Prädikatenlogik

Übersicht:

1. Teil 1: Syntax und Semantik
2. **Teil 2: Normalformen und Grenzen der Prädikatenlogik 1. Stufe**
3. Teil 3: Modellierung und Beweise
4. Teil 4: Substitution, Unifikation und Resolution

Negationsnormalform (NNF)

Definition 1 (Negationsnormalform (NNF))

Eine Formel α ist in Negationsnormalform (NNF) genau dann, wenn jedes Negationszeichen direkt vor einer Primformel steht.

Algorithmus: (Erweiterung des aussagenlogischen Verfahrens auf PL1)

1. Ersetze $\neg\neg\sigma$ durch σ (Negation)
2. Ersetze $\neg(\alpha \wedge \beta)$ durch $\neg\alpha \vee \neg\beta$ (de Morgan)
3. Ersetze $\neg(\alpha \vee \beta)$ durch $\neg\alpha \wedge \neg\beta$ (de Morgan)
4. Ersetze $\neg(\exists y \alpha)$ durch $\forall y \neg\alpha$ (Quantoren)
5. Ersetze $\neg(\forall x \alpha)$ durch $\exists x \neg\alpha$ (Quantoren)

Satz 2

Zu jeder prädikatenlogischen Formel α gibt es eine logisch äquivalente Formel in Negationsnormalform.

Transformation in NNF

Beispiel:

$$
\begin{array}{lll}
 & \neg((\exists x P(x) \wedge \neg \forall y S(x,y)) \vee \neg \forall z S(z,z)) & \\
\approx & (\neg(\exists x P(x) \wedge \neg \forall y S(x,y))) \wedge \neg\neg \forall z S(z,z) & \text{(De Morgan)} \\
\approx & (\neg \exists x P(x) \vee \neg\neg \forall y S(x,y)) \wedge \neg\neg \forall z S(z,z) & \text{(De Morgan)} \\
\approx & (\neg \exists x P(x) \vee \forall y S(x,y)) \wedge \forall z S(z,z) & \text{(Negation)} \\
\approx & \forall x \neg P(x) \vee \forall y S(x,y) \wedge \forall z S(z,z) & \text{(Quantoren)}
\end{array}
$$

Pränexe Normalform (PNF)

Definition 3 (Pränexe Normalform (PNF))

Eine Formel α ist in pränexer Normalform (PNF), falls sie die Form $Q_1 x_1 \dots Q_n x_n\ \beta$ hat, wobei $Q_i \in \{\exists, \forall\}$ für $1 \leq i \leq n$ und β quantorenfrei ist.
β wird auch als **(quantorenfreier) Kern** der Formel bezeichnet.
Die Quantorenfolge $Q_1 x_1 \dots Q_n x_n$ wird auch als **Präfix** bezeichnet.

Beispiel:

$$\forall x \exists y \forall z\, (P(x,y) \wedge S(y,z,z))$$

ist in pränexer Normalform. Der Kern ist $(P(x,y) \wedge S(y,z,z))$ und das Präfix ist $\forall x \exists y \forall z$.

Aufgrund der Quantoreliminationsregel können wir immer davon ausgehen, dass alle Variablen im Präfix paarweise verschieden sind.

Transformation in PNF

Algorithmus: Transformation in eine äqivalente PNF

Input: Formel in NNF

1. Führe eine konsistente Umbenennung durch, so dass verschiedene Quantoren sich auf verschiedene Variablen beziehen und keine Variable sowohl frei als auch gebunden vorkommt.
2. Wende die folgenden Ersetzungsregeln so lange wie möglich an:
   1. Ersetze $(\forall x\alpha) \wedge \beta$ durch $\forall x(\alpha \wedge \beta)$
   2. Ersetze $(\exists x\alpha) \wedge \beta$ durch $\exists x(\alpha \wedge \beta)$
   3. Ersetze $(\forall x\alpha) \vee \beta$ durch $\forall x(\alpha \vee \beta)$
   4. Ersetze $(\exists x\alpha) \vee \beta$ durch $\exists x(\alpha \vee \beta)$

   (Vorkommen wie $\alpha \wedge (\forall x\beta)$ werden analog behandelt.)

Satz 4

Zu jeder prädikatenlogische Formel gibt es eine logisch äquivalente Formel in PNF.

Transformation in PNF

Beispiel:

$$\forall x \neg P(x) \land \forall x (P(x) \lor R(x)) \land \forall x (Q(x) \land \exists y R(y))$$

$$\approx \forall x_1 \neg P(x_1) \land \forall x_2 (P(x_2) \lor R(x_2)) \land \forall x (Q(x) \land \exists y R(y))$$

durch Umbenennung

$$\approx \forall x_1 \neg P(x_1) \land \forall x_2 (P(x_2) \lor R(x_2)) \land \forall x \exists y (Q(x) \land R(y))$$

nach Ersetzungsregel

$$\approx \forall x_1 \neg P(x_1) \land \forall x_2 \forall x \exists y \big((P(x_2) \lor R(x_2)) \land (Q(x) \land R(y))\big)$$

nach Ersetzungsregel

$$\approx \forall x_1 \forall x_2 \forall x \exists y \big(\neg P(x_1) \land (P(x_2) \lor R(x_2)) \land (Q(x) \land R(y))\big)$$

nach Ersetzungsregel

Skolem Normalform (SKNF)

Definition 5

Eine Formel α ist in **Skolem Normalform (SKNF)**, falls α die Form hat $\alpha = \forall x_1 \ldots \forall x_n \beta$, wobei β quantorenfrei ist. Das Präfix von α enthält also keine Existenzquantoren.

Analog zur Aussagenlogik:

Definition 6 (Erfüllbarkeits-Äquivalenz)

Zwei prädikatenlogische Formeln α und β heißen **erfüllbarkeits-äquivalent**, in Zeichen $\alpha \approx_{sat} \beta$, genau dann, wenn gilt: (α ist erfüllbar gdw. β ist erfüllbar).

Skolem Normalform

Beispiele:

$\exists x P(x) \not\approx P(a)$

Interpretation $\Im : \omega = \{1, 2\}, a_\omega = 1, P_\omega = \{2\}$, dann gilt:
$\Im(\exists x P(x)) = 1$, aber $\Im(P(a)) = 0$, da $1 \notin P_\omega$

Aber es gilt:
$\exists x P(x) \approx_{sat} P(a)$ (beide Formeln sind erfüllbar)

$\forall x \exists y\ P(x, y) \not\approx \forall x\ P(x, f(x))$

Aber:
$\forall x \exists y\ P(x, y) \approx_{sat} \forall x\ P(x, f(x))$

Idee: Erzeuge erfüllbarkeits-äquivalente Formeln.
Ersetze Existenzquantoren durch neue Funktionszeichen.

Transformation in SKNF

Algorithmus: Trans-SKNF

Input: Formel α

1. Erstelle zu α eine äquivalente Formel Φ in pränexer Normalform.
2. Alle Existenzquantoren werden von links nach rechts wie folgt eliminiert:
   1. $\Phi = \exists x \beta$: Wähle ein neues Konstantensymbol a_x und ersetze in β alle Vorkommen von x durch a_x (d.h. bilde $\beta[x/a_x]$).

      Setze $\Phi := \beta[x/a_x]$
   2. $\Phi = \forall x_1 \ldots \forall x_n \exists y \beta$: Wähle neues n-stelliges Funktionssymbol f_y und ersetze in β alle Vorkommen von y durch $f_y(x_1, \ldots, x_n)$, (d.h. bilde $\beta[y/f_y(x_1, \ldots, x_n)]$).

      Setze $\Phi := \forall x_1 \ldots \forall x_n \beta[x/f_y(x_1, \ldots, x_n)]$

Transformation in SKNF

Satz 7

Das Verfahren Trans-SKNF transformiert jede Formel α in eine erfüllbarkeits-äquivalente Formel in Skolem Normalform.

Beispiel:

$$\alpha = \exists x_1 \forall x_2 \forall x_3 \exists x_4 \ (\neg P(x_1) \wedge (P(x_2) \vee S(g(x_4, x_1, x_2))) \wedge Q(x_3) \wedge R(x_4))$$

neues Konstantensymbol a für x_1

$$\alpha \approx_{sat} \forall x_2 \forall x_3 \exists x_4 \ (\neg P(a) \wedge (P(x_2) \vee S(g(x_4, a, x_2))) \wedge Q(x_3) \wedge R(x_4))$$

neues 2-stelliges Funktionssymbol f für x_4

$$\alpha \approx_{sat} \forall x_2 \forall x_3 \ (\neg P(a) \wedge (P(x_2) \vee S(g(f(x_2, x_3), a, x_2))) \wedge Q(x_3) \wedge R(f(x_2, x_3)))$$

Kern in Konjunktiver Normalform

Voraussetzung: Formeln in Skolem Normalform

Gesucht: Transformation des Kerns in eine logisch äquivalente KNF.

Verfahren: (wie in der Aussagenlogik)

1. Transformation des Kerns in eine Negationsnormalform
2. Anwendung der Distributivgesetze

Beispiel:

$$\forall x \forall y \forall z \, (P(x) \vee \neg(S(y,x) \vee T(z,y)))$$

$$\approx \forall x \forall y \forall z \, (P(x) \vee (\neg S(y,x) \wedge \neg T(z,y)))$$

$$\approx \forall x \forall y \forall z \, ((P(x) \vee \neg S(y,x)) \wedge (P(x) \vee \neg T(z,y)))$$

Modellbildung

Hinweise: Wenn möglich, dann

1. Formeln in Normalformen, z.B. pränexe oder Skolem Normalform
2. Vermeiden von Existenzquantoren
3. Vermeiden von Funktionssymbolen

Effekt:

1. Leichtere Lesbarkeit
2. Einfachere Verarbeitung (z.B. Komplexität des Erfüllbarkeitsproblems)

Modellbildung

Beispiel:

Formeln in Skolem Normalform ohne Funktionssymbole und Existenzquantoren, aber mit Konstantensymbolen.

Beispielformel (mit nur einem Konstantensymbol)

$$\alpha = \forall x \forall y \forall z\ (P(x,a) \wedge S(z,a) \wedge (\neg S(z,y) \vee \neg P(y,y)))$$

Frage: Ist α widerspruchsvoll?

Modellbildung

Beispiel:

Formeln in Skolem Normalform ohne Funktionssymbole und Existenzquantoren, aber mit Konstantensymbolen.

Beispielformel (mit nur einem Konstantensymbol)

$$\alpha = \forall x \forall y \forall z \; (P(x,a) \wedge S(z,a) \wedge (\neg S(z,y) \vee \neg P(y,y)))$$

Frage: Ist α widerspruchsvoll?

α ist widerrspruchsvoll gdw.

$(P(a,a) \wedge S(a,a) \wedge (\neg S(a,a) \vee \neg P(a,a)))$ widerspruchsvoll ist gdw.

die aussagenlogische Formel $p \wedge s \wedge (\neg s \vee \neg p)$ widerspruchsvoll ist.

p steht für $P(a,a)$ und s steht für $S(a,a)$.

Unendliche Formelmengen

Satz 8 (Endlichkeitssatz)

Sei M eine (unendliche) Menge von Formeln. Dann ist M widerspruchsvoll genau dann, wenn es eine endliche Teilmenge M' von M gibt, so dass M' widerspruchsvoll ist.

Satz 9

Die Endlichkeit ist nicht axiomatisierbar.
D.h. es gibt keine Formelmenge M, die nur durch Interpretationen mit endlichem Grundbereich erfüllt wird.