Applikative Grammatiken und AB

Kilian Evang, Christian Wurm

Düsseldorf, 20.10.2022

Applikative Kategorialgrammatiken

Applikative Kategorialgrammatiken sind die einfachste Form. Man braucht folgende Zutaten (sowieso und immer):

1. Var, die Menge der atomaren Variablen (wie in jeder Aussagenlogik)
2. $Cat(Var)$, die Menge der Typen (entspricht den Formeln)
3. $\mathtt{Lex}$, das Lexikon (entspricht Terminalen)
4. eine Zuordnung ζ

Die letztere müssen wir genauer betrachten.

Applikative CG

Gegeben eine Menge von Kategorien $Cat(Var)$ und ein Lexikon $\texttt{Lex}$ definieren wir eine Kategorienzuweisung ζ als Funktion

$$\zeta : \texttt{Lex} \to \wp(Cat(Var)) \tag{1}$$

Sprich: jedes Wort unseres Lexikons bekommt eine Menge von Kategorien zugewiesen. Üblicherweise ist $|\zeta(\texttt{Lex})| \leq \omega$

- Falls $A \in Var$, dann $A \in Cat(Var)$
- Falls $\alpha, \beta \in Cat(Var)$, dann α/β, $\beta\backslash\alpha \in Cat(Var)$
- Sonst ist nichts in $Cat(Var)$

Das gilt unabhängig von Var; also falls Var definiert ist, dann ist auch $Cat(Var)$ definiert.

Applikative CG

Definition CG

Eine (applikative) Kategorialgrammatik ist ein Tupel $(Var, \texttt{Lex}, S, \zeta)$

Das bedeutet: wir haben

1. Var und damit $Cat(Var)$, also die Menge der (atomaren) syntaktischen Kategorien
2. $\texttt{Lex}$, das Lexikon. Das können Worte sein (eher linguistisch) oder Buchstaben (eher formal-sprachlich). Im letzteren Fall wäre $\texttt{Lex} = \Sigma$.
3. S ist eine *ausgezeichnete Kategorie*, wie man sie von CFG kennt
4. ζ ist die Typzuweisung.

Applikative CG

Als nächstes müssen wir den Begriff der Ableitung definieren. Applikative CG haben zwei Regeln:

$$\frac{\alpha/\beta \quad \beta}{\alpha}\ /E \qquad \frac{\beta \quad \beta\backslash\alpha}{\alpha}\ \backslash E$$

Eine **Ableitung** ist induktiv definiert:

- jede Instanz von $/E$, $\backslash E$ ist eine Ableitung, und
- falls $\frac{T}{\beta}$, $\frac{T'}{\alpha/\beta}$ Ableitungen sind, dann auch $\dfrac{\dfrac{T'}{\alpha/\beta} \quad \dfrac{T}{\beta}}{\alpha}$,
- ebenso mit $\backslash$

Applikative CG

Beachte

Der Begriff der Ableitung ist unabhängig von der Grammatik definiert!

Wir sagen: eine Grammatik $G = (Var, \mathtt{Lex}, S, \zeta)$ **generiert** ein Wort/Satz

$$\vec{w} = w_1 w_2 ... w_i \in \mathtt{Lex}^+$$

falls gilt:

- es gibt einen Ableitungsbaum mit Wurzel S,
- Blättern $\alpha_1, ..., \alpha_i$, und
- f.a. $j \in \{1, ..., i\}$ gilt: $\alpha_j \in \zeta(w_j)$

Wir schreiben $L(G)$ für die Menge der Worte die von G generiert werden.

Applikative CG: Übung

Übung 1

Schreiben Sie eine CG für die Sprache $\{a^n b^n : n \in \mathbb{N}\}$

Übung 2

Schreiben Sie eine CG für die Dyk-Sprache über a, b ($a \cong$ Klammer auf, $b \cong$ Klammer zu)

Applikative CG: Übung

Übung 1

Schreiben Sie eine CG für die Sprache $\{a^n b^n : n \in \mathbb{N}\}$

Lösung:

- $\zeta(a) = \{S/N, (S/N)/S\}$,
- $\zeta(b) = \{N\}$

Applikative CG: Übung

Übung 2

Schreiben Sie eine CG für die Dyk-Sprache über a, b ($a \cong$ Klammer auf, $b \cong$ Klammer zu)

- $\zeta(a) = \{S/N, (S/N)/S, (S/S)/N, ((S/S)/N)/S\}$,
- $\zeta(b) = \{N\}$

Einige Ergebnisse

Sei $L \subseteq \Sigma^+$ eine formale Sprache (ohne leeres Wort!). Dann gilt:

Theorem

$L = L(G)$ für eine Applikative CG genau dann wenn $L = L(G_{CF}$ für eine kontextfreie Grammatik G_{CF}.

Sprich: wir generieren genau die kontextfreien Sprachen.

Wir werden uns dieses Ergebnis nächste Woche genauer anschauen!

Der AB-Kalkül

Benannt nach den Entdeckern Ajduckiewicz und Bar-Hillel ist der **AB-Kalkül** eine echte und mächtige Erweiterung der applikativen Regeln.

Wichtig: hier wird zunächst eine Logik definiert, und dann die Grammatik!

Der AB-Kalkül wird hier präsentiert als **Gentzen-Kalkül**. Schauen wir uns das mal an:

Der AB-Kalkül

(*ax*) $\alpha \vdash \alpha$

$(\mathbf{I}-/)\quad \dfrac{\Gamma, \alpha \vdash \beta}{\Gamma \vdash \beta/\alpha}$

$(\mathbf{I}-\backslash)\quad \dfrac{\alpha, \Gamma \vdash \beta}{\Gamma \vdash \alpha\backslash\beta}$

$(/-\mathbf{I})\quad \dfrac{\Delta, \beta, \Theta \vdash \gamma \qquad \Gamma \vdash \alpha}{\Delta, \beta/\alpha, \Gamma, \Theta \vdash \gamma}$

$(\backslash-\mathbf{I})\quad \dfrac{\Delta, \beta, \Theta \vdash \gamma \qquad \Gamma \vdash \alpha}{\Delta, \Gamma, \alpha\backslash\beta, \Theta \vdash \gamma}$

(schnitt) $\dfrac{\Gamma, \alpha, \Gamma' \vdash \beta \qquad \Delta \vdash \alpha}{\Gamma, \Delta, \Gamma' \vdash \beta}$

Der AB-Kalkül

Übung 1

Finden Sie eine Ableitung für $\beta \vdash \alpha/(\beta\backslash\alpha)$

Übung 2

Finden Sie eine Ableitung für $\alpha/\beta, \beta/\gamma \vdash \alpha/\gamma$

Übung 3

Finden Sie eine Ableitung für $\gamma\backslash(\alpha/\beta) \vdash (\gamma\backslash\alpha)/\beta$

AB-Grammatiken

Eine AB Grammatik $G = (Var, \mathtt{Lex}, S, \zeta)$ generiert ein Wort/Satz $\vec{w} = w_1 w_2 ... w_i \in \mathtt{Lex}^+$, falls gilt:

- es gibt einen Beweis im AB-Kalkül von $\alpha_1, ..., \alpha_i \vdash S$, und
- f.a. $j \in \{1, ..., i\}$ gilt: $\alpha_j \in \zeta(w_j)$

Das ist natürlich praktisch und schnell erledigt (wir lassen einfach die Logik für uns arbeiten).

Der Linguist wird aber die Bäumchen vermissen! Also präsentieren wir noch eine andere Variante von AB.

Natürliches Schließen

- Praktisch jede Logik, die einen Gentzen Kalkül hat, hat auch einen Kalkül des nat. Schließens.
- Die Übersetzung ist mehr oder weniger Standard
- Die Philosophen lieben das, Mathematiker und Informatiker weniger.
- Der Kalkül beweist *keine* Sequenten, sondern *Formeln*, und zwar von *Annahmen*.

Und so sieht das aus:

Natürliches Schließen

$$(\mathsf{E}/)\ \frac{\alpha/\beta \quad \beta}{\alpha} \qquad\qquad (\mathsf{E}\backslash)\ \frac{\beta \quad \beta\backslash\alpha}{\alpha}$$

$$(\mathsf{I}/)\ \frac{\dfrac{\dfrac{\dots \quad [\beta]}{\vdots}}{\alpha}}{\alpha/\beta} \qquad\qquad (\mathsf{I}\backslash)\ \frac{\dfrac{\dfrac{[\beta] \quad \dots}{\vdots}}{\beta}}{\beta\backslash\alpha}$$

Die Intro-Regeln sind etwas schwer zu interpretieren: β ist hier eine **Annahme**, die wir mit der Regel **auflösen**. Die Annahme muss dabei am linken/rechten Rand des Beweises stehen!

Der AB-Kalkül

Diese Formulierung ist äquivalent, erlaubt uns aber die geohnte Form des Beweisbaumes zu erhalten.

Übung 1

Finden Sie eine Ableitung von $\alpha/(\beta\backslash\alpha)$ von Annahme β

Übung 2

Finden Sie eine Ableitung von α/γ von Annahmen α/β β/γ

Natürliches Schließen

Auch hier sind Beweise Bäume. In dieser Form gilt:
$G = (Var, \mathrm{Lex}, S, \zeta)$ generiert ein Wort/Satz
$\vec{w} = w_1 w_2 ... w_i \in \mathrm{Lex}^+$, falls gilt:

- es gibt im (n.s.) im AB-Kalkül einen Beweis von S von den (geordneten) Annahmen $\alpha_1, ..., \alpha_i \vdash S$, und
- f.a. $j \in \{1, ..., i\}$ gilt: $\alpha_j \in \zeta(w_j)$

Die Ableitungen sind hier ähnlich den applikativen Form, der einzige Unterschied ist: wir können Annahmen machen und auflösen!

Der AB-Kalkül: ein linguistisches Beispiel

Folgendes Beispiel:

(1) Jede Katze schläft.

(2) Ein Hund sieht die Katze.

Sprachsemantik

Wir können definieren (gegeben $G = (Var, \mathtt{Lex}, S, \zeta)$)

$$\|\beta\|_G = \{w_1 ... w_i \in \mathtt{Lex}^+ : \alpha_1, ... \alpha_i \vdash \beta,\ \alpha_j \in \zeta(w_j)\},$$

Dann gilt qua Definition: $L(G) = \|S\|_G$. Allgemeiner gilt folgendes Ergebnis:

Lemma

$\alpha \vdash \beta$ *impliziert* $\|\alpha\|_G \subseteq \|\beta\|_G$

Das bedeutet: die Semantik ist **korrekt**. Beweis ist Induktion über die Länge der Ableitung.

Zusammenfassung

Kategorialgrammatiken sind superspannend!