

析取演绎数据库否定信息的推理规则^{*}

余 金 山

(华侨大学计算机科学系, 泉州 362011)

摘要 引入否定和析取式头部扩展常规演绎数据库或逻辑程序, 这被认为是增强知识表示能力、支持常识推理和非单调推理的有效的、重要的措施。但是该类问题的处理又是困难的, 至今尚无满意的解决办法。其难点之一可表述为“如何从析取演绎数据库或逻辑程序中, 推导出否定信息”。首先, 给出一种关于否定信息推理规则的评价准则, 并用该准则分析比较已有的若干相关规则。然后, 再给出一种不同的推理规则, 并证明它的一些重要性质。这些性质指出, 该规则能更好地满足给出的评价准则。

关键词 演绎数据库, 析取数据库, 否定, 推理, 逻辑程序

分类号 TP 311.13

为了增强知识表示能力, 并对常识推理和非单调推理提供支持, 故而引入否定和析取式子句扩展常规的演绎数据库或逻辑程序。这是一项有效、重要的措施^[1-3]。但是, 经这样扩展的演绎数据库或逻辑程序的语义处理又是十分困难的, 至今尚无满意的解决方法。本文关心的是其中的难题之一, 它可表述为: 如何从给定的析取演绎数据库或逻辑程序, 推导出否定信息。首先, 给出一种关于否定信息推理规则的评价准则, 用该准则对已有的若干相关规则进行分析比较。然后, 再给出一种新的推理规则, 并证明它的若干性质。这些性质指出, 该规则能更好地满足给定的评价准则。

1 基本概念

首先, 定义几个相关的基本概念。

定义 1 数据库子句是形如

$$A_1 \quad A_2 \quad \dots \quad A_m \quad B_1 \quad B_2 \quad \dots \quad B_n$$

的全称量化封闭公式。若 $m=0$, 即其头部为空, 则称该子句是负子句或否定子句; 若 $m=1$, 则称该子句是确定的(definite); 若其头部非空且某个原子 A_i 是特异的, 则称该子句是一般的并记为

$$A_i \quad \sim A_1 \quad \dots \quad \sim A_{i-1} \quad \sim A_{i+1} \quad \dots \quad \sim A_m \quad B_1 \quad \dots \quad B_n.$$

定义 2 (析取) 数据库是数据库子句的有限集, 确定数据库是确定子句的有限集。一般数据库是一般子句的有限集。

定义3 设 D 是数据库, R 是关于负子句的推理规则. 则 $R(D)$ 是单独使用规则 R 可从 D 中推导得到的负子句的集合.

定义4 询问 Q 是任意的一阶谓词公式. Q 的回答是指对 Q 中的某些变量的一种置换. 设 D 是数据库, Q 是询问, R 是关于负子句的推理规划, θ 是 Q 的回答. 那么, 如果 $\forall (Q\theta)$ 在 D 的每一个 Herbrand 模型中均为真, 则称 θ 是 D 和 Q 关于 R 的一种正确回答.

根据以上定义不难看出, 每个不确定 (indefinite) 数据库均是析取数据库; 反之, 则不然.

2 几种推理规则的比较

2.1 比较准则

从理论上讲, 关于负子句的推理规则 R 应具有以下几种性质. (1) 相容性. 若数据库 D 是相容的, 那么 $D \cup R(D)$ 也应该是相容的. (2) 简明性. R 的应用能有效地减少必须在 D 中显式存储的否定信息的信息量. (3) 清晰性. R 中使用的逻辑符号应具有标准的语义解释. 例如, 析取符号 “ \vee ” 不应该限于是排它的. (4) 递减性. 若 $D \subseteq D'$, 那么必有 $R(D) \supseteq R(D')$.

相容性、简明性和清晰性的要求是显然的. 注意到 R 的使用的重要目的, 是要从 D 中得到隐含的否定信息. 因此, 当 D 中显式表达的信息增加时, 必须经推导才能得到的否定信息不应增加.

2.2 封闭世界假设规则 CWA

根据 CWA 及前面的定义, 有

$$CWA(D) = \{ \sim A \mid \text{基原子 } A \text{ 不是 } D \text{ 的逻辑推论} \}.$$

由此可见, 对确定的数据库而言 CWA 是相容的、递减的, 且显然是简明的. 对析取数据库, 由简单的实例 $D = \{p(a) \vee p(b)\}$, 即可知它是非相容的.

2.3 失败即否定规则 NAF

根据 NAF, 对确定的数据库 D , 有

$$NAF(D) = \{ \sim A \mid \text{存在一棵关于 } A \text{ 的有限失败的 SLD 树} \}.$$

由 SLD 的有关理论^[8]可知, 除无限推导外, NAF 接近于 CWA. 因此, 对确定的数据库 D , NAF 是相容的和递减的, 但对析取数据库它是不相容的. 实际上, 对一般的数据库 NAF 也是不相容的, 因为对一般的数据库而言, 相容数据库的完整化未必是相容的^[8].

2.4 GCWA, EGCWA 和 ECWA

GCWA 是对 CWA 的推广. 由定义

$$GCWA(D) = \{ \sim A \mid D \text{ 的任何最小模型均不包含 } A \}$$

可知, GCWA 是相容的和简明的, 但不具清晰性和递减性.

例1 设数据库 $D = \{p(a) \vee q(a)\}$, 其一阶语言仅含谓词 p, q 和常数 a . 那么, 可确定 $GCWA(D) = \varnothing$. 让 $D' = D \cup \{p(a)\}$, 则 $GCWA(D') = \{\sim q(a)\}$. 因而 GCWA 不是递减的. 另外, 还可看出, 在 D 中只有 $p(a)$ 可为真, 即它把析取解释为排它的, 因而 GCWA 是不清晰的. 自然地, 当知道 $p(a)$ 为真时, 不应导出 $\sim q(a)$.

EGCWA 除了允许推导出形如 $B_1 \vee \dots \vee B_n$ 的否定基子句外, 其它的性质与 GCWA 相同, 因而也不具备清晰性和递减性. ECWA 也类似. 关于 EGCWA 和 ECWA 的详细定义可参阅文献 [5].

3 最大闭集规则 GCSR

本节我们给出一种称之为最大闭集的新规则 GCSR, 并分析证明它的若干重要性质.

3.1 GCSR 的语法定义

定义5 设 D 是数据库, S 是 B_D 的一个子集. 如果对任一 $A \in S$ 和 D 中的某个子句的每一个基实例 C , 若 A 出现在 C 的头部, 则必存在原子 B , 使得 $B \in S$, 且出现在 C 的体中, 则称 S 是 D 的一个闭集.

例1 设有数据库 D_1 , 则

$$\begin{array}{l} p(x, y) \quad q(x) \quad r(y) \quad s(x) \\ p(a, y) \quad v(y) \quad t(y) \\ r(y) \quad u(y) \\ s(a) \quad q(a) \\ t(b) \end{array}$$

且 D_1 的语言仅包含常数 a 和 b . 那么, 由定义可知 $\varnothing, \{r(a), r(b), u(a), u(b)\}, \{p(a, b), s(a)\}, \{s(b)\}, \{s(a), q(a)\}$ 等均是 D_1 的闭集.

例3 设有数据库 D_2 , 则

$$\begin{array}{l} p(f(x)) \quad p(x) \\ p(f(f(a))) \\ q(x) \quad q(f(x)) \end{array}$$

且 D_2 的语言只包含常数 a 和函数符号 f . 那么, $\{p(a)\}, \{p(a), p(f(a))\}, \{q(a), q(f(a)), q(f(f(a))), \dots\}$ 等均是 D_2 的闭集.

显然, 若 S_1, S_2, \dots 均是 D 的闭集, 则 $S = S_1 \cup S_2 \cup \dots$ 也是 D 的闭集.

定义6 设 S_1, S_2, S_3, \dots 是 D 的所有闭集, 称 $S = S_1 \cup S_2 \cup \dots$ 是 D 的最大闭集, 可记为 $\text{gcs}(D)$.

例如, 对上面的例2, 我们可得 $\text{gcs}(D_1) = \{p(a, a), p(b, a), p(b, b), v(a), q(a), q(b), s(a), s(b), t(a), r(a), r(b), u(a), u(b)\}$.

定义7 设 D 是数据库, A 是某个基原子, 那么

$$\text{GCSR}(D) = \{ \sim A \mid A \in \text{gcs}(D) \}.$$

3.2 GCSR 的不动点定义

定义8 设 D 是数据库, 那么 $T_D(I) = \{A \in B_D \mid C \text{ 是 } D \text{ 中某个子句的基实例, } A \text{ 出现在 } C \text{ 的头部, 且对 } C \text{ 的体中的每个 } B, B \in I\}$. 其中 B_D 是 D 的 Herbrand 基.

类似于文献 [4] 中的有关命题, 可得如下推论.

推论1 $T_D \omega$ 是 T_D 的最小不动点.

引理1 $T_D \omega$ 是 D 中非负子句的一个模型.

证明 去掉 D 中的负子句, 并把每一形如 $A_1 \dots A_m \neg B_1 \dots \neg B_n$ 的非负子句, 替换成如下的一组子句:

$$\begin{array}{ccccccc} A_1 & B_1 & & \dots & & B_n \\ A_2 & B_1 & & \dots & & B_n \end{array}$$

$$A_m \quad B_1 \quad \dots \quad B_n$$

由此, 所得的确定数据库记为 D^* . 显然, D^* 的任一模型也是 D 中非负子句的模型. 由于 T_D ω 是 D^* 的最小 Herbrand 模型^[6], 因而它是 D 的非负子句的一个模型.

值得注意的是, T_D ω 未必是 D 的模型. 关于 D 的模型, 我们有如下结论.

引理 2 如果 M 是 D 的模型, 那么 $M \cup T_D$ ω 也是 D 的模型.

证明 设 $N = M \cup T_D$ ω . 因为 $N \subseteq M$, 而 M 是 D 的模型, 故 N 是 D 中所有负子句的模型. 现设 C 是 D 中某非负子句的基实例, 若 C 的体中的某个原子不在 N 中. 那么, 显然 N 是 C 的一个模型. 假定 C 的体中的所有原子均在 N 中, 因 $N \subseteq T_D$ ω , T_D ω 是 D 中非负子句的模型, 故 C 头部中的所有原子均在 T_D ω 中. 又因为 M 是 C 的模型, 且 $N \subseteq M$, 则 C 的头部中必有某个原子出现在 N 中. 因此, N 是 C 的模型. 再注意到, N 是 D 中所有负子句的模型, 而 C 又是 D 的任一非负子句的基实例, 即可证得 N 是 D 的模型.

定义 9 设 D 是数据库, A 是某个基原子, 那么 $GCSR(D) = \{ \sim A \mid A \in T_D$ $\omega \}$.

3.3 GCSR 的性质

定理 1 $GCSR$ 的两种定义是等价的, 即有 $gcs(D) = B_D \setminus T_D$ ω .

证明 先证明 $gcs(D) \subseteq B_D \setminus T_D$ ω . 因为, 若有 $A \in T_D$ ω , 则必存在非负整数 j , 使得 $A \in T_D$ j . 因此, 只要对 j 施行归纳法证明, 若 $A \in T_D$ j 则 $A \notin gcs(D)$ 即可. 显然, 当 $j=0$ 时所要的结论为真. 假定结果对 $j-1$ 为真, 设 $A \in T_D$ j . 那么, 存在 D 中某个子句的基实例 C , 使得 A 在 C 的头中且对 C 的体中的任一原子 $B \in T_D$ j . 由 T_D ω 的单调性可知, $B \in T_D$ $j-1$, 由归纳假设 $B \notin gcs(D)$. 根据闭集的定义可得 $A \notin gcs(D)$.

再证明 $gcs(D) \supseteq B_D \setminus T_D$ ω . $B_D \setminus T_D$ ω 是闭的. 若不然, 则必存在某一基原子 A 和 D 中某一子句的基实例 C , 使得 $A \in B_D \setminus T_D$ ω 且 A 出现在 C 的头部中. 同时, 对 C 的体中的每一个原子 B , $B \in T_D$ ω , 从而可得 $A \in T_D$ ω . 但是, $A \in T_D(T_D$ $\omega)$, 这与推论 1 矛盾. 从而根据 $gcs(D)$ 的定义可知, $gcs(D) \supseteq B_D \setminus T_D$ ω .

定理 2 若 D 是相容的, 那么 $D \cup GCSR(D)$ 也是相容的.

证明 由引理 2 可知, 若 D 是相容的, 那么 D 有一个模型且该模型是 T_D ω 的一个子集.

定理 3 若 $D \subseteq D'$, 那么 $gcs(D) \supseteq gcs(D')$.

证明 用归纳法容易证明 T_D $j \subseteq T_{D'}$ j ($j=0$). 再根据定理 1 即可得所要的结果.

定理 4 如果 $K = \sim K_1 \sim K_2 \dots \sim K_n$ 是基负文字的析取, 且 $D \cup GCSR(D) \vdash K$, 但 $D \not\vdash K$, 那么存在某个 i 使得 $\sim K_i \in GCSR(D)$.

证明 因为 $D \not\vdash K$, 那么必存在 D 的某个模型 M_0 , 它包含了所有的 K_i ($i=1, 2, \dots, n$). 记 $M_0 \cup T_D$ ω 为 M_1 . 由此可知 M_1 是 D 的模型. 因为 $M_1 \subseteq T_D$ ω 故 M_1 是 $D \cup GCSR(D)$ 的模型. 那么, 即然有 $D \cup GCSR(D) \vdash K$, 则 K 关于 M_1 为真. 从而存在某个 i , $K_i \in T_D$ ω , 进而由定义知 $\sim K_i \in GCSR(D)$.

定理 5 $gcs(D) \subseteq \{A \mid D \text{ 的任一最小模型均不包含 } A\}$.

证明 由引理 2 知 D 的任一最小模型均是 T_D ω 的子集. 再由 $gcs(D)$ 的定义和定理 1 即得所要的结果.

根据文 [4] 的有关定理, 可得

定理 6 若 D 是确定的数据库, 那么 $g_{cs}(D) = \{A \mid B \mid D \models A\}$.

4 结论

定理 2~4 分别证明了 GCSR 的相容性、递减性和清晰性. 定理 5 指出, 任何可由 GCSR 推导出的否定信息也可由 GCWA 推导出, 而定理 6 指出对确定的数据库而言, GCSR 与 CWA 是等价的, 因而具有类似于 CWA 和 GCWA 的简明性. 另外, 值得指出的是, 对 GCSR 只要简单地在 D 中增加适当的负子句, 便可把某个子句的析取解释为排它的. 如增加 $A_1 \mid A_2 \mid B$ 后, 即可把原子句 $A_1 \mid A_2 \mid B$ 解释为排它的. 因而, GCSR 也适合于某些需要排它析取和可兼析取相混合的应用. 而现有的规则, 包括完美模型规则, 均做不到这一点.

参 考 文 献

- 1 陈 荣, 孙吉贵. 逻辑程序中的否定问题与非单调逻辑. 计算机科学, 1997, 24(6): 16~22
- 2 周傲英, 施伯乐. 演绎数据库和逻辑程序中的否定. 计算机科学, 1996, 23(5): 14~17
- 3 余金山. 逻辑程序设计对软件开发的影响. 华侨大学学报(自然科学版), 1992, 13(3): 418~423
- 4 Lloyd J W. Foundations of logic programming. Berlin: Springer-Verlag, 1987. 20~57
- 5 Gelfond M. The extended closed world assumption and its relationship to parallel circumscription. In: Minker J. Proc. 5th ACM Symp Principles Database Systems, Cambridge: Mass., 1986: 133~139

Rules for Inferring Negative Information from Disjunctive Database and Deductive Database

Yu Jinshan

(Dept. of Comput. Sci., Huaqiao Univ., 362011, Quanzhou)

Abstract It is the effective and important measure for strengthening the capability of knowledge representation and for supporting common sense reasoning and non-monotonous reasoning to extend conventional deductive database or logic programs by introducing negation and disjunctive clauses. However, such issues as how to derive negative information from disjunctive database or logic program, satisfactory settlement remains to be found. For evaluating the reasoning rule of negative information, the author gives firstly a criterion by which several existing correlated rules have been analysed and compared. And then, a different reasoning rule is given and some of its relevant properties are proved, these properties point out that the rule is better than the previous rules in satisfying the evaluation criteria given above.

Keywords deductive database, disjunctive database, negation, reasoning, logic program