Logische Programmierung

7. Logik

Elmar Eder

22. März 2021

Grundzeichen

- Aussagensymbole $p, q, r, \ldots$
- Junktoren
  - $\neg$ nicht
  - $\wedge$ und
  - $\vee$ oder
- Runde Klammern (,)

Formeln

Definition

Formeln sind spezielle Zeichenreihen aus Grundzeichen, und zwar

1. Jedes Aussagensymbol ist eine Formel
2. Wenn F eine Formel ist, dann ist auch $\neg F$ eine Formel.
3. Wenn F und G Formeln sind, dann sind auch $(F \wedge G)$ und $(F \vee G)$ Formeln.

Interpretationen

Definition

Die Zeichen w und f heißen Wahrheitswerte. Sie stehen für „wahr“ bzw. „falsch“.

Definition

Eine Interpretation ist eine Abbildung von der Menge der Aussagensymbole in die Menge der Wahrheitswerte.

Der Wahrheitswert einer Formel bei einer Interpretation

Definition

Der Wahrheitswert F^I einer Formel F bei einer Interpretation I ist induktiv definiert durch

1. $p^I := I(p)$ für jedes Aussagensymbol p.
2. $(\neg F)^I := \begin{cases} \mathrm{w}, & \text{wenn } F^I = \mathrm{f} \\ \mathrm{f} & \text{sonst} \end{cases}$
3. $(F \wedge G)^I := \begin{cases} \mathrm{w}, & \text{wenn } F^I = \mathrm{w} \text{ und } G^I = \mathrm{w} \\ \mathrm{f} & \text{sonst} \end{cases}$
4. $(F \vee G)^I := \begin{cases} \mathrm{w}, & \text{wenn } F^I = \mathrm{w} \text{ oder } G^I = \mathrm{w} \\ \mathrm{f} & \text{sonst} \end{cases}$

Einige Begriffe der Semantik

Definition

Ein Modell einer Formel F ist eine Interpretation I so, dass $F^I = \mathrm{w}$ ist.

Definition

- Eine Formel F heißt allgemeingültig, wenn für jede Interpretation I gilt $F^I = \mathrm{w}$.
- Eine Formel F heißt erfüllbar, wenn es eine Interpretation I gibt mit $F^I = \mathrm{w}$.
- Andernfalls heißt sie unerfüllbar.

Einige Begriffe der Semantik

Definition

- G folgt semantisch aus F, in Zeichen $F \models G$, wenn jedes Modell von F auch Modell von G ist.
- F und G heißen semantisch äquivalent, in Zeichen $F \sim G$, wenn $F^I = G^I$ ist für alle Interpretationen I.

$\sim$ ist eine Äquivalenzrelation auf der Menge der Formeln.

Einige Begriffe der Semantik

Definition

Sei S eine Menge von Formeln und G eine Formel.

- I heißt Modell von S, wenn I Modell jeder Formel $F \in S$ ist.
- G folgt semantisch aus S, in Zeichen $S \models G$, wenn jedes Modell von S ein Modell von G ist.
- S heißt erfüllbar, wenn S ein Modell hat.
- Andernfalls heißt S unerfüllbar.

Einige semantische Äquivalenzen

$F \wedge G \sim G \wedge F$	Kommutativität von $\wedge$
$F \vee G \sim G \vee F$	Kommutativität von $\vee$
$\neg\neg F \sim F$	
$(F \wedge G) \wedge H \sim F \wedge (G \wedge H)$	Assoziativität von $\wedge$
$(F \vee G) \vee H \sim F \vee (G \vee H)$	Assoziativität von $\vee$
$F \wedge (G \vee H) \sim (F \wedge G) \vee (F \wedge H)$	Distributivgesetz
$F \vee (G \wedge H) \sim (F \vee G) \wedge (F \vee H)$	Distributivgesetz
$F \wedge F \sim F$	
$F \vee F \sim F$	
$\neg(F \wedge G) \sim \neg F \vee \neg G$	DeMorgan'sche Regel
$\neg(F \vee G) \sim \neg F \wedge \neg G$	DeMorgan'sche Regel

Eigenschaften semantischer Begriffe

Satz

Sei F eine Formel. Dann gilt:

- F *ist allgemeingültig* $\iff$ $\neg F$ *ist unerfüllbar.*
- F *ist unerfüllbar* $\iff$ $\neg F$ *ist allgemeingültig.*

Satz

Sei S eine Formelmenge und F eine Formel. Dann gilt:

$$S \models F \iff S \cup \{\neg F\} \text{ ist unerfüllbar.}$$

Literal, Konjunktion, Disjunktion

Definition

Ein Literal ist ein Aussagensymbol p oder die Negation $\neg p$ eines Aussagensymbols.

Definition (Konjunktion $F_1 \wedge \cdots \wedge F_n$ von Formeln $F_1, \ldots, F_n$)

- Die Konjunktion von F_1 ist F_1.
- $F_1 \wedge \cdots \wedge F_{n+1} :\equiv (F_1 \wedge \cdots \wedge F_n) \wedge F_{n+1}$.

Definition (Disjunktion $F_1 \vee \cdots \vee F_n$ von Formeln $F_1, \ldots, F_n$)

- Die Disjunktion von F_1 ist F_1.
- $F_1 \vee \cdots \vee F_{n+1} :\equiv (F_1 \vee \cdots \vee F_n) \vee F_{n+1}$.

Konjunktive und disjunktive Normalform

Definition

Eine Formel F ist in konjunktiver Normalform (KNF), wenn sie eine Konjunktion von Disjunktionen von Literalen ist.

Definition

Eine Formel F ist in disjunktiver Normalform (DNF), wenn sie eine Disjunktion von Konjunktionen von Literalen ist.

Verwandlung in Normalform

Satz

Zu jeder Formel F gibt es eine Formel F' in KNF mit $F \sim F'$.

Satz

Zu jeder Formel F gibt es eine Formel F' in DNF mit $F \sim F'$.

Beweis mit

- DeMorgan'schen Regeln
- Elimination doppelter Negationen
- Ausmultiplizieren mit Distributivgesetz

KNF in Prolog

```
p :- q, r.
q.
r.
?- p.
```

$$(p \vee \neg q \vee \neg r) \wedge$$
$$q \wedge$$
$$r \wedge$$
$$\neg p$$

Klauseln beim automatischen Beweisen

Definition

Eine Klausel beim automatischen Beweisen ist eine endliche Menge von Literalen.

Im Resolutionskalkül

steht eine Klausel für eine Disjunktion von Literalen.
Die leere Klausel $\square = \emptyset$ ist immer falsch (entspricht einem Widerspruch).

Beispiel

$\{p, \neg q, \neg r\}$ für die Formel $p \vee \neg q \vee \neg r$

Resolution

Die Resolutionsregel

$$\frac{c \qquad\qquad d}{(c \setminus \{L\}) \cup (d \setminus \{\overline{L}\})} \quad , \qquad \text{wenn } L \in c \text{ und } \overline{L} \in d.$$

- Elternklauseln: c und d
- Resolvente: $(c \setminus \{L\}) \cup (d \setminus \{\overline{L}\})$

Beispiel

$$\frac{\{\neg p\} \qquad \{p, \neg q, \neg r\}}{\{\neg q, \neg r\}} \qquad \begin{array}{c}\text{Elternklauseln}\\ \text{Resolvente}\end{array}$$

Hornklauseln

Definition

- Ein positives Literal ist ein Aussagensymbol (also ein Literal, das kein Negationszeichen $\neg$ enthält).
- Ein negatives Literal ist die Negation eines Aussagensymbols (also ein Literal, das das Negationszeichen $\neg$ enthält).

Definition

- Eine Hornklausel (oder definite Klausel) ist eine Klausel, die höchstens ein positives Literal enthält.
- Eine Programmklausel ist eine Klausel, die genau ein positives Literal enthält.
- Eine Anfrageklausel oder Zielklausel (engl. query clause oder goal clause ist eine Klausel, die kein positives Literal enthält.

SLD-Resolution

Definition

SLD-Resolution (Selection rule driven Linear resolution for Definite clauses) ist Resolution für Hornklauseln mit folgenden Einschränkungen

- Klauselmenge $S = P \cup \{G\}$
- P ist eine Menge von Programmklauseln.
- G ist eine Zielklausel.
- lineare Resolution:
  In jedem Schritt wird eine Zielklausel mit einer Klausel des Programms P resolviert.
- In der beteiligten Zielklausel wird das wegzuresolvierende Literal durch eine "selection function" ausgewählt.
  In Prolog standardmäßig das erste Literal der Zielklausel.

SLD-Resolution

Eine SLD-Resolutionsableitung (Ableitungsbaum) hat die Form

$$\begin{tikzcd}
G_0 \arrow[dr, dash] & & c_0 \arrow[dl, dash] & & & \\
& G_1 \arrow[dr, dash] & & c_1 \arrow[dl, dash] & & \\
& & G_2 \arrow[dr, dash] & & c_2 \arrow[dl, dash] & \\
& & & G_3 \arrow[dr, dash] & & \\
& & & & \ddots &
\end{tikzcd}$$

wobei $G_0 \equiv G$ ist und $c_0, c_1, c_2, \dots \in P$ sind.

Wenn sie mit der leeren Klausel $\Box$ endet, sprechen wir von einer SLD-Widerlegung oder einem Widerlegungsbaum.

SLD-Resolution

Dabei muss jeder Resolutionsschritt die Form haben:

$$\frac{\texttt{?-}\ A_1, \ldots, A_j, \ldots, A_m. \qquad\qquad A_j\ \texttt{:-}\ B_1, \ldots, B_n.}{\texttt{?-}\ A_1, \ldots, A_{j-1}, B_1, \ldots, B_n, A_{j+1}, \ldots, A_m.}$$

Das Literal A_j heißt das ausgewählte Literal. Es wird angenommen, dass eine Auswahlfunktion (englisch: selection function) gegeben sei, die zu einer Zielklausel das ausgewählte Literal findet.

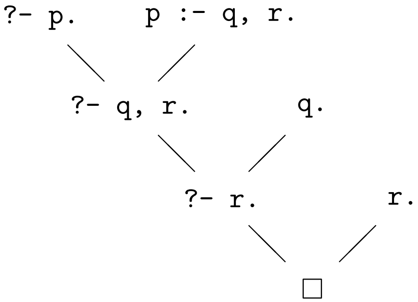
Beispiel

(mit dem ersten Literal der Zielklausel als ausgewähltem Literal):

$$\frac{\texttt{?-p.} \qquad \texttt{p:-q,r.}}{\texttt{?-q,r.}}$$

Beispiel für einen Widerlegungsbaum

```
p :- q, r.
q.
r.
?- p.
```



Sprache der Prädikatenlogik erster Stufe

Grundzeichen

- Variable
- Konstanten (in Prolog: Atome und Zahlen)
- Funktionszeichen (in Prolog: Funktoren Atome)
- Prädikatszeichen (in Prolog: Prädikate Atome)
- Junktoren $\neg$, $\wedge$, $\vee$
- Quantoren $\forall$, $\exists$
- Runde Klammern (,) und Komma ,

Sprache der Prädikatenlogik erster Stufe

Terme

- Variable
- Konstanten
- $f(t_1, \ldots, t_n)$

Formeln

- $p(t_1, \ldots, t_n)$
- $\neg F$, $F \wedge G$, $F \vee G$
- $\forall X\, F$, $\exists X\, F$

Semantik der Prädikatenlogik erster Stufe

Interpretation I gegeben durch

- Individuenbereich (engl. domain) D (nichtleere Menge)
- Funktion ordnet jeder/jedem
  - Konstanten a ein $a^I \in D$
  - n-stelligen Funktionszeichen f eine Funktion $f^I \colon D^n \to D$
  - n-stelligen Prädikatszeichen p eine Relation $p^I \subseteq D^n$

  zu.

Variablenbelegung V ordnet jeder Variablen X ein $X^V \in D$ zu.

Semantik der Prädikatenlogik erster Stufe

Wert t^{IV} eines Terms t bei I und V

- $X^{IV} := X^{V}$, wenn X Variable
- $a^{IV} := a^{I}$, wenn a Konstante
- $f(t_1, \ldots, t_n)^{IV} := f^{I}(t_1^{IV}, \ldots, t_n^{IV})$

$$V_{\xi}^{x}(y) := \begin{cases} \xi, & \text{wenn } y \equiv x \\ V(y) & \text{sonst} \end{cases}$$

Semantik der Prädikatenlogik erster Stufe

Wahrheitswert F^{IV} einer Formel F bei I und V

- $p(t_1, \ldots, t_n)^{IV} := \begin{cases} \text{w}, & \text{wenn } (t_1^{IV}, \ldots, t_n^{IV}) \in p^I \\ \text{f} & \text{sonst} \end{cases}$
- $(\neg F)^{IV} := \begin{cases} \text{w}, & \text{wenn } F^{IV} = \text{f} \\ \text{f} & \text{sonst} \end{cases}$
- $(F \wedge G)^{IV} := \begin{cases} \text{w}, & \text{wenn } F^{IV} = \text{w und } G^{IV} = \text{w} \\ \text{f} & \text{sonst} \end{cases}$
- $(F \vee G)^{IV} := \begin{cases} \text{w}, & \text{wenn } F^{IV} = \text{w oder } G^{IV} = \text{w} \\ \text{f} & \text{sonst} \end{cases}$
- $(\forall X\, F)^{IV} := \begin{cases} \text{w}, & \text{wenn } F^{IV_\xi^x} = \text{w für alle } \xi \in D \\ \text{f} & \text{sonst} \end{cases}$
- $(\exists X\, F)^{IV} := \begin{cases} \text{w}, & \text{wenn } F^{IV_\xi^x} = \text{w für mindestens ein } \xi \in D \\ \text{f} & \text{sonst} \end{cases}$

Semantik der Prädikatenlogik erster Stufe

Eine Formel heißt geschlossen, wenn

- jede Variable durch einen Quantor gebunden ist.

Dann ist F^{IV} unabhängig von V und wir schreiben einfach F^{I}.

Eine geschlossene Formel F heißt

- allgemeingültig, wenn $F^{I} = \mathrm{w}$ für alle I.
- erfüllbar, wenn $F^{I} = \mathrm{w}$ für mindestens ein I.

Prologklauseln als Formeln

Beispiel

```
grossvater(X,Z) :-
     vater(X,Y),
     kind(Z,Y).
```

$$\forall \mathtt{X} \forall \mathtt{Y} \forall \mathtt{Z} \big(\mathtt{grossvater}(\mathtt{X},\mathtt{Z}) \leftarrow \mathtt{vater}(\mathtt{X},\mathtt{Y}) \land \mathtt{kind}(\mathtt{Z},\mathtt{Y})\big)$$

$$\forall \mathtt{X} \forall \mathtt{Y} \forall \mathtt{Z} \big(\mathtt{grossvater}(\mathtt{X},\mathtt{Z}) \lor \neg \mathtt{vater}(\mathtt{X},\mathtt{Y}) \lor \neg \mathtt{kind}(\mathtt{Z},\mathtt{Y})\big)$$

Prologklauseln als Formeln

Regel

A:-B_1, ..., B_n.

$$\forall[A \vee \neg B_1 \vee \cdots \vee \neg B_n]$$

Faktum

A.

$$\forall[A]$$

Anfrage

?-B_1, ..., B_n.

$$\forall[\neg B_1 \vee \cdots \vee \neg B_n]$$

Prologklauseln als Formeln

Beispiel

```
?- grossvater(X,gabi).
```

$$\forall \mathtt{X} \neg \mathtt{grossvater}(\mathtt{X}, \mathtt{gabi})$$

Prologklauseln als Formeln

Beispiel

```
kind(peter,hans).
kind(gabi,peter).
maennlich(hans).
vater(X,Y) :- kind(Y,X), maennlich(X).
grossvater(X,Z) :- vater(X,Y), kind(Z,Y).
?- grossvater(X,gabi).
```

Prologklauseln als Formeln

Beispiel

$$\begin{gathered}
\texttt{kind}(\texttt{peter}, \texttt{hans}) \wedge \\
\texttt{kind}(\texttt{gabi}, \texttt{peter}) \wedge \\
\texttt{maennlich}(\texttt{hans}) \wedge \\
\forall \texttt{X} \forall \texttt{Y} (\texttt{vater}(\texttt{X}, \texttt{Y}) \vee \neg \texttt{kind}(\texttt{Y}, \texttt{X}) \vee \neg \texttt{maennlich}(\texttt{X})) \wedge \\
\forall \texttt{X} \forall \texttt{Y} \forall \texttt{Z} (\texttt{grossvater}(\texttt{X}, \texttt{Z}) \vee \neg \texttt{vater}(\texttt{X}, \texttt{Y}) \vee \neg \texttt{kind}(\texttt{Z}, \texttt{Y})) \wedge \\
\forall \texttt{X} \neg \texttt{grossvater}(\texttt{X}, \texttt{gabi})
\end{gathered}$$