Einführung in die Computerlinguistik

Merkmalsstrukturen (Feature Structures)

Laura Kallmeyer
Heinrich-Heine-Universität Düsseldorf

Sommersemester 2013

Überblick

1. Einleitung
2. Merkmalsstrukturen als Graphen
3. Subsumption und Unifikation
4. Getypte Merkmalsstrukturen
5. Erweiterungen

Einleitung (1)

Die in CFGs verwendeten Nichtterminalen sind in der Regel nicht ausreichend, um linguistische Generalisierungen auszudrücken.

Bsp. Kongruenz

S → NPSg VPSg S → NPPl VPPl

Besser:

S → NP VP Bedingung: gleicher Numerus in NP und VP

Einleitung (2)

Um diese Generalisierung auszudrücken, faktorisieren wir die Nichtterminalen:

- Ein Nichtterminales ist nicht mehr atomar, sondern hat eine Struktur.
- Der Inhalt des Nichtterminalen wird durch Attribute beschrieben, die bestimmte Werte haben können.

$$\begin{bmatrix} \text{CAT} & \text{NP} \\ \text{NUM} & \text{Pl} \end{bmatrix}$$

- Man kann sich an verschiedenen Stellen auf denselben Attributwert beziehen (Structure Sharing)

$$\begin{bmatrix} \text{CAT} & \text{S} \end{bmatrix} \rightarrow \begin{bmatrix} \text{CAT} & \text{NP} \\ \text{NUM} & \boxed{1} \end{bmatrix} \begin{bmatrix} \text{CAT} & \text{VP} \\ \text{NUM} & \boxed{1} \end{bmatrix}$$

Die Variable $\boxed{1}$ denotiert immer den gleichen Wert.

Einleitung (3)

Unterspezifikation: Es sind nicht immer alle Merkmale bekannt. Anstatt die verschiedenen Möglichkeiten explizit aufzulisten, gibt man nur die Merkmale an, die feststehen.

$\begin{bmatrix} \text{CAT} & \text{N} \\ \text{NUM} & \text{Sg} \\ \text{GEN} & \text{m} \end{bmatrix} \rightarrow \text{man}$ $\quad$ $\begin{bmatrix} \text{CAT} & \text{N} \\ \text{GEN} & \text{n} \end{bmatrix} \rightarrow \text{fish}$

$\begin{bmatrix} \text{CAT} & \text{Det} \\ \text{NUM} & \text{Sg} \end{bmatrix} \rightarrow \text{a}$ $\quad$ $\begin{bmatrix} \text{CAT} & \text{Det} \end{bmatrix} \rightarrow \text{the}$

$\begin{bmatrix} \text{CAT} & \text{NP} \\ \text{NUM} & \boxed{1} \\ \text{GEN} & \boxed{2} \\ \text{PERS} & 3 \end{bmatrix} \rightarrow \begin{bmatrix} \text{CAT} & \text{Det} \\ \text{NUM} & \boxed{1} \end{bmatrix} \begin{bmatrix} \text{CAT} & \text{N} \\ \text{NUM} & \boxed{1} \\ \text{GEN} & \boxed{2} \end{bmatrix}$

Einleitung (4)

Attribute müssen nicht notwendig atomare Werte haben. Der Wert eines Attributs kann wiederum eine Merkmalsstruktur sein:

$\begin{bmatrix} \text{CAT} & \text{N} \\ \text{AGR} & \begin{bmatrix} \text{GEN} & \text{n} \end{bmatrix} \end{bmatrix} \rightarrow \text{fish}$

$\begin{bmatrix} \text{CAT} & \text{Det} \\ \text{AGR} & \begin{bmatrix} \text{NUM} & \text{Sg} \end{bmatrix} \end{bmatrix} \rightarrow \text{a}$

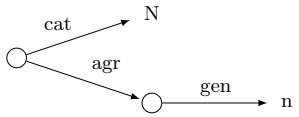
$\begin{bmatrix} \text{CAT} & \text{NP} \\ \text{AGR} & \boxed{1}\begin{bmatrix} \text{PERS} & 3 \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \text{CAT} & \text{Det} \\ \text{AGR} & \boxed{1} \end{bmatrix} \begin{bmatrix} \text{CAT} & \text{N} \\ \text{AGR} & \boxed{1} \end{bmatrix}$

Merkmalsstrukturen als Graphen (1)

Merkmalsstrukturen werden in der Regel als gerichtete Graphen formalisiert.

Es gibt zwei Möglichkeiten: Man kann eine Attribut-Wert-Matrix wie $\begin{bmatrix} \text{CAT} & \text{N} \\ \text{AGR} & \begin{bmatrix} \text{GEN} & \text{n} \end{bmatrix} \end{bmatrix}$ als

1. einen gerichteten Graphen auffassen



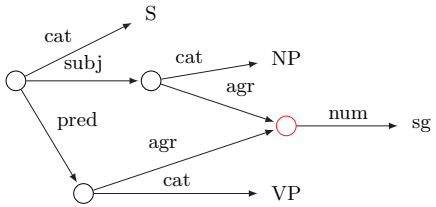
2. oder als eine Beschreibung derartiger Graphen, die im Prinzip von einer unendlichen Menge von Graphen erfüllt wird.

Merkmalsstrukturen als Graphen (2)

Merkmalsstrukturen als Graphen sind nicht notwendig Bäume:

$\begin{bmatrix} \text{CAT} & \text{S} \\ \text{SUBJ} & \begin{bmatrix} \text{CAT} & \text{NP} \\ \text{AGR} & \boxed{1}\begin{bmatrix} \text{NUM} & \text{sg} \end{bmatrix} \end{bmatrix} \\ \text{PRED} & \begin{bmatrix} \text{CAT} & \text{VP} \\ \text{AGR} & \boxed{1} \end{bmatrix} \end{bmatrix}$

structure sharing
"Strukturteilung"



Merkmalsstrukturen als Graphen (3)

Wichtig: Die Attribute sind funktional, d.h., sie müssen, wenn vorhanden, eindeutige Werte haben.

⇒ Bedingung auf die Graphen: für jeden Knoten v und jedes Attribut a gibt es nur höchstens eine von v ausgehende Kante, die das Label a hat.

Mit anderen Worten: Jedes Attribut a kann aufgefasst werden als eine partielle Funktion auf der Menge der Knoten des Graphen einer Merkmalsstruktur.

Subsumption und Unifikation (1)

Subsumption: Relation auf Merkmalsstrukturen: S_1 subsumiert S_2 ($S_1 \sqsubseteq S_2$), wenn in S_2 mindestens die Information aus S_1 enthalten ist.

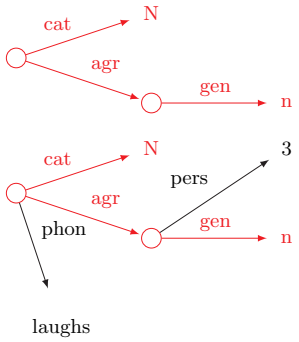
Bsp. S_1: $\begin{bmatrix} \text{CAT} & \text{V} \\ \text{AGR} & \begin{bmatrix} \text{NUM} & \text{Sg} \end{bmatrix} \end{bmatrix}$ S_2: $\begin{bmatrix} \text{ORTH} & \text{laughs} \\ \text{CAT} & \text{V} \\ \text{AGR} & \begin{bmatrix} \text{PERS} & 3 \\ \text{NUM} & \text{Sg} \end{bmatrix} \end{bmatrix}$

$S_1 \sqsubseteq S_2$

Mit anderen Worten: Es gibt eine injektive Abbildung der Knoten aus S_1 in die Knoten aus S_2, die kanten- und labelerhaltend ist, also einen Homomorphismus.

Subsumption und Unifikation (2)

S_1 als Graph und sein homomorphes Bild in S_2:



Subsumption und Unifikation (3)

Subsumption ist eine partielle Ordnung, d.h. sie ist

1. reflexiv: Jede Struktur subsumiert sich selbst $S \sqsubseteq S$ für alle S;
2. transitiv: aus $S_1 \sqsubseteq S_2$ und $S_2 \sqsubseteq S_3$ folgt $S_1 \sqsubseteq S_3$ für alle S_1, S_2, S_3;
3. antisymmetrisch: aus $S_1 \sqsubseteq S_2$ und $S_2 \sqsubseteq S_1$ folgt $S_1 = S_2$.

Die leere Merkmalsstruktur [ ] subsumiert alle anderen Merkmalsstrukturen.

Subsumption und Unifikation (4)

Eine Merkmalsstruktur S heißt Unifikation von S_1 und S_2 ($S_1 \sqcup S_2$) , wenn S sowohl von S_1 als auch von S_2 subsumiert wird und wenn außerdem S alle anderen Merkmalsstrukturen subsumiert, die ebenfalls von S_1 und von S_2 subsumiert werden.

$$\begin{bmatrix} \text{CAT} & \text{V} \\ \text{AGR} & \begin{bmatrix} \text{NUM} & \text{Sg} \end{bmatrix} \end{bmatrix} \sqcup \begin{bmatrix} \text{CAT} & \text{V} \\ \text{AGR} & \begin{bmatrix} \text{PERS} & 3 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \text{CAT} & \text{V} \\ \text{AGR} & \begin{bmatrix} \text{NUM} & \text{Sg} \\ \text{PERS} & 3 \end{bmatrix} \end{bmatrix}$$

Damit $\sqcup$ immer definiert ist, führen wir ein Symbol $\bot$ für die inkonsistente Merkmalsstruktur ein, die von allen Merkmalsstrukturen subsumiert wird.

$$\begin{bmatrix} \text{CAT} & \text{NP} \\ \text{AGR} & \begin{bmatrix} \text{NUM} & \text{Sg} \end{bmatrix} \end{bmatrix} \sqcup \begin{bmatrix} \text{CAT} & \text{V} \\ \text{AGR} & \begin{bmatrix} \text{NUM} & \text{Sg} \\ \text{PERS} & 3 \end{bmatrix} \end{bmatrix} = \bot$$

Subsumption und Unifikation (5)

Die Menge der Merkmalsstrukturen, versehen mit der Relation $\sqsubseteq$ bildet dann einen Verband (lattice): $\sqsubseteq$ ist eine partielle Ordnung und zusätzlich gilt für alle Merkmalsstrukturen S_1, S_2:

(sup) Es gibt eine Merkmalsstruktur S, so dass $S_1 \sqsubseteq S$ und $S_2 \sqsubseteq S$ und so dass S alle anderen Merkmalsstrukturen subsumiert , die ebenfalls von S_1 und S_2 subsumiert werden. S heißt dann das Supremum von $\{S_1, S_2\}$.

(inf) Es gibt eine Merkmalsstruktur S, so dass $S \sqsubseteq S_1$ und $S \sqsubseteq S_2$ und so dass S von allen anderen Merkmalsstrukturen subsumiert wird, die ebenfalls S_1 und S_2 subsumieren. S heißt dann das Infimum von $\{S_1, S_2\}$.

Das bezüglich $\sqsubseteq$ kleinste Element ist $[\,]$, das größte Element ist $\bot$.

Getypte Merkmalsstrukturen (1)

Die obigen Beispiele sind implizit davon ausgegangen, dass CAT die syntaktitsche Kategorie angibt und AGR für Kongruenz zuständig ist. D.h., folgende Merkmalsstrukturen solten nicht möglich sein:

$$\begin{bmatrix} \text{CAT} & \text{Sg} \\ \text{AGR} & \begin{bmatrix} \text{NUM} & 3 \\ \text{PERS} & \text{V} \end{bmatrix} \end{bmatrix} \qquad \begin{bmatrix} \text{CAT} & \begin{bmatrix} \text{AGR} & \begin{bmatrix} \text{PERS} & 3 \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Bisher haben wir diese Merkmalsstrukturen aber nicht ausgeschlossen, da wir die entsprechende Generalisierung nicht definieren können.

Ziel: Einschränkungen formulieren von der Art “eine Agreement Merkmalsstruktur kann höchstens die Attribute NUM, PERS und GEN haben”.

Getypte Merkmalsstrukturen (2)

Daher führen wir Typen für Merkmalsstrukturen ein:

- Jede Merkmalsstruktur hat eine Typ τ.
- Für jeden Typ τ ist festgelegt, welche Attribute für ihn zugelassen sind und von welchem Typ wiederum die Werte dieser Attribute sind.
- Typen sind in einer Typenhierarchie organisiert, in der spezifischere Typen von allgemeineren erben können.
- Unifikation wird erweitert, so dass Typen berücksichtigt werden.

Getypte Merkmalsstrukturen (3)

Die Festlegung von Typen und ihren möglichen Attributen setzt sich aus einer Typenhierarchie und Attributangaben für einzelne Typen zusammen.

$$\begin{bmatrix} \textit{agr-structure} \\ \text{AGR} \quad \textit{agr} \end{bmatrix} \quad \begin{bmatrix} \textit{agr} \\ \text{NUM} \quad \textit{num} \\ \text{GEN} \quad \textit{gen} \\ \text{PERS} \quad \textit{pers} \end{bmatrix}$$

$$\begin{bmatrix} \textit{determiner} \\ \text{QUANT} \quad \textit{quant} \end{bmatrix} \quad \begin{bmatrix} \textit{noun} \\ \text{CASE} \quad \textit{case} \end{bmatrix} \quad \begin{bmatrix} \textit{syncat} \\ \text{CAT} \quad \textit{cat} \end{bmatrix}$$

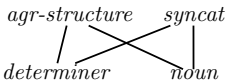
Typ *quant*: {every, most, some, none}, Typ *num*: {Sg, Pl}, Typ *gen*: {m,f,n}, Typ *pers*: $\{1, 2, 3\}$, Typ *case*: {nom, acc, dat}, Type *cat*: {N, V, NP, VP, S, ...}

Getypte Merkmalsstrukturen (4)

$$\begin{bmatrix} \textit{agr-structure} \\ \text{AGR} \quad \textit{agr} \end{bmatrix} \quad \begin{bmatrix} \textit{syncat} \\ \text{CAT} \quad \textit{cat} \end{bmatrix} \quad \begin{bmatrix} \textit{agr} \\ \text{NUM} \quad \textit{num} \\ \text{GEN} \quad \textit{gen} \\ \text{PERS} \quad \textit{pers} \end{bmatrix}$$

$$\begin{bmatrix} \textit{determiner} \\ \text{QUANT} \quad \textit{quant} \end{bmatrix} \quad \begin{bmatrix} \textit{noun} \\ \text{CASE} \quad \textit{case} \end{bmatrix}$$

Typenhierarchie:



D.h., die Attribute von *noun* werden bestimmt durch die Klassen *agr-structure*, *syncat* und *noun*.

$$\begin{bmatrix} \textit{noun} \\ \text{CAT} \quad \text{N} \\ \text{CASE} \quad \text{acc} \\ \text{AGR} \quad \begin{bmatrix} \textit{agr} \\ \text{NUM} \quad \text{Sg} \\ \text{GEN} \quad \text{m} \\ \text{PERS} \quad 3 \end{bmatrix} \end{bmatrix}$$

Erweiterungen (1)

Einige linguistische Theorien verwenden auch mengen- oder listenwertige Attribute.

Bsp.: Head-Driven Phrase Structure Grammar (HPSG) kodiert Syntaxbäume in Form von Merkmalsstrukturen, wobei die Töchter eines Knotens in Form einer Liste als Wert des Attributs DTRS ("daughters") formalisiert werden.

$$\begin{bmatrix} \textit{phrase} \\ \text{DTRS} \quad \langle \begin{bmatrix} \text{CAT} \quad \text{PRO} \\ \text{ORTH} \quad \text{I} \end{bmatrix}, \begin{bmatrix} \text{CAT} \quad \text{VP} \\ \text{DTRS} \quad \langle \begin{bmatrix} \text{CAT} \quad \text{V} \\ \text{ORTH} \quad \text{love} \end{bmatrix}, \begin{bmatrix} \text{CAT} \quad \text{NP} \\ \text{ORTH} \quad \text{New York} \end{bmatrix} \rangle \end{bmatrix} \rangle \end{bmatrix}$$

Erweiterungen (2)

- Manche Systeme arbeiten direkt mit Merkmalsstrukturen, also mit Graphen.
- Andere verwenden Beschreibungen von Merkmalsstrukturen.

Vorteil von Beschreibungen: Je nach verwendeter Logik große Expressivität (die man sich allerdings in der Regel auch mit einer entsprechenden Komplexität erkauft). Dinge, die nützlich sein könnten:

1. Disjunktion: CASE = acc $\vee$ CASE = dat
2. Negation: $\neg$(CASE = nom)
3. Pfadungleichheiten: SUBJ [CASE] $\neq$ OBJ [CASE]