# 一种基于自然语言的模式推理算法*)

王树西　赵星秋　刘瑞林　黄健青
（对外经济贸易大学信息学院　北京 100029）

**摘　要**　传统的基于谓词模式推理算法，需要把自然语言表示的知识，人工转换为机器可以理解的谓词，这就需要耗费大量的人力物力。本文提出一种基于自然语言的模式推理算法，可以基于自然语言进行模式推理，不需要将自然语言表示的知识转换为谓词，从而大大节省了人力物力。实验结果表明，本算法可以基于自然语言，有效的进行模式推理。

**关键词**　自然语言处理，模式合一，模式推理

## A Pattern Inference Algorithm Based on Natural Language

WANG Shu-Xi　ZHAO Xing-Qiu　LIU Rui-Lin　HUANG Jian-Qing
(Information Academy of the University of International Business and Economics, Beijing 100029)

**Abstract**　Traditional parrern inference algorithm is based upon predication. Traditional algorithm need to change natural language into predicate, which is understood by the computing system. So traditional algorithm need plentiful manpower and material resources. This paper proposes a pattern inference algorithm which is based upon natural language. The new algorithm need not to change natural language into predicate. So the new algorithm save plentiful manower and material resources. Experiment result indicates that the new algorithm is very successful.

**Keywords**　Natural language processing, Pattern unification, Pattern inference

模式推理，又称为推理[1]。“推理”作为动词，是描述依据事实库和规则库，通过一定的推理机制，从前提出发，产生一个或者多个推理结果的一系列活动；作为名词，是“〈推理结果，依据〉”二元组系列，每一个推理结果，或者是事实，或者是规则作用于事实、先前的推理结果之后，而得到的推理结果。后一种意义上的“推理”，也叫做“证明”[3]。传统的模式推理算法（例如 Prolog、Lisp 等人工智能语言的推理机制），是针对谓词进行推理，需要把自然语言表示的知识，人工转换为机器可以理解的谓词。所以，传统的模式推理算法，需要耗费大量的人力物力，是需要改进的[2,4]。

本文就试图对传统的模式推理算法进行改进。文中提出一种基于自然语言的模式推理算法，可以基于自然语言进行模式推理，而不需要将自然语言表示的知识转换为谓词，从而大大节省了人力物力。实验结果表明，本算法可以基于自然语言，有效地进行模式推理。

本文的结构如下：首先提出一系列相关的定义；然后提出基于自然语言的模式推理算法，并对算法进行了改进；实验结果表明，本算法可以基于自然语言，有效地进行模式推理；最后指出了下一步的工作方向。

## 1　相关定义

**①联立合一**　一个模式 $n$ 元组$\langle P_1, P_2, \cdots, P_n\rangle$是另一个模式 $n$ 元组$\langle Q_1, Q_2, \cdots, Q_n\rangle$的联立细化，如果存在一个代换 $n$ 元组$\langle x_1/y_1, x_2/y_2, \cdots, x_n/y_n\rangle$($n\geqslant 1, x_i\in V, y_i\in(\sum\cup V)^*$，$1\leqslant i\leqslant n$)，使得把 $Q_1$，$Q_2$，…，$Q_n$ 中的所有 $x_i$ 的出现，都同时替换成 $y_i$，结果恰好得到 $P_1, P_2, \cdots, P_n$($1\leqslant i\leqslant n$)。两个模式 $n$ 元组$\langle P_1, P_2, \cdots, P_n\rangle$和$\langle Q_1, Q_2, \cdots, Q_n\rangle$的公共联立细化，称为它们的联立合一[8]。

**②变量代换**　又称为“代换”，是指将模式中的变量 $x$，替换成字符串 $\alpha$，其它字符保持不变。变量代换可以形式化表示为 $\alpha/x$，其中，$\alpha\in(\sum\cup V)^*$，$x\in V$。一系列的代换，构成形如$\{\alpha_1/x_1, \alpha_2/x_2, \cdots, \alpha_n/x_n\}$的有限集合，其中，$\alpha_1, \alpha_2, \cdots, \alpha_n$ 是字符串，又称为“项”，$x_1, x_2, \cdots, x_n$ 是互不相同的变量，$\alpha_i/x_i$ 表示用 $\alpha_i$ 替换 $x_i$[7]。

**③模式推理规则**　又称推理规则，是指为了进行模式推理，所必需的推理依据，一条模式推理规则，包括前件和后件两大部分。推导符号（“→”）前面的部分，称为前件；推导符号右面的部分，称为后件。构成前件的模式，称为前提；构成后件的模式，称为结论。在推理规则中，前提和结论都是命题，模式推理是由前提推出结论。一般来说，一条规则中，前提的个数$\geqslant 0$，结论只有 1 个[5,6,9]。

**④事实库**　又称知识库。是指由预先给定的事实，所构成的集合。

**⑤规则库**　又称推理规则库。是指由预先给定的模式推理规则，所构成的集合。

**⑥目标模式**　对用户查询中的疑问词，进行变量代换之后，所构成的新的模式。例如，对用户查询“《祝福》的作者是谁”进行变量代换，用变量“X”，代换用户查询中的疑问词“谁”，得到目标模式“《祝福》的作者 X”。

*)本文有关研究得到“中俄经贸合作网”项目资助。**王树西**　博士，讲师，主要研究领域为计算语言学等；**赵星秋**　副教授，硕士研究生导师，主要研究领域为人工智能、数据库等；**刘瑞林**　副教授，硕士研究生导师，主要研究领域为电子商务、数据挖掘等；**黄健青**　副教授，硕士研究生导师，主要研究领域为电子商务等。

## 2 基于自然语言的模式推理算法

### 2.1 算法的输入

①事实库(表示为 sFactBase)。对事实库、规则库进行预处理,得到事实库中事实的个数(计作 iFactNum);

②规则库(sRuleBase)。规则库中规则的个数(iRuleNum);每条事实、其中字的个数、每个字的偏移量(数组 aFact[]);每条规则左部中的每个模式、模式个数(计作数组 aRuleLeft[]);每条规则右部、其中字的个数、每个字的偏移量(计作数组 aRuleRight[])。

③目标模式(表示为 sPattern)。

### 2.2 算法的输出

①判断目标模式 sPattern 能否推理成功。如果是,则返回 true;否则,返回 false。

②如果目标模式 sPattern 推理成功,那么得到模式推理的结果。

③如果目标模式 sPattern 推理成功,那么可能有多个模式推理的结果。

### 2.3 具体算法

①两个模式之间进行合一。根据双模式合一的"减首去尾"算法,将目标模式 sPattern,分别与事实库中的每一条事实 sFact,进行"双模式合一"的计算。

②如果 sPattern 与 sFact 合一失败,那么,转①。

否则 sPattern 与 sFact 合一成功,目标模式 sPattern 匹配事实库成功,得到合一结果 sUnifyP(sFact),并作为目标模式 sPattern 的一个模式推理结果,加入到目标模式 sPattern 的模式推理结果数组 aDeduceResult[]。转①。

③如果目标模式 sPattern 与事实库中所有事实之间的双模式合一计算全都失败,那么 sPattern 匹配事实库失败。

④根据双模式合一的"减首去尾"算法,将目标模式 sPattern,分别与规则库中每一条规则(不妨令其为 $R_j, j\geqslant 1$)的右部,进行双模式合一的计算。

⑤如果双模式合一计算失败,那么转④。

否则双模式合一计算成功。得到合一结果 sRuleRightUnify,并根据规则 $R_j$ 右部、合一结果 sRuleRightUnify,对规则 $R_j$ 中的所有变量,分别进行变量代换。

不妨令变量代换之后,规则 $R_j$ 变为 $R_j'$。分别得到规则 $R_j'$ 的所有 $n$ 个前提:$SG_1, SG_2, \cdots, SG_n (n\geqslant 1)$,作为规则 $R_j'$ 派生出来的 n 个子目标。

⑥分别对每一个子目标 $SG_i (n\geqslant i\geqslant 1)$,进行模式推理的计算。

⑦如果其中一个子目标 $SG_i$ 模式推理失败,那么目标模式 sPattern 匹配规则 $R_j$ 失败,转④。否则所有子目标全都模式推理成功,目标模式 sPattern 匹配规则 $R_j$ 成功,从而目标模式 sPattern 匹配规则库成功,并分别得到每个子目标 $SG_i$ 的模式推理结果。

⑧分别考察每一个子目标 $SG_i$。不妨令 $SG_i$ 有 $m$ 个模式推理结果:$PR_1, PR_2, \cdots, PR_m$。

分别考察子目标 $SG_i$ 的每一个模式推理结果 $PR_j (m\geqslant j\geqslant 1)$。

根据子目标 $SG_i$ 与它的模式推理结果 $PR_j$,得到 $SG_i$ 中每个变量所对应的字符串。将这种变量代换方式,施加到其它子目标 $SG_k (n\geqslant k\geqslant 1, k\neq i)$,$SG_k$ 变为 $SG_k'$。

$SG_k'$ 与子目标 $SG_k$ 的所有模式推理结果,分别进行模式合一的计算。

如果 $SG_k'$ 与 $SG_k$ 的所有模式推理结果全都模式合一失败,那么从 $SG_i$ 的模式推理结果中,去除 $PR_j$。否则 $SG_k'$ 与 $SG_k$ 的其中一个模式推理结果模式合一成功,将 $SG_i$ 的模式推理结果 $PR_j$ 保留。

⑨考察每一个子目标 $SG_i$ 及其模式推理结果。如果 $SG_i$ 的模式推理结果的个数为 0,那么子目标 $SG_i$ 模式推理失败,从而目标模式 sPattern 匹配规则 $R_j$ 失败,转④。

⑩根据每一个子目标 $SG_i$ 及其模式推理结果,得到目标模式 sPattern 与规则 $R_j$ 的模式推理结果,具体方法如下:

建立一个新的"规则右部推理结果数组",用来存储目标模式 sPattern 与规则 $R_j$ 的模式推理结果,并记作 aRuleRightDeduceResult[]。建立一个字符串存储器,用来存储模式推理的中间结果。

考察第一个子目标 $SG_1$ 及其所有模式推理结果。

考察 $SG_1$ 的每一个模式推理结果 $PR_i$。将 $SG_1$ 与 $PR_i$ 进行模式合一计算,从而得到 $SG_1$ 中每个变量所对应的字符串。

将这种变量代换方式,施加到合一结果 sRuleRightUnify,然后添加到"规则右部推理结果数组";并将这种变量代换方式,分别施加到其它的子目标 $SG_2, SG_3, \cdots, SG_n$,得到 $SG_2', SG_3', \cdots, SG_n'$。

分别考察 $SG_2', SG_3', \cdots, SG_n'$,以及子目标 $SG_2, SG_3, \cdots, SG_n$ 的所有模式推理结果。

如果 $SG_i' (n\geqslant i\geqslant 2)$,与 $SG_i$ 的模式推理结果 PR 合一成功,那么根据 $SG_i'$ 和 PR,得到 $SG_i'$ 中每个变量所对应的字符串,并将这种变量代换方式,分别施加到"规则右部推理结果数组"中的每一个元素,然后将结果加入存储器。

将存储器中的所有元素取出,代换"规则右部推理结果数组",从而得到新的"规则右部推理结果数组"。

分别考察"规则右部推理结果数组"中每一个元素 sRuleRightDeduceResult,以及目标模式推理结果数组 aDeduceResult[]。

如果 sRuleRightDeduceResult 是常量字符串,并且不在数组 aDeduceResult[]中出现,那么,将 sRuleRightDeduceResult 加入到数组 aDeduceResult[]中。

转④。

⑪如果目标模式 sPattern 匹配所有的规则失败,那么目标模式 sPattern 匹配规则库失败。

如果目标模式 sPattern 匹配事实库失败,并且匹配规则库失败,那么目标模式 sPattern 模式推理失败,返回 false。

否则,目标模式 sPattern 模式推理成功,并得到推理结果,返回 true。

### 2.4 算法分析

在本算法中,①~③这 3 个步骤,是目标模式 sPattern 匹配事实库 sFactBase 的过程,比较直观。④~⑩这 7 个步骤,是目标模式 sPattern 匹配规则库 sRuleBase 的过程,其中,步骤⑥递归调用算法本身。步骤⑪是判断目标模式 sPattern 是否模式推理成功。

模式推理最终归结为目标模式 sPattern 与事实、规则右部之间的模式合一。在本算法中,步骤①是目标模式 sPattern 与事实之间的模式合一,步骤④是目标模式 sPattern 与规则右部之间的模式合一。

本算法的难点是步骤⑧、步骤⑩,这两个步骤相当复杂。

步骤⑧是对子目标模式的模式推理结果如何进行取舍。对于每个子目标 $SG_i$ 来说，根据它的每一个模式推理结果 $PR_j$，得到一种变量代换方式，并将这种变量代换方式，分别施加到其它子目标。如果所有其它子目标均可以接受这种变量代换方式，那么，子目标 $SG_i$ 就保留模式推理结果 $PR_j$；否则，子目标 $SG_i$ 就去除模式推理结果 $PR_j$。

经过步骤⑧之后，每个子目标 $SG_i$ 及其模式推理结果，与其它子目标 $SG_j$ 及其模式推理结果之间，存在着变量代换的一致性。也就是说，考察任意子目标 $SG_i$ 及其每一个模式推理结果 $PR_i$、$SG_j$ 及其每一个模式推理结果 $PR_j$，$SG_i$ 与模式推理结果 $PR_i$ 之间，存在一种变量代换方式。将这种变量代换方式，施加到子目标 $SG_j$，子目标 $SG_j$ 变为 $SG_j'$。那么，$SG_j$ 至少存在一个模式推理结果 $PR_j$，使得 $SG_j'$ 与 $PR_i$ 能够合一成功。子目标 $SG_i$ 的模式推理结果 $PR_i$、子目标 $SG_j$ 的模式推理结果 $PR_j$，属于同一个集合，称之为子模式推理结果一致性集合。

步骤⑩的作用，就是找到所有这些子模式推理结果一致性集合(不妨令它们分别为 $SET_1, SET_2, \cdots, SET_n$)，并分别根据这些集合，得到目标模式的模式推理结果。

### 2.5 算法测试

【事实库 FactBase(31 条事实)】：

F1：汉文帝是汉景帝的父亲　F2：汉景帝是汉武帝的父亲
F3：汉武帝是汉元帝的父亲　F4：毛泽东是毛岸英的父亲
F5：刘邦是吕雉的丈夫　F6：吕雉是汉惠帝的母亲
F7：汉惠帝是汉文帝的父亲　F8：曹操是曹植的父亲
F9：曹操是曹丕的父亲　F10：曹丕是曹睿的父亲
F11：贾代善是贾赦的父亲　F12：贾赦是贾琏的父亲
F13：贾赦是贾琮的父亲　F14：贾政是贾珠的父亲
F15：贾政是贾宝玉的父亲　F16：贾珠是贾兰的父亲
F17：贾巧姐是贾琏的女儿　F18：贾元春是贾政的女儿
F19：贾探春是贾政的女儿　F20：贾惜春是贾政的女儿
F21：贾琏是男的　F22：贾政是男的
F23：贾代善是贾母的丈夫　F24：贾母是贾环的祖母
F25：贾代化是贾敬的父亲　F26：贾珍是贾敬的儿子
F27：贾敬是男的　F28：贾代善是贾政的父亲
F29：贾宝玉是薛宝钗的丈夫　F30：贾政是王夫人的丈夫
F31：王夫人是薛宝钗的姨妈　F32：伯益是皋陶的儿子
F33：启是禹的儿子　F34：公子纠是齐襄公的兄弟
F35：公子小白是齐襄公的兄弟　F36：齐桓公是齐襄公的兄弟

【规则库 RuleBase(10 条事实)】：

R1：$_1_ $是$_2_ $的父亲 &&. $_2_ $是$_3_ $的父亲 → $_1_ $是$_3_ $的祖父
R2：$_1_ $是$_2_ $的母亲 &&. $_2_ $是$_3_ $的父亲 → $_1_ $是$_3_ $的祖母
R3：$_1_ $是$_2_ $的女儿 &&. $_2_ $是男的 → $_2_ $是$_1_ $的父亲
R4：$_1_ $是$_2_ $的丈夫 &&. $_2_ $是$_3_ $的祖母 → $_1_ $是$_3_ $的祖父
R5：$_1_ $是$_2_ $的儿子 &&. $_2_ $是男的 → $_2_ $是$_1_ $的父亲
R6：$_1_ $是$_2_ $的祖父 &&. $_2_ $是$_3_ $的父亲 → $_1_ $是$_3_ $的曾祖父
R7：$_1_ $是$_2_ $的丈夫 &&. $_2_ $是$_3_ $的姨妈 → $_1_ $是$_3_ $的姨父
R8：$_1_ $是$_2_ $的父亲 &&. $_2_ $是$_3_ $的丈夫 → $_1_ $是$_3_ $的公公
R9：$_1_ $是$_2_ $的丈夫 &&. $_1_ $是$_3_ $的父亲 → $_2_ $是$_3_ $的母亲
R10：$_1_ $是$_2_ $的母亲 &&. $_2_ $是$_3_ $的丈夫 → $_1_ $是$_3_ $的婆婆

【测试 1】：

目标模式 sPattern = “刘邦是$_1_ $的祖父”(相应的用户查询：“刘邦是谁的祖父”)

【测试结果】：

刘邦是汉惠帝的祖父

【测试 2】：

目标模式 sPattern=“汉文帝是汉元帝的$_1_ $”(用户查询：“汉文帝是汉元帝的什么人”)

【测试结果】：“汉文帝是汉元帝的曾祖父”

【测试结果分析】：

在本例中，首先根据事实：“汉文帝是汉景帝的父亲”、事实：“汉景帝是汉武帝的父亲”，以及推理规则：“$_1_ $是$_2_ $的父亲 &&. $_2_ $是$_3_ $的父亲 → $_1_ $是$_3_ $的祖父”，从而得到中间推理结果“汉文帝是汉武帝的祖父”。

然后，根据这个中间推理结果，以及事实：“汉武帝是汉元帝的父亲”、推理规则 R6：“$_1_ $是$_2_ $的祖父 &&. $_2_ $是$_3_ $的父亲 → $_1_ $是$_3_ $的曾祖父”，得到目标模式的推理结果：“汉文帝是汉元帝的曾祖父”。

【测试 3】：

sPattern = “贾政是薛宝钗的$_1_ $”(相应的用户查询：“贾政是薛宝钗的什么人”)

【测试结果】：

贾政是薛宝钗的姨父　‖　贾政是薛宝钗的公公

【测试结果分析】：

在本例中，根据事实“贾政是王夫人的丈夫”、“王夫人是薛宝钗的姨妈”，以及推理规则 R7：“$_1_ $是$_2_ $的丈夫 &&. $_2_ $是$_3_ $的姨妈 → $_1_ $是$_3_ $的姨父”，得到目标模式的推理结果：“贾政是薛宝钗的姨父”。

根据事实“贾政是贾宝玉的父亲”、“贾宝玉是薛宝钗的丈夫”，以及推理规则 R8：“$_1_ $是$_2_ $的父亲 &&. $_2_ $是$_3_ $的丈夫 → $_1_ $是$_3_ $的公公”，得到目标模式的推理结果：“贾政是薛宝钗的公公”。

## 3 算法的改进

下面提出两种模式推理的改进方法：首先给出结论，然后加以证明。

### 3.1 改进方法 1

【结论】：如果目标模式中没有变量，并且目标模式匹配事实库成功，那么，目标模式推理成功，不必继续去匹配规则库。

【证明】：目标模式中没有变量，说明目标模式是一个常量模式。目标模式匹配事实库成功，说明目标模式自身就是一条事实。

目标模式的模式推理，最终归结为目标模式(及其派生的子模式)与事实库的匹配。既然目标模式自身是一条事实，所以目标模式不必去匹配规则库。

(下转第 173 页)

($1\leqslant i\leqslant m$)。(c)$T_i(S_i)$(对于模式$(S_i)$?)和$(T_i(S_i)|T_j(S_j))$(对于模式$(S_i|S_j)$)定义类似;(d)$T_u$ 表示基本类型 $U$ 的原子模板。

根据上面的模板代数,模板递归构造过程的算法 ConstT 可以实现如下。

输入:Set{ε1,ε2,…,εm},根 MFEC
输出:对应于根 MFEC 的模板 $T(S)$,$S$ 是对应于根 MFEC 的模式,$S=[S_1,S_2,\cdots,S_n]_\Psi$
过程:令 $\varepsilon_i=\langle d_1,d_2,\cdots,d_n\rangle$,$t_i$ 表示与 $d_i$ 对应的 token;$p_i$(表示 $\varepsilon_i$ 中非空位置,定义 $q+1$ 个字符串$\langle C_{i1},C_{i2},\cdots,C_{iq+1}\rangle$,其中 $C_{i1}=t_1\cdots t_{p1}$,$C_{ij}=t_{p(j-1)+1}\cdots t_{pj}$ 和 $C_{q+1}=t_{pq}\cdots t_n$。

```
εi=根 MFEC,k=1;
While k<=n-1
{
  求 PosStr(εi,k);//通过判断 dk,dk+1 是否总连续
  If PosStr(εi,k)空 {k++;continue;}//进入下一次循环
  If PosStr(εi,k)内存在等价类 εj
  { If(PosStr(εi,k)不存在对应的已有模式)
      ConstT(Set,εj);//求 Sk
    Sk=Tεj }
  Else
    Sk=U;
Cij=tp(j-1)+1…tpj;
K++;}
Sεi=[S1,S2,…,Sq]Ψ;//组合 S1,S2,…,Sq 得到模式 S//并通过等价向量判断类型
Tεi=⟨Tεip1,…,Tεipq⟩(Ci1,Ci2,…,Ciq+1);
```

(3)抽取数据,得到最终的值$\{v_1,\cdots,v_n\}$。在本文中忽略通过模板抽取数据的过程,因为这部分内容相对容易实现。

## 5 实验

在本文所述算法基础上,我们实现了一个原型系统,本文的实验全在该系统上进行。在实验中所用的页面(见表 1 第 1 列)全部来自于实际的网站。

为了比较的目的,用 $S_m$ 表示手工确定的数据模式,$S_e$ 表示通过系统抽取得到的模式。$A_m$ 表示 $S_m$ 中的数据项,$A_e$ 表示 $S_e$ 中的数据项。为方便评估,考虑 $S_m$ 中的每个属性 $A_m$,并将评估结果分为三种类型:完全正确 #$c$,部分正确 #$p$,不正确 #$i$。#$c$ 表示 $S_e$ 中所有 $A_e$ 与 $S_m$ 中的 $A_m$ 完全相同;#$p$ 表示 $S_e$ 中 $A_e$ 与 $S_m$ 中的 $A_m$ 部分相同;#$i$ 表示 $S_e$ 中所有 $A_e$ 与 $S_m$ 中的 $A_m$ 完全不相同。$n$ 表示某个数据源用于测试的页面数,$a$ 表示 $S_m$ 中属性的个数。

在实验中成功实现了 Web 页面信息自动抽取。与其它抽取算法比较,它可以不用任何人工输入的信息。所得到的结果如表 1 所示。

表 1 实验结果

| Data Source | n | a | #c | #p | #i |
|---|---|---|---|---|---|
| aAmazon(Book) | 20 | 5 | 5 | 0 | 0 |
| Yahoo Shopping | 10 | 10 | 8 | 2 | 0 |
| eBay(auction) | 20 | 9 | 9 | 0 | 0 |
| Bigbook | 50 | 8 | 6 | 2 | 0 |
| Webcrawler | 10 | 10 | 10 | 0 | 0 |
| Buy. com(prod) | 10 | 12 | 9 | 2 | 0 |
| LA Weekly | 10 | 7 | 6 | 1 | 0 |
| Amazon(Cars) | 21 | 13 | 13 | 0 | 0 |
| Exite | 10 | 11 | 11 | 0 | 0 |
| Openfind | 5 | 6 | 4 | 2 | 0 |
| Total | 166 | 90 | 81 | 9 | 0 |

实验的目的主要是检验抽取算法的有效性。表 1 概括了实验的结果,它说明算法 EBMFEC 用于抽取数据是非常有效的。其中完全正确的有 56%,部分正确的是 44%。按全部属性平均统计,抽取的正确率大概为 98.9%。

**结论** 本文提出了一种从数据导向型页面中自动抽取数据的算法 EBMFEC。先给出了基于模板的页面模型,在这个模型之上,分析给定页面的 tokens,产生有效的最大频繁等价类 MFEC,然后推导出页面的模板并利用模板进行数据的抽取。在实际页面上的实验已经证明该方法可以达到较高的正确率。实验中发现算法使用的假设存在不成立的情况,但只是少量属性存在这种问题。在以后的工作中,将进一步对数据导向型页面的检索和查询支持以及对从 Web 中抽取得到的数据进行数据模式挖掘开展研究。

(下转第 202 页)

(上接第 141 页)

### 3.2 改进方法 2

【结论】:考察目标模式中的常量字 sConstWord。如果 sConstWord 不在事实库、规则库中出现,那么,目标模式推理失败。

【证明】:目标模式的模式推理,最终归结为目标模式(及其派生的子模式)与事实库的匹配。目标模式中的常量字 sConstWord 不在事实库中出现,所以,目标模式匹配事实库失败。

如果目标模式匹配规则库失败,那么目标模式推理失败。

否则,不妨令目标模式匹配规则 R 的右部。因为常量字 sConstWord 不在规则库中出现,所以,常量字 sConstWord 代换规则 R 的一个变量。对于一条规则来说,规则右部的变量,肯定在规则左部出现。所以,在派生出来的子目标中,至少有一个子目标 SGi 含有常量字 sConstWord。

子目标 SGi 匹配事实库失败。即使子目标 SGi 匹配规则库成功,派生出来的子目标 SGii,也会含有常量字 sConstWord。

总之,在目标模式的模式推理过程中,无法最终归结为目标模式(及其派生的子目标)与事实库的匹配。所以,目标模式推理失败。

**结论和下一步的工作** 本文首先介绍了相关定义,然后根据现有模式推理的特点,提出了一种基于自然语言的模式推理算法,并提出了改进方法。实验结果表明,本算法可以有效地解决基于自然语言的模式推理问题。下一步的工作,将致力于减小算法的时间复杂性,进一步提高算法的效率。

## 参考文献


1 王树西,白硕,姜吉发. 模式合一的"斩首"算法及其应用. 计算机工程,2004,21:22~24
2 王树西,白硕,等. 模式合一的"斩首"算法. 见:中国人工智能学会第 10 届全国学术年会论文集(上),广东,2003.528~532
3 白硕. 大规模内容计算. 见:语言计算与基于内容的文本处理. 北京:清华大学出版社,2003.13~15
4 王树西,白硕. 事实库、规则库的一体化全文索引算法. 计算机科学,2006,33(4):174~176
5 Post E L. A variant of a recursively unsolvable problem. Bull of the Am Math Soc,1946, 52
6 Ehrenfeucht A, Karhumaki J, Rozenberg G. The (generalized) post correspondece problem with lists consisting of two words is decidable. Theoret Comput Sci,1982,21(2)
7 Matiyasevich Y, Senizergues G. Decision problems for semi-Thue systems with a few rules. In: Proceedings, 11th Anual IEEE Symposium on Logic in Computer Science,1996
8 Halava V, Harju T, Hirvensalo M. Binary (Generalized) Post Correspondence Problem: [Technical Report No. 357]. TUCS, August 2000
9 Simmons R E. Natural Language Question-Answering Systems: 1969. Communications of the ACM,1970, 13(1):15~30