# Vybrané aktuální projekty Laboratoře NLP

Vojtěch Kovář, Jan Pomikálek

E-mail: xkovar3@fi.muni.cz, xpomikal@fi.muni.cz
http://nlp.fi.muni.cz/poc_lingv/

Obsah:

- SET – syntaktická analýza pomocí postupné segmentace věty
- Corpus Architect

# Syntaktická analýza přirozeného jazyka

Syntaktická analýza:

- odhalení povrchové struktury věty
- základ pro analýzu jazyka na vyšších úrovních

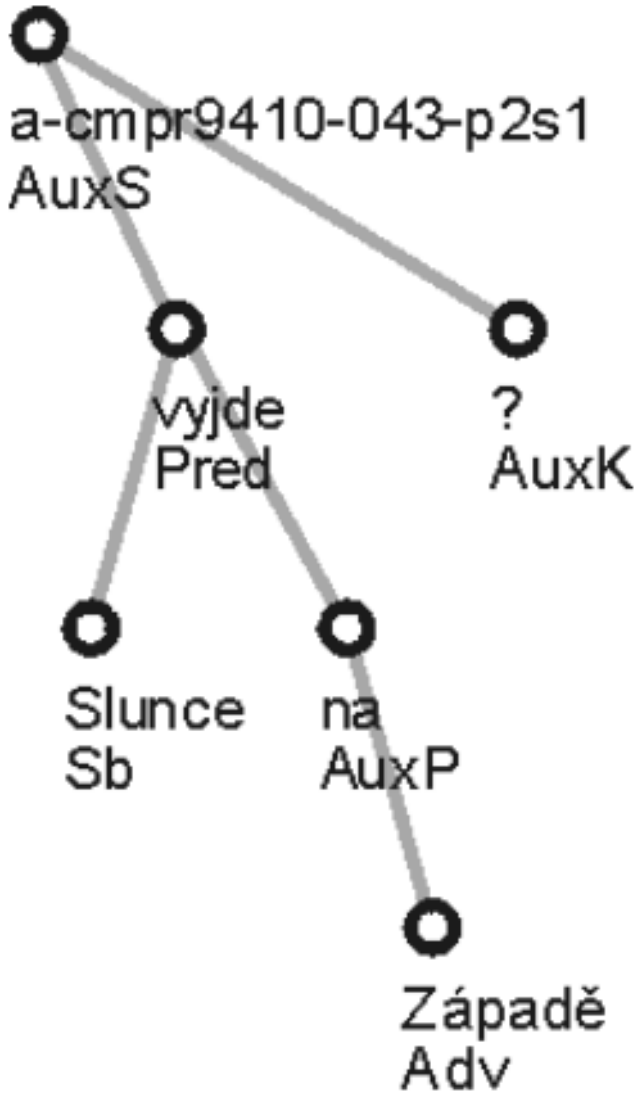
Závislostní formalismus:

- strukturní vztahy kódovány závislostmi mezi slovy na vstupu
- pražský korpus závislostních stromů PDT

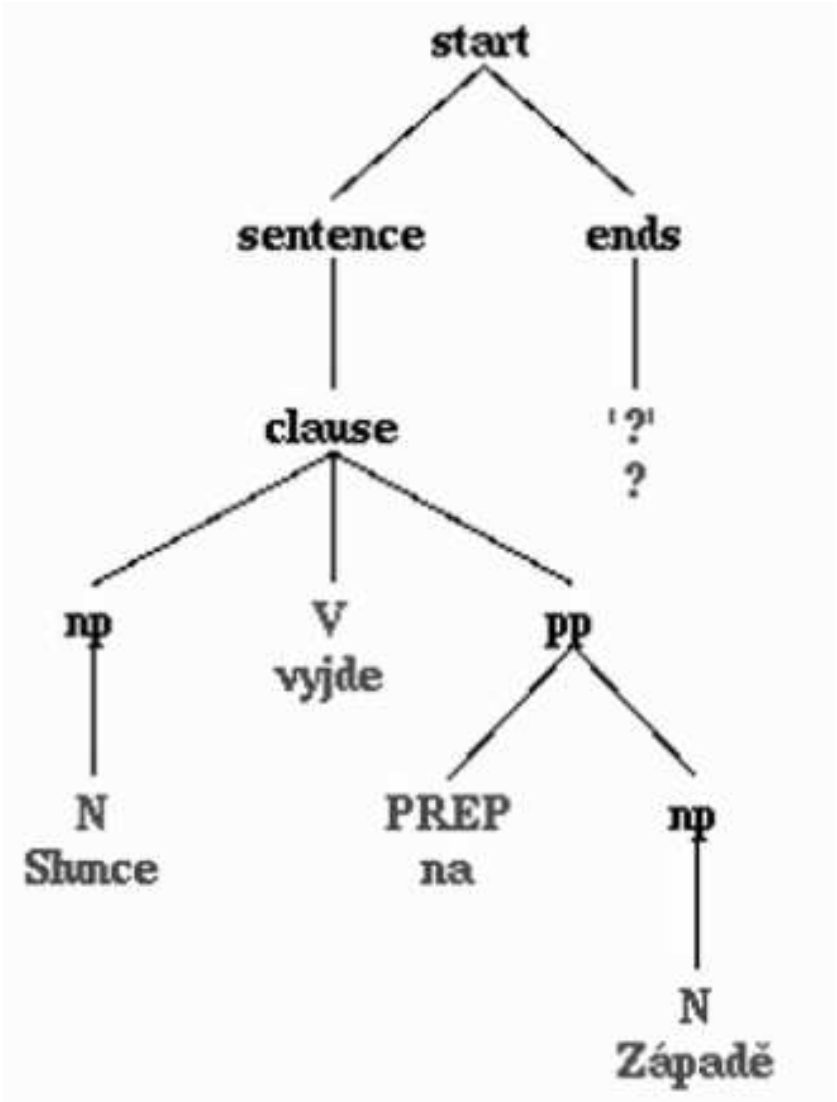
Složkový formalismus:

- strukturální vztahy popisovány stromem odvození z gramatiky
- brněnský analyzátor `synt`

# Závislostní strom – příklad



# Složkový strom – příklad



# Syntaktická analýza přirozeného jazyka

Parciální syntaktická analýza:

- nezajímá nás kompletní strom, jen některé vztahy
- např. systém `VaDis`, Word Sketches

Použití syntaktické analýzy:

- jakékoli pokročilejší zpracování jazyka
- např. vztahy mezi slovy $\rightarrow$ logické konstrukce
- identifikace frází v textu
- ...

# Metoda postupné segmentace věty

Základní myšlenky:

- některé syntaktické jevy jsou lépe rozpoznatelné než jiné
- nejprve určíme snadnější vztahy, dále pokračujeme složitějšími
- analýza probíhá v několika vrstvách (úrovních)
- z každé úrovně dostaneme parciální syntaktickou informaci

Principy:

- využití principů parciální analýzy pro analýzu úplnou
- rozdělení procesu analýzy do několika vrstev
- pravidlový systém – množina vzorků
- pattern matching – vyhledávání vzorků v textu

# Jazyk pro definici pravidel

Každé pravidlo obsahuje dvě části – šablonu a akce

- šablona určuje, co se v textu má hledat
- akce určují, jaké syntaktické vztahy mají být vyznačeny
- pravděpodobnostní ohodnocení nalezených vzorků – délka, pravděpodobnost pravidla apod.

Příklady pravidel:

```
prep ... noun       AGREE 0 2 c MARK 2 DEP 0
noun ... noun2      MARK 2 DEP 0
verb ... comma conj ... verb ... bound      MARK 2 7 <relclause>
```

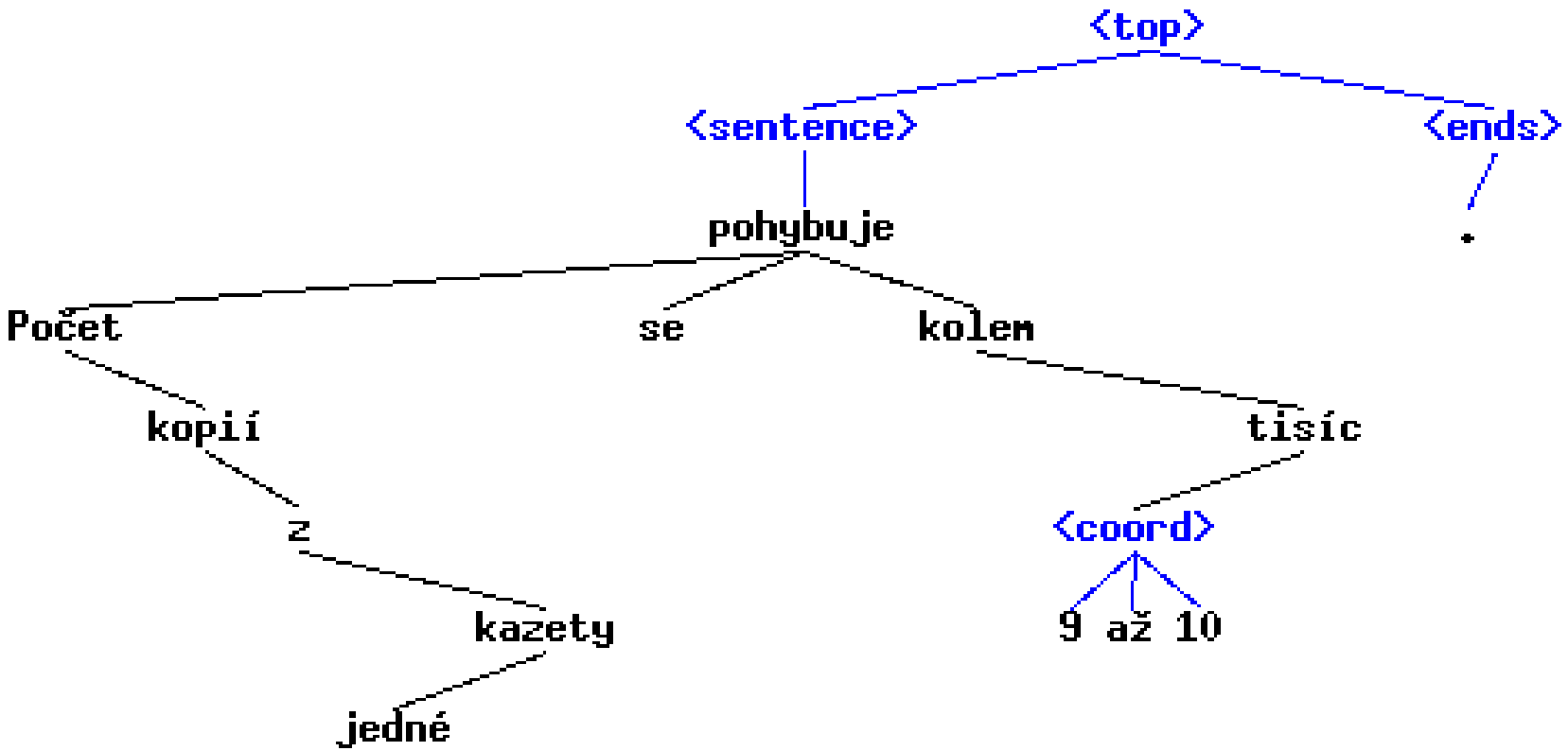
# Výstup analýzy

Tzv. hybridní stromy – kombinují závislostní a složkové prvky

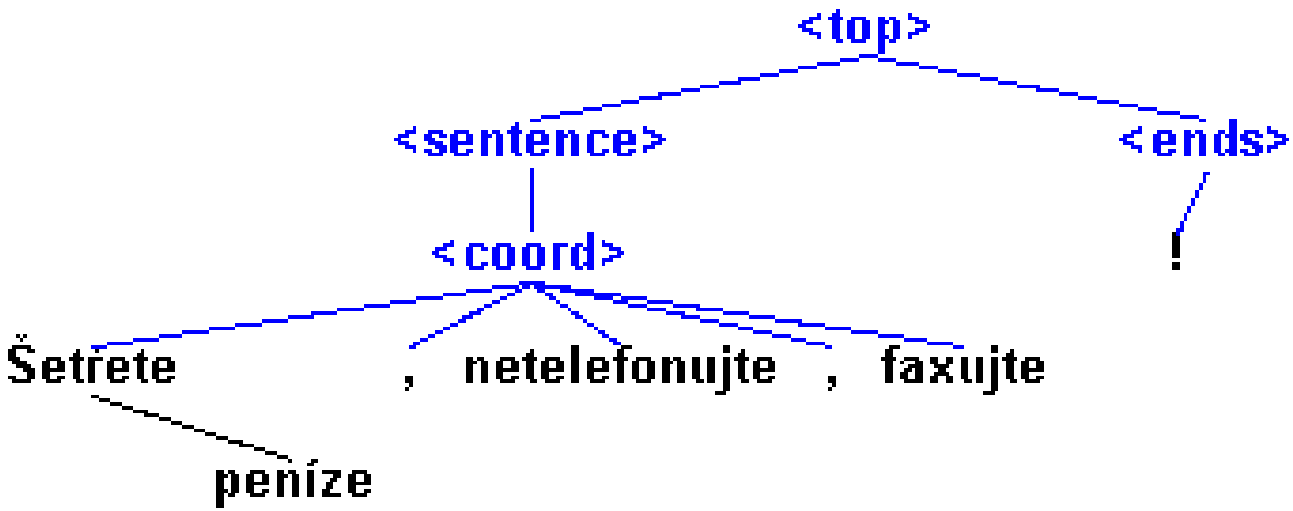
- čitelnější pro člověka
- rozlišování složkových a závislostních jevů je výhodou při analýze
- možnost převodu do čistě závislostního formátu

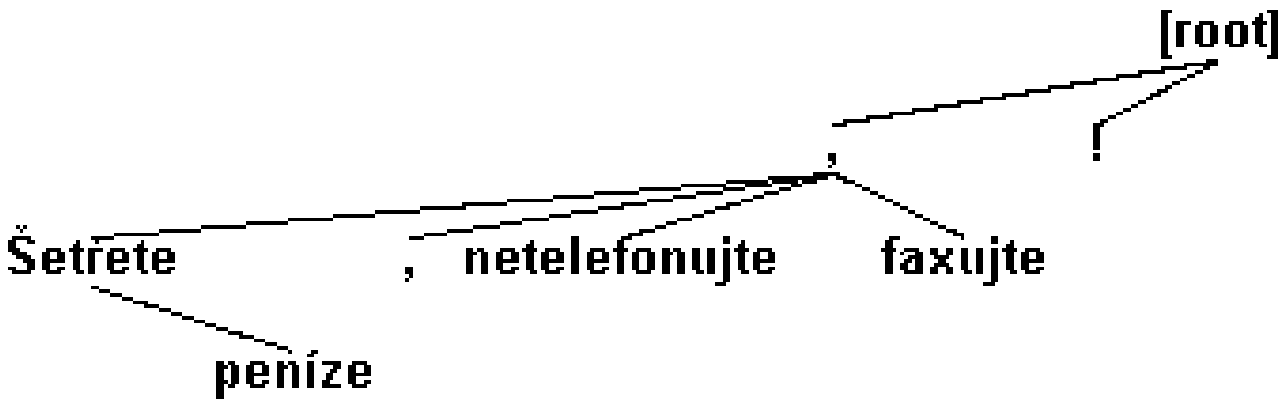
Na výstupu analýzy je vždy jediný strom, na stderr se vypisují všechny nalezené vzorky – zachycení možné víceznačnosti

# Hybridní strom – příklad



# Hybridní a závislostní strom



# Implementace – systém SET

„Syntax as Elements of Text"

- implementace v jazyce Python
- objektový model věty, pravidel a syntaktických vztahů
- ucelený soubor pravidel pro analýzu syntaxe češtiny
- 2000 řádků kódu, 150 pravidel

Funkce:

- analýza morfologicky označkovaného textu
- výstup ve formě hybridních i závislostních stromů
- reprezentace víceznačnosti ve formě výpisů na `stderr`
- grafická vizualizace výstupu

# Přesnost a rychlost

Přesnost závislostního výstupu (vzhledem k datům z PDT):

| Testovací sada | Přesnost – průměr | Přesnost – medián |
|---|---:|---:|
| PDT e-test | 76,14 % | 78,26 % |
| BPT2000 | 83,02 % | 87,50 % |
| PDT50 | 92,68 % | 94,99 % |

Rychlost:

- asymptoticky $O(R\,N\,log(R\,N))$
- v praxi 0.14 sekundy na větu

# Shrnutí

Syntaktická analýza metodou postupné segmentace věty:

- postupně vyhledáváme vzorky v textu (pattern matching)
- vybíráme a vyznačujeme nejpravděpodobnější z nich

Výhody navrženého přístupu:

- jednoduchost a průhlednost ve srovnání s formálními přístupy
- čitelnost kódu (Python vs. C)
- čitelnost množiny pravidel
- nezávislost na anotovaných datech

`http://nlp.fi.muni.cz/projects/set`

# Obsah

# Co je Corpus Architect?

- program pro tvorbu korpusů
  - z vlastních textů (.txt, .html, .pdf, .ps, .doc)
  - z webu (comming soon)
- nástupce dvou používaných nástrojů
  - CorpusBuilder
  - WebBootCaT

# Proč potřebujeme nový software?

- integrace CB a WBC (můžu vytvořit korpus částečně ze svých textů, částečně z webu)
- zlepšení uživatelské přívětivosti (zejména potřeba u CB)
- robustnější implementace; rozšíření o nové funkce bez vnášení hacků

# Filosofie návrhu

- maximální jednoduchost použití
- systém průvodců (wizards), uživatel dělá jen jednoduché volby
- neptat se uživatele na to, co dokážeme spolehlivě autodetekovat
- co dokážeme detekovat méně spolehlivě, nabídnout jako výchozí volbu

# Implementované funkce

- zpracování textových dokumentů v několika různých formátech
- zpracování vertikálních souborů
- automatická detekce kódování s vysokou úspěšností (téměř 100%)
- detekce strukturálních značek
- POS-tagging, lemmatizace (TreeTagger + vlastní pre/post-processing)
- detekce hranic vět
- kompilace korpusu pro Sketch Engine
- kompilace word sketches a statistického thesauru
  - pomocí předpřipravených gramatik
  - pomocí vlastních gramatik
- paralelní zpracování, procesy na pozadí, sledování průběhu (progress bary)
- automatický bug-reporting

# TO-DO list

- získávání dat z webu (obdoba WebBootCaT)
- sdílení korpusů mezi uživateli
- zpracování velkých korpusů
  - optimalizace některých procedur, zpracování na pozadí
  - upload dat přes FTP
- upload balíků souborů (.zip)
- přidávání nástrojů/dat
  - značkovače
  - tokenizéry pro některé asijské jazyky (japonština, čínština)
  - sketch grammars (gramatiky pro tvorbu WS)
- sjednocení vzhledu se Sketch Engine (+ lepší propojení)