

Miriam Butt und Maribel Romero
G220/5109, G212/2728
{miriam.butt|maribel.romero}@uni-konstanz.de

Advanced Computational Semantics
Sommersemester 2008, Hauptseminar

Lösung zu Übung 9, 26.06.2008

Aufgabe 1)

1. Das Programm `pronounDRT.pl` stützt sich auf die beiden Dateien `englishLexicon.pl` und `englishGrammar.pl`. Das heißt, dass Sätze nur Wörter enthalten dürfen, die im Lexikon enthalten sind:

```
> mia knows jody. she smokes.  
1 drs(...
```

```
> maggie knows jody. she smokes.  
fail.
```

Außerdem gehen manche einfachen Konstruktionen nicht, obwohl die entsprechend notwendigen Wörter im Lexikon enthalten sind:

```
> mia loves jody. she dies.  
1 drs(...
```

```
> the boxer loves jody. she dies.  
fail.
```

In diesen Fällen ist dann wohl die Grammatik schuld.
Das Programm arbeitet jedoch sehr gut an einfachen Hauptsätzen und kommt auch mit mehreren Pronomen zurecht:

```
> mia knows vincent. she loves him.  
1 drs(...
```

2. — (Die Testsuite lässt sich mit `pronounDRTTestSuite.` aufrufen.)

3. Entweder weil der Datei `englishLexicon.pl` Lexikoneinträge des zu verarbeitenden Satzes fehlen, oder weil die Grammatik nicht mitspielt — siehe oben. Oder aber weil dem Lexikon semantische Information über seine Einträge fehlt, es also linguistisch nicht besonders viel weiß (siehe Aufgabe 4.).

4. Prolog gibt keinen Fehler aus, weil es offensichtlich nicht weiß, dass `jody` ein Frauenname ist, und darauf deshalb nicht mit `he` referiert werden darf. Aber betrachte dieses Beispiel:

```
> mia collapses. he smokes.
fail.
```

```
> jody collapses. he smokes.
1 drs(...
```

Prolog weiß offenbar bescheid über das Geschlecht von `mia`, aber nicht über das von `jody`. In der Datei `englishLexicon.pl` werden jedoch alle Eigennamen gleich behandelt, nämlich ohne Information über das Geschlecht. Diese muss in der Datei `lexicalKnowledge.pl` eingetragen werden; dort ist bereits ein Eintrag für `mia` vorhanden, analog braucht man einen für `jody`:

```
lexicalKnowledge(jody,0,Axiom):-
    Axiom = all(A,imp(eq(A,jody),woman(A))).
```

Weiter oben in der Datei wird dann festgelegt, dass Elemente, die die Eigenschaft `woman()` besitzen, auch immer automatisch `female()` sind.

5. Reflexiva funktionieren:

```
> jimmy knows himself.
1 drs(...
```

6. Prolog gibt auch hier keinen Fehler aus, und auch hier reicht es, dem Eigennamen `jimmy` in der Datei `lexicalKnowledge.pl` ein Geschlecht zuzuweisen, damit der Satz so nicht mehr von Prolog akzeptiert wird:

```
lexicalKnowledge(jimmy,0,Axiom):-
    Axiom = all(A,imp(eq(A,jimmy),man(A))).
```

7. Im ersten Teilsatz gibt es nur ein Individuum, und zwar jimmy. Für dieses Individuum wird eine Variable erstellt, auf die sich `himself` dann rückbeziehen kann:

```
> jimmy knows himself.
1 drs([$VAR(0)], [male($VAR(0)):[], eq($VAR(0), jimmy):[],
know($VAR(0), $VAR(0)): [ref:yes]])
true ;
fail.
```

Für `he` im zweiten Teilsatz gilt das gleiche; es kann sich rückbeziehen auf `jimmy`. Das ist nicht der Fall für `him`, welches ein zusätzliches vorher als Objekt eingeführtes Individuum erfordert. Somit wird der Satz nicht akzeptiert.

Aufgabe 2)

JANI's Lösung für den deutschen Text: This solution counts as correct for the purposes of this course. But see Maribel's solution below for the bigger picture.

Diskurssegment; Anapherauflösung	Cb(U_i, D)	Cp(U_i, D)	Transition
Der Adler arbeitete wieder an dem Pfeil.	—	Adler	—
Endlich gelang es ihm, die Spitze herauszuziehen. ihm → Adler	ihm	ihm	CONTINUE
Der Pfeil fiel ins feuchte Gras.	—	Pfeil	ROUGH-SHIFT
Harka sprang hin und holte ihn. ihn → Pfeil	ihn	Harka	RETAIN
Als er wieder bei seinem Vater saß, er → Harka	er	er	SMOOTH-SHIFT
saß auch der Adler auf seinem Stein, ruhig wie ein Bild.	—	Adler	ROUGH-SHIFT
Das Fleisch hielt er zwischen den Fängen. er → Adler	Adler	Fleisch	RETAIN

MARIBEL's Lösung für den deutschen Text

Note 1: Referents that are inferrable in a given sentence count as realized in that sentence. E.g., in the second sentence, from the reference to "Spitze" we can infer the reference to "Pfeil".

Note 2: Adjunct clauses (e.g. "Als er wieder bei seinem Vater saß") count as any other adjunct. Since we did not discuss the ordering of different adjuncts in the Cf list, for the sake of this exercise we will consider "Stein", "Bild" and the sub-list "er, Vater" unranked with respect to each other.

Note 3: Since we did not say otherwise, we assume that German ranks the Cf list according to grammatical function. With this assumption, the Cf list for the final sentence is as indicated below.

Diskurssegment; Anapherauflösung	Cb(U_i, D)	Cf(U_i, D)	Transition
Der Adler arbeitete wieder an dem Pfeil.	—	Adler, Pfeil	—
Endlich gelang es ihm, die Spitze herauszuziehen. ihm → Adler	ihm	ihm, Spitze (vom Pfeil)	CONTINUE
Der Pfeil fiel ins feuchte Gras.	Pfeil	Pfeil, Gras	SMOOTH-SHIFT
Harka sprang hin und holte ihn. ihn → Pfeil	ihn	Harka, ihn	RETAIN
Als er wieder bei seinem Vater saß, saß auch der Adler auf seinem Stein, ruhig wie ein Bild. er → Harka	er	Adler, Stein / Bild / [er, Vater]	ROUGH-SHIFT
Das Fleisch hielt er zwischen den Fängen. er → Adler	Adler	er, Fleisch, Fängen	SMOOTH-SHIFT