语候选部分，我们定义了一些过滤规则：

- 1) 在遍历分词列表的时候，如果词语的类别不是汽车实体、汽车特征或已经消解的代

词，则过滤该词语。

- 2) 如果该词语的单复数信息与指代语的单复数信息不一致，则过滤该词语。
- 3) 如果该词语的语法角色与指代语的语法角色不兼容，则过滤该词语。比如：如果一个句子的动词引导一个子句作为宾语，那么当句子的主语是人称代词时，被引导的子句名词不能作为该代词的先行语。

在进行先行语权值计算部分，我们使用了如下的设置方案：

- 1) 角色权值设置：如果指代语和先行语在同一个句子中，而且彼此语法角色兼容（相同角色总是兼容，主语和宾语兼容等等），则权值设置为 2，否则为 0；如果不在同一个句子中，彼此语法角色兼容的话，权值为 1，否则为 0。
- 2) 距离权值设置：首先在句子内部确定候选时，在同一个子句中权值为 1，每相隔一个子句，权重加 1；考虑上下文环境时，在同一个句子中权值为 1，每相隔一个句子，权重加 1。
- 3) 单复数权值设置：如果先行语和指代语单复数信息兼容，则权重为 2，否则为 0。
- 4) 先行语总权值：总权值 = 角色权值 + 单复数权值 - 距离权值。

首先是先对句子内部的先行语候选进行处理：

- 1) 首先进行预处理，过滤掉一些噪音，比如：“其他”、“不管三七二十一”等；
- 2) 对句子层次进行划分，并找到先行语候选；
- 3) 对每个候选计算各自的语法角色权重、单复数权重和性别权重（本系统中不考虑性别权重，设置为 0）；
- 4) 计算每个候选的距离权重；
- 5) 确定该句子层次中最有可能的先行语候选，并输出，进行下一步处理。

上述算法部分主要目的是确定句子内部的先行语候选，接下来要考虑上下文环境，向前文搜索并不断扩充候选集合，并最后通过权值计算规则确定最有可能的先行语。

- 1) 重新设置距离权重，考虑句子之间的距离，而不是子句；
- 2) 指针向左移动，考虑上一个包含先行语候选的句子，同时距离权重加 1；
- 3) 同时结合语法角色、单复数信息计算该先行语的权重；
- 4) 如果前文的先行语候选值不会大于现在的先行语候选集合中的最大值，则终止指代消解算法，具有最大权值的先行语被确定为最终候选；
- 5) 指针继续向左移动；
- 6) 重复 3)~5)。

省略恢复算法主要分为两个部分：首先是进行省略判断，确认待处理的元素是否需要进行主语省略补充，主要是对元素结构中的语法依存关系树进行分析。然后，对于待省略恢复的元素，用空谓词记录待补充的位置，并保存相应的约束信息，利用上下文环境进行主语省略补充。

同时我们还制定了一些省略恢复的规则：

- 1) 对于主语省略，省略所跨越的句子数目不能超过 10 个，也就是说省略的位置和被省略的成分之间相隔不能超过 10 个句子。如果超过了这个限制就只能利用默认主题进

行主语省略恢复。

- 2) 由于实验中只考虑对汽车领域进行意见挖掘,因此只考虑那些出现了极性词但是没有出现对应的汽车实体和特征的句子。因此被补充的成分只是汽车品牌、汽车技术和汽车性能。

省略恢复算法如下:

- 1) 首先分析元素的语法依存关系树,判断是否需要省略恢复;
- 2) 如果不需要,则终止;如果需要省略补充,则设置空谓词,记录需要补充的位置,并保存相关信息;
- 3) 考虑上下文环境,对空谓词进行复原;
- 4) 如果没有空谓词剩余,则终止程序;否则,重复(3)~(4)。

语法关系分析模块算法:

- 1) 首先判断在语法关系树中是否存在实体结点,如存在,转2);否则,转8);
- 2) 构造冗余实体和特征集;
- 3) 构造极性词-修饰词的映射关系表;
- 4) 构造特征-实体的映射关系表;
- 5) 抽取特征-极性词对;
- 6) 抽取实体-极性词对;
- 7) 终止;
- 8) 判断在语法关系树中是否存在特征结点或极性结点,如存在,转9);否则,转7);
- 9) 使用默认主题实体,转2)。

上述算法的目的是从构建好的语法关系树中找到有修饰关系的词对。它使用事先训练好的六组匹配规则(见3.5节图6)来抽取这样的词对。而这六组匹配规则在算法中各司其职。

4 系统实现

4.1 资源开发

在本系统中所使用到的资源有汽车本体、极性词词典和匹配规则。

开发汽车本体使用了protégé-2000工具[21],主要采用手工的方式进行。实体分类模型涉及汽车品牌,特征分类模型涉及汽车部件、汽车综合和汽车技术等。由于本体中同一个实体的各个特征可能会有不同的重要性,应该按照重要程度给予每个特征以不同的权值。本系统中已经定义了本体结点权值功能,但是目前为止限于汽车本体的研究尚未最终完成,实际的本体文件中所有结点的权值均为1.0。另外,考虑到描述的严格和运行效率,采用了XML方式来定义汽车本体。

在极性词词典的开发中,我们采用了将英文极性词词典 Harvard &Lasswell[22]借助于HowNet[23]进行转换并人工修正的方法。转换的过程是先从 Harvard &Lasswell 词典中获取英文极性词,然后对每个英文极性词在 HowNet 中查找 W_E 域为该英文极性词的词条,获取对应的中文极性词,词性,极性,概念定义。接下来在已生成的中文极性词词典中查找,

看对应的中文极性词是否已经包含在其中, 如果没有, 则添加入中文极性词词典。

通过这种方法总共得到 3120 个褒义词, 3485 个贬义词。随机抽取了 10% 进行人工检查, 发现 2/3 的结果是准确的。其中不准确的情况大多发生在中性词误分为贬褒词 (且多发生在名词), 很少有贬褒词极性置反的情况发生。其中中性词被误分为贬褒词的情况就需要人工检查来排除。另外, 在针对语料的词频统计试验中, 曾提取了一些高频的名词, 副词和形容词, 用手工的方式在这些提取的词中选取了极性词。并将这一部分词也加入了极性词词典。

在此基础上, 将英文词典 Harvard & Lasswell 的 weak 域和 strong 域信息转换到中文词典的强度域。若为 strong 则极性强度为 “3”, 若为 weak, 则极性强度为 “1”, 其余极性词强度为 “2”。除了从 Harvard & Lasswell 词典中获取的极性词外, 其余的极性词采取手工标注极性强度的方法。

极性词的异常特征和动态极性以及修饰词的强化度通过手工方式添加。特征极性词也通过手工方式添加。

系统的匹配规则的获得是采用手工标注基础上的半自动生成方法。通过一个我们自己开发的训练/测试工具, 对输入语料进行标注 (标注方法见 5.1 节。)。然后根据模式匹配方法自动导出规则文件。在导出规则文件时, 是否选中了 “使用泛型词性” 将直接影响导出的文件的内容。如果选择了 “使用泛型词性”, 那么此工具将对所有标注的信息只在语素级别上处理词性。例如把所有标注的名语素都作为名词来处理, 如语料中某条路径是 “专有名词-主谓-专有名词”, 那么使用了泛型词性后就变成了 “名词-主谓-名词”, 即范化了路径的。由于直接从训练/测试工具导出的规则非常多, 通常有上百条, 导致使用这样的规则运行系统速度很慢, 而且过于死板, 不适合扩展, 于是我们又开发了规则优化器来优化规则。规则优化器的优化算法为:

- 1) 剔除重复规则;
- 2) 将含有包含关系的两个规则合并;
- 3) 将相似的两个规则合并;
- 4) 将名语素和非汉字语素合并;
- 5) 迭代操作以上各项直到没有规则变化为止。

通过优化, 减少了许多不必要的匹配规则。

4.2 系统界面

意见挖掘系统在 Windows XP 操作系统下采用 Java 语言开发 (JDK 版本为 1.5.6)。并采用 CVS (Cuncurrent Versions System) 版本控制工具进行系统集成, 包括处理模块和资源。先通过各个系统模块和资源模块各自独立调试, 测试, 再进行集成, 总的调试和测试。在系统也集成了哈尔滨工业大学信息检索研究室提供的汉语句法分析器 Deparser。

除了主要系统的上述部分以外, 还开发了四个辅助工具, 即测试/训练语料标注工具、资源生成器、规则优化器、准确度测试工具。

系统界面主要分为两个部分: 测试/训练语料标注界面, 分析结果演示界面。在此做一个简单介绍。

1) 测试/训练语料标注界面

主要功能:

- 导入文本语料——所有的语料都可以以文本文件导入;
- 保存标注信息——可以在已标注的句子前打勾,保存后下次打开这些勾仍然存在;
- 合成文件——可以将多个已标注的文件合并到一起;
- 导出测试配置文件——可以导出测试配置文件以便运行测试;
- 导出规则文件——可以导出系统匹配规则文件;
- 单句、单篇文章实际测试——可以当场测试某句句或某篇文章。

图 8 所示是导入语料后的界面。

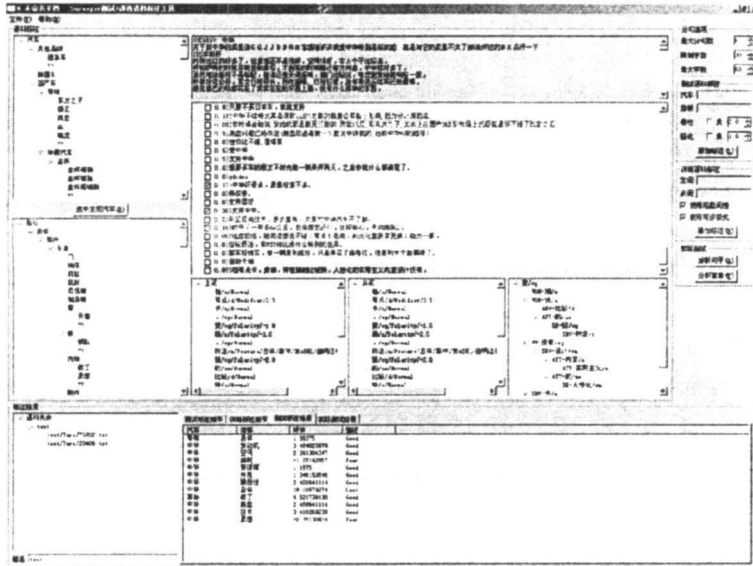


图 8 测试/训练语料标注界面导入语料后

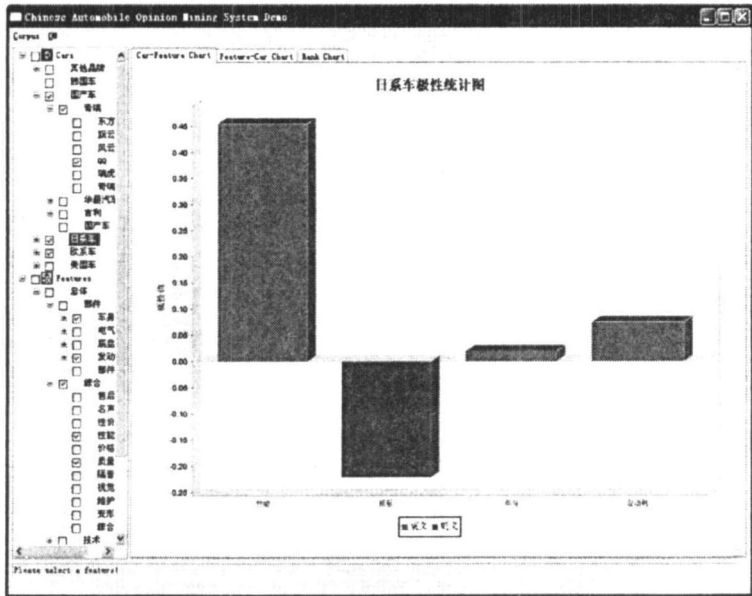


图 9 实体-特征图

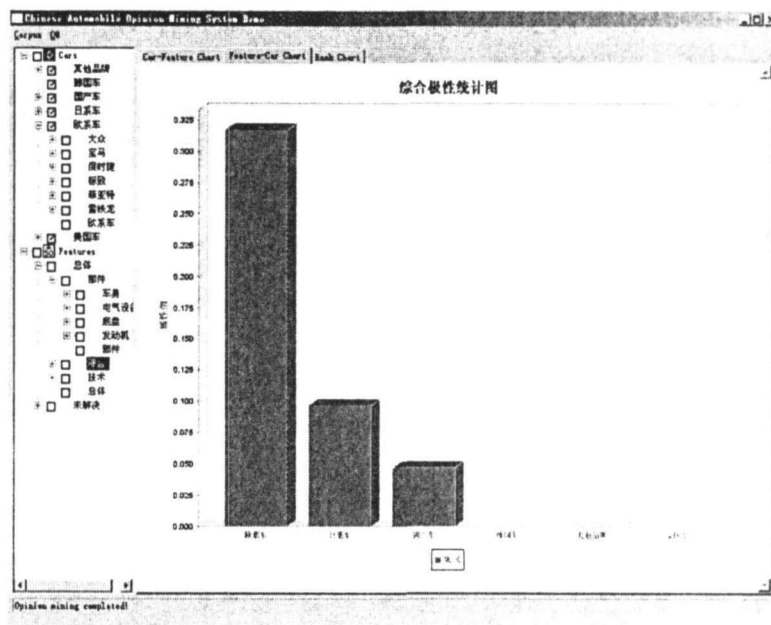


图 10 特征-实体图

2) 分析结果演示界面

主要功能:

- 选择实体-特征图，显示某种车的特征评价;
- 选择特征-实体图，显示不同汽车的同种特征，以便进行对比。

图 9 所示是实体-特征图，该图显示的是欧系车的所选特征的评价信息。图 10 所示是特征-实体图，该图显示的是不同车型（欧系车、日系车和国产车）的综合特征的评价信息。

5 测试方法和结果

5.1 测试语料的标注

如上所述，意见挖掘系统包含了一个测试/训练语料标注工具，标注者可以选择导入测试语料，系统会根据分句准则对语料进行分句，然后标注者可在每句话中选择实体词和特征词，最后对特定汽车的特定指标进行极性标注，这个极性强度可以是 $(-\infty, +\infty)$ 之间的任意一个整数数值。

例如：“Polo的性能很好，而且它的外形很好看，但是价格很贵。”

标注者可以根据标注工具提供的便利选择实体“Polo”，特征词“性能”，然后手工标注极性“+4”。即标注者可以手工添加如下记录，该记录会保存在标注结果文件中。如图11所示。

Polo	性能	+4
------	----	----

图 11 标注记录

5.2 测试方法

意见挖掘系统利用汽车本体、极性词词典、以及各种语法关系匹配准则（自动提取测试语料中的评价信息，提取结果如上图所示）。我们将测试语料的人工标注结果和系统自动提取结果进行对比，以此来对系统进行测试。

由于手工标注的主观性，以及极性词词典、句子权重计算的相对主观性，我们很难做到手工标注的结果和系统自动抽取结果的精确匹配。还拿上图为例，如果我们要求精确匹配，自动抽取结果必须是极性+4，才能认为匹配，如是+3或者+5都将出现匹配错误，而事实上人很难主观的区分+3和+4在极性上的微小差异。对极性等级使用过小的颗粒度，最终会导致系统的精确度很低，而这种区分又显得太过主观化。

针对这种情况，我们对极性进行了粗颗粒的量化处理，具体的量化等级如下所示：

Disgusting: $(-\infty, -10)$; Awful: $[-10, -5)$; Poor: $[-5, -0.5)$; OK: $[-0.5, 0.5)$;
Good: $[0.5, 5)$; Great: $[5, 10)$; Cool: $[10, +\infty)$ 。

经过量化以后，我们认为只要手工标注和自动抽取结果处在同一个等级上，即为匹配。

5.3 测试结果及分析

测试报告包含了测试语料中正确的记录数日，错误的记录数日，遗失的记录数日和冗余的记录数日。由此得到了系统的召回率和准确率。测试报告同时详细记录了每一个错误记录中手工标注和自动抽取的差异；也记录了所有的冗余记录，包括实体词和特征词在汽车本体中的路径，自动提取的极性；还记录了每一条遗失记录。

我们总共手工标注了大约30篇语料，每篇语料的长短不一，短的只有几百字，而长的有六千字。事先已对这些语料进行了手工筛选，即选择含有评价信息较多的语料作为测试语料。由于标注语料需要耗费大量的人力，我们没有进行大规模的测试语料标注。我们选用单篇（一个主题的所有帖子，包括主帖和所有的回帖）测试的方法。由于各篇语料的相对独立性和平行性，这种单篇测试的结果能从一定程度上反映系统对整个语料集的适应度。系统对单篇语料的测试结果，召回率通常在80%左右，精确率在60%左右。

影响系统精确率的主要因素是：极性量化等级的颗粒度和极性匹配规则。通过粗化极性颗粒度和精化极性匹配规则可以提高系统的精确率。举一个极端的例子，如果最后只对评价分为褒义和贬义两个等级，那么系统的精确率可以上升到80%以上。由于系统的极性匹配规则是建立在手工标注基础上的半自动生成方法，而标注又只有正例没有反例，所以匹配规则的生成是完全建立在正例基础上的。如果我们能有对应的反例，对每个匹配规则赋予一定的置信度，在最终匹配的时候考虑这些置信度因素，将有可能改善系统的性能，这部分工作我们准备继续尝试。

影响系统召回率和冗余度的主要因素都是极性匹配规则，解决这两个问题最重要的方法仍是优化匹配规则，考虑匹配规则置信度因素。

6 结论

本文介绍了一个用于汉语汽车论坛的意见挖掘系统的设计和实现方法。系统的测试结果显示了系统总平均召回率达到了80%，总平均精确率达到了60%。因此，这种用于汉语汽车评论的意见挖掘系统是合理和有效的。

这项研究对于汉语主观性文本处理是一次有益的探索，特别是在文本各个层次处理所涉及的词汇、句子和文本极性和强度判断以及它们之间关系的分析方面。

归纳本系统的特点如下：

- 借助于极性词词典，可以有效处理修饰词、特征极性词、异常特征和动态极性词；
- 借助于汽车本体，可以有效处理实体和实体、实体和特征以及特征和特征之间的关系；
- 可以有效处理句子中实体、特征、极性词之间多对多关系；
- 可以有效处理缺省主题的文本；
- 可以有效地处理代词和成分缺省的文本；
- 系统具有很好地领域适应性等。

当然，这个系统还存在许多不完善的地方，如匹配规则的训练只针对正例匹配规则而没有考虑反例匹配规则；对于语料的格式还是有一定的要求，且需要大量的人工标注语料，实用性不强；对于动态极性的问题没有提出彻底的解决办法；性能还稍显不足等。

下一步的系统完善可以考虑以下几个方面：

- 改进极性词词典，将动态极性的概念完全作为极性词的基本属性来对待；
- 进一步强化本体的作用，使之不仅仅作为领域实体和特征的上下位关系的模型；
- 对依存对类型进行抽象，使系统能够处理Deparser分析错误的情况；
- 增加系统的预处理功能，使文本在进入语法分析阶段前尽量规范化（这一点对于论坛一类的以口语为主的语料尤其重要）；
- 添加领域无关的常用词语库（例如：论坛常用词语库）；
- 提高分词系统的准确率，如采用机器学习的方法对错误进行训练。

感谢： 哈尔滨工业大学信息检索研究室为我们的研究工作提供了汉语句法分析器Deparser。

参考文献

- [1] A.-H. Tan. 1999. Text Mining: The state of the art and the challenges. In Proc. of the PAKDD 1999 Workshop on Knowledge Discovery from Advanced Databases, pages 65-70. Beijing, China.
- [2] 吴欣明, 李春伟. 文本挖掘探析. 《廊坊师范学院学报》, 2004年, 20 (4), P54-56.
- [3] J. Wiebe. 2000. Learning Subjective Adjectives from Corpora. In Proc. of AAAI-00, 17th Conference of the American Association for Artificial Intelligence, pages 735-740. Austin, USA.

- [4] E. Riloff, J. Wiebe, and T. Wilson. 2003. Learning subjective nouns using extraction pattern bootstrapping. In Proc. of CONLL-03, 7th Conference on Natural Language Learning, pages 25-32. Edmonton, Canada.
- [5] M. Baroni, and S. Vegnaduzzo. 2004. Identifying Subjective Adjectives through Web-based Mutual Information. In Proc. of KONVENS-04, pages 17-24. Vienna, Austria.
- [6] B. Tsou et al. 2005. Polarity Classification of Celebrity Coverage in the Chinese Press. In Proc. of the International Conference on Intelligence Analysis. McLean, USA.
- [7] L. Dini and G. Mazzini. 2002. Opinion Classification Through Information Extraction. In A. Zanasi, C. Brebbia, N. Ebecken and P. Melli (eds.): Data Mining III, pages 299-310. WIT Press. Southampton, UK.
- [8] H. Yu and V. Hatzivassiloglou. 2003. Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences. In M. Collins and M. Steedman (eds): Proc. of EMNLP-03, 8th Conference on Empirical Methods in Natural Language Processing, pages 129-136. Sapporo, Japan.
- [9] J. Yi and W. Niblack. 2005. Sentiment Mining in WebFountain. In Proc. ICDE-05, the 21st International Conference on Data Engineering, pages 1073-1083, IEEE Computer Society. Tokyo, Japan.
- [10] C. Whitelaw, N. Garg, and S. Argamon. 2005. Using Appraisal Taxonomies for Sentiment Analysis. In Proc. of MCLC-05, the 2nd Midwest Computational Linguistic Colloquium. Columbus, USA.
- [11] S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushima. 2002. Mining product reputations on the Web. In Proc. of KDD-02, 8th ACM International Conference on Knowledge Discovery and Data Mining, pages 341-349. Edmonton, Canada.
- [12] R. Agrawal, S. Rajagopalan, R. Srikant, and Y. Xu. 2003. Mining Newsgroups Using Networks Arising From Social Behavior. In Proc. of the 12 WWW Conference, pages 529-535. Budapest, Hungary.
- [13] M. Gamon, A. Aue, S. Corston-Oliver, and E. Ringger. 2005. Pulse: Mining Customer Opinions from Free Text. In Proc. of IDA-05, the 6th International Symposium on Intelligent Data Analysis. Lecture Notes in Computer Science, Springer-Verlag. Madrid, Spain.
- [14] 邱立坤, 程薇, 龙志伟, 孙娇华. 面向BBS的话题挖掘初探. 见: 孙茂松, 陈群秀主编《自然语言理解与大规模内容计算》. 清华大学出版社. 2005年. 401-407页.
- [15] P. D. Turney and M. L. Littman. 2003. Measuring Praise and Criticism: Inference of Semantic Orientation from Association. ACM Transactions on Information Systems (TOIS), Vol. 21, No. 4, pages 315-346.
- [16] R. Yuan et al. 2004. Morpheme-based Derivation of Bipolar Semantic Orientation of Chinese Words. In Proc. of the 20th International Conference on Computational Linguistics (COLING-2004), pages 1008-1014. Geneva, Switzerland.
- [17] 朱嫣岚, 闵锦, 周雅倩, 黄萱菁, 吴立德. 基于HowNet的词汇语义倾向计算. 《中文信息学报》, 2006年, 20(1), 14-20页.
- [18] L. Dini and G. Mazzini. 2002. Opinion Classification Through Information Extraction. In A. Zanasi, C.

Brebbia, N. Ebecken and P. Melli (eds.): Data Mining III, pages 299-310, WIT Press, Southampton, UK.

- [19] T. Wilson, J. Wiebe, and P. Hoffmann. 2005. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In Proc. of Human Language Technologies Conference/Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005), pages 347-354. Vancouver, Canada.
- [20] B. Liu, M. Hu, and J. Cheng. 2005. Opinion observer: analyzing and comparing opinions on the Web. In Proc. of WWW '05, the 14th international conference on World Wide Web, pages 342-351. Chiba, Japan.
- [21] Stanford Medical Informatics. 2003. The Protégé Ontology Editor and Knowledge Acquisition System. the School of Medicine, Stanford University. Stanford, USA. (<http://protege.stanford.edu/>.)
- [22] P. J. Stone, D. C. Dunphy, M. S. Smith, and D. M. Ogilvie. 1966. The General Inquirer: A Computer Approach to Content Analysis. MIT Press. (<http://www.webuse.umd.edu:9090/>)
- [23] Dong Z, Dong Q. 2000. HowNet. (http://www.keenage.com/zhiwang/e_zhiwang.html).