Zpracování přirozeného jazyka

Aleš Horák

E-mail: hales@fi.muni.cz
http://nlp.fi.muni.cz/uui/

Obsah:

- Komunikace
- Gramatiky
- Analýza přirozeného jazyka
- PA026 – Projekt z umělé inteligence

Přirozený jazyk – prostředek komunikace

komunikace = cílená výměna informace pomocí produkce a vnímání (sdílených) pokynů

– zvířata – až stovky pokynů (šimpanz, delfín, . . .)

– člověk – potenciálně neomezené množství, díky přirozenému jazyku

2 náhledy na přirozený jazyk:

- klasický (před 1953) – jazyk se skládá z vět, které jsou buď pravdivé nebo nepravdivé (srovnej s logikou)
- moderní (po 1953) – užití jazyka je jedna z možných akcí
  Wittgenstein (1953) Philosophical Investigations
  Searle (1969) Speech Acts

Turingův test založen na jazyku ⇐ jazyk je pevně spojen s myšlením

komunikace se tvoří pomocí řečových aktů (*speech acts*) jako jeden z typů agentových akcí
cíl komunikace – změnit akce ostatních agentů

Řečové akty

Komunikační situace
Mluvčí (*speaker*) → Promluva (*utterance*) → Posluchač (*hearer*)

řečové akty směřují k naplnění cílů mluvčího:

- informovat (inform) "Před tebou je jáma."
- ptát se (query) "Vidíš zlato?"
- přikázat/žádat (command/request) "Zvedni to."
- slíbit/svěřit se s plánem (promise, commit to plan) "Rozdělím se s tebou o zlato."
- potvrdit (acknowledge) "OK"

plánování řečových aktů vyžaduje znalosti:

- komunikační situace
- sémantiky a syntaxe (sdílených konvencí)
- informace o Posluchači – cíle, znalosti, rozumnost

Komunikační fáze (při informování)

průběh promluvy je možné rozložit na fáze:

– záměr (intention)	M chce informovat Po, že Pr
– generování (generation)	M vybírá slova W pro vyjádření Pr
– syntéza (synthesis)	M říká slova W
– vnímání (perception)	Po vnímá W'
– analýza (analysis)	Po odvozuje možné významy $Pr_1, \ldots, Pr_n$
– zjednoznačnění (disambiguation)	Po vybírá zamýšlený význam Pr_i
– zahrnutí (incorporation)	Po zahrne Pr_i do své báze znalostí

Může přitom vzniknout chyba?

- neupřímnost (Po nevěří Pr)
- víceznačnost promluvy (Po zvolí špatné Pr_i)
- různé pochopení aktuální situace (zamýšlený význam mezi Pr_i není)

Komunikační fáze – příklad

záměr
$Vědět(Po, \neg Na_živu(Wumpus_1, S_3))$

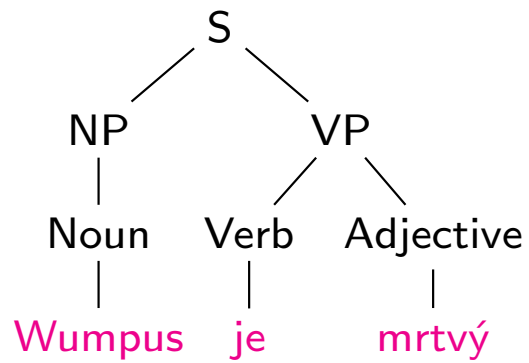
generování
"Wumpus je mrtvý."

syntéza MLUVČÍ
[v u m p u s j e m r t v i:]

vnímání
"Wumpus je mrtvý."

analýza

syntaktická analýza:



sémantická interpretace:
$\neg Na_živu(Wumpus, Teď)$
$Unavený(Wumpus, Teď)$

pragmatická interpretace:
$\neg Na_živu(Wumpus_1, S_3)$
$Unavený(Wumpus_1, S_3)$

zjednoznačnění POSLUCHAČ
$\neg Na_živu(Wumpus_1, S_3))$

zahrnutí
$Tell(KB, \neg Na_živu(Wumpus_1, S_3))$

Gramatiky

zvířata používají místo vět izolované symboly ⇒ omezená sada komunikovatelných situací → žádná generativní kapacita

gramatika specifikuje skladební strukturu složených pokynů – definuje formální jazyk pokynů

formální jazyk = množina řetězců (vět) teminálních symbolů (slov)

2 náhledy na vztah věty a gramatiky:

- S je správný řetězec/věta z jazyka ⇔ S je analyzovatelný danou gramatikou
- příslušná gramatika generuje S ⇔ S je správný řetězec/věta z jazyka

gramatika je zadána jako množina přepisovacích pravidel

$$S \rightarrow NP \; VP$$
$$Pronoun \rightarrow já \mid ty \mid on \mid \ldots$$

v tomto příkladu:

S	větný symbol – kořenový symbol gramatiky
NP, VP	neterminály
já, ty, . . .	terminály

Typy gramatik

- regulární (regular) neterminál → **terminál**[neterminál]

  $S \rightarrow \mathbf{a}S$
  $S \rightarrow \mathbf{b}$

  ekvivalentní síle konečných automatů, neumí $a^n b^n$
- bezkontextové (context-free) neterminál → cokoliv

  $S \rightarrow \mathbf{a}S\mathbf{b}$

  ekvivalentní síle zásobníkových automatů, umí $a^n b^n$, neumí $a^n b^n c^n$
- kontextové (context-sensitive) – víc termů na levé straně (*kontext* neterminálu)

  $\underline{A}S\underline{B} \rightarrow \underline{A}A\mathbf{a}B\underline{B}$

  umí $a^n b^n c^n$
- rekurzivně vyčíslitelné (recursively enumerable) – bez omezení

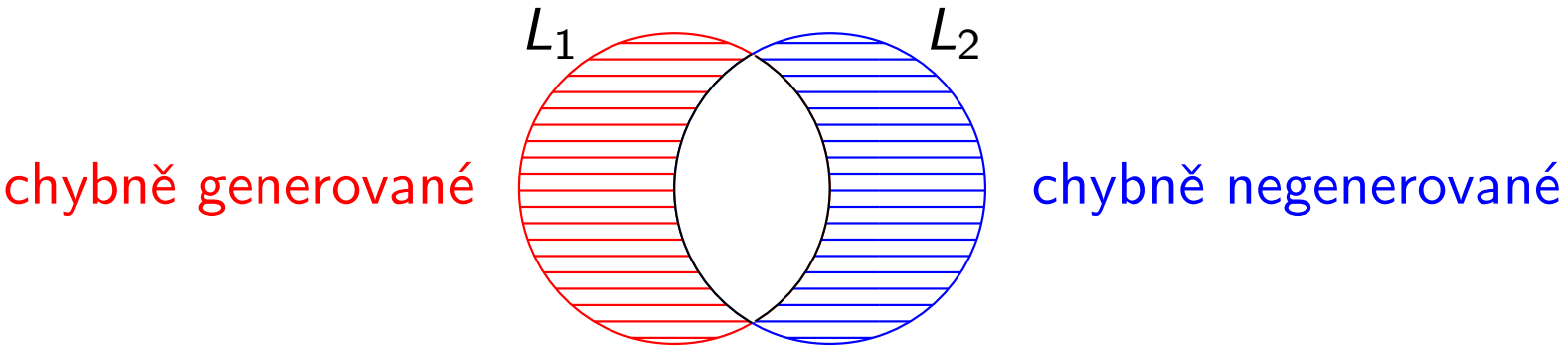
  ekvivalentní síle Turingova stroje

přirozený jazyk byl dlouho pokládán za bezkontextový → nyní prokázáno, že obsahuje kontextové prvky

Přesnost a pokrytí gramatiky

u složitějších jazyků (např. přirozených)

→ jazyk L_1 (generovaný gramatikou) se liší od zamýšleného jazyka L_2



kvalita gramatiky:

- pokrytí – procento vět jazyka L_2 generovatelných gramatikou ($|L_1 \cap L_2|/|L_2|$)
- přesnost – procento generovaných vět, které jsou správné věty jazyka L_2 ($|L_1 \cap L_2|/|L_1|$)

tvorba gramatiky . . . postupný proces zvyšování pokrytí a přesnosti gramatiky přirozených jazyků – velmi rozsáhlé a přesto většinou nepopisují plně ani angličtinu ☹

DC gramatiky – gramatiky uspořádaných klauzulí

Gramatiky uspořádaných klauzulí:

- *Definite-Clause Grammars*, DCG
- významná aplikace Prologu – *syntaktická analýza*
- DCG jsou rozšířením bezkontextových gramatik (CFG)
- jejich implementace využívá *rozdílových seznamů*

Formální podobnosti mezi DCG a CFG:

- CFG: pravidla tvaru $x \to y$, kde $x \in N$ je neterminál a $y \in (N \cup T)^*$ je konečná posloupnost terminálů a neterminálů
- DCG: pravidla tvaru ⟨**hlava**⟩ −−> ⟨**tělo**⟩, kde ⟨**hlava**⟩ je opět neterminál a ⟨**tělo**⟩ je opět konečná posloupnost terminálů a neterminálů
- pravidlo ⟨**hlava**⟩ −−> ⟨**tělo**⟩ znamená, že jedním z možných tvarů ⟨**hlavy**⟩ je **tělo**, neboli: ⟨**hlavu**⟩ je možno přepsat na ⟨**tělo**⟩

Rozdíly a rozšíření DCG oproti CFG

DCG:

1. Neterminál může být téměř libovolný term, kromě *seznamu*, *proměnné* a *čísla*.
2. Terminál může být libovolný term, s tím, že terminály a posloupnosti terminálů uzavíráme do hranatých závorek – jako seznamy.
3. Pravá strana pravidla může obsahovat dodatečné podmínky v podobě prologovských podcílů. Tyto podmínky uzavíráme do složených závorek.
4. Levá strana pravidla může dokonce vypadat i tak, že neterminál je následován posloupností terminálů.
5. Tělo pravidla smí obsahovat řez.

DC gramatika – příklad 1

gramatika vět typu "The young boy sings a song."

```
% 1. část -- pravidla
sentence --> noun_phrase, verb_phrase.

noun_phrase --> determiner, noun_phrase2.
noun_phrase --> noun_phrase2.

noun_phrase2 --> adjective, noun_phrase2.
noun_phrase2 --> noun.

verb_phrase --> verb.
verb_phrase --> verb, noun_phrase.

% 2. část -- lexikon
determiner --> [the].        noun --> [boy].
determiner --> [a].          noun --> [song].

verb --> [sings].            adjective --> [young].
```

Analýza v Prologu pomocí append

- věta = seznam slov **[the,young,boy,sings,a,song]**
- pravidlová část – neterminál chápeme jako unární predikát, jehož argumentem je ta větná složka, kterou daný neterminál popisuje

```
sentence(S) :- append(NP,VP,S),
               noun_phrase(NP), verb_phrase(VP).
...
```

- slovníková část, lexikon – reprezentujeme pomocí faktů:

```
determiner([the]).        noun([boy]).
determiner([a]).          ...
```

Efektivněji – rozdílové seznamy

přepis gramatiky do Prologu pomocí rozdílových seznamů:

```
sentence(S,S0) :- noun_phrase(S,S1), verb_phrase(S1,S0).

noun_phrase(S,S0) :- determiner(S,S1), noun_phrase2(S1,S0).
noun_phrase(S,S0) :- noun_phrase2(S,S0).
noun_phrase2(S,S0) :- adjective (S,S1), noun_phrase2(S1,S0).
noun_phrase2(S,S0) :- noun(S,S0).
verb_phrase(S,S0) :- verb(S,S0).
verb_phrase(S,S0) :- verb(S,S1), noun_phrase(S1,S0).

determiner([the|S],S).          noun([boy|S],S).
determiner([a|S],S).            noun([song|S],S).
verb([sings|S],S).              adjective ([young|S],S).

?- sentence([the, young,boy,sings , a, song], [] ).
    Yes
```

Lexikon pro agenta ve Wumpusově jeskyni

Gramatika přímo na slovech je příliš rozsáhlá. Řešením je rozdělení slov do kategorií:

podst. jméno:	*Noun*	→	zápach | vánek | třpyt | nic | wumpuse | jáma | zlato | ...
sloveso:	*Verb*	→	jsem | je | vidím | cítím | působí | zapáchá | jdu | ...
příd. jméno:	*Adjective*	→	levý | pravý | východní | jižní | ...
příslovce:	*Adverb*	→	tady | tam | blízko | vpředu | vpravo | vlevo | východně | jižně | vzadu | ...
vl. jméno:	*Name*	→	Petr | Honza | Brno | FI MU | ...
zájmeno:	*Pronoun*	→	já | ty | mě | toho | ten | ta ...
předložka:	*Preposition*	→	do | v | na | u | ...
spojka:	*Conjunction*	→	a | nebo | ale | ...
číslice:	*Digit*	→	0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

kategorie můžeme dělit na otevřené (vyvíjející se) a uzavřené (stálé)

Morfologická analýza

- v češtině u lexikonu nestačí prostý výčet tvarů – je nutná morfologická analýza (morfologie=tvarosloví)
- skloňovaná a časovaná slova se rozkládají na segmenty

  pří-lež-it-ost-n-ými:
  pří – prefix; *lež* – kořen; *it*, *ost*, *n* – suffixy; *ými* – koncovka

- základní tvar slova (*lemma*), podle koncovky se určují gramatické kategorie

```
% slovník základních gramatických kategorií -- pád, číslo, rod
% adj(+Slovo, +Lemma, +Pad, +Cislo, +Rod)
adj(chytrý, chytrý, 1, sg, mz).    adj(chytrého, chytrý, 2, sg, mz).
adj(chytří, chytrý, 1, pl, mz).
```

- reálná morfologická analýza ČJ – program MAJKA na FI MU
  `http://nlp.fi.muni.cz/projekty/wwwajka/`

```
ajka>nejneuvěřitelněji
<s> nej-ne=uvěřiteln==ěji= (1022)
    <l>uvěřitelně
    <c>k6xMeNd3
```

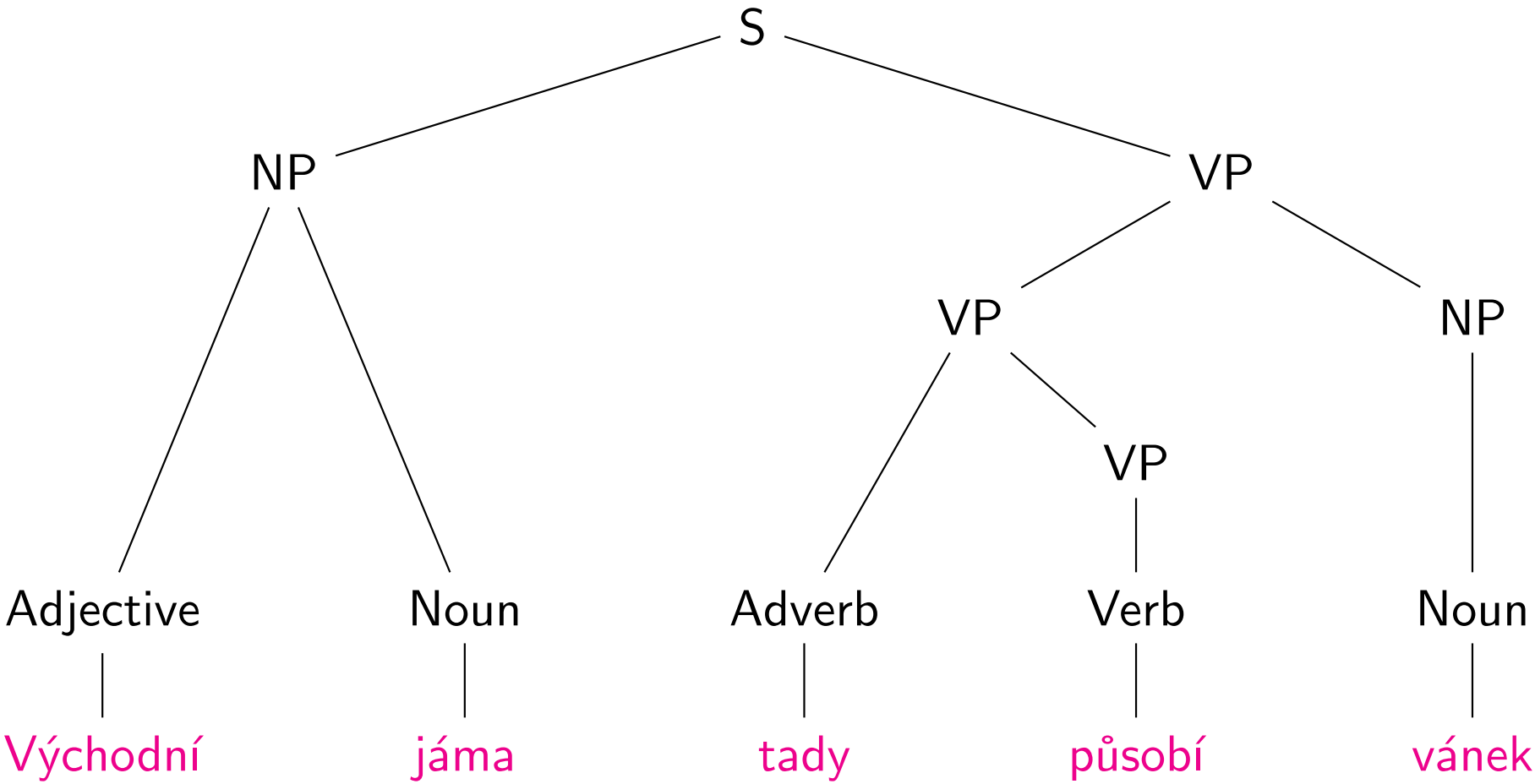
```
ajka>hnát
<s> ==hná=t= (618)
    <l>hnát
    <c>k5eAmFaI
<s> =hnát=== (1030)
    <l>hnát
    <c>k1gInSc1,k1gInSc4
```

Gramatická pravidla pro agenta ve Wumpusově jeskyni

S	→	*NP VP*	%	já + cítím vánek
	|	*S Conjunction S*	%	já cítím vánek + a + já jdu
			%	na východ
NP	→	*Pronoun*	%	já
	|	*Noun*	%	jáma
	|	*Adjective Noun*	%	levá jáma
	|	*Pronoun NP*	%	toho + wumpuse
	|	*Noun Digit* ',' *Digit*	%	pole + 3,4
	|	*NP PP*	%	jáma + na východě
	|	*NP RelClause*	%	toho wumpuse + ,který
			%	zapáchá
VP	→	*Verb*	%	zapáchá
	|	*VP NP*	%	cítím + vánek
	|	*VP Adjective*	%	je + třpytivý
	|	*VP PP*	%	jdu + na východ
	|	*VP Adverb* | *Adverb VP*	%	jdu + dopředu
PP	→	*Preposition NP*	%	na + východ
RelClause	→	', který' *VP*	%	,který + zapáchá

Syntaktický strom

syntaktický strom vzniká během syntaktické analýzy a dává záznam o jejím průběhu:



Konstrukce derivačního stromu

Neterminály opatříme argumentem:

```
sentence(sentence(NP,VP)) --> noun_phrase(NP), verb_phrase(VP).
```

Převod do podoby klauzulí:

```
sentence(sentence(NP,VP),S,S0) :- noun_phrase(NP,S,S1), verb_phrase(VP,S1,S0).
```

DC gramatika s konstrukcí stromu analýzy

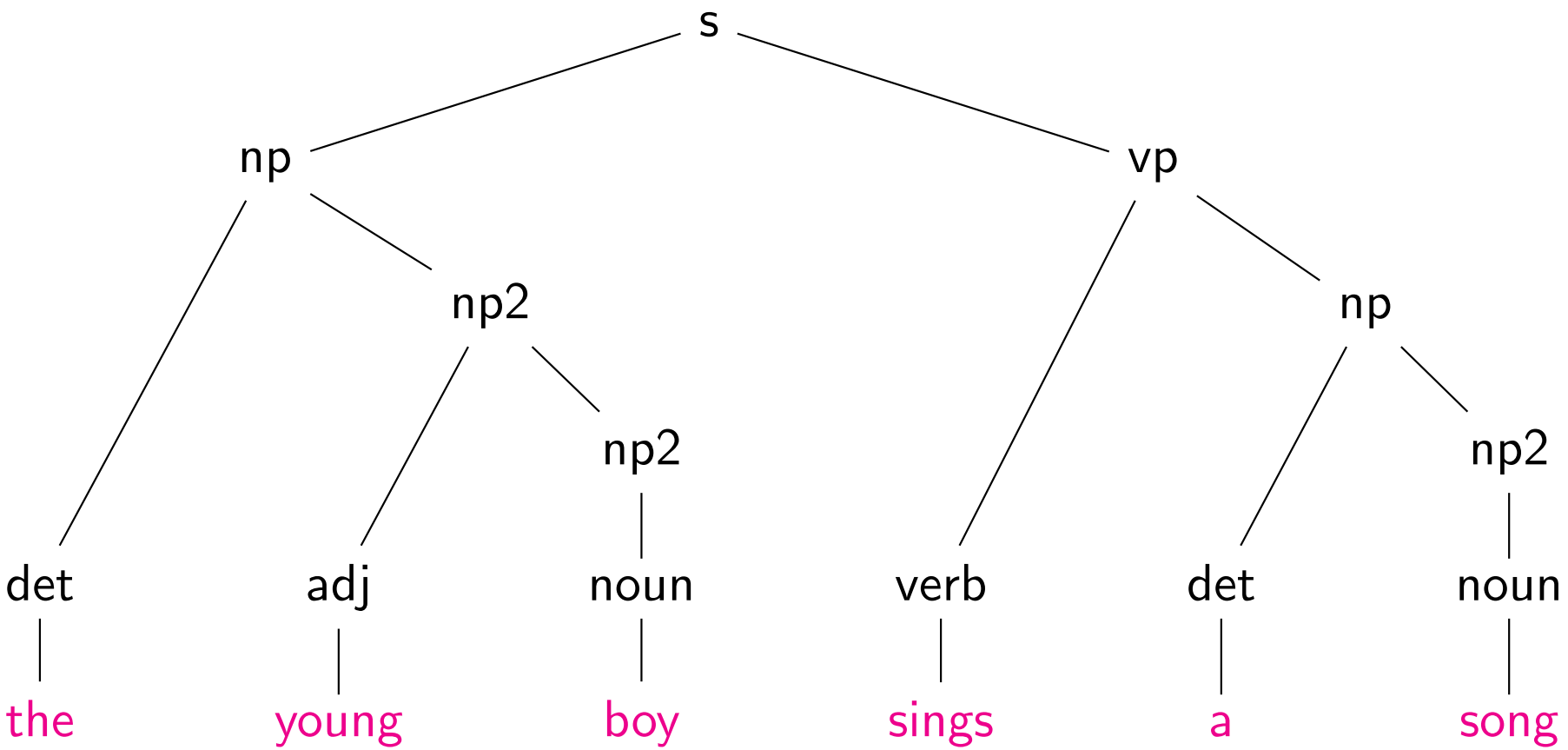
```
sentence(s(N,V)) --> noun_phrase(N), verb_phrase(V).
noun_phrase(np(D,N)) --> determiner(D), noun_phrase2(N).
noun_phrase(np(N)) --> noun_phrase2(N).
noun_phrase2(np2(A,N)) --> adjective(A), noun_phrase2(N).
noun_phrase2(np2(N)) --> noun(N).
verb_phrase(vp(V)) --> verb(V).
verb_phrase(vp(V,N)) --> verb(V), noun_phrase(N).

determiner(det(the)) --> [the].
determiner(det(a)) --> [a].
adjective(adj(young)) --> [young].
noun(noun(boy)) --> [boy].
noun(noun(song)) --> [song].
verb(verb(sings)) --> [sings].

?- sentence(Tree, [the,young,boy,sings,a,song],[]).
    Tree=s(np(det(the),np2(adj(young),np2(noun(boy)))),
           vp(verb(sings),np(det(a),np2(noun(song)))))
```

Derivační strom analýzy v DC gramatikách

```
?- sentence(Tree, [the, young, boy, sings, a, song], []).
   Tree=s(np(det(the), np2(adj(young), np2(noun(boy)))),
          vp(verb(sings), np(det(a), np2(noun(song)))))
```



Test na shodu

Pokud však rozšíříme slovník:

```
noun(noun(boys)) --> [boys].
verb(verb(sing))  --> [sing].
```

Narazíme na problém se shodou v čísle:

```
?- sentence(_,[a, young, boys, sings],[]).
Yes

?- sentence(_,[a, boy, sing],[]).
Yes
```

Proto rozšíříme neterminály o další argument **Num**, ve kterém můžeme testovat shodu:

```
sentence(sentence(NP,VP)) --> noun_phrase(NP,Num), verb_phrase(VP,Num).
```

DC gramatika s testy na shodu

```
sentence(sentence(N,V)) --> noun_phrase(N,Num), verb_phrase(V,Num).
noun_phrase(np(D,N),Num) --> determiner(D,Num), noun_phrase2(N,Num).
noun_phrase(np(N),Num) --> noun_phrase2(N,Num).
noun_phrase2(np2(A,N),Num) --> adjective(A), noun_phrase2(N,Num).
noun_phrase2(np2(N),Num) --> noun(N,Num).
verb_phrase(vp(V),Num) --> verb(V,Num).
verb_phrase(vp(V,N),Num) --> verb(V,Num), noun_phrase(N,Num1).

determiner(det(the),_) --> [the].
determiner(det(a),sg) --> [a].
verb(verb(sings),sg) --> [sings].
verb(verb(sing),pl) --> [sing].
adjective(adj(young)) --> [young].

noun(noun(boy),sg) --> [boy].
noun(noun(song),sg) --> [song].
noun(noun(boys),pl) --> [boys].
noun(noun(songs),pl) --> [songs].

?- sentence(_,[a, young, boys, sings],[]).
   No
?- sentence(_,[the,boys,sings,a,song],[]).
   No
?- sentence(_,[the,boys,sing,a,song],[]).
```

Podmínky v těle pravidel

DC gramatiky mohou mít pomocné podmínky v těle pravidel – libovolný Prologovský kód

CFG pro vyhodnocení artimetického výrazu:

$$E \to T + E \mid T - E \mid T$$
$$T \to F * T \mid F / T \mid F$$
$$F \to (E) \mid f$$

zapíšeme včetně výpočtu hodnoty výrazu:

```
expr(X) --> term(Y), [+], expr(Z), {X is Y+Z}.
expr(X) --> term(Y), [-], expr(Z), {X is Y-Z}.
expr(X) --> term(X).

term(X) --> factor(Y), [*], term(Z), {X is Y*Z}.
term(X) --> factor(Y), [/], term(Z), {X is Y/Z}.
term(X) --> factor(X).

factor(X) --> ['('], expr(X), [')'].
factor(X) --> [X], {integer(X)}.

?- expr(X,[3,+,4,/,2,-,'(',2,*,6,/,3,+,2,')'],[]).      %   3 + 4/2 - (2*6/3 + 2) = -1
    X = -1
```

Generativní síla DCG

Generativní (rozpoznávací) síla DCG je větší než CFG
např. jazyk $a^n b^n c^n$:

```
abc --> a(N), b(N), c(N).

a(0) --> [].
a(s(N)) --> [a], a(N).

b(0) --> [].
b(s(N)) --> [b], b(N).

c(0) --> [].
c(s(N)) --> [c], c(N).

?- abc(X,[]).
X = [] ;
X = [a, b, c] ;
X = [a, a, b, b, c, c] ;
X = [a, a, a, b, b, b, c, c, c] ;
...
```

Význam syntaktické analýzy

- analýza syntaxe je nutná pro analýzu významu
- většina teorií analýzy významu využívá princip kompozicionality:

  Význam složeného výrazu je funkcí významu jednotlivých podvýrazů

- proces sémantické analýzy:
  - buď vychází z výsledků syntaktické analýzy
  - nebo probíhá současně se syntaktickou analýzou; pak může zasahovat i do tvorby syntaktického stromu

Problémy při analýze přirozeného jazyka

- víceznačnost
- anaforické výrazy
- indexické výrazy
- nejasnost
- nekompozicionalita
- struktura promluvy
- metonymie
- metafory

Víceznačnost

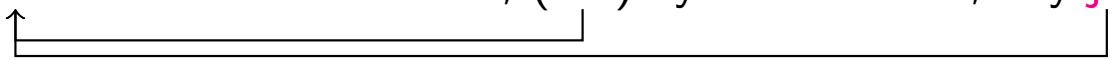
- *ambiguity*
- víceznačnost může být lexikální, syntaktická, sémantická a referenční
- lexikální – "stát," "žena," "hnát"
- syntaktická – "Jím špagety s masem."
  "Jím špagety se salátem."
  "Jím špagety s použitím vidličky."
  "Jím špagety se sebezapřením."
  "Jím špagety s přítelem."
- sémantická – "Jeřáb je vysoký." "Viděli jsme veliké oko."
- referenční – "Oni přišli pozdě." "Můžeš mi půjčit knihu?"
  "Ředitel vyhodil dělníka, protože (on) byl agresivní."

Anaforické a indexické výrazy

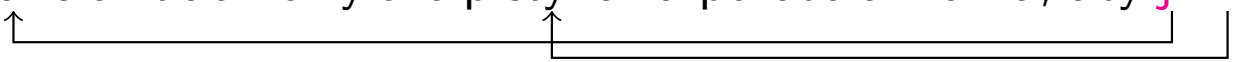
anaforické výrazy:

- *anaphora*
- používají zájmena pro odkazování na objekty zmíněné dříve

"Poté co se Honza s Marií rozhodli se vzít, (oni) vyhledali kněze, aby je oddal."



"Marie uviděla ve výloze prstýnek a požádala Honzu, aby jí ho koupil."



indexické výrazy:

- *indexicals*
- odkazují se na údaje v jiných částech promluvy nebo mimo promluvu

"Já jsem tady."
"Proč jsi to udělal?"

Metafora a metonymie

metafora:

- *metaphor*
- použití slov v přeneseném významu (na základě podobnosti), často systematicky

> "Zkoušel jsem ten proces zabít, ale nešlo to."
> "Bouře se vzteká."

metonymie:

- *metonymy*
- používání jména jedné věci pro (často zkrácené) označení věci jiné

> "Čtu Shakespeara."
> "Chrysler oznámil rekordní zisk."
> "Ten pstruh na másle u stolu 3 chce další pivo."

Nekompozicionalita

- *noncompositionality*
- příklady porušení pravidla kompozicionality u ustálených termínů nebo přednost jiného možného významu při určitých spojeních

  "aligátoří boty," "basketbalové boty," "dětské boty"

  "pata sloupu"

  "červená kniha," "červené pero"

  "bílý trpaslík"

  "dřevěný pes," "umělá tráva"

  "velká molekula"

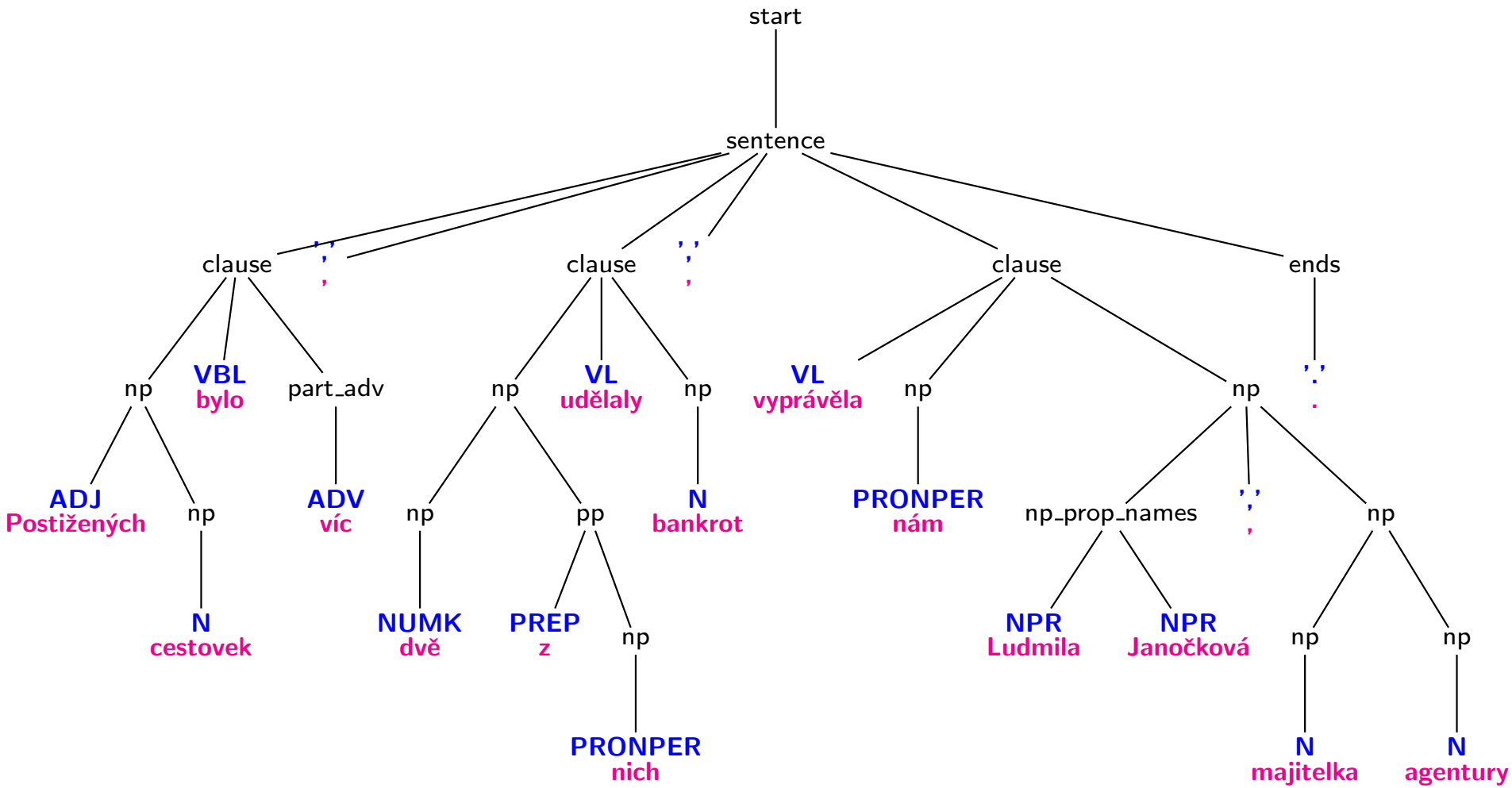
Reálná syntaktická analýza přirozeného jazyka

- velice rozsáhlé gramatiky (desítky až stovky tisíc pravidel)
- silná víceznačnost – někdy až obrovské množství (>milióny) možných syntaktických stromů

  Obehnat Šalounův pomník mistra Jana Husa na pražském Staroměstském náměstí živým plotem z hustých keřů s trny navrhuje občanské sdružení Společnost Jana Jesenia.

- existují efektivní algoritmy pro takové gramatiky
  např. tabulkový analyzátor (*chart parser*), beží v $O(n^3)$, tisíce slov/sekundu

Příklad stromu analýzy v systému synt



http://nlp.fi.muni.cz/projekty/wwwsynt/

Příklad logické analýzy v systému synt

Když je pořádná zima s množstvím sněhu, ani velký nával návštěvníků přírodě příliš nevadí.

$$\lambda w_1 \lambda t_2 \Big[\mathbf{když_ani}_{w_1 t_2},$$
$$\lambda w_3 \lambda t_4 (\exists i_5) \Big([\mathbf{pořádný}_{w_3 t_4}, i_5] \ \wedge \ [\mathbf{zima}_{w_3 t_4}, i_5] \ \wedge$$
$$\Big[\big[\mathbf{s}_{w_3 t_4}, [\mathbf{Of}, \mathbf{množství}, \mathbf{sníh}] \big]_{w_3 t_4}, i_5 \Big] \Big),$$
$$\lambda w_6 \lambda t_7 \Big[\mathbf{Not}, \Big[\mathbf{True}_{w_6 t_7}, \lambda w_8 \lambda t_9 (\exists x_{10}) (\exists i_{11}) (\exists i_{12}) \Big($$
$$\big[\mathbf{Does}_{w_8 t_9}, i_{12}, [\mathbf{Imp}_{w_8}, x_{10}] \big] \ \wedge \ [\mathbf{příroda}_{w_8 t_9}, i_{11}] \ \wedge$$
$$x_{10} \ \subset \ [\mathbf{vadit}, i_{11}]_{w_8} \ \wedge \ [\mathbf{příliš}, x_{10}] \ \wedge$$
$$\Big[\big[\mathbf{velký}, [\mathbf{Of}, \mathbf{nával}, \mathbf{návštěvník}] \big]_{w_8 t_9}, i_{12} \Big] \Big) \Big] \Big] \Big] \ldots o_{\tau\omega}$$

PA026 – Projekt z umělé inteligence

- navazuje na předmět *PB016 Úvod do umělé inteligence*
- volba programovacího jazyka ovšem není nijak omezena
- samostatná volba tématu v rozsahu ≥ 1 semestru
- předmět probíhá jako konzultace
- zajímavé výsledky (`http://nlp.fi.muni.cz/uiprojekt/`)
  - projekt elnet – > 5 let spolupráce na grantových projektech simulace elektrorozvodných sítí
  - projekt plagiaty_z_webu – reálné a funkční vyhledávání shod s dokumenty na celém webu
  - projekt robot_johnny_5 – sestavení a "oživení" robota – mobilního počítače