Gramatyki wykorzystywane w analizie języka naturalnego

PCFG=Probabilistic Context-Free Grammars

HLPCFG=Head-Lexicalised PCFG

HG=Head Grammar

HPSG=Head Grammar

IG=Indexed Grammar

LIG=Linear Indexed Grammar

PCFG składa się z:

- Zbioru terminali $\{w^k\}$, gdzie k=1…V
- Zbioru nieterminali $\{N^i\}$, gdzie i=1..n
- Wyróżnionego symbolu startowego
- Zbioru produkcji $N^i \rightarrow A^j$, gdzie A^j to sekwencja terminali i nieterminali
- Zbiorów prawdopodobieństw produkcji takich, że $\forall i \sum_j P(N^i \rightarrow A^j) = 1$

Notacja

- Zdanie: sekwencja słów $w_1 \ldots w_m$
- w_{ab}:podłańcuch $w_a \ldots w_b$
- Prawdopodobieństwo łańcucha

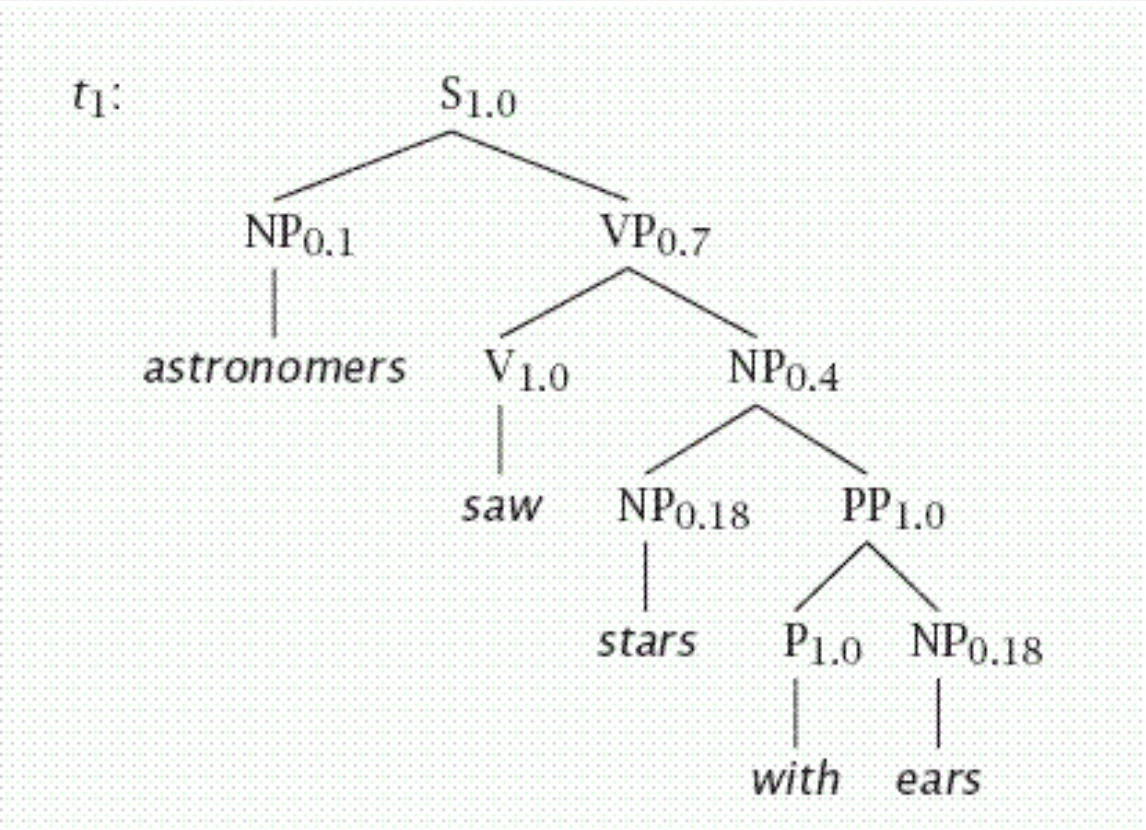
$$P(w_{1n}) = \sum_t P(w_{1n}, t) = \sum_t P(t)$$

gdzie t to wyprowadzenie w_{1n}

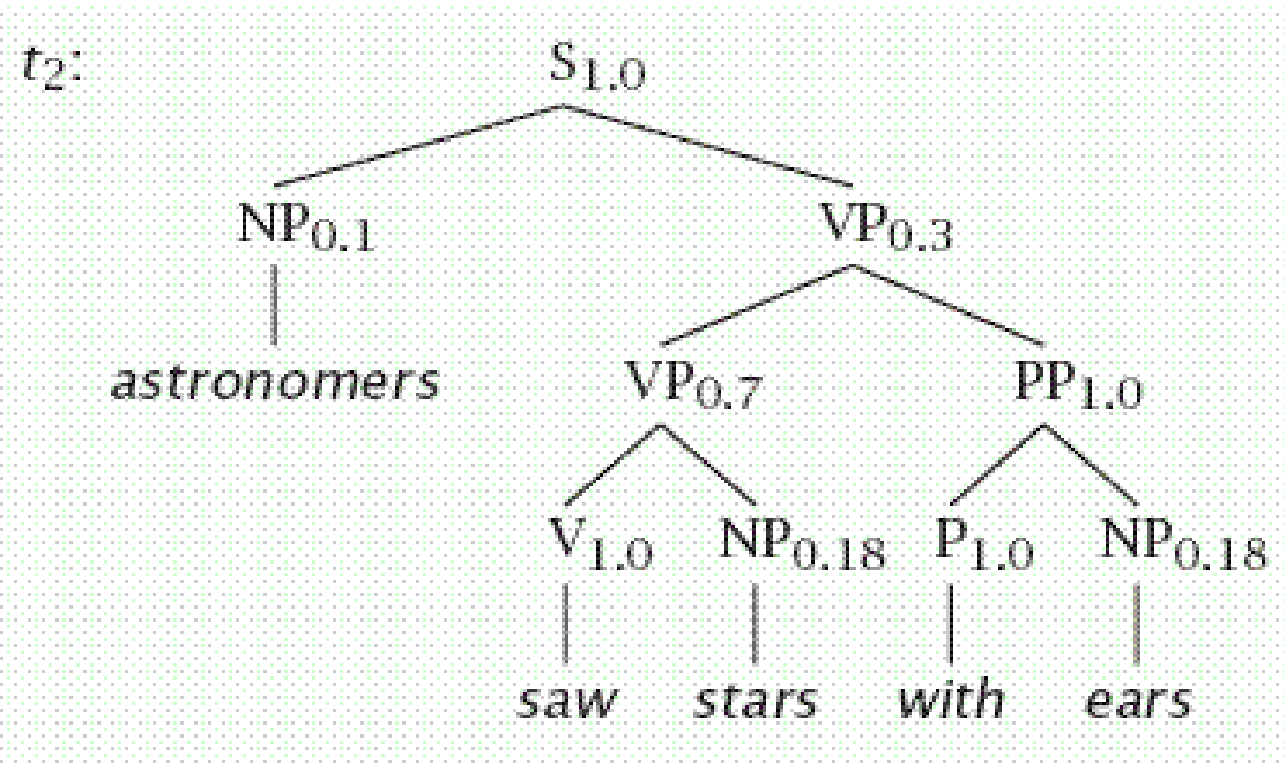
PCFG (prosty przykład)

- S →NP VP (1.0)
- PP →P NP (1.0)
- VP →V NP (0.7)
- VP →VP PP (0.3)
- P →with (1.0)
- V →saw (1.0)
- NP →NP PP (0.4)
- NP →astronomers (0.1)
- NP →ears (0.18)
- NP →saw (0.04)
- NP →stars (0.18)
- NP →telescopes (0.1)

Drzewa rozbioru:



Drzewa rozbioru cd.



Prawdopodobieństwa

- Drzew rozbioru:

  $P(t1)=1.0\times0.1\times0.7\times1.0\times0.4\times0.18\times1.0\times1.0\times0.18$

  $=0.0009072$

  $P(t2)=1.0\times0.1\times0.3\times0.7\times1.0\times0.18\times1.0\times1.0\times0.18$

  $=0.0006804$

- Zdania:

  $P(w15)=P(t1)+P(t2)=0.0015876$

Niektóre cechy

- Stanowią częściowe rozwiązanie dla gramatyk niejednoznacznych:dają prawdopodobieństwo danego zdania
- Elastyczność (przyjąć wszystko z małym prawdopodobieństwem)
- Dają probabilistyczny model językowy języka angielskiego
- W praktyce gorsze od modelu trigram
- Dopuszczają np. że małe drzewa rozbioru są bardziej prawdopodobne

Sprzeczne rozkłady

- S→rhubarb (1/3)

  S→S S (2/3)
- rhubarb (1/3)
- rhubarb rhubarb 2/3×1/3 ×1/3=2/27
- rhubarb rhubarb rhubarb (2/3)^2 ×(1/3)^3 ×2=8/243
- P(L)=1/3+2/27+8/243+…=1/2
- nie jest to problem jeśli liczymy z parsed treebank

Zadania PCFG

- Modelowanie języka:

  nadanie prawdopodobieństwa każdemu łańcuchowi generowanemu przez gramatykę

  $$P(w_1 \ldots w_m \mid G)$$

- Najlepsze drzewo rozbioru:

  wybranie najbardziej prawdopodobnego drzewa rozbioru dla danego łańcucha

$$\arg_{tree} \max P(tree \mid w_1 \ldots w_m, G)$$

Zadania PCFG cd.

- Zoptymalizowanie prawdopodobieństw reguł danej gramatyki dla niektórych zdań

$$\arg_G \max P(w_1 \ldots w_m \mid G)$$

Najbardziej prawdopodobne drzewo rozbioru

- Najprostsze rozwiązanie: znalezienie wszystkich możliwych i wybranie maksimum
- Jest to rozwiązanie mało wydajne dla dłuższych zdań w gramatykach niejednoznacznych(złożoność wykładnicza)
- Użycie algorytmu Viterbi dla PCFG lub inside algorithm

CNF PCFG

- Produkcje są postaci

$$N^i \rightarrow N^j N^k$$

$$N^i \rightarrow w^j$$

- Parametry:

$$P(N^j \rightarrow N^r N^s \mid G)$$ (n^3 macierz param)

$$P(N^j \rightarrow w^k \mid G)$$ (nt macierz param)

Dla j=1..n

$$\sum_{r,s} P(N^j \rightarrow N^r N^s) + \sum_k P(N^j \rightarrow w^k) = 1$$

Założenia PCFG

- Niezmienniczość miejsca:

  Identyczne poddrzewa mają takie same prawdopodobieństwa niezależnie od miejsca występowania w drzewie syntaktycznym
- Bezkontekstowość

  Prawdopodobieństwo poddrzewa nie bierze pod uwagę słów występujących ani przed ani za

Założenia PCFG cd.

- Ancestor-free

  Dominujące węzły poddrzewa nie mają wpływu na jego prawdopodobieństwo

Użyteczność PCFG

- W modelowaniu języka

  Modele probabilistyczne oparte na samym PCFG są zbyt proste:

  - założenia niezależności są zbyt silne
  - potrzeba leksykalizacji i kontekstualizacji
  - istnieją różne metody na rozszerzenie PCFG

Użyteczność w rozbiorze

- Korzystne jeśli nadamy niskie prawdopodobieństwa
- W niektórych przypadkach może pomóc w wyeliminowaniu niejednoznaczności
- Jednak występują typowe ograniczenia np.uprzywilejowanie mniejszych drzew

Head-Lexicalised Probabilistic Context-Free Grammar

- Każda reguła PCFG jest uzupełniana aby identyfikować jeden ze składników prawej strony produkcji jako jej head
- Następnie headword węzła przechodzi na jego syna
- Teraz wszystkie nieterminale są postaci X(x)

Przykład

- Produkcja gramatyki bezkontekstowej

  $VP \rightarrow V\ NP\ PP$

- Produkcje gramatyki zleksykalizowanej

  $VP(throw) \rightarrow V(throw)\ NP(ball)\ PP(into)$

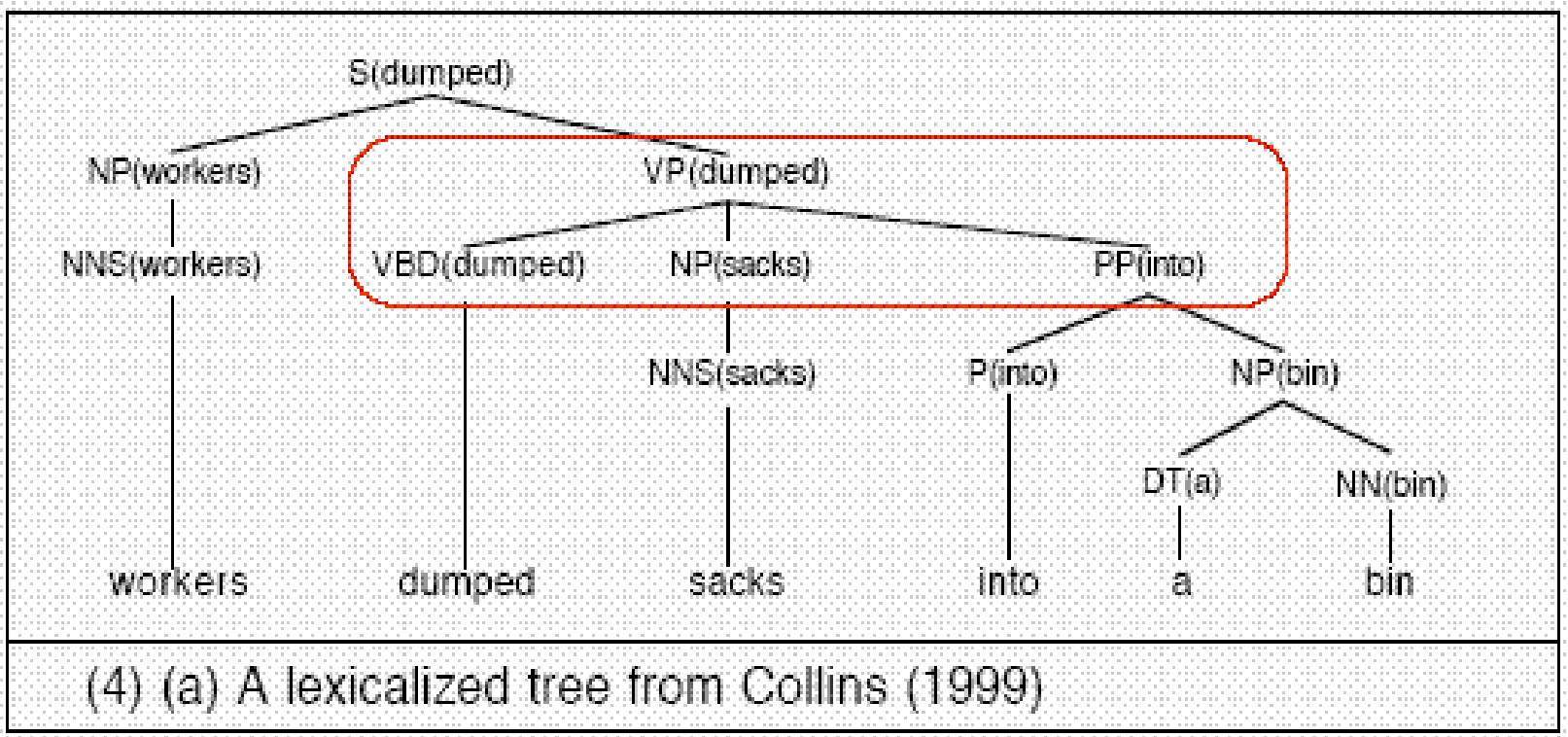
  $VP(send) \rightarrow V(send)\ NP(soldiers)\ PP(into)$

  $VP(send) \rightarrow V(send)\ NP(gift)\ PP(to)$

  $VP(put) \rightarrow V(put)\ NP(ball)\ PP(below)$

  itd.

Przykład



(4) (a) A lexicalized tree from Collins (1999)

Przypisanie prawdopodobieństw

- Upraszczamy założenia niezależności
- W standardowej PCFG prawd.,że $X \rightarrow \beta$ uwarunkowane jedynie syntaktyczną kategorią X: $P(X \rightarrow \beta | X)$
- Wprowadzamy współczynnik uwarunkowania headword węzła X (head(X))
- Dla reguły VP →VBD NP PP
  P(VP →VBD NP PP |VP, dumped)

Head Grammar

- Pollard, 1984
- Każdy łańcuch zawiera wyróżniony symbol head
- Posiada operatory konkatenacji i opakowania (wrapping) do tworzenia nowych stringów
- Gdy dwa łańcuchy zostają skonkatenowane to lewa lub prawa głowa zostaje głową łańcucha wynikowego

HG-operacja opakowania

- Operacja opakowania oddziela łańcuch od głowy, a następnie umieszcza inny łańcuch pomiędzy
- 4 rodzaje:drugi łańcuch jest umieszczany z prawej lub lewej strony głowy pierwszego łańcucha, głowa pierwszego lub drugiego łańcucha zostaje głową łańcucha wynikowego

HG-produkcje

- Produkcje są postaci $A \rightarrow$ α1 lub

  $A \rightarrow$ f(α1… αn) gdzie f to konkatenacja lub operator opakowania
- np. $C_1(w_1 \uparrow w_2, u_1 \uparrow u_2) \rightarrow w_1 \uparrow w_2 u_1 u_2$

  $C_1(w_1 \uparrow w_2, u_1 \uparrow u_2) \rightarrow w_1 w_2 u_1 \uparrow u_2$

  $W(w_1 \uparrow w_2, u_1 \uparrow u_2) \rightarrow w_1 w_2 \uparrow u_1 u_2$

HPSG=Head-Driven Phrase Structure Grammar

Z punktu widzenia lingwistycznego może obejmować komponenty:

- Słownik dostarczający podstawowych form wyrazowych
- Reguły leksykalne
- immediate dominance (ID) schemata, struktura składniowa zdań
- Linear precedence (LP) statements, określające szyk wyrazów w zdaniu
- Zbiór ograniczeń wyrażających generalizaje na temat cech gramatycznych obiektów językowych

HPSG
w sensie formalnym składa się z:

1. Sygnatury
2. Teorii

Sygnatura określa domenę obejmującą:

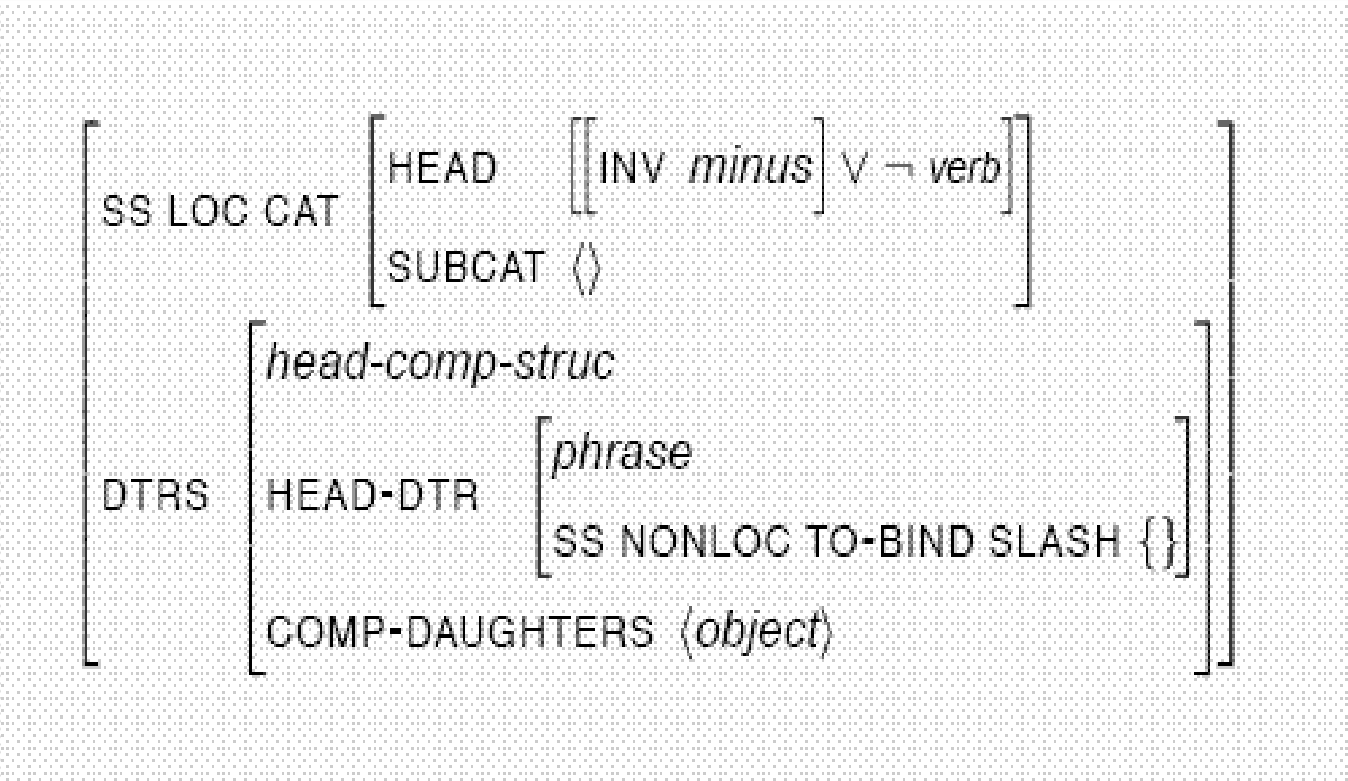
- Zbiór symboli typów
- Zbiór symboli atrybutów
- Zbiór symboli relacyjnych
- Funkcję określającą które atrybuty mogą występować z określonymi typami
- Hierarchię typów

Teoria ograniczająca tę domenę

- Jest zbiorem ograniczeń (zasad), będących formułami języka służącego do zapisu teorii HPSG
- Opisuje każdy obiekt w modelu
- Ograniczenia mogą mieć formę:prostej specyfikacji typu, złożonej deskrypcji z przydzieleniem typu i równością ciągów atrybutów, koniunkcji, negacji, implikacji.

$$word \rightarrow (LE_1 \vee \ldots \vee LE_n)$$

Przykład schematu



Indexed Grammars

- Uogólnienie gramatyk bezkontekstowych
- Ustalona liczba symboli może być zdjęta ze stosu lewych stron produkcji
- Stosy nieterminali po prawej stronie produkcji:

  *mają ustalony rozmiar albo

  *nieograniczony stos z lewej strony produkcji z ustaloną liczbą odłożonych symboli

IG-definicja

- $G=(V_n,V_t,V_s,S,P)$
- V_n-niepusty skończony zbiór nieterminali
- V_t -skończony zbiór terminali
- P-skonczony zbiór produkcji

$$A[..x]\rightarrow\alpha_1..\alpha_n$$

gdzie $x\in V_s^*$ i $\forall 1\le i\le n,\alpha_i=A[..y],\alpha_i=A[z]$, lub $\alpha_i=\omega$

gdzie $A\in V_n,\omega\in V_t^*;y,z\in V_s^*$

Umowa: [..l] -dowolny stos z l na wierzchołku

- V_s zbiór symboli stosowych

IG-przykład języka $a^n b^n c^n d^n e^n$

- $V_n = \{S, A, B, C, D, E\}$
- $V_t = \{a, b, c, d, e\}$
- $V_s = \{i\}$

Gramatyka

- S[..]→S[..i]
- S[…] →A[…]B[…]C[…]D[…]E[…]
- A[..ii] →aA[..i]
- A[i] →a
- B[..ii] →bB[..i]
- B[i] →b
- C[..ii] →cC[..i]
- C[i] →c
- D[..ii] →dD[..i]
- D[i] →d
- E[..ii] →eE[..i]
- E[i] →e

LIG-Linear Indexed Grammar

- Szczególne ograniczenie IG
- Kluczowa zmiana: tylko jeden nieterminal z prawej strony produkcji dziedziczy stos
- Zależności pomiędzy niezwiązanymi gałęziami drzewa nie są możliwe

LIG-przykład języka $a^n b^n c^n d^n$

- $V_n = \{S, T\}$
- $V_t = \{a, b, c, d\}$
- $V_s = \{i\}$

- S[..]→aS[..i]d
- S[..] →T[..]
- T[..i] →bT[..]c
- T[] →ε

LoPar-a left corner parser for head-lexicalised probabilistic context-free grammars

- Przykład gramatyki języka angielskiego wykorzystywany przez ten parser

http://www.ims.uni-stuttgart.de/projekte/gramotron/SOFTWARE/English-HLPCFG-en.html

Selective Magic HPSG Parsing

- Wykorzystuje zarówno parsing bottom-up jak i top-down
- http://arxiv.org/PS_cache/cs/pdf/9907/9907012.pdf

ASSERT (Automatic Statistical SEmantic Role Tagger)

- http://oak.colorado.edu/assert/

References

- www.icsi.berkeley.edu/śtolcke/papers/c
- www.cog.brown.edu/~mj/Publications.htm
- http://www-nlp.stanford.edu/fsnlp/pcfg
- http://arxiv.org/PS_cache/cs/pdf/9907/9907012.pdf
- http://portal.acm.org