

doi:10.3969/j.issn.1007-7162.2014.04.003

基于分类的 ESGS 矛盾问题建模的研究

汪中飞,李卫华,陶 星

(广东工业大学 计算机学院,广东 广州 510006)

摘要: 以往的可拓策略生成系统软件只能解决一类矛盾问题,通过分析已有矛盾问题建模方法及其不足,提出了一种基于分类思想的建模方法,并详细描述了矛盾问题分类建模的步骤,具体给出了分类建模所需库的建立方法,最后在软件中实现.实验结果表明,对某领域可拓策略生成系统中的矛盾问题进行分类,基本能够解决对该领域不同类矛盾问题的建模,实现了可拓策略生成系统问题输入界面的通用性,对可拓策略生成系统通用性的研究具有一定的现实意义.

关键词: 可拓策略生成系统;矛盾问题;分类;建模

中图分类号: TP311

文献标志码: A

文章编号: 1007-7162(2014)04-0014-06

Research on ESGS Contradiction Problem Modeling Based on Classification

Wang Zhong-fei, Li Wei-hua, Tao Xing

(School of Computer Science, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: The previous extension generating system software can only be used to solve some kind of contradiction problems. Through the analysis of the existing problem modeling methods and their shortcomings, it presents a new modeling method, based on the idea of classification. A detailed description of the problem classification modeling steps was given, and finally implemented in programming. Experimental results show that the new modeling method can be used to classify different kinds of contradiction problems in some areas. As a result, it not only creates a universal input interface to the extension strategy generating system, but also has great practical significance for the versatility of research on the Extension Strategy Generating System.

Key words: Extension Strategy Generating System(ESGS); contradiction problem; classification; modeling

目前,可拓学^[1]理论的研究不断完善和成熟,而以其为支撑的可拓策略生成系统^[2]软件的研发相对来说却滞后很多,并且都只是针对某一类矛盾问题.例如,文献[3-6],从这些可拓策略生成系统可以看出,每一个系统只提供给用户求解该系统定义的矛盾问题,界面都固定了矛盾问题所需的属性或者特征.这主要表现出两个方面的不足:一是代码复用性不强,每个系统只解决了该系统已有的矛盾问题;二是界面通用性不强,每个系统都需要单独设计一个该系统支持的界面.文献[7]虽然在代码复用性上做

了初步研究,但实际上还是只能解决一类矛盾问题.

目前,急需利用可拓学理论来设计开发出某个领域的可拓策略生成系统.一个领域里的矛盾问题肯定不止一个,由此看来,以往各种可拓策略生成系统的开发模式肯定是不能满足的.究其原因是界面限制了矛盾问题的输入,即没有通用的矛盾问题输入界面.本文针对这一问题,提出了利用分类的思想,来对某个领域矛盾问题进行分类,从而设计出一个通用矛盾问题输入界面,然后对输入问题进行分类,进而对不同矛盾问题建立形式化的模型,有了矛

盾问题的可拓模型,最后利用已有的可拓策略生成系统通用框架便能很好地解决某个领域的不同矛盾问题.

1 已有建模方法的不足

解决界面通用问题的方法之一是在界面上设计一个与百度搜索类似的文本框,用户输入自己的矛盾问题,即可生成该问题的可拓模型.然而,如何从用户的自然语言中解析出或者挖掘出该矛盾问题的可拓模型成为解决可拓策略生成系统界面通用性的关键.

通过查看文献,发现李卫华教授利用 Agent 技术的通信能力与反应性来智能引导用户输入矛盾问题建立矛盾问题可拓模型^[8].但在文献[8]中并没有给出具体的实现方法,文献[8]中作者以上班交通问题为例,Agent 可以引导用户输入起点、目的地、希望上班时限等,作者再以住房问题为例,Agent 可以引导用户输入承租要求(区域,周围环境,面积等),针对上班交通问题与租房这两种矛盾问题甚至更多其他的矛盾问题,Agent 是如何区别这些不同的矛盾问题的,另外每种矛盾问题要输入哪些属性是如何得来的,作者并没有详细讨论.作者也指出对于过于广泛的问题,由于无法预料全部输入,因此也无法为 Agent 准备有效的应对措施,这个问题仍需要进一步的研究.文献[9]虽然给出了实现 Agent 智能引导的方法,但仍然没有做区分不同矛盾问题的研究.

另外,赵杰在文献[10]中,提出了利用知网的相关知识,结合分词技术来解决可拓策略生成系统对矛盾问题进行界定和建模.作者给出了一个以自然语言描述问题的例子,“我有 2 000 块钱,能不能买一部苹果手机?”然后通过分词系统,结果为:“我/r 有/v2000/m 块/钱/n,能/v 不/d 能/v 买/v 一/m 部/q 苹果/n 手机/n”.最后对分词结果对比基础库中存在的基元.作者只是描述到了这里,但没有深入论述如何得到所需要的基元,如何建立可拓模型.

2 基于分类思想的建模方法

2.1 矛盾问题分类思想的产生

上述系统的设计者们一般都是知道某些地方存在矛盾问题,并以此设计出矛盾问题模型,同时也收集相关的数据来支撑可拓变换形成最后的决策.

由此,要设计出在某个领域较通用的可拓策略生成系统,应该事先收集该领域可能存在的矛盾问

题,并且把收集起来的问题进行分类,然后对每类问题建立一个解决该类问题的问题可拓模型,同时广泛收集每类问题相关的数据.

2.2 利用分类思想对矛盾问题进行建模

2.2.1 分类建模的流程

在上一节中提出了分类建模的思想,具体对矛盾问题分类建模的流程如图 1 所示.其步骤为:

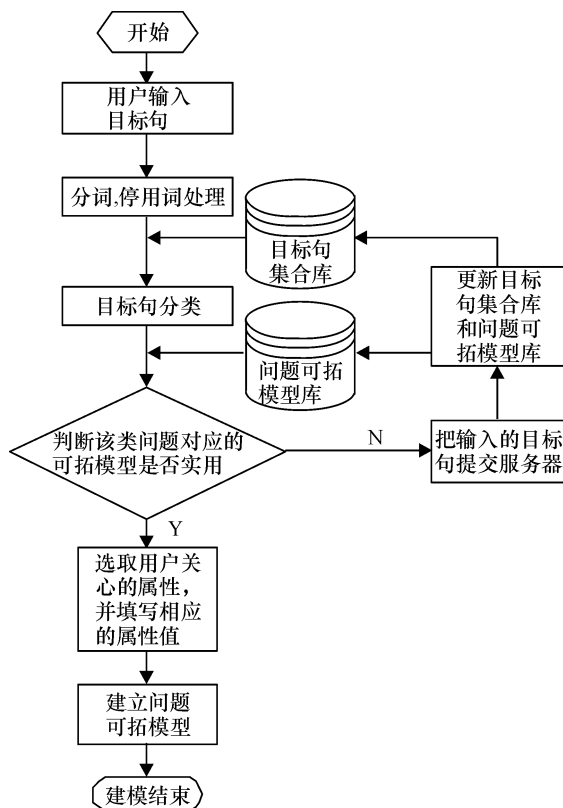


图 1 分类方法建模流程图

Fig. 1 Flow diagram of classification modeling

(1) 用户在通用界面输入自己想要达到的目标语句.例如教育领域中的高考志愿问题,用户只需简单输入“我想报考清华”,而不必要把目标和条件都详细地描述,如“我理科考了 620 分,想报考清华大学,清华大学去年的理科录取线是 630 分”.由于大多数情况用户是不知道具体条件是什么的,如用户可能不知道清华去年录取线是多少,用户只关心自己的分数能不能被清华录取.而且根据对以往可拓策略生成系统的分析知道,条件数据一般是存储在数据库中的.

(2) 对用户输入的目标句进行分词,并且去掉停用词^[11-13].

(3) 把(2)处理的结果与目标句集合库进行比对,判断用户输入目标句所属类型.

(4) 系统界面将从问题可拓模型库中调用目标

句所属类型的目标 G 的模型.

(5) 用户判断该模型是否适用,如果适用则选择自己关心属性并填入相应的属性值,如果不适用则提交给服务器,相关人员再分析是否需要更新目标句集合库和问题可拓模型库.

(6) 用户填入相关属性值后,便可生成目标模型,然后搜索后台数据库便可找到相应的条件,从而可以建立矛盾问题的问题模型.

(7) 建立好了问题模型即可通过文献[7]的通用框架解决某领域的各种矛盾问题.

2.2.2 分类建模需要做的工作

在上述流程中,除了要实现上面的 7 个步骤外还需要做以下 3 个方面的工作.

1) 目标句集合库及矛盾问题模型库的构建

对目标句的收集主要可以公开目标句输入接口,让用户输入目标句以此来采集目标句,其次可以做调查问卷来收集.对采集好的目标句子进行筛选、优化、分类,然后存入目标句集合库.与此同时,对于每类问题,由该类问题专家分析归纳出该类问题用户一般关心的属性,然后就可以对该类问题进行建模.另外,在数据库中也需存储该类问题相关的数据,以支持矛盾问题的可拓变换.

例如,对于教育领域目标句的采集,采集的目标句为“我想报考广工”、“我的志愿想填报广工”等等,然后判断出这些目标句属于报考类.该类问题的专家可以据此总结出报考矛盾问题,可拓模型为:

$$P = G * L = \left[\begin{array}{cc} \text{报考, 支配对象, 某高校} & \\ & \text{施动对象, 学生 } M \end{array} \right] * \left[\begin{array}{cc} \text{某高校, 理科分数线, } v1 & \\ & \text{文科分数线, } v2 \\ & \text{年份, } v3 \\ & \text{地区, } v4 \\ & \text{批次 } v5 \\ \dots & \dots \end{array} \right]$$

其中,

$$M = \left[\begin{array}{cc} \text{某学生, 报考学校, } v1 & \\ & \text{文科或理科分数, } v2 \\ & \text{报考年份, } v3 \\ & \text{报考地区, } v4 \\ & \text{报考批次 } v5 \\ \dots & \dots \end{array} \right]$$

因此,该类问题专家进而可以归纳出该类问题用户关心的一些属性,比如报考学校、高考分数、报考批次等等,并且在数据库中存储所有高校相应的数据.

然而,建立好的问题模型并不一定能满足该类

问题的所有矛盾,这就需要对模型进行更新添加新的属性.例如报考类,用户不一定是高考志愿填报,还有可能是研究生志愿填报,或者其他的报考矛盾问题等等.对于这种情况,解决并不难,因为报考矛盾的整体模型框架是一定的,只需要该类问题专家添加新的用户关心的属性.不同用户根据不同的情况选取自己关心的属性并赋值即可.

2) 领域知识库的构建

文献[14]中提出计算机理解自然语言的能力非常低,例如“苹果”这个词,计算机不知道苹果到底是水果还是手机,因此可以利用知网^[15]来对自然语言消除歧义,文献[10]也对此作了很好的应用.然而知网只是针对通用的知识,并没有某个特有领域的知识,专门领域的知识需要专门领域的专家来研究和建设^[15].因此做某一个领域的可拓策略生成系统遇到自然语言语义处理而通用知网知识不能满足时就需要本领域的知识库.例如,教育领域,广工或者广东工业大学知网是没有定义的,这就需要我们自己添加,构建出教育领域的的知识库.

NO. = 1000000
W_C = 广工
G_C = N
E_C = ~ 很大, ~ 教学质量不错
W_E = Guangdong University of Technology
G_E = N
E_E =
DEF = InstitutePlace | 场所, @ teach | 教, @ study | 学, education | 教育, ProperName | 专, (China | 中国)

3) 目标句的处理

对用户输入的目标句处理主要包括对输入句子的分词处理、停用词过滤处理、目标句分类处理,其中最重要的就是对目标句分类处理.本文主要采用朴素贝叶斯算法^[16-17]进行分类.其分类步骤如下:

(1) 设定目标句类别集合 $T = \{t_1, t_2, \dots, t_n\}$, 其中 n 为目标句类型个数.

(2) 根据用户输入的目标句 g 来判断目标句 g 所属的类型,即求解 $P(t_i | g)$, 其中 $i = (1, 2, \dots, n)$, 选择其中最大的 $P(t_i | g)$ 所对应的 t_i , 则目标句 g 的类别就是类别 t_i . 计算公式如下^[16]:

$$P(t_i | g) = \frac{P(g | t_i) P(t_i)}{P(g)} \tag{1}$$

(3) 在比较不同类别后验概率时,分母 $P(g)$ 相当于一个常数,可以不参与计算.所以我们只要求解分子 $P(g | t_i) P(t_i)$ 即可,其中 $P(t_i)$ 为先验概率,是容易求得的,这里采用多项式模型进行计算^[17]:

$$P(t_i) = \frac{wCount_i}{wCount}, \tag{2}$$

其中, $wCount_i$ 为目标句集合库中类 t_i 下单词出现的总数, $wCount$ 为目标句集合库中所有单词总数.

(4) 求解类条件概率 $P(g | t_i)$. 因为采用的是朴素贝叶斯分类器, 可以认为目标句 g 的各个关键词是条件独立的, 所以求解 $P(g | t_i)$ 又可以转换成求解 $P(w_j | t_i)$, 其中 w_j 为目标句中的某个关键词, 计算公式如下^[16-17]:

$$P(g | t_i) = P(w_1 | t_i)P(w_2 | t_i) \cdots P(w_m | t_i) \\ = \prod_{j=1}^m P(w_j | t_i). \tag{3}$$

(5) 最后求解 $P(w_j | t_i)$. 设 $wTCount_j$ 为目标句在类别 t_i 下时关键词 w_j 出现的次数, $wTCount$ 为目标句在类别 t_i 下时所有关键词出现的次数之和, 所以有^[17]:

$$P(w_j | t_i) = \frac{wTCount_j}{wTCount}. \tag{4}$$

(6) 当类别 t_i 下时关键词 w_j 出现的次数为 0 时, 即此时类条件概率为 0, 因此是无法分类的, 为了解决这一问题可以采用一种估计概率的贝叶斯方法, 即 $m -$ 估计, 因此式(4)可以修改为^[16]:

$$P(w_j | t_i) = \frac{wTCount_j + 1}{wTCount + |Vocabulary|}, \tag{5}$$

其中, $Vocabulary$ 是目标句集合库中的关键词表, 即对集合库中所有句子分词去停用词后所有的关键词, 重复单词只算一次. $|Vocabulary|$ 则表示关键词表中词语的个数.

因此, 目标句分类即可以得到解决.

3 领域矛盾问题分类建模通用输入界面的实现

3.1 开发环境

本系统的开发环境为:

- 1) 操作系统: windows 7 32 位;
- 2) 开发平台: MyEclipse 8.5;
- 3) JDK 版本: JDK 7.0;
- 4) 数据库服务器: Microsoft SQL Server 2000.

3.2 数据库及各模型的建立

3.2.1 目标句集合库的建立

通过对教育领域目标句的采集, 然后进行筛选和分类, 存入数据库, 作为训练的样本. 实验时采集了两类数据, C001(报考类)和 C002(求职类), 每类分别有 10 条语句, 然后存入目标句集合表 $tb_gSentencesSet$ 中, 如图 2 所示. 其中字段 $GSentences$ 是目标句, $CType$ 是目标句所属类.

Id	GSentences	CType
1	我想报考广东工业大学.	C001
2	我想报考广工.	C001
3	我志愿填报华工.	C001
4	我想填报北大的志愿.	C001
5	高考志愿我想填报武汉大学.	C001
6	我高考分数580分, 能不能填报广东工业大学.	C001
7	我的分数只有480分, 能读广工的本科吗?	C001
8	我考了580分, 能否过华工的分数线吗?	C001
9	我是文科生, 考了540分, 能读华师吗?	C001
10	我是理科生, 高考510分, 能报清华吗?	C001
11	我想找工作.	C002
12	我想找软件开发的工作.	C002
13	我想从事计算机方面的工作.	C002
14	我想找广州的工作.	C002
15	我打算去国企上班.	C002
16	我本科毕业能去国企上班吗?	C002
17	我想求得薪水在5000左右的职位.	C002
18	我想进腾讯上班.	C002
19	我想应聘华为的职位.	C002
20	我想求得高薪职位.	C002

图2 目标句集合表 $tb_gSentencesSet$
Fig.2 Table of the goal sentence set

3.2.2 问题可拓模型库的建立

问题可拓模型库由若干表组成, 其中有个总表 $tb_ProblemModelSet$, 主要用于存放每类问题目标模型和条件模型的表名, 如图 3 所示. 其中字段 $CType$ 为问题所属类型编码, 字段 $CName$ 为类型对应的中文说明, 字段 $GModel$ 为目标模型存放的表名, 表名格式为 $tb_类型代码_G$, 字段 $LModel$ 为条件模型存放的表名, 表名格式为 $tb_类型代码_L$, 字段 $gBasicElementName$ 与 $lBasicElementName$ 分别表示默认目标基元名与条件基元名, 基元名是可以根据用户需要改变的.

Id	CType	CName	GModel	LModel	gBasicElementName	lBasicElementName
1	C001	报考类	tb_C001_G	tb_C001_L	学生	学校
2	C002	求职类	tb_C002_G	tb_C002_L	求职学生	职位

图3 问题可拓模型总表 $tb_ProblemModelSet$
Fig.3 Table of the problem extension model set $tb_ProblemModelSet$

目标模型表是不需要数据的, 只是存放目标基元所需的属性. 此处以报考类目标基元表 tb_C001_G 为例, 如图 4 所示. 其中字段 $AttributeName_E$ 表示为属性英文名, $AttributeName_C$ 表示为属性中文名, 字段 $isCompoundElement$ 表示是否为复合元. 此处为了简化说明, 仅仅讨论非复合元. 对于复合元只需再建立一张表, 用户选择属性时, 对复合元的属性再做选择即可.

AttributeName_E	AttributeName_C	isCompoundElement
SchoolName	报考学校	0
ScienceScores	理科分数	0
LiberalArts	文科分数	0
SchoolProvince	报考省	0
SchoolCity	报考市	0
SchoolRank	学校批次	0
StudentProvince	考生所在省	0
StudentCity	考生所在市	0

图4 报考类目标基元表 tb_C001_G
Fig.4 Table of Basic-Element for enter oneself for an examination goal tb_C001_G

前面已经讨论过 ESGS 解决每类问题时是需要该类问题大量相关数据的,矛盾问题的条件往往又存在于这些数据中. 因此条件基元表不仅仅表现出一个模型,而且还有大量的数据. 此处接着以上述报考类目标基元对应的条件基元表 tb_C001_L 为例,如图 5 所示. 该表的每个字段名都来自目标模型的属性,即用户一般关心的属性,对于这些属性后台管理员需要大量收集相关数据以支持可拓变换. 另外这些属性在界面上需要中文表示,而目标基元表中的属性中文名不一定适合,此处解决方法可以利用 Microsoft SQL Server2000 设计表时给每个字段加上中文说明.

Id	SchoolName	ScienceScores	Literalscores	SchoolProvince	SchoolCity	SchoolRank	StudentProvince	StudentCity
1	广工	548	557	广东省	广州市	一本	广东省	<未知>
2	清华	650	640	<未知>	北京市	一本	北京市	<未知>
3	华工	630	620	广东省	广州市	一本	广东省	<未知>

图 5 报考类条件基元表 tb_C001_L

Fig.5 Basic-Element table of the conditions for entering oneself for an examination in tb_C001_L

3.2.3 目标句集合库的训练模型

由于建立好的目标句集合库在一段时间内是不会修改变动的,为了不让每次分类都重复计算先验概率、每个词的条件概率等,可以把这些重要的而且每次计算都固定的数值存放到一个模型文件中,以便提高分类的速度. 此处利用 Java 的 ObjectOutputStream.writeObject 方法将如图 6 所示相关数据存放至 classData.mdl 文件中.

```
public class TrainedModel implements Serializable {  
  
    public String[] classifications;//类别名  
    HashSet<String> vocabulary; //目标句集合库中所出现过的单词 (不重复)  
    private double[] pC;// 类别的先验概率  
    @SuppressWarnings("unchecked")  
    private HashMap[] pXC;// 每个词的条件概率
```

图 6 训练模型中存放的数据

Fig.6 The data stored in the training model

3.3 案例演示

通过设计和 Java Swing 编程技术^[18]基本实现了矛盾问题分类建模软件界面通用. 现以教育领域为例来演示如何通过通用界面实现矛盾问题建模的.

(1) 输入目标句,生成目标模型. 例如输入“我想报考广工”,点击“确定”按钮,就会对目标句类别进行判断,归类后调用该类别的目标模型,如图 7 所示. 此处目标句的输入在界面可以给出目标句的例子,也可以利用目标句集合库进行联想输入.

(2) 判断模型是否满足,如不满足点击“模型不满足”按钮,此时将会把用户输入的目标句提交到数据库,管理员再判断是否需要更新模型库. 如满足则用户选择自己关心的属性并填写相应属性值,点击“=>”按钮将删除目标模型中没用的属性,即生

成目标的关键属性,也就是评价特征. 如图 8 所示. 这里输入属性值时可以利用文献[9]界面 Agent 技术引导用户输入. 如果用户需要添加新属性时,可以点击“点击这里添加新属性”按钮,将会弹出如图 9 所示对话框. 其添加方法参考文献[19].



图 7 输入目标句生成目标模型界面

Fig.7 Interface of goal model generating after entering the goal sentence

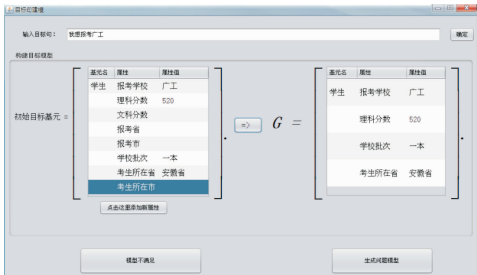


图 8 生成关键属性界面

Fig.8 Interface of generate key attributes



图 9 添加新属性界面

Fig.9 Interface of adding a new feature

(3) 建立问题可拓模型. (2) 中评价特征选好赋值后点击“生成问题模型”按钮,后台将根据所选评价特征在条件模型表中搜索相关的条件,并把问题可拓模型显示在界面上,如图 10 所示.

至此,一个问题模型成功建立. 如果要判断该问题是否矛盾,必须建立关联函数后才能计算相容度,关联度小于零就是矛盾问题.



图 10 生成问题模型界面

Fig. 10 The generated problem model interface

可以在同一输入界面建立该领域的另一个模型,如求职问题.按照上述相同步骤,在界面输入“我想求得一份Java程序员的工作”,然后生成目标模型,用户选择其关心的属性并填写相应属性值,生成关键属性模型,如图 11 所示.最后生成该问题的问题模型,如图 12 所示.



图 11 求职问题生成关键属性界面

Fig. 11 Key attributes interface of job problem generating



图 12 生成求职问题模型界面

Fig. 12 Model interface of job problem generating

4 结论

本文利用分类的思想对可拓策略生成系统中的矛盾问题进行分类,从而解决了可拓策略生成系统问题输入界面的通用性,这有重要的现实意义.(1)本研究再结合通用的 ESGS 框架,便能实现可拓策略生成系统的通用性,这使得软件开发者不再在软件编码和界面设计上耗费时间;(2)利用分类的思想还能对 ESGS 需要检索的地方提高检索效率.例如对做问题相似度计算时,利用分类思想可以节省大量计算时间;(3)利用分类思想还能加强对系统数据的有效管理.

另外,本文重点论证了利用分类思想能够解决 ESGS 问题输入界面通用性的可行性,对于选取何种分类算法及如何提高分类性能,如何建立领域知识库以及问题可拓模型库中属性如何用最有效的方法获取等问题,仍需要进一步研究.

参考文献:

[1] 杨春燕,蔡文. 可拓工程[M]. 北京:科学出版社,2007.

[2] 李立希,杨春燕,李铨汶. 可拓策略生成系统[M]. 北京:科学出版社,2006.

[3] 方卓君,李卫华,李承晓. 自助游可拓策略生成系统的研究与实现[J]. 广东工业大学学报,2009,26(2):83-89.

Fang Z J, Li W H, Li C X. Research and realization of extension strategy generating system for independent travel[J]. Journal of Guangdong University of Technology, 2009, 26(2): 83-89.

[4] 李承晓,李卫华. 租房可拓策略生成系统[J]. 智能系统学报, 2011,6(3):272-278.

Li C X, Li W H. Research on a tenement extension strategy generation system[J]. Transactions on Intelligent Systems, 2011,6(3): 272-278.

[5] 陈亚男. 解决求职矛盾的可拓策略生成系统研究[D]. 广州:广东工业大学计算机学院,2012.

[6] 许阳. 面向企业人才流失问题的可拓策略生成方法研究[D]. 广州:广东工业大学计算机学院,2012.

[7] 李承晓. 可拓策略生成系统框架设计与复用研究[D]. 广州:广东工业大学计算机学院,2011.

[8] 李卫华. Agent 协助建立矛盾问题的可拓模型研究[J]. 数学的实践与认识, 2009,39(4):173-177.

Li W H. The research of extension model for contradiction problems with agent[J]. Mathematics in practice and theory, 2009, 39(4): 173-177.

[9] 方卓君. 基于 Agent 的用户信息智能引导的研究[D]. 广州:广东工业大学计算机学院,2010.

[10] 赵杰,李卫华. 基于知网的矛盾问题语义二义性研究[J]. 广东工业大学学报, 2014,31(2):1-6.

Zhao J, Li W H. Research on the ambiguity of contradiction base on hownet[J]. Journal of Guangdong University of Technology, 2014, 31(2): 1-6.

[11] 刘件,魏程. 中文分词算法研究[J]. 微计算机应用, 2008,29(8):11-16.

Liu J, Wei C. Arithmetic research on Chinese segmentation[J]. MicroComputer Applications, 2008, 29(8):11-16.