Resolution in der Prädikatenlogik

Der Algorithmus von Gilmore funktioniert zwar, ist in der Praxis aber unbrauchbar.

Daher ist unser Programm der nächsten Stunden:

> Wie sieht Resolution in der Prädikatenlogik aus?

Wiederholung: Resolution in der Aussagenlogik

Resolutionsschritt:

$$\{L_1, \ldots, L_n, A\} \qquad \{L'_1, \ldots, L'_m, \neg A\}$$

$$\{L_1, \ldots, L_n, L'_1, \ldots, L'_m\}$$

Mini-Beispiel:

$$\{\neg A, B\} \qquad \{A\} \qquad \{\neg B\}$$

$$\{B\}$$

$$\square$$

Eine Klauselmenge ist unerfüllbar genau dann, wenn die leere Klausel abgeleitet werden kann.

Anpassung des Algorithmus von Gilmore

Algorithmus von Gilmore:

Sei F eine prädikatenlogische Aussage in Skolemform und sei $\{F_1, F_2, F_3, \ldots,\}$ eine Aufzählung von $E(F)$.

Eingabe: F

$n := 0$;

repeat $n := n + 1$;

until $(F_1 \wedge F_2 \wedge \ldots \wedge F_n)$ ist unerfüllbar;

(dies kann mit Mitteln der Aussagenlogik, z.B. Wahrheitstafeln, getestet werden)

Gib "unerfüllbar" aus und stoppe.

"Mittel der Aussagenlogik" $\rightsquigarrow$ wir verwenden Resolution für den Unerfüllbarkeitstest

Definition von $Res(F)$ (Wiederholung)

Definition: Sei F eine Klauselmenge. Dann ist $Res(F)$ definiert als

$$Res(F) = F \cup \{R \mid R \text{ ist Resolvent zweier Klauseln in } F\}.$$

Außerdem setzen wir:

$$\begin{aligned} Res^0(F) &= F \\ Res^{n+1}(F) &= Res(Res^n(F)) \qquad \text{für } n \geq 0 \end{aligned}$$

und schließlich sei

$$Res^*(F) = \bigcup_{n \geq 0} Res^n(F).$$

Grundresolutionsalgorithmus

Sei $F_1, F_2, F_3, \ldots$ weiterhin die Aufzählung der Herbrand-Expansion.

Eingabe: eine Aussage F in Skolemform mit der Matrix F^* in **KNF**

$i := 0$;
$M := \emptyset$;
repeat
    $i := i + 1;\ M := M \cup \{F_i\};\ M := \mathit{Res}^*(M)$
until $\square \in M$

Gib "unerfüllbar" aus und stoppe.

Warum der Name Grundresolution? Im Gegensatz zu späteren Verfahren werden Terme ohne Variable ($=$ Grundterme) substituiert, um die Formeln der Herbrand-Expansion zu erhalten.

Grundresolutionssatz

Den Grundresolutionsalgorithmus kann man auch in folgenden Grundresolutionssatz umformulieren:

Grundresolutionssatz: Eine Aussage in Skolemform $F = \forall y_1 \dots \forall y_k F^*$ mit der Matrix F^* in **KNF** ist unerfüllbar genau dann, wenn es eine Folge von Klauseln $K_1, \dots, K_n$ gibt mit der Eigenschaft:

- K_n ist die leere Klausel
- Für $i = 1, \dots, n$ gilt:
  - *entweder* ist K_i eine Grundinstanz einer Klausel $K \in F^*$,d.h. $K_i = K[y_1/t_1] \dots [y_k/t_k]$ mit $t_i \in D(F)$
  - *oder* K_i ist (aussagenlogischer) Resolvent zweier Klauseln K_a, K_b mit $a < i$ und $b < i$

Weglassen von Klauseln und Resolutionsschritten, die nicht zur Herleitung der leeren Klausel beitragen.

Wiederholung: Substitutionen

Eine Substitution sub ist eine Abbildung von Variablen auf Terme.

$F\,sub$: Anwendung der Substitution sub auf die Formel F

$t\,sub$: Anwendung der Substitution sub auf den Term t

Eine Substitution kann auch als Folge von Ersetzungen beschrieben werden:

$$[x/f(z)]\,[y/g(a,z)]\,[z/h(w)]$$

entspricht folgender Abbildung (entflochtene Substitution):

$$x \mapsto f(h(w)) \quad y \mapsto g(a,h(w)) \quad z \mapsto h(w).$$

Ersetzungen werden von links nach rechts durchgeführt!

Verknüpfung von Substitutionen: $sub_1 sub_2$ (zuerst wird sub_1 angewandt, anschließend sub_2).

Vertauschen von Substitutionen

Regel für das Vertauschen von Substitutionen:

$$[x/t]sub = sub[x/t\,sub],$$

falls x in sub nicht vorkommt.

Beispiele:

- $[x/f(y)]\underbrace{[y/g(z)]}_{sub} = [y/g(z)][x/f(g(z))]$
- aber $[x/f(y)]\underbrace{[x/g(z)]}_{sub} \neq [x/g(z)][x/f(y)]$
- und $[x/z]\underbrace{[y/x]}_{sub} \neq [y/x][x/z]$

Unifikator/Allgemeinster Unifikator

Gegeben sei eine Menge $\mathbf{L} = \{L_1, \ldots, L_k\}$ von Literalen. Eine Substitution sub heißt *Unifikator* von $\mathbf{L}$, falls

$$L_1 sub = L_2 sub = \ldots = L_k sub$$

Das ist gleichbedeutend mit $|\mathbf{L} sub| = 1$, wobei $\mathbf{L} sub = \{L_1 sub, \ldots, L_k sub\}$.

Ein Unifikator sub von $\mathbf{L}$ heißt *allgemeinster Unifikator* von $\mathbf{L}$, falls für jeden Unifikator sub' von $\mathbf{L}$ gilt, daß es eine Substitution s gibt mit $sub' = sub\, s$.

Aufgabe

Unifizierbar?			Ja	Nein
	$P(f(x))$	$P(g(y))$		
	$P(x)$	$P(f(y))$		
	$P(x, f(y))$	$P(f(u), z)$		
	$P(x, f(y))$	$P(f(u), f(z))$		
	$P(x, f(x))$	$P(f(y), y)$		
	$P(x, g(x), g^2(x))$	$P(f(z), w, g(w))$		
$P(x, f(y))$	$P(g(y), f(a))$	$P(g(a), z)$		

Unifikationsalgorithmus

Eingabe: eine Literalmenge $\mathbf{L} \neq \emptyset$

$sub := []$; (leere Substitution)

while $|\mathbf{L}sub| > 1$ **do**

   Suche die erste Position, an der sich zwei Literale L_1, L_2 unterscheiden

   if keines der beiden Zeichen ist eine Variable

   then stoppe mit "nicht unifizierbar"

   else Sei x die Variable und t der Term in anderen Literal (möglicherweise auch eine Variable)

      if x kommt in t vor

      then stoppe mit "nicht unifizierbar"

      else $sub := sub\,[x/t]$

Ausgabe: sub

Korrektheit des Unifikationsalgorithmus

Der Unifikationsalgorithmus terminiert immer und gibt bei Eingabe einer nicht-unifizierbaren Literalmenge “nicht unifizierbar” aus.

Wenn eine Menge $\mathbf{L}$ von Literalen unifizierbar ist, dann findet der Unifikationsalgorithmus immer den allgemeinsten Unifikator von $\mathbf{L}$.

Das bedeutet unter anderem auch, daß jede unifizierbare Menge von Literalen einen allgemeinsten Unifikator hat.

Prädikatenlogische Resolution

Eine Klausel R heißt *prädikatenlogischer Resolvent* zweier Klauseln K_1, K_2, wenn folgendes gilt:

- Es gibt Substitutionen s_1, s_2, die Variablenumbenennungen sind, so dass $K_1 s_1$ und $K_2 s_2$ keine gemeinsamen Variablen enthalten.
- Es gibt Literale $L_1, \ldots, L_m$ aus $K_1 s_1$ und Literale $L'_1, \ldots, L'_n$ aus $K_2 s_2$, so dass

  $$\mathbf{L} = \{\overline{L_1}, \ldots, \overline{L_n}, L'_1, \ldots, L'_n\}$$

  unifizierbar ist. Sei sub der allgemeinste Unifikator von $\mathbf{L}$.
  ($\overline{L}$ bezeichnet das negierte Literal L.)
- Es gilt

  $$R = ((K_1 s_1 - \{L_1, \ldots, L_m\}) \cup (K_2 s_2 - \{L'_1, \ldots, L'_n\}))sub.$$

Korrektheit und Vollständigkeit

Zwei Fragen:

- Wenn man mit prädikatenlogischer Resolution aus einer Formel F die leere Klausel $\square$ ableiten kann, ist F dann unerfüllbar? (Korrektheit)
- Kann man für eine unerfüllbare Formel F immer durch prädikatenlogische Resolution die leere Klausel herleiten? (Vollständigkeit)

Aufgabe

Sind diese Klauseln resolvierbar?
Wieviele mögliche Resolventen gibt es?

K_1	K_2	Möglichkeiten
$\{P(x), Q(x,y)\}$	$\{\neg P(f(x))\}$	
$\{Q(g(x)), R(f(x))\}$	$\{\neg Q(f(x))\}$	
$\{P(x), P(f(x))\}$	$\{\neg P(y), Q(y,z)\}$	

Lifting-Lemma

Seien K_1, K_2 zwei prädikatenlogische Klauseln und seien K'_1, K'_2 zwei Grundinstanzen hiervon, die aussagenlogisch resolvierbar sind und den Resolventen R' ergeben.

Dann gibt es einen prädikatenlogischen Resolventen R von K_1, K_2, so daß R' eine Grundinstanz von R ist.

$$
\begin{tikzcd}
K_1 \arrow[d] \arrow[dr, dash] & & K_2 \arrow[d] \arrow[dl, dash] \\
K'_1 \arrow[dr, dash] & R \arrow[d] & K'_2 \arrow[dl, dash] \\
& R' &
\end{tikzcd}
$$

—: Resolution

$\rightarrow$: Substitution

Beispiel zum Lifting-Lemma

$$\begin{tikzcd}
\{\neg P(f(x)), Q(x)\} \arrow[d, "{[x/g(a)]}"] \arrow[dr, dash] & & \{P(f(g(y)))\} \arrow[d, "{[y/a]}"] \arrow[dl, dash] \\
\{\neg P(f(g(a))), Q(g(a))\} \arrow[dr, dash] & \{Q(g(y))\} \arrow[d, "{[y/a]}"] & \{P(f(g(a)))\} \arrow[dl, dash] \\
& \{Q(g(a))\} &
\end{tikzcd}$$

Resolutionssatz

Resolutionssatz der Prädikatenlogik:

Sei F eine Aussage in Skolemform mit einer Matrix F^* in **KNF**.
Dann gilt: F ist unerfüllbar genau dann, wenn $\square \in Res^*(F^*)$.

Allabschluss

Für eine Formel H mit freien Variablen $x_1, \ldots, x_n$ bezeichnen wir mit

$$\forall H = \forall x_1 \forall x_2 \ldots \forall x_n H$$

ihren *Allabschluss*.

Sei F eine Aussage in Skolemform und sei F^* deren Matrix in **KNF**, so gilt:

$$F \equiv \forall F^* \equiv \bigwedge_{K \in F^*} \forall K$$

Beispiel:

$$\begin{array}{rcl} F^* & = & P(x,y) \wedge \neg Q(y,x) \\ F & \equiv & \forall x \forall y (P(x,y) \wedge \neg Q(y,x)) \equiv \forall x \forall y P(x,y) \wedge \forall x \forall y (\neg Q(y,x)) \end{array}$$

Beispiel

Ist die Klauselmenge

$$\{\{P(f(x))\}, \{\neg P(f(x)), Q(f(x), x)\}, \{\neg Q(f(a), f(f(a)))\}, \\ \{\neg P(x), Q(x, f(x))\}\}$$

unerfüllbar?

Demo

Wir betrachten folgende Klauselmenge (Beispiel aus dem Schöning):

$$\begin{aligned} F &= \{\{\neg P(x), Q(x), R(x, f(x))\}, \{\neg P(x), Q(x), S(f(x))\}, \{T(a)\}, \\ &\quad \{P(a)\}, \{\neg R(a, x), T(x)\}, \{\neg T(x), \neg Q(x)\}, \{\neg T(x), \neg S(x)\}\} \end{aligned}$$

und zeigen ihre Unerfüllbarkeit mit Hilfe von `otter`.

Verfeinerung der Resolution (Ausblick)

Probleme bei der prädikatenlogischen Resolution:

- Zu viele Wahlmöglichkeiten
- Immer noch zu viele Sackgassen
- Kombinatorische Explosion des Suchraums

Lösungsansätze:

Strategien und Heuristiken: Verbieten bestimmter Resolutionsschritte, Suchraum wird dadurch eingeschränkt

Vorsicht: Die Vollständigkeit darf dadurch nicht verloren gehen!