Artificial Intelligence

2. 命题逻辑的机器推理

罗晓鹏

xpluo@nju.edu.cn

工管 · 南京大学 · 2022 秋

Outline

引言

目标

- 机器推理
- 判定是否存在 $\Gamma \vdash_L \mathscr{A}$ 的有效证明
- 找出 $\Gamma \vdash_L \mathscr{A}$ 的一个有效证明
- 直接构造的困难?
- 反证法

反证法与不可满足性

定义2.1 (空合取式)

$\varnothing := \neg(\mathscr{A} \to \mathscr{A})$, $\mathscr{A}$ 是任意公式. (即, $\varnothing = \mathscr{A} \wedge \neg\mathscr{A}$).

命题2.2 (反证法)

设 Γ 是公式集且 $\mathscr{A}$ 是公式, $\Gamma \vdash_L \mathscr{A}$ 等价于 $\Gamma \cup \{\neg\mathscr{A}\} \vdash_L \varnothing$.

Proof.

(练习·板书) □

定义2.3

公式集 $\Gamma \cup \{\neg\mathscr{A}\}$ 被称为不可满足的, 若 $\Gamma \cup \{\neg\mathscr{A}\} \vdash_L \varnothing$.

不可满足和矛盾是等价的概念

(回顾)命题1.17

对于任意的真值表，存在仅含连接词 $\neg, \wedge, \vee$ 的命题形式与之相对应.

- $\Gamma \vdash_L \mathscr{A}$ 成立等价于命题形式 $\Gamma \cup \{\neg \mathscr{A}\}$ 是矛盾式
- 可用某种标准式来表示 $\Gamma \cup \{\neg \mathscr{A}\}$
- 其有利于从中推演出 $\varnothing$

归结规则

概述

- 1965 年，Robinson 针对一阶逻辑提出了归结原理
- 有效提高了机器定理证明的实际效率
- 当前的讨论仅针对命题逻辑

基本概念$_1$

定义2.4 (子句)

原子公式及其否定称为<u>文字</u>

若干个文字的一个析取式称为一个<u>子句</u>

<u>空子句</u> $\varnothing := \neg A \wedge A$，其中 A 可以是任何文字

定义2.5 (子句集)

子句集是以子句为元素组成一个集合，其中元素间关系默认为合取，元素无量词约束且否定符只作用于单个文字

定义2.6 (不可满足)

一个子句集是不可满足的，如果它能够推演出空子句

定义2.7 (互补文字)

设 L 是一个文字，则 L 和 $\neg L$ 为一对互补文字

定义2.8 (归结式·亲本子句)

设子句 C_1, C_2 具有形式

$$C_1 = C_1' \vee L, \quad C_2 = C_2' \vee \neg L,$$

则称子句 $C_1' \vee C_2'$ 为 C_1, C_2 的归结式，称 C_1, C_2 为子句 $C_1' \vee C_2'$ 的亲本子句.

命题2.9 (归结推理规则·重述)

归结式 $C_1' \vee C_2'$ 是相应亲本子句 $(C_1' \vee L) \wedge (C_2' \vee \neg L)$ 的逻辑结果，即 $(C_1' \vee L) \wedge (C_2' \vee \neg L) \quad \to \quad (C_1' \vee C_2')$ 是重言式.

命题2.10 (使用归结的辅助规则)

对于任意的 $\mathscr{A}, \mathscr{B}, \mathscr{C}$，有:

(1) 交换律: $\mathscr{A} \vee \mathscr{B} \quad \leftrightarrow \quad \mathscr{B} \vee \mathscr{A}$;

(2) 结合律: $(\mathscr{A} \vee \mathscr{B}) \vee \mathscr{C} \quad \leftrightarrow \quad \mathscr{A} \vee (\mathscr{B} \vee \mathscr{C})$;

(3) 重言律: $\mathscr{A} \vee \mathscr{A} \quad \leftrightarrow \quad \mathscr{A}$.

例：使用辅助规则后的归结

(1) $(A \vee \neg B \vee C) \wedge (D \vee B \vee E) \quad \to \quad A \vee C \vee D \vee E$;

(2) $(A \vee \neg B \vee C) \wedge (A \vee B \vee C) \quad \to \quad A \vee C$.

归结论证

定义2.11 (归结证明)

令 $\mathscr{F}$ 是和公式集 $\Gamma\cup\{\neg\mathscr{A}\}$ 逻辑等价的字句集。

公式序列 $\mathscr{A}_1,\cdots,\mathscr{A}_n=\varnothing$ 被称为从 Γ 到 $\mathscr{A}$ 的归结证明，如果其中每个公式 $\mathscr{A}_i$:

(i) 或者是 $\mathscr{F}$ 中的一个字句,

(ii) 或者是由它前面的两个字句和 归结规则推出的结论.

记为 $\Gamma\cup\{\neg\mathscr{A}\}\vdash_{Res}\varnothing$.

归结论证$_2$

例

$\{A \vee B,\ \neg A \vee B,\ A \vee \neg B,\ \neg A \vee \neg B\} \vdash_{Res} \varnothing$.

Proof.

1.	$A \vee B$	
2.	$\neg A \vee B$	
3.	$A \vee \neg B$	
4.	$\neg A \vee \neg B$	
5.	B	1, 2, Res
6.	$\neg B$	3, 4, Res
7.	$\varnothing$	5, 6, Res

□

命题2.13 (归结的有限步终止)

若子句集的原子公式数目有限，则归结证明在有限步内终止。

Proof.

(简述) □

例

设子句集仅含命题变元 A, B，则所有可能的子句为：

0. $\varnothing$
1. $A,\ \neg A,\ B,\ \neg B$
2. $A \vee B,\ \neg A \vee B,\ A \vee \neg B,\ \neg A \vee \neg B$

命题2.12 (归结证明的可靠性)

若 $\Gamma \cup \{\neg \mathscr{A}\} \vdash_{Res} \varnothing$，则 $\Gamma \vdash_L \mathscr{A}$，即 $\Gamma \cup \{\neg \mathscr{A}\}$ 不可满足.

Proof.

(命题2.2和2.9的推论) □

命题2.14 (归结系统的一致性)

不存在 L 内的公式 $\mathscr{A}$ 使得

$$\Gamma \cup \{\neg \mathscr{A}\} \vdash_{Res} \varnothing \quad \text{且} \quad \Gamma \cup \{\mathscr{A}\} \vdash_{Res} \varnothing.$$

Proof.

(L 的一致性) □

命题2.15 (归结系统的完备性)

如果公式集 $\Gamma \cup \{\neg \mathscr{A}\}$ 不可满足, 那么使用归结证明能够推演出空子句.

Proof.

(简述)

1. 有限个子句的情况;
2. 无穷个子句的情况：紧致性定理 (依赖于选择公理) □

论证示例

示例$_1$

例 1

设 $P,\ P \wedge Q \to R,\ S \vee T \to Q,\ T$. 求证：$R$

写出 $\Gamma \cup \{\neg\mathscr{A}\}$ 对应的子句集

1. $(P \wedge Q \to R) \quad \leftrightarrow \quad (\neg(P \wedge Q) \vee R) \quad \leftrightarrow \quad (\neg P \vee \neg Q \vee R)$
2. $(S \vee T \to Q) \quad \leftrightarrow \quad (\neg(S \vee T) \vee Q) \quad \leftrightarrow \quad (\neg S \wedge \neg T) \vee Q$

   $\leftrightarrow \quad (\neg S \vee Q) \wedge (\neg T \vee Q)$
3. $\{P,\ \neg P \vee \neg Q \vee R,\ \neg S \vee Q,\ \neg T \vee Q,\ T,\ \neg R\}$

示例$_2$

Proof.

1. P
2. $\neg P \vee \neg Q \vee R$
3. $\neg S \vee Q$
4. $\neg T \vee Q$
5. T
6. $\neg R$
7. $\neg Q \vee R$ 1, 2, Res
8. Q 4, 5, Res
9. R 7, 8, Res
10. $\varnothing$ 6, 9, Res □

再次回到说谎者问题

- 有 A, B, C 三人，要么说谎要么说真话

  A 说：B 和 C 都是说谎者

  B 说：A 和 C 都是说谎者

  C 说：A 和 B 中至少有一个说谎者

  问：谁是说谎者？

- 试用用字句集描述待证命题
- 完成归结证明

归结证明的优点

- 无需进行公理模式的构造
- 归结式的长度本质上是缩减的
- 便于程序化实现

基于搜索的归结算法

Algorithm 1 (基于宽度优先搜索的)归结算法

将原问题转化为子句集.
while 子句集不包含$\varnothing$ **do**
    穷举当前的子句集中的两两组合.
    for $i = 1 : \text{numOfComb}$ **do**
        if 第 i 对组合存在互补文字 **then**
            利用归结规则，生成新子句($= \varnothing$ 即终止).
        end if
    end for
    将 for loop 中生成的全部新子句添加到子句集中.
end while

Algorithm 1 中的宽度优先搜索

- 状态空间中的搜索
- 最短路径问题
- 宽度优先搜索
- 启发式搜索
- Deepblue 与 Stockfish

Algorithm 1 的终止

命题2.16

设Algorithm 1 的子句集包含 n 个变元, 那么 Algorithm 1 能够在 2^{2n-1} 次归结内完成论证.

Proof.

(练习) $\square$

提示:

0. $\binom{2n}{0}$: $\varnothing$
1. $\binom{2n}{1}$: $\cdots$

$\vdots$

n. $\binom{2n}{n}$: $\cdots$

Algorithm 1 的终止与复杂度

命题2.16

设初始的子句集不可满足且有 N 个子句. 如果存在一个深度为 m 的归结树导出空子句，那么 Algorithm 1 能够在 $\mathcal{O}(N^{2^m})$ 次互补文字的搜索和归结推理内完成论证.

Proof.

(练习) $\square$

补充：抽象的(Gentzen)自然演绎系统

- 语言：
  - ▷ 命题符号：$A_i, \bar{A}_i$
  - ▷ 逻辑符号：$\vee, \wedge$
  - ▷ 辅助符号：$(,)$
- 注意：符号$\neg, \to$不再是原始符号，它们被定义为
  - ▷ $\neg A = \bar{A},\ \neg\bar{A} = A$
  - ▷ $\neg(p \vee q) = \neg p \wedge \neg q,\ \neg(p \wedge q) = \neg p \vee \neg q$
  - ▷ $p \to q = \neg p \vee q$
- 注意：用 Γ, p 表示公式集 $\Gamma \vee \{p\}$，其中，

$$\Gamma = \{p_1, \cdots, p_n\} = p_1 \vee \cdots \vee p_n$$

补充：抽象的(Gentzen)自然演绎系统

- 公理：$\Gamma, p, \neg p$
- 规则：
  - $\vee : \frac{\Gamma, p}{\Gamma, p \vee q}$
  - $\wedge : \frac{\Gamma, p \quad \Gamma, q}{\Gamma, p \wedge q}$
  - $\text{cut} : \frac{\Gamma, p \quad \Gamma, \neg p}{\Gamma}$
- 推论(常常视为规则直接使用)：
  - $\text{rw} : \frac{\{p, q, \gamma\}}{\{p, q\}, \gamma}$
  - $\vee' : \frac{\Gamma, p, q}{\Gamma, p \vee q}$

例

试证明假言三段论：$[(\alpha \to \beta) \wedge (\beta \to \gamma)] \to (\alpha \to \gamma)$

I 利用摩根率和实质蕴涵率，假言三段论可转换为

$$[(\alpha \wedge \neg\beta) \vee (\beta \wedge \neg\gamma)] \vee (\neg\alpha \wedge \gamma)$$

II 推演树如下：

$$
\begin{prooftree}
\AxiomC{$\{\gamma,\beta\},\alpha,\neg\alpha$}
\RightLabel{(rw)}
\UnaryInfC{$\{\gamma,\beta,\neg\alpha\},\alpha$}
\AxiomC{$\{\neg\alpha,\gamma\},\neg\beta,\beta$}
\RightLabel{(rw)}
\UnaryInfC{$\{\gamma,\beta,\neg\alpha\},\neg\beta$}
\RightLabel{$(\wedge)$}
\BinaryInfC{$\{\gamma,\neg\alpha,\beta\},\alpha\wedge\neg\beta$}
\RightLabel{(rw)}
\UnaryInfC{$\{\gamma,\neg\alpha,\alpha\wedge\neg\beta\},\beta$}
\AxiomC{$\{\neg\alpha,\alpha\wedge\neg\beta\},\gamma,\neg\gamma$}
\RightLabel{(rw)}
\UnaryInfC{$\{\gamma,\neg\alpha,\alpha\wedge\neg\beta\},\neg\gamma$}
\RightLabel{$(\wedge)$}
\BinaryInfC{$\{\gamma,\neg\alpha,\alpha\wedge\neg\beta\},\beta\wedge\neg\gamma$}
\RightLabel{(rw)}
\UnaryInfC{$(\alpha\wedge\neg\beta),(\beta\wedge\neg\gamma),\neg\alpha,\gamma$}
\RightLabel{$(\vee')$}
\UnaryInfC{$((\alpha\wedge\neg\beta)\vee(\beta\wedge\neg\gamma)),(\neg\alpha\vee\gamma)$}
\RightLabel{$(\vee')$}
\UnaryInfC{$((\alpha\wedge\neg\beta)\vee(\beta\wedge\neg\gamma))\vee(\neg\alpha\vee\gamma)$}
\end{prooftree}
$$

归结推理

一阶逻辑的归结概述(后面用)

- 一阶逻辑是命题逻辑的拓展
- 1930 年的Herbrand 定理：
  - ▷ 保证了一阶逻辑可转换到命题逻辑
  - ▷ 给出了一阶逻辑的半可判定算法
- 1936 年，Turing 和 Church 证明了一阶逻辑的不可判定性
- 1965 年，Robinson 针对一阶逻辑提出了归结原理，有效地提高了机器定理证明的实际效率

END