Gramatyka Kategorialna Języka Polskiego

Wojciech Jaworski

Instytut Informatyki
Uniwersytet Warszawski

3 października 2011

Logika liniowa posiada następujące zalety jako narzędzie do analizy składniowej:

- każde założenie musi być wykorzystanie w dowodzie dokładnie raz, co odpowiada temu, że każde słowo musi pojawić się dokładnie raz w drzewie wywodu;
- spójniki logiczne w tej logice w naturalny sposób wyrażają pojęcia używane przy opisie języka;
- istnieje pełny system dowodowy i wyniki teoretyczne dotyczące własności logiki liniowej.

Spis treści

Typy

- W wyniku segmentacji zdanie zostaje zamienione na ciąg tokenów $\gamma_1, \gamma_2, \ldots, \gamma_n$,
- a w wyniku analizy morfologicznej każdemu tokenowi zostaje przypisana formuła (typ) $\varphi_1, \ldots, \varphi_n$.
- na przykład zdanie „Jan widzi stół.” zostanie podzielone na tokeny:

$$\text{Jan}_1,\ \text{widzi}_2,\ \text{stół}_3,\ ._4$$

  którym przyporządkowane zostaną typy

$$\mathrm{NP},\ (\mathrm{VP} \setminus \mathrm{NP}) \,/\, \mathrm{NP},\ \mathrm{NP},\ \mathrm{S} \setminus \mathrm{VP}$$

- Typy składają się z symboli atomowych reprezentujących rodzaj frazy (np. NP, VP) i kategorie gramatyczne takie jak przypadek, liczba, rodzaj czy osoba (np. $\mathrm{nom}, \mathrm{sg}, \mathrm{m}_3$, sec)
- Symbole te łączone są za pomocą spójników logicznych.

Wywód gramatyczny

- drzewem wyprowadzenia (wywodu gramatycznego) jest dla nas dowód w intuicjonistycznej niekomutatywnej logice liniowej (rachunku Lambeka).
- Założeniami w dowodzie są dla nas sekwenty utworzone z wyodrębnionych w wyniku segmentacji tokenów i przypisanych im typów:

$$\gamma_1 \vdash \varphi_1, \quad \gamma_2 \vdash \varphi_2, \quad \dots \quad , \gamma_n \vdash \varphi_n.$$

- Tezą w dowodzie jest sekwent

$$\gamma_1, \gamma_2, \dots, \gamma_n \vdash \mathrm{S}$$

oznaczający, że z ciągu złożonego ze wszystkich tokenów potrafimy wywieść całe zdanie.

Przykładowy wywód gramatyczny

- będziemy korzystać z aksjomatów

$$\text{Jan}_1 \vdash \mathrm{NP},\ \text{widzi}_2 \vdash (\mathrm{VP} \backslash \mathrm{NP}) / \mathrm{NP},\ \text{stół}_3 \vdash \mathrm{NP},\ ._4 \vdash \mathrm{S} \backslash \mathrm{VP}$$

- oraz z reguł wnioskowania *modus ponens*:

$$\frac{\Gamma \vdash \psi / \varphi \qquad \Delta \vdash \varphi}{\Gamma, \Delta \vdash \psi} \qquad\qquad \frac{\Delta \vdash \varphi \qquad \Gamma \vdash \psi \backslash \varphi}{\Delta, \Gamma \vdash \psi}$$

- otrzymamy następujące drzewo wywodu:

$$\cfrac{\cfrac{\text{Jan}_1 \vdash \mathrm{NP} \qquad \cfrac{\text{widzi}_2 \vdash (\mathrm{VP} \backslash \mathrm{NP}) / \mathrm{NP} \qquad \text{stół}_3 \vdash \mathrm{NP}}{\text{widzi}_2, \text{stół}_3 \vdash \mathrm{VP} \backslash \mathrm{NP}}}{\text{Jan}_1, \text{widzi}_2, \text{stół}_3 \vdash \mathrm{VP}} \qquad ._4 \vdash \mathrm{S} \backslash \mathrm{VP}}{\text{Jan}_1, \text{widzi}_2, \text{stół}_3, ._4 \vdash \mathrm{S}}$$

Rachunek sekwentów Gentzena

- aksjomat

$$\varphi \vdash \varphi \text{ (Axiom)}$$

  gdzie φ jest dowolną formułą
- reguła cięcia

$$\frac{\Gamma \vdash \varphi \qquad \Delta, \varphi, \Delta' \vdash \sigma}{\Delta, \Gamma, \Delta' \vdash \sigma} \text{ (Cut)}$$

- reguły wprowadzania każdego z operatorów po lewej i prawej stronie sekwentu.

Implikacja

- Implikacje liniowe $\psi \,/\, \varphi$ i $\psi \setminus \varphi$ reprezentują typ funkcyjny, w którym przesłanka φ jest przetworzona w konkluzję ψ, przy czym zostaje całkowicie „zużyta", czy też „skonsumowana".
- Implikacja / pobiera przesłankę prawej strony, a \ z lewej.
- Za pomocą implikacji będziemy wyrażać fakt, że fraza funktorem pobierającym jako argument inną frazę.
- Reguły wnioskowania dla implikacji w rachunku sekwentów Gentzena:

$$\frac{\Delta, \psi, \Delta' \vdash \sigma \qquad \Gamma \vdash \varphi}{\Delta, \psi \,/\, \varphi, \Gamma, \Delta' \vdash \sigma}\ (\mathrm{L}/) \qquad \frac{\Gamma, \varphi \vdash \psi}{\Gamma \vdash \psi \,/\, \varphi}\ (\mathrm{R}/)$$

$$\frac{\Gamma \vdash \varphi \qquad \Delta, \psi, \Delta' \vdash \sigma}{\Delta, \Gamma, \psi \setminus \varphi, \Delta' \vdash \sigma}\ (\mathrm{L}\setminus) \qquad \frac{\varphi, \Gamma \vdash \psi}{\Gamma \vdash \psi \setminus \varphi}\ (\mathrm{R}\setminus)$$

Dowód reguły *modus ponens*

- Za pomocą reguł

$$\varphi \vdash \varphi \text{ (Axiom)} \qquad \frac{\Gamma \vdash \varphi \qquad \Delta, \varphi, \Delta' \vdash \sigma}{\Delta, \Gamma, \Delta' \vdash \sigma} \text{ (Cut)}$$

$$\frac{\Delta, \psi, \Delta' \vdash \sigma \qquad \Gamma \vdash \varphi}{\Delta, \psi / \varphi, \Gamma, \Delta' \vdash \sigma} \text{ (L/)} \qquad \frac{\Gamma, \varphi \vdash \psi}{\Gamma \vdash \psi / \varphi} \text{ (R/)}$$

- dowodzimy

$$\frac{\Delta \vdash \psi / \varphi \qquad \Gamma \vdash \varphi}{\Delta, \Gamma \vdash \psi}$$

- Dowód:

$$\dfrac{\Gamma \vdash \varphi \qquad \dfrac{\Delta \vdash \psi / \varphi \qquad \dfrac{\psi \vdash \psi \qquad \varphi \vdash \varphi}{\psi / \varphi, \varphi \vdash \psi} \text{ (L/)}}{\Delta, \varphi \vdash \psi} \text{ (Cut)}}{\Delta, \Gamma \vdash \psi} \text{ (Cut)}$$

Tensor

- Obiekt typu $\varphi \otimes \psi$ to para obiektów: pierwszy typu φ, drugi typu ψ.
- Utworzenie takiej pary wymaga odrębnych zasobów dla każdego członu.
- Skonsumowanie takiej pary polega na zużyciu obu członów
- Operacja $\otimes$ pozwala otypować token za pomocą listy jego kategorii gramatycznych:

$$\text{Jan}_1 \vdash \mathrm{NP} \otimes (\mathrm{sg} \otimes (\mathrm{nom} \otimes \mathrm{m}_1))$$

- Reguły wnioskowania:

$$\frac{\Gamma, \varphi, \psi, \Delta \vdash \sigma}{\Gamma, \varphi \otimes \psi, \Delta \vdash \sigma} \ (\mathrm{L}\otimes) \qquad \frac{\Gamma \vdash \varphi \qquad \Delta \vdash \psi}{\Gamma, \Delta \vdash \varphi \otimes \psi} \ (\mathrm{R}\otimes)$$

- Operacja $\otimes$ jest łączna, więc nawiasy w zapisie będziemy pomijać.

Wraz

- Obiekt typu $\varphi \,\&\, \psi$ to „wirtualna” para obiektów (pierwszy typu φ, drugi typu ψ), z których każdy może potencjalnie zostać utworzony (z tych samych zasobów).
- Jest to „prawo wyboru” pomiędzy obiektem typu φ i obiektem typu ψ.
- Operator & pozwala otypować tokeny mające niejednoznaczne kategorie gramatyczne:

$$\text{stół}_3 \vdash \mathrm{NP} \otimes \mathrm{sg} \otimes (\mathrm{nom} \,\&\, \mathrm{acc}) \otimes \mathrm{m}_3$$

- Reguły wnioskowania:

$$\frac{\Gamma, \varphi, \Delta \vdash \sigma}{\Gamma, \varphi \,\&\, \psi, \Delta \vdash \sigma} \text{(L\&)} \quad \frac{\Gamma, \psi, \Delta \vdash \sigma}{\Gamma, \varphi \,\&\, \psi, \Delta \vdash \sigma}$$

$$\frac{\Gamma \vdash \varphi \qquad \Gamma \vdash \psi}{\Gamma \vdash \varphi \,\&\, \psi} \text{(R\&)}$$

Zero

- Rozpatrzmy wyraz „panaceum”. Możemy nadać mu typ

  $$\text{panaceum}_i \vdash \mathrm{NP} \otimes \mathrm{sg} \otimes (\mathrm{nom} \,\&\, \mathrm{gen} \,\&\, \mathrm{dat} \,\&\, \mathrm{acc} \,\&\, \mathrm{inst} \,\&\, \mathrm{loc} \,\&\, \mathrm{voc}) \otimes \mathrm{n}_2,$$

- Ale typ ten będzie skomplikowany i nie będzie oddawać istoty zjawiska jakim jest to, że ’panaceum’ jest w liczbie pojedynczej nieodmienne, czyli posiada każdą możliwą wartość przypadka.
- Logika liniowa dostarcza nam stałą 0 (typ pusty) o następującej własności:

  $$0 \vdash \varphi \text{ (L0)}$$

- Możemy teraz zapisać na nowo typ dla ’panaceum’:

  $$\text{panaceum}_i \vdash \mathrm{NP} \otimes \mathrm{sg} \otimes 0 \otimes \mathrm{n}_2.$$

- Pierwotny typ uzyskamy za pomocą następującego wnioskowania:

$$\dfrac{\text{panaceum}_i \vdash \mathrm{NP} \otimes 0 \qquad \dfrac{\dfrac{\mathrm{NP} \vdash \mathrm{NP} \qquad 0 \vdash \mathrm{nom} \,\&\, \cdots \,\&\, \mathrm{voc}}{\mathrm{NP}, 0 \vdash \mathrm{NP} \otimes (\mathrm{nom} \,\&\, \cdots \,\&\, \mathrm{voc})}}{\mathrm{NP} \otimes 0 \vdash \mathrm{NP} \otimes (\mathrm{nom} \,\&\, \cdots \,\&\, \mathrm{voc})}}{\text{panaceum}_i \vdash \mathrm{NP} \otimes (\mathrm{nom} \,\&\, \cdots \,\&\, \mathrm{voc})}$$

Plus

- Obiekt typu $\varphi \oplus \psi$ to para złożona z obiektu typu φ lub obiektu typu ψ oraz wskaźnika pokazującego, o który obiekt typ faktycznie chodzi.
- Utworzenie obiektu typu $\varphi \oplus \psi$ polega na utworzeniu jednego obiektu (typu φ bądź ψ) i zaopatrzeniu go w odpowiednią flagę.
- Skonsumowanie obiektu typu $\varphi \oplus \psi$ polega na jego otwarciu i spożyciu zawartości.
- Reguły dla $\oplus$:

$$\frac{\Gamma, \varphi, \Delta \vdash \sigma \qquad \Gamma, \psi, \Delta \vdash \sigma}{\Gamma, \varphi \oplus \psi, \Delta \vdash \sigma} \text{(L}\oplus\text{)}$$

$$\frac{\Gamma \vdash \varphi}{\Gamma \vdash \varphi \oplus \psi} \text{(R}\oplus\text{)} \frac{\Gamma \vdash \psi}{\Gamma \vdash \varphi \oplus \psi}$$

- $\oplus$ jest operatorem dualnym do &. Pozwala zapisać niejednoznaczność argumentów implikacji:

$$\text{jest}_i \vdash \mathrm{VP}/((\mathrm{NP} \otimes \mathrm{inst}) \oplus (\mathrm{AdjP} \otimes \mathrm{nom}))$$

- Jest też stosowany przy koordynacji (o czym później)

Top

- Słowo „widzi” akceptuje podmiot w dowolnym rodzaju to warto zastąpić wyliczenie wszystkich możliwości przez uniwersalną stałą.
- Wyliczenie wszystkich możliwych rodzajów warto zastąpić przez uniwersalną stałą.
- Stałą tą będzie $\top$ o następującej właściwości:

$$\varphi \vdash \top \ (\mathrm{R}\top)$$

- $\top$ pozwala w zwarty sposób zdefiniować typ słowa „widzi”:

$$\text{widzi}_i \vdash (\mathrm{VP} \setminus \mathrm{NP} \otimes \mathrm{sg} \otimes \mathrm{nom} \otimes \top) / \mathrm{NP} \otimes \top \otimes \mathrm{acc} \otimes \top$$

- $\top$ jest stałą dualną do 0
- 0 jest to $\star$ modulująca, a $\top$ jest to $\star$ selektywna.

Jedynka

- Stała 1 pozwala reprezentować białe znaki.
- Związane są z nią następujące reguły:

$$\frac{\Gamma \vdash \sigma}{\Gamma, 1 \vdash \sigma}\ (\mathrm{L1})\ \frac{\Gamma \vdash \sigma}{1, \Gamma \vdash \sigma} \qquad\qquad \vdash 1\ (\mathrm{R1})$$

- Na przykład, niech $\gamma_i \vdash \varphi$ oraz $\gamma_{i+1} \vdash 1$.
- Pokarzemy, że wtedy $\gamma_i, \gamma_{i+1} \vdash \varphi$:

$$\frac{\gamma_{i+1} \vdash 1 \qquad \dfrac{\gamma_i \vdash \varphi}{\gamma_i, 1 \vdash \varphi}}{\gamma_i, \gamma_{i+1} \vdash \varphi}$$

Argumenty opcjonalne

- Wprowadzimy operator unarny $\cdot^{0-1}$ wskazujący, że dany argument implikacji może być użyty 0 lub 1 raz.
- Jest one szczególnym przypadkiem stosowanego w logice liniowej operatora ?.
- Reguły wnioskowania:

$$\frac{\Gamma, \psi \,/\, \varphi, \Delta \vdash \sigma}{\Gamma, \psi \,/\, \varphi^{0-1}, \Delta \vdash \sigma} \ (\mathrm{L}/^{0-1}) \ \frac{\Gamma, \psi, \Delta \vdash \sigma}{\Gamma, \psi \,/\, \varphi^{0-1}, \Delta \vdash \sigma}$$

$$\frac{\Gamma, \psi \setminus \varphi, \Delta \vdash \sigma}{\Gamma, \psi \setminus \varphi^{0-1}, \Delta \vdash \sigma} \ (\mathrm{L}\backslash^{0-1}) \ \frac{\Gamma, \psi, \Delta \vdash \sigma}{\Gamma, \psi \setminus \varphi^{0-1}, \Delta \vdash \sigma}$$

- Powyższe reguły oznaczają, że formuła $\psi \,/\, \varphi^{0-1}$ zachowuje się jak ψ & $(\psi \,/\, \varphi)$.

Argumenty wielokrotne

- Wprowadzimy operator unarny $\cdot^{1-\infty}$ oznaczający, że dany argument implikacji może być użyty 1 lub więcej razy.
- Jest one szczególnym przypadkiem stosowanego w logice liniowej operatora ?.
- Reguły wnioskowania:

$$\frac{\Gamma, (\psi \,/\, \varphi^{1-\infty}) \,/\, \varphi, \Delta \vdash \sigma}{\Gamma, \psi \,/\, \varphi^{1-\infty}, \Delta \vdash \sigma} \;(\mathsf{L}/^{1-\infty})\; \frac{\Gamma, \psi \,/\, \varphi, \Delta \vdash \sigma}{\Gamma, \psi \,/\, \varphi^{1-\infty}, \Delta \vdash \sigma}$$

$$\frac{\Gamma, (\psi \setminus \varphi^{1-\infty}) \setminus \varphi, \Delta \vdash \sigma}{\Gamma, \psi \setminus \varphi^{1-\infty}, \Delta \vdash \sigma} \;(\mathsf{L}\setminus^{1-\infty})\; \frac{\Gamma, \psi \setminus \varphi, \Delta \vdash \sigma}{\Gamma, \psi \setminus \varphi^{1-\infty}, \Delta \vdash \sigma}$$

- Opcjonalny argument wielokrotny $\varphi^{0-\infty}$ definiujemy jako $(\varphi^{1-\infty})^{0-1}$.

Implikacja obustronna

- Implikację obustronną $\psi \mid \varphi$ definiujemy jako $(\psi / \varphi) \,\&\, (\psi \backslash \varphi)$.
- Zauważmy, że $(\psi \mid \varphi) \mid \varphi_2$ jest równoważne $((\psi / \varphi_1) / \varphi_2) \,\&\, ((\psi \backslash \varphi_1) / \varphi_2) \,\&\, ((\psi / \varphi_1) \backslash \varphi_2) \,\&\, ((\psi \backslash \varphi_1) \backslash \varphi_2)$.
- Możemy wyprowadzić następującą regułę dla implikacji obustronnej:

$$\frac{\psi / \varphi \vdash \sigma}{\psi \mid \varphi \vdash \sigma}\ (\mathrm{L}|)\ \frac{\psi \backslash \varphi \vdash \sigma}{\psi \mid \varphi \vdash \sigma}$$

Zbiór argumentów

- Zbiór argumentów $\psi\{|_1 \varphi_1, |_2 \varphi_2, \dots, |_n \varphi_n\}$, gdzie $|_i \in \{/, \backslash, |\}$, jest to zbiór połączonych za pomocą spójnika & formuł postaci $(\dots((\psi \,|_{\sigma(1)}\, \varphi_{\sigma(1)}) \,|_{\sigma(2)}\, \varphi_{\sigma(2)}) \dots \,|_{\sigma(n)}\, \varphi_{\sigma(n)})$, gdzie σ jest dowolną permutacją zbioru *n*-elementowego.
- Będziemy korzystać z następującej reguły wnioskowania:

$$\frac{(\psi\{|_1 \varphi_1, |_2 \varphi_2, \dots, |_{i-1} \varphi_{i-1}, |_{i+1} \varphi_{i+1}, \dots, |_n \varphi_n\}) \,|_i\, \varphi_i \vdash \sigma}{\psi\{|_1 \varphi_1, |_2 \varphi_2, \dots, |_n \varphi_n\} \vdash \sigma} \;(\mathrm{L}\{\})$$

Wraz z parametrem

- Notacja $\&_{\alpha\in\{\gamma_1,\dots,\gamma_n\}}\,\tau$ jest skrótem dla zapisu formuły $\tau[\alpha := \gamma_1]\ \&\cdots\&\ \tau[\alpha := \gamma_n]$, czyli oznacza potencjalną możliwość utworzenia formuły τ ze zmienną α zastąpioną przez każdą z formuł γ_i.
- Mamy następujące reguły:

$$\frac{\tau[\alpha := \gamma_i] \vdash \sigma}{\&_{\alpha\in\{\gamma_1,\dots,\gamma_n\}}\,\tau \vdash \sigma}\ (\mathrm{L}\&)$$

$$\frac{\tau \vdash \psi\,/\,\varphi \qquad \left(\&_{\alpha\in\{\gamma_1,\dots,\gamma_n\}}\,\psi\right)/\,\varphi \vdash \sigma}{\&_{\alpha\in\{\gamma_1,\dots,\gamma_n\}}\,\tau \vdash \sigma}\ (\mathrm{L}\&/\mathrm{a})\ ,$$

gdy α nie występuje w φ

$$\frac{\tau \vdash \psi\,/\,\varphi \qquad \psi[\alpha := \gamma_i]\,/\,\varphi[\alpha := \gamma_i] \vdash \sigma}{\&_{\alpha\in\{\gamma_1,\dots,\gamma_n\}}\,\tau \vdash \sigma}\ (\mathrm{L}\&/\mathrm{b})\ ,$$

gdy α występuje w φ
oraz analogiczne reguły L$\&\backslash$a i L$\&\backslash$b

Spis treści

Przykładowy wywód gramatyczny zaopatrzony w semantykę

- będziemy korzystać z założeń

$$\mathsf{Jan}_1 \vdash \mathrm{NP} : \begin{bmatrix} \textsc{lexeme} & \mathsf{Jan} \\ \textsc{cat} & \mathrm{subst} \end{bmatrix}$$

$$\mathsf{widzi}_2 \vdash \mathrm{VP} \backslash \mathrm{NP} : \lambda x. \begin{bmatrix} \textsc{lexeme} & \mathit{lexeme} \\ \textsc{cat} & \mathrm{verb} \\ \textsc{args} & \begin{bmatrix} \textsc{Subj} & x \end{bmatrix} \end{bmatrix}$$

- oraz z reguły wnioskowania *modus ponens*:

$$\frac{\Delta \vdash \varphi : M \qquad \Gamma \vdash \psi \backslash \varphi : N}{\Delta, \Gamma \vdash \psi : NM}$$

- otrzymamy następujące drzewo wywodu:

$$\frac{\mathsf{Jan}_1 \vdash \mathrm{NP} : \begin{bmatrix} \textsc{lexeme} & \mathsf{Jan} \\ \textsc{cat} & \mathrm{subst} \end{bmatrix} \qquad \mathsf{widzi}_2 \vdash \mathrm{VP} \backslash \mathrm{NP} : \lambda x. \begin{bmatrix} \textsc{lexeme} & \mathit{lexeme} \\ \textsc{cat} & \mathrm{verb} \\ \textsc{args} & \begin{bmatrix} \textsc{Subj} & x \end{bmatrix} \end{bmatrix}}{\mathsf{Jan}_1, \mathsf{widzi}_2 \vdash \mathrm{VP} : \begin{bmatrix} \textsc{lexeme} & \mathit{lexeme} \\ \textsc{cat} & \mathrm{verb} \\ \textsc{args} & \begin{bmatrix} \textsc{Subj} & \begin{bmatrix} \textsc{lexeme} & \mathsf{Jan} \\ \textsc{cat} & \mathrm{subst} \end{bmatrix} \end{bmatrix} \end{bmatrix}}$$

Semantyka dla rachunku sekwentów Gentzena

$$\varphi : M \vdash \varphi : M \text{ (Axiom)}, \qquad \text{o ile } \varphi \text{ jest poprawnym typem dla } M$$

$$\varphi : M \vdash \top : M \text{ (R}\top\text{)}, \qquad \text{o ile } \varphi \text{ jest poprawnym typem dla } M$$

$$\frac{\Gamma \vdash \varphi : N \qquad \Delta, \varphi : N, \Delta' \vdash \sigma : S}{\Delta, \Gamma, \Delta' \vdash \sigma : S} \text{ (Cut)}$$

- Reguły są napisane w sposób bezkontekstowy ze względu na semantykę.
- W lewym dolnym rogu reguły znajduje się wejściowa semantyka, w prawym dolnym wyliczona.
- $\top$ jest najbardziej ogólnym typem.
- Każdy term, który ma typ jest również typu $\top$.

Zero

$$0 : \textbf{emp} \vdash \varphi : \varepsilon_\varphi(\textbf{emp}) \text{ (L0)}$$

- **emp** jest jedynym istniejącym termem typu 0.
- ε_φ jest kanoniczną funkcją z typu pustego w typ φ.
- Zdefiniujemy

$$\varepsilon_{\text{gen\&acc}}(0) = \langle [\text{CASE: gen}], [\text{CASE: acc}] \rangle,$$

  i analogicznie dla każdego innego typu, dla którego będzie to potrzebne.
- Z uwagi na to, że nie ma reguły wprowadzania 0, wszystkie typy φ, dla których potrzebna jest funkcja ε_φ są wyznaczone przez leksykon.

Wraz

$$\frac{\Gamma, \varphi : \pi_1(M), \Delta \vdash \sigma : S}{\Gamma, \varphi \,\&\, \psi : M, \Delta \vdash \sigma : S} \text{(L\&)} \frac{\Gamma, \psi : \pi_2(M), \Delta \vdash \sigma : S}{\Gamma, \varphi \,\&\, \psi : M, \Delta \vdash \sigma : S}$$

$$\frac{\Gamma \vdash \varphi : M \qquad \Gamma \vdash \psi : N}{\Gamma \vdash \varphi \,\&\, \psi : \langle M, N \rangle} \text{(R\&)}$$

- $\langle s, t \rangle$ jest parą dwu obiektów, z których dokładnie jeden może zostać utworzony.
- Operacje π_i to wybór obiektu, czyli rzutowanie na i-tą współrzędną pary.
- Formuła $\pi_i(\langle M_1, M_2 \rangle)$ redukuje się do M_i.

Tensor

$$\frac{\Gamma, \varphi : x, \psi : y, \Delta \vdash \sigma : S}{\Gamma, \varphi \otimes \psi : M, \Delta \vdash \sigma : \mathbf{let}\ x \otimes y = M\ \mathbf{in}\ S}\ (\mathrm{L}\otimes)$$

$$\frac{\Gamma \vdash \varphi : M \qquad \Delta \vdash \psi : N}{\Gamma, \Delta \vdash \varphi \otimes \psi : M \otimes N}\ (\mathrm{R}\otimes)$$

- Obiekt postaci $M \otimes N$ to para, której oba człony muszą być wykorzystane.
- Instrukcja **let** $x \otimes y = M$ **in** S oznacza, że w termie S w miejsce wystąpień zmiennych x i y podstawiamy odpowiednio pierwszy i drugi obiekt z pary, którą jest term M.
- Liniowość wymaga, żeby w termie S było dokładnie jedno wystąpienie zmiennej wolnej x i dokładnie jedno zmiennej wolnej y.
- Formuła **let** $x \otimes y = M \otimes N$ **in** S redukuje się do $S[x := M, y := N]$.

Plus

$$\frac{\Gamma, \varphi : x, \Delta \vdash \sigma : S \qquad \Gamma, \psi : y, \Delta \vdash \sigma : T}{\Gamma, \varphi \oplus \psi : M, \Delta \vdash \sigma : \textbf{case } M \textbf{ of inl}(x) \Rightarrow S \mid \textbf{inr}(y) \Rightarrow T} \ (\mathrm{L}\oplus)$$

$$\frac{\Gamma \vdash \varphi : M}{\Gamma \vdash \varphi \oplus \psi : \textbf{inl}(M)} \ (\mathrm{R}\oplus) \ \frac{\Gamma \vdash \psi : M}{\Gamma \vdash \varphi \oplus \psi : \textbf{inr}(M)}$$

- Operacje **inl** oraz **inr** to lewe i prawe włożenie.
- Instrukcja $\textbf{case } M \textbf{ of inl}(x) \Rightarrow S \mid \textbf{inr}(y) \Rightarrow T$ działa następująco:
  - jeżeli term M jest postaci $\textbf{inl}(R)$ to zwraca term S z R podstawionym na zmienną x,
  - jeżeli zaś M jest postaci $\textbf{inr}(R)$ to zwraca term T z R podstawionym na zmienną y.
- Mamy tu następujące redukcje:

$$\textbf{case inl}(R) \textbf{ of inl}(x) \Rightarrow S \mid \textbf{inr}(y) \Rightarrow T \ \rightarrow \ S[x := R]$$

$$\textbf{case inr}(R) \textbf{ of inl}(x) \Rightarrow S \mid \textbf{inr}(y) \Rightarrow T \ \rightarrow \ T[y := R]$$

Implikacja

$$\frac{\Gamma \vdash \varphi : N \qquad \Delta, \psi : MN, \Delta' \vdash \sigma : S}{\Delta, \psi \,/\, \varphi : M, \Gamma, \Delta' \vdash \sigma : S} \text{ (L/)}$$

$$\frac{\Gamma, \varphi : x \vdash \psi : M}{\Gamma \vdash \psi \,/\, \varphi : \lambda x.M} \text{ (R/)}$$

$$\frac{\Gamma \vdash \varphi : N \qquad \Delta, \psi : MN, \Delta' \vdash \sigma : S}{\Delta, \Gamma, \psi \backslash \varphi : M, \Delta' \vdash \sigma : S} \text{ (L}\backslash\text{)}$$

$$\frac{\varphi : x, \Gamma \vdash \psi : M}{\Gamma \vdash \psi \backslash \varphi : \lambda x.M} \text{ (R}\backslash\text{)}$$

- Semantyką implikacji jest aplikacja argumentu do funkcji oraz λ-abstrakcja.
- Formuła $(\lambda x.M)N$ redukuje się do $M[x := N]$.

Jedynka

$$\frac{\Gamma \vdash \sigma : S}{\Gamma, 1 : P \vdash \sigma : \textbf{let} \bullet = P \textbf{ in } S} \text{(L1)} \quad \frac{\Gamma \vdash \sigma : S}{1 : P, \Gamma \vdash \sigma : \textbf{let} \bullet = P \textbf{ in } M}$$

$$\vdash 1 : \bullet \text{ (R1)}$$

- Formuła **let** $\bullet = \bullet$ **in** M redukuje się do M

Argumenty opcjonalne i wielokrotne

$$\frac{\Gamma, \psi \,/\, \varphi : \lambda x.M\, \mathbf{add}(x, \mathbf{null}), \Delta \vdash \sigma : S}{\Gamma, \psi \,/\, \varphi^{0-1} : M, \Delta \vdash \sigma : S}\ (\mathrm{L}/^{0-1})$$

$$\frac{\Gamma, \psi : M\, \mathbf{null}, \Delta \vdash \sigma : S}{\Gamma, \psi \,/\, \varphi^{0-1} : M, \Delta \vdash \sigma : S}$$

$$\frac{\Gamma, (\psi \,/\, \varphi^{1-\infty}) \,/\, \varphi : \lambda x \lambda y.M\, \mathbf{add}(x, y), \Delta \vdash \sigma : S}{\Gamma, \psi \,/\, \varphi^{1-\infty} : M, \Delta \vdash \sigma : S}\ (\mathrm{L}/^{1-\infty})$$

$$\frac{\Gamma, \psi \,/\, \varphi : \lambda x.M\, \mathbf{add}(x, \mathbf{null}), \Delta \vdash \sigma : S}{\Gamma, \psi \,/\, \varphi^{1-\infty} : M, \Delta \vdash \sigma : S}$$

- Semantykę dla argumentów opcjonalnych i wielokrotnych definiuję za pomocą list.
- **null** jest listą pustą.
- **add**(M, L) jest operacją dodania elementu M do listy L.
- Elementy listy możemy traktować jako połączone koniunkcją (tą z klasycznego rachunku zdań)

Argumenty obustronne i listy argumentów

$$\frac{\psi\,/\,\varphi : M \vdash \sigma : S}{\psi \mid \varphi : M \vdash \sigma : S}\ (\mathrm{L}|)\ \frac{\psi\,\backslash\,\varphi : M \vdash \sigma : S}{\psi \mid \varphi : M \vdash \sigma : S}$$

$$\frac{\begin{array}{c}(\psi\{|_1\varphi_1,|_2\varphi_2,\ldots,|_{i-1}\varphi_{i-1},|_{i+1}\varphi_{i+1},\ldots,|_n\varphi_n\})|_i\varphi_i\\ :\lambda x_i\lambda x_1\ldots x_{i-1}x_{i+1}\ldots x_n.Mx_1\ldots x_n\vdash\sigma:S\end{array}}{\psi\{|_1\ \varphi_1,|_2\ \varphi_2,\ldots,|_n\ \varphi_n\} : M \vdash \sigma : S}\ (\mathrm{L}\{\})$$

- Semantyką dla zbioru argumentów jest ciąg λ-abstrakcji.
- Ich kolejność jest taka jak kolejność argumentów w zapisie zbioru.

Wraz z parametrem

$$\frac{\tau : T \vdash \psi \,/\, \varphi : M \qquad \left(\&_{\alpha\in\{\gamma_1,\dots,\gamma_n\}} \psi\right) / \varphi : \langle M\rangle_{x\in\{P_1,\dots,P_n\}} \vdash \sigma : S}{\&_{\alpha\in\{\gamma_1,\dots,\gamma_n\}} \tau : \langle T\rangle_{x\in\{P_1,\dots,P_n\}} \vdash \sigma : S} \text{ (L\& /a}$$

$$\frac{\tau : T \vdash \psi \,/\, \varphi : M \qquad \psi[\alpha := \gamma_i] \,/\, \varphi[\alpha := \gamma_i] : T[x := P_i] \vdash \sigma : S}{\&_{\alpha\in\{\gamma_1,\dots,\gamma_n\}} \tau : \langle T\rangle_{x\in\{P_1,\dots,P_n\}} \vdash \sigma : S} \text{ (L\& /b)}$$

$$\frac{\tau[\alpha := \gamma_i] : T[x := P_i] \vdash \sigma : S}{\&_{\alpha\in\{\gamma_1,\dots,\gamma_n\}} \tau : \langle T\rangle_{x\in\{P_1,\dots,P_n\}} \vdash \sigma : S} \text{ (L\&)}$$

- Notacja $\langle T\rangle_{x\in\{P_1,\dots,P_n\}}$ oznacza, że
  - T jest termem mającym zmienną wolną x, na którą należy podstawić wartość ze zbioru $P_1, \dots, P_n$.
  - x jest termem przypisanym zmiennej typowej α.
  - $P_1, \dots, P_n$ są poprawnymi termami dla typów $\gamma_1, \dots, \gamma_n$.

Spis treści

Typ rzeczownika

$\mathrm{NP} \otimes \mathit{number} \otimes \mathit{case} \otimes \mathit{gender} \otimes \mathrm{ter}\{$

$\quad | \, (\mathrm{AdjP} \otimes \mathit{number} \otimes \mathit{case} \otimes \mathit{gender} \otimes \top)^{0-\infty},$ — przydawka przymiotna

$\quad / (\mathrm{NP} \otimes \top \otimes \mathrm{gen} \otimes \top \otimes \top)^{0-\infty},$ — przydawka dopełniaczowa

$\quad / (\mathrm{NP} \otimes \top \otimes \mathrm{nom} \otimes \top \otimes \top)^{0-1},$ — przydawka rzeczowna

$\quad / (\mathrm{NP} \otimes \top \otimes \mathit{case} \otimes \top \otimes \top)^{0-1},$ — apozycja

$\quad / (\mathrm{PP} \otimes \top)^{0-1},$ — przydawka przyimkowa

$\} :$

$$\lambda x_1 x_2 x_3 x_4 x_5 . \begin{bmatrix} \textsc{lexeme} & \mathit{lexeme} \\ \textsc{cat} & \mathrm{subst} \\ \textsc{args} & \begin{bmatrix} \mathrm{ADJP} & x_1 \\ \mathrm{NP\text{-}GEN} & x_2 \\ \mathrm{NP\text{-}NOM} & x_3 \\ \mathrm{NP\text{-}APOZ} & x_4 \\ \mathrm{PP} & x_5 \end{bmatrix} \end{bmatrix} \otimes \begin{bmatrix} \textsc{number} & \mathit{number} \end{bmatrix} \otimes$$

$$\otimes \begin{bmatrix} \textsc{case} & \mathit{case} \end{bmatrix} \otimes \begin{bmatrix} \textsc{gender} & \mathit{gender} \end{bmatrix} \otimes \begin{bmatrix} \textsc{person} & \mathrm{ter} \end{bmatrix}$$

Typ rzeczownika

- W sytuacji, w której kategorie fleksyjne nie są jednoznacznie wyznaczone korzystamy ze spójnika $\&$.
- Na przykład, dla rzeczownika „stół” otrzymamy typ:

$$\begin{array}{l}
\&_{\mathit{case}\in\{\mathrm{nom},\mathrm{acc}\}}\ \mathrm{NP}\otimes\mathrm{sg}\otimes \mathit{case}\otimes\mathrm{m}_3\otimes\mathrm{ter}\{\\
\qquad |\,(\mathrm{AdjP}\otimes\mathrm{sg}\otimes \mathit{case}\otimes\mathrm{m}_3\otimes\top)^{0-\infty},\\
\qquad /(\mathrm{NP}\otimes\top\otimes\mathrm{gen}\otimes\top\otimes\top)^{0-\infty},\\
\qquad /(\mathrm{NP}\otimes\top\otimes\mathrm{nom}\otimes\top\otimes\top)^{0-1},\\
\qquad /(\mathrm{NP}\otimes\top\otimes \mathit{case}\otimes\top\otimes\top)^{0-1},\\
\qquad /(\mathrm{PP}\otimes\top)^{0-1},\\
\}:
\end{array}$$

$$\left\langle \lambda x_1 x_2 x_3 x_4 x_5.\left[\begin{array}{ll}
\text{LEXEME} & \text{stół}\\
\text{CAT} & \mathrm{subst}\\
\text{ARGS} & \left[\begin{array}{ll}
\text{ADJP} & x_1\\
\text{NP-GEN} & x_2\\
\text{NP-NOM} & x_3\\
\text{NP-APOZ} & x_4\\
\text{PP} & x_5
\end{array}\right]
\end{array}\right]\otimes\left[\ \text{NUMBER}\quad \mathrm{sg}\ \right]\otimes\left[\ \text{CASE}\quad \mathit{case}\ \right]\otimes\right.$$

$$\left.\otimes\left[\ \text{GENDER}\quad \mathrm{m}_3\ \right]\otimes\left[\ \text{PERSON}\quad \mathrm{ter}\ \right]\right\rangle_{\mathit{case}\in\{\mathrm{nom},\mathrm{acc}\}}$$

Typ przymiotnika

$$\mathrm{AdjP} \otimes \mathit{number} \otimes \mathit{case} \otimes \mathit{gender} \otimes \mathit{grad}\{$$
$$\quad |\, \mathrm{AdvP}^{0-1},$$
$$\quad /(\mathrm{PP} \otimes \top)^{0-1},$$
$$\}:$$

$$\lambda x_1 x_2 . \begin{bmatrix} \text{LEXEME} & \mathit{lexeme} \\ \text{CAT} & \mathrm{adj} \\ \text{ARGS} & \begin{bmatrix} \text{ADVP} & x_1 \\ \text{PP} & x_2 \end{bmatrix} \end{bmatrix} \otimes \begin{bmatrix} \text{NUMBER} & \mathit{number} \end{bmatrix} \otimes$$
$$\otimes \begin{bmatrix} \text{CASE} & \mathit{case} \end{bmatrix} \otimes \begin{bmatrix} \text{GENDER} & \mathit{gender} \end{bmatrix} \otimes \begin{bmatrix} \text{GRAD} & \mathit{grad} \end{bmatrix}$$

Typ przymiotnika

- W sytuacji, w której kategorie fleksyjne nie są jednoznacznie wyznaczone korzystamy ze spójnika &
- Na przykład, dla przymiotnika 'duży' otrzymamy typ:

$$\begin{array}{l} \mathrm{AdjP} \otimes \mathrm{sg} \otimes ((\mathrm{nom} \,\&\, \mathrm{voc}) \otimes (\mathrm{m}_1 \,\&\, \mathrm{m}_2 \,\&\, \mathrm{m}_3) \otimes \mathrm{pos}) \,\&\, (\mathrm{acc} \otimes \mathrm{m}_3 \otimes \mathrm{pos}))\{ \\ \qquad |\, \mathrm{AdvP}^{0-1}, \\ \qquad /(\mathrm{PP} \otimes \top)^{0-1}, \\ \} : \end{array}$$

$$\lambda x_1 x_2 . \begin{bmatrix} \text{LEXEME} & \textit{lexeme} \\ \text{CAT} & \mathrm{adj} \\ \text{ARGS} & \begin{bmatrix} \text{ADVP} & x_1 \\ \text{PP} & x_2 \end{bmatrix} \end{bmatrix} \otimes \begin{bmatrix} \text{NUMBER} & \mathrm{sg} \end{bmatrix} \otimes$$

$$\otimes (((\begin{bmatrix} \text{CASE} & \mathrm{nom} \end{bmatrix} \,\&\, \begin{bmatrix} \text{CASE} & \mathrm{voc} \end{bmatrix}) \otimes$$

$$\otimes (\begin{bmatrix} \text{GENDER} & \mathrm{m}_1 \end{bmatrix} \,\&\, \begin{bmatrix} \text{GENDER} & \mathrm{m}_2 \end{bmatrix} \,\&\, \begin{bmatrix} \text{GENDER} & \mathrm{m}_3 \end{bmatrix}) \otimes \begin{bmatrix} \text{GRAD} & \mathrm{pos} \end{bmatrix}) \,\&$$

$$\& \, (\begin{bmatrix} \text{CASE} & \mathrm{acc} \end{bmatrix} \otimes \begin{bmatrix} \text{GENDER} & \mathrm{m}_3 \end{bmatrix} \otimes \begin{bmatrix} \text{GRAD} & \mathrm{pos} \end{bmatrix}))$$

Typ przyimka

$$\mathrm{PP} \otimes \mathit{case}_p \,/\, \mathrm{NP} \otimes \mathit{case}_n \otimes \top \otimes \top :$$

$$\lambda x. \begin{bmatrix} \text{LEXEME} & \mathit{lexeme} \\ \text{CAT} & \mathrm{prep} \\ \text{ARG} & x \end{bmatrix} \otimes \begin{bmatrix} \text{CASE} & \mathit{case}_p \end{bmatrix}$$

- case_n to przypadek frazy rzeczownikowej,
- case_p to przypadek analityczny utworzony przez połączenie leksemu przyimka i przypadku frazy rzeczownikowej.

Czasownik w formie osobowej

$$
\begin{array}{ll}
\mathrm{VP} \otimes \mathit{number}_v \otimes \mathit{gender}_v \otimes \mathit{person}_v \{ & \\
\quad \mid (\mathrm{NP} \otimes \mathit{number}_v \otimes \mathrm{nom} \otimes \mathit{gender}_v \otimes \mathit{person}_v)^{0-1} & \text{— wiązanie podmiotu} \\
\quad \mid (\mathrm{NP} \otimes \mathit{case}_o \otimes \top \otimes \top \otimes \top)^{0-1} & \text{— wiązanie dopełnienia bliższego} \\
\quad \mid (\mathrm{NP} \otimes \mathit{case}_{o_2} \otimes \top \otimes \top \otimes \top)^{0-1} & \text{— wiązanie dopełnienia dalszego rzecz} \\
\quad \mid (\mathrm{PP} \otimes \mathit{case}_{o_3})^{0-\infty} & \text{— wiązanie dopełnienia dalszego przyi} \\
\quad \mid \mathrm{AdvP}^{0-\infty} & \\
\quad \mid \text{się}^{0-1} & \\
\} : &
\end{array}
$$

$$
\lambda x_1 x_2 x_3 x_4 x_5 x_6 . \begin{bmatrix} \text{LEXEME} & \mathit{lexeme}_v \\ \text{CAT} & \mathrm{verb} \\ \text{ASPECT} & \mathit{aspect}_v \\ \text{ARGS} & \begin{bmatrix} \text{SUBJ} & x_1 \\ \text{DIROBJ} & x_2 \\ \text{OBJ-NP} & x_3 \\ \text{OBJ-PP} & x_4 \\ \text{ADVP} & x_5 \\ \text{SIE} & x_6 \end{bmatrix} \end{bmatrix} \otimes \begin{bmatrix} \text{NUMBER} & \mathit{number}_v \end{bmatrix} \otimes
$$

$$
\otimes \begin{bmatrix} \text{GENDER} & \mathit{gender}_v \end{bmatrix} \otimes \begin{bmatrix} \text{PERSON} & \mathit{person}_v \end{bmatrix}
$$

Spis treści

System dowodowy

Pełny system dowodowy

- określa jakie konstrukcje należą do języka
- niewielka ilość reguł dowodzenia
- własności zbadane teoretycznie
- duża wyrażalność i złożoność obliczeniowa

Efektywny system dowodowy

- upraszczające przybliżenie pełnego systemu dowodowego
- mała złożoność obliczeniowa
- każdy dowód zapisany za pomocą efektywnego systemu dowodowego da się zapisać za pomocą pełnego systemu dowodowego
- ale nie na odwrót

Wniosek filozoficzny: język jest się bytem idealnym, a konkretne jego realizacje (implementacja w mózgu, czy komputerze) są przybliżeniami.
Wniosek psychologiczny: ludzie mają różne kompetencje językowe - z różną dokładnością są w stanie przybliżać pełny system dowodowy.

Aplikacja

Efektywny system dowodowy opierać się będzie na regułach prawo i lewostronnej aplikacji:

$$\frac{\Gamma \vdash \tau : T \qquad \tau : T \vdash_1 \psi \,/\, \varphi : M \qquad \sigma : S \vdash_2 \varphi : N \qquad \Delta \vdash \sigma : S}{\Gamma, \Delta \vdash \psi : MN} \; (>)$$

$$\frac{\Delta \vdash \sigma : S \qquad \sigma : S \vdash_2 \varphi : N \qquad \tau : T \vdash_1 \psi \backslash \varphi : M \qquad \Gamma \vdash \tau : T}{\Delta, \Gamma \vdash \psi : MN} \; (<)$$

- Powyższe reguły odpowiadają aplikacji stosowanej w CCG, rozszerzonej o „unifikację” realizowaną przez sekwenty $\sigma \vdash_2 \varphi$ oraz $\tau \vdash_1 \psi \mid \varphi$.
- Sekwenty $\sigma \vdash_2 \varphi$ oraz $\tau \vdash_1 \psi \mid \varphi$ dowodzimy za pomocą uproszczonego systemu dowodowego logiki liniowej.
- Stosując powyższe reguły zakładamy, że Γ i Δ to ciągi tokenów.
- Kolejność wykonywania dowodu.

Dowody $\vdash_1$

Aby skonstruować wywód $\vdash_1$ korzystamy jedynie z

- aksjomatu postaci

$$\psi \,/\, \varphi : M \vdash \psi \,/\, \varphi : M \text{ lub } \psi \setminus \varphi : M \vdash \psi \setminus \varphi : M,$$

- reguł L&, $\mathrm{L}/^{0-1}$, $\mathrm{L}\setminus^{0-1}$, L|, L{}, $\mathrm{L}\&\,/$a, $\mathrm{L}\&\,/$b, $\mathrm{L}\&\setminus$a i $\mathrm{L}\&\setminus$b
- oraz reguł

$$\frac{(\psi \,/\, \varphi^{0-\infty}) \,/\, \varphi : \lambda x \lambda y.M\ \mathbf{add}(x, y) \vdash \sigma : S}{\psi \,/\, \varphi^{1-\infty} : M \vdash \sigma : S}\ (\mathrm{L}/^{0-\infty})$$

$$\frac{(\psi \setminus \varphi^{0-\infty}) \setminus \varphi : \lambda x \lambda y.M\ \mathbf{add}(x, y) \vdash \sigma : S}{\psi \setminus \varphi^{1-\infty} : M \vdash \sigma : S}\ (\mathrm{L}\setminus^{0-\infty})$$

Dowody $\vdash_2$

Aby udowodnić sekwent typu $\vdash_2$ stosujemy aksjomat oraz reguły L&, L$\&$, L0, R$\top$, R$\oplus$,L$|$,L$\{\}$,L$/^{0-1}$,L$\backslash^{0-1}$,L$|^{0-1}$, L$/^{1-\infty}$,L$\backslash^{1-\infty}$,L$|^{1-\infty}$,

$$\frac{\varphi : x \vdash \sigma : S \qquad \psi : y \vdash \tau : T}{\varphi \otimes \psi : M \vdash \sigma \otimes \tau : \textbf{let } x \otimes y = M \textbf{ in } S \otimes T} \text{(LR}\otimes\text{)}$$

$$\frac{\sigma : x \vdash \varphi : P \qquad \psi : MP \vdash \tau : N}{\psi \,/\, \varphi : M \vdash \tau \,/\, \sigma : \lambda x.N} \text{(LR/)}$$

$$\frac{\sigma : x \vdash \varphi : P \qquad \psi : MP \vdash \tau : N}{\psi \backslash \varphi : M \vdash \tau \backslash \sigma : \lambda x.N} \text{(LR}\backslash\text{)}$$

$$\frac{\sigma : x \vdash \varphi : P \qquad \psi : MP \vdash \tau : N}{\psi \mid \varphi : M \vdash \tau \mid \sigma : \lambda x.N} \text{(LR}|\text{)}$$

Ustalony kierunek: po lewej stronie znajduje się dopasowywane wyrażenie a po prawej wzorzec.

Kompozycja

Reguły aplikacji uzupełnimy o reguły kompozycji:

$$\frac{\Gamma \vdash \tau : T \qquad \tau : T \vdash_1 \psi / \varphi : M \qquad \rho : Nx \vdash_2 \varphi : P \qquad \omega : S \vdash_1 \rho / \sigma : N \qquad \Delta \vdash \omega : S}{\Gamma, \Delta \vdash \psi / \sigma : \lambda x.MP} \ (>\mathbf{B}$$

$$\frac{\Delta \vdash \omega : S \qquad \omega : S \vdash_1 \rho \backslash \sigma : N \qquad \rho : Nx \vdash_2 \varphi : P \qquad \tau : T \vdash_1 \psi \backslash \varphi : M \qquad \Gamma \vdash \tau : T}{\Delta, \Gamma \vdash \psi \backslash \sigma : \lambda x.MP} \ (<\mathbf{B}$$

gdzie Γ i Δ to ciągi tokenów.

Eliminacja jedynki

Oprócz tego potrzebować będziemy reguły eliminacji 1:

$$\frac{\Gamma \vdash \sigma : S \qquad \Delta \vdash 1 : N}{\Gamma, \Delta \vdash \sigma : \mathbf{let} \bullet = N \ \mathbf{in} \ S} \ (-1)$$

gdzie Γ i Δ to ciągi tokenów.
Reguła eliminacji 1 po lewej stronie termu nie jest konieczna, a wprowadzenie jej zwiększa złożoność obliczeniową systemu dowodowego.

Parser

- Algorytmem wnioskującym jest tu parser tablicowy oparty na algorytmie CYK.
- Korzystamy z tego ze reguły zawsze łączą dwa sąsiadujące ciągi tokenów w jeden.
- Algorytm wnioskuje wszystkie możliwe konkluzje wynikające z powyższego systemu dowodowego.
- Zaproponowany efektywny system dowodowy można uzupełniać o nowe reguły zwiększając jego siłę.

Niejednoznaczność

- Operator & reprezentuje niejednoznaczność składniową.
- Odpowiadający mu semantyczny operator pary $\langle\cdot,\cdot\rangle$ umożliwia wyrażenie niejednoznaczności semantycznej.
- Algorytm redukcji niejednoznaczności działa następująco:
  - Jeżeli podczas parsowania w jednym polu tablicy znajdą się dwa identyczne typy, są one uznawane za jeden, a ich semantyka jest łączona operatorem $\langle\cdot,\cdot\rangle$.
  - Identyczne poddrzewa są wyciągane przed operator $\langle\cdot,\cdot\rangle$, np:

$$\langle\varphi\otimes\sigma,\psi\otimes\sigma\rangle \;\longrightarrow\; \langle\varphi,\psi\rangle\otimes\sigma$$

Spis treści

Koordynacja

- Na poziomie typów koordynacja jest realizowana przez nadanie spójnikowi typu conj i dodaniu nowej reguły

$$\frac{\Gamma \vdash \varphi \qquad \Omega \vdash \mathrm{conj} \qquad \Delta \vdash \psi}{\Gamma, \Omega, \Delta \vdash \varphi \oplus \psi} \ (\mathrm{R}\oplus\mathrm{conj})$$

- Spójnik może łączyć dwie frazy o dowolnych typach, a fraza powstała w rezultacie jest sumą prostą tych typów.
- Jeżeli zarówno φ jak i ψ są funktorami mającymi ten sam argument, możemy wyciągnąć ten argument przed $\oplus$.
- Faktyczny typ frazy powstałej w wyniku koordynacji wyznaczany jest gdy staje się ona argumentem funktora.
- Dzięki temu nie trzeba wyszukiwać najmniej ogólnego typu, który jest bardziej ogólny od typów koordynowanych fraz.

Pat is wealthy and a Republican

$$\begin{aligned} \text{is} &\vdash (\mathrm{VP} \backslash \mathrm{NP}) / (\mathrm{AdjP} \oplus \mathrm{NP}) \\ \text{wealthy} &\vdash \mathrm{AdjP} \\ \text{and} &\vdash \mathrm{conj} \\ \text{a Republican} &\vdash \mathrm{NP} \end{aligned}$$

$$\dfrac{\text{is} \vdash (\mathrm{VP} \backslash \mathrm{NP}) / (\mathrm{AdjP} \oplus \mathrm{NP}) \qquad \dfrac{\text{wealthy} \vdash \mathrm{AdjP} \qquad \text{and} \vdash \mathrm{conj} \qquad \text{a Republican} \vdash \mathrm{NP}}{\text{wealthy, and, a Republican} \vdash \mathrm{AdjP} \oplus \mathrm{NP}}}{\text{is, wealthy, and, a Republican} \vdash (\mathrm{VP} \backslash \mathrm{NP})}$$

Er findet und hilft Frauen

$$\begin{array}{rcl} \text{findet} & \vdash & VP\,/(NP \otimes acc) \\ \text{und} & \vdash & conj \\ \text{hilft} & \vdash & VP\,/(NP \otimes dat) \\ \text{Frauen} & \vdash & NP \otimes (acc \,\&\, dat) \end{array}$$

$$\dfrac{\dfrac{\dfrac{\text{findet} \vdash VP\,/(NP \otimes acc) \qquad \text{und} \vdash conj \qquad \text{hilft} \vdash VP\,/(NP \otimes dat)}{\text{findet, und, hilft} \vdash (VP\,/(NP \otimes acc)) \oplus (VP\,/(NP \otimes dat))}}{\text{findet, und, hilft} \vdash VP\,/(NP \otimes (acc \,\&\, dat))}\,(\star) \qquad \text{Frauen} \vdash NP \otimes (acc \,\&\, dat)}{\text{findet, und, hilft, Frauen} \vdash VP}$$

Dowód $(\star)$, po wykonaniu cięcia:

$$\dfrac{\dfrac{\dfrac{VP \vdash VP \qquad \dfrac{NP \vdash NP \qquad \dfrac{acc \vdash acc}{acc \,\&\, dat \vdash acc}}{NP \otimes (acc \,\&\, dat) \vdash NP \otimes acc}}{VP\,/(NP \otimes acc), NP \otimes (acc \,\&\, dat) \vdash VP}}{(VP\,/(NP \otimes acc)) \vdash VP\,/(NP \otimes (acc \,\&\, dat))} \qquad \dfrac{\text{analogicznie jak po lewej}}{(VP\,/(NP \otimes dat)) \vdash VP\,/(NP \otimes (acc \,\&\, dat))}}{(VP\,/(NP \otimes acc)) \oplus (VP\,/(NP \otimes dat)) \vdash VP\,/(NP \otimes (acc \,\&\, dat))}$$

Kogo Janek lubi a Jerzy nienawidzi

$$\begin{array}{rcl} \text{Kogo} & \vdash & \mathrm{NP} \otimes (\mathrm{gen} \ \& \ \mathrm{acc}) \\ \text{Janek lubi} & \vdash & \mathrm{VP} \mid (\mathrm{NP} \otimes \mathrm{acc}) \\ \text{a} & \vdash & \mathrm{conj} \\ \text{Jerzy nienawidzi} & \vdash & \mathrm{VP} \mid (\mathrm{NP} \otimes \mathrm{gen}) \end{array}$$

Przypadek analogiczny do poprzedniego.
Podobnie: Dajcie wina i całą świnię

Zjadł ogórek i panaceum

$$\begin{array}{rcl}
\text{Zjadł} & \vdash & \mathrm{VP} / (\mathrm{NP} \otimes \top \otimes \mathrm{acc} \otimes \top) \\
\text{ogórek} & \vdash & \mathrm{NP} \otimes \mathrm{sg} \otimes (\mathrm{nom} \mathbin{\&} \mathrm{acc}) \otimes \mathrm{m}_3 \\
\text{i} & \vdash & \mathrm{conj} \\
\text{panaceum} & \vdash & \mathrm{NP} \otimes \mathrm{sg} \otimes 0 \otimes \mathrm{n}_2
\end{array}$$

$$\dfrac{\text{ogórek} \vdash \mathrm{NP} \otimes \mathrm{sg} \otimes (\mathrm{nom} \mathbin{\&} \mathrm{acc}) \otimes \mathrm{m}_3 \qquad \text{i} \vdash ((\alpha \oplus \beta) \backslash \alpha) / \beta \qquad \text{panaceum} \vdash \mathrm{NP} \otimes \mathrm{sg} \otimes 0 \otimes \mathrm{n}_2}{\text{ogórek, i, panaceum} \vdash (\mathrm{NP} \otimes \mathrm{sg} \otimes (\mathrm{nom} \mathbin{\&} \mathrm{acc}) \otimes \mathrm{m}_3) \oplus (\mathrm{NP} \otimes \mathrm{sg} \otimes 0 \otimes \mathrm{n}_2)}$$

$$\dfrac{\dfrac{\mathrm{NP} \vdash \mathrm{NP} \qquad \mathrm{sg} \vdash \top \qquad \dfrac{\mathrm{acc} \vdash \mathrm{acc}}{\mathrm{nom} \mathbin{\&} \mathrm{acc} \vdash \mathrm{acc}} \qquad \mathrm{m}_3 \vdash \top}{\mathrm{NP} \otimes \mathrm{sg} \otimes (\mathrm{nom} \mathbin{\&} \mathrm{acc}) \otimes \mathrm{m}_3 \vdash \mathrm{NP} \otimes \top \otimes \mathrm{acc} \otimes \top} \qquad \dfrac{\mathrm{NP} \vdash \mathrm{NP} \qquad \mathrm{sg} \vdash \top \qquad 0 \vdash \mathrm{acc} \qquad \mathrm{n}_2 \vdash \top}{\mathrm{NP} \otimes \mathrm{sg} \otimes 0 \otimes \mathrm{n}_2 \vdash \mathrm{NP} \otimes \top \otimes \mathrm{acc} \otimes \top}}{(\mathrm{NP} \otimes \mathrm{sg} \otimes (\mathrm{nom} \mathbin{\&} \mathrm{acc}) \otimes \mathrm{m}_3) \oplus (\mathrm{NP} \otimes \mathrm{sg} \otimes 0 \otimes \mathrm{n}_2) \vdash \mathrm{NP} \otimes \top \otimes \mathrm{acc} \otimes \top}$$

Semantyka koordynacji

$$\frac{\Gamma \vdash \varphi : M \qquad \Omega \vdash \text{conj} : C \qquad \Delta \vdash \psi : N}{\Gamma, \Omega, \Delta \vdash \varphi \oplus \psi : \mathbf{inb}(M, C, N)} \ (\mathrm{R}\oplus\text{conj})$$

- $\mathbf{inb}(M, C, N)$ oznacza, że mamy dwa obiekty połączone spójnikiem C.
- Semantyka spójnika „i”: $\lambda xy.x \wedge y$
- Semantyka spójnika „lub”: $\lambda xy.x \vee y$
- Semantyka dla reguły L$\oplus$:

$$\frac{\Gamma, \varphi : x, \Delta \vdash \sigma : S \qquad \Gamma, \psi : y, \Delta \vdash \sigma : T}{\Gamma, \varphi \oplus \psi : M, \Delta \vdash \sigma : \mathbf{case}\ M\ \mathbf{of}\ \mathbf{inl}(x) \Rightarrow S \mid \mathbf{inr}(y) \Rightarrow T \mid \mathbf{inb}(x, c, y) \Rightarrow cST} \ (\mathrm{L}\oplus)$$

Semantyka dla reguły LR$\oplus$:

Konkluzje

- Zaproponowany formalizm jest logiką i posiada system dowodowy, co pozwala elegancko reprezentować w nim składnię i semantykę języka.
- Pozwala też oddzielić zagadnienia formalnej reprezentacji zjawisk lingwistycznych od problemu efektywnego przetwarzania.
- Możliwe kierunki rozwoju:
  - dodanie kontekstu,
  - implementacja zjawiska elipsy,
  - gramatyka semantyczna.

Dziękuję za uwagę!