Vybrané aktuální projekty Centra ZPJ

Jan Rygl, Vojtěch Kovář

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

Obsah:

- Rozpoznávání autorství anonymních dokumentů na Internetu
- SET – aplikačně orientovaná syntaktická analýza

Motivace

Proč se začalo mluvit o určování autorství?

- Může za to Shakespearovo dílo:
  - Gale Ecco, 1787:
    A Dissertation on the Three Parts of King Henry VI. Tending to Shew That Those Plays Were Not Written Originally by Shakspeare.

Motivace

Koho autorství dokumentů zajímá?

- Ministerstvo vnitra
  - Anonymní výhrůžky (ohlášení bomby, hrozby sousedům, kolegům)
  - Publikování ilegálních textů pod pseudonymem
- Soudy
  - Zpochybnění závěti (ověření shody autorství)
  - Falešné doznání obžalovaného napsané policisty (**1. použití před soudem**)
- Firmy
  - Kdo pomlouvá na Internetu firmu
  - Publikují kritikové/pochlebovači pod více účty?

Motivace

Prostředí Internetu?

- anonymita, nejsou k dispozici logy a používají se proxy servery
- prostor pro extremismus, podporu terorismu, podvody
- velké množství dat znemožňuje manuální analýzu lingvisty

Dennis Bayley, 2004:
Anonymity is the single most important enabler of criminal activity.

Praktické zadání

1. **Verifikace**: Máme množinu dokumentů psaných pod dvěma pseudonymy. Mají dokumenty jednoho autora?
2. **Shlukování**: Máme příspěvky od několika autorů, lze některé autory ztotožnit?
3. **Přiřazování s kandidáty**: Máme anonymní dokumenty a množinu potenciálních autorů. Pokud dokumenty patří někomu z potenciálních autorů, kterému?
4. **Přiřazování bez kandidátů**: Máme anonymní dokument, chceme zjistit autora.
   - Až zde potřebujeme Internet.
   - Pokud předem neomezíme množinu (autor je registrovaný na webu, bydlí v nějaké vesnici apod.), úloha je velmi "ambiciózní".
   - Předpokládá se, že skutečný **autor někdy publikoval pod svým pravým jménem** (bakalářská práce, inzerát, ...)

Jak na ověření autorství

Mějme dokumenty psané pod dvěma pseudonymy *A* a *B*. *Jaká je pravděpodobnost, že autor A a B je jedna osoba?*

Postup:

1. Analýza textů
   - Detekce jazyka (např. `langid.py`)
   - Detekce kódování (laboratorní `chared`)
2. Zpracování textů
   - Odstranění šumu (text a formátování, které nevytvořil autor)
   - Tokenizace
   - Morfologická analýza a desambiguace (značky, lemmata, doplnění diakritiky), pro češtinu `majka`, pro jiné jazyky např. `Stanford POSTagger`
   - Možná syntaktická analýza (pro češtinu a angličtinu laboratorní `SET`)
3. Vlastní analýza autorství dokumentů.

Příklad zpracování textu

```
<p align="justify"> <img style="margin: 3px; float:
left" src="otaznik.jpg" alt=""> Společnost se za
svou historii dokázala
```

↓

```
<s>
<img src="otaznik.jpg"/>
Společnost   společnost   k1gFnSc1
se           sebe         k3xPyFc4
za           za           k7c4
svou         svůj         k3xOyFpXgFnSc4
historii     historie     k1gFnSc4
dokázala     dokázat      k5eAaPmAgFnS
```


Metody autorství: charakteristické rysy autora

Vychází se z předpokladu, že každý autor má inviduální:

- aktivní slovní zásobu,
- oblíbené fráze a posloupnosti slov,
- znalost gramatiky
- a typografické znalosti.

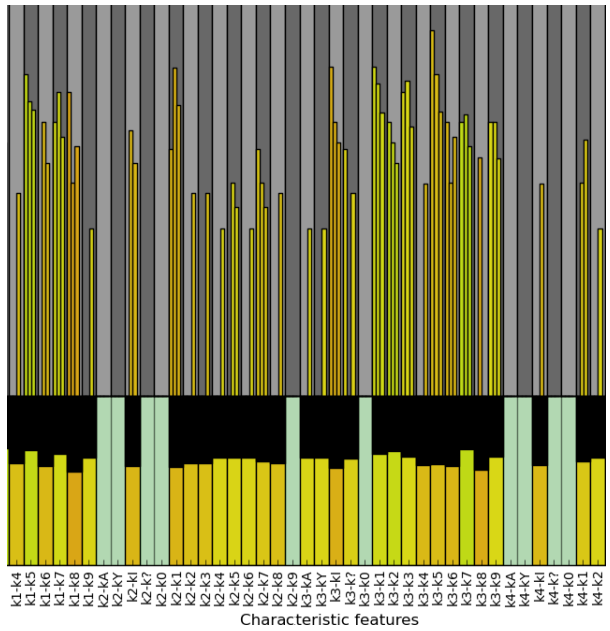
Jelikož tyto své návyky používá autor podvědomě, lze pomocí nich vytvořit rysy autora, které ho charakterizují.

Metody autorství: charakteristické rysy autora

- Jazykově závislé
  - Osoba mluvčího (pohlaví, číslo)
  - Analýza gramatických značek v textu
  - Analýza počtu vět (hlavní, vedlejší, ...)
  - Chyby v textu (překlepy, hrubky, syntax)
- Jazykově nezávislé (stačí tokenizace)
  - Analýza délky vět (počet slov, znaků)
  - Analýza délky slov (porovnání histogramů)
  - Frekvence slov, bigramů, ... (ovlivněna tématem)
  - Frekvence stopslov (tematicky nezávislá)
  - δ-score (srovnání frekvencí slov v textu s běžnou frekvencí slova v korpusech)
  - Bohatost slovní zásoby

Extrakce rysů z textu

- Pro každou kategorii (např. bigramy slovních druhů) a pro každý text spočítáme hodnoty jednotlivých atributů.
- Atribut je vyčíslitelná charakteristika autora s hodnotou $a_i^{cat}(d) \in \langle 0,1 \rangle \cup \{undef\}$
- Např. pro bigramy slovních druhů je atributem relativní frekvence bigramu k2-k1 (adj-noun)



Př. pro 3 dokumenty od 1 autora

Porovnání dvou dokumentů

- Pro každou kategorii *cat* a dokument *d* máme atributy $a_i^{cat}(d)$.
- Podobnost dvou dokumentů vzhledem ke kategorii *cat* definujeme jako vektor invertovaných vzdálenosti atributů dvou dokumentů:

$$sim^{cat}(d_1, d_2) = [1 - |a_i^{cat}(d_1) - a_i^{cat}(d_2)|, \ldots]$$

  Rozdíl atributů bereme v absolutní hodnotě a normujeme podle rozložení hodnot v korpuse.
- Podobnost dvou dokumentů lze charakterizovat jako vektor podobností kategorií $[sim^{cat_1}(d), \ldots]$, kde každá podobnost kategorií je vektor podobností atributů.

Porovnání dvou dokumentů – strojové učení

- S autory a dokumenty, u nichž známe autorství, vytvoříme velké sady dvojic dokumentů tak, abychom měli stejný případ shod i neshod (např. 10000 od každého). Každé dvojice bude reprezentována n-ticí n-tic.
- Použijeme **strojové učení**, aby se naučilo rozpoznávat n-tice signalizující shodu a rozdílnost autorů.
  Získáme tak model M takový, že:

$$M\Big(\big[sim^{cat_1}(d), \ldots, sim^{cat_n}(d)\big]\Big) = P\big(autor(A) == autor(D)\big)$$

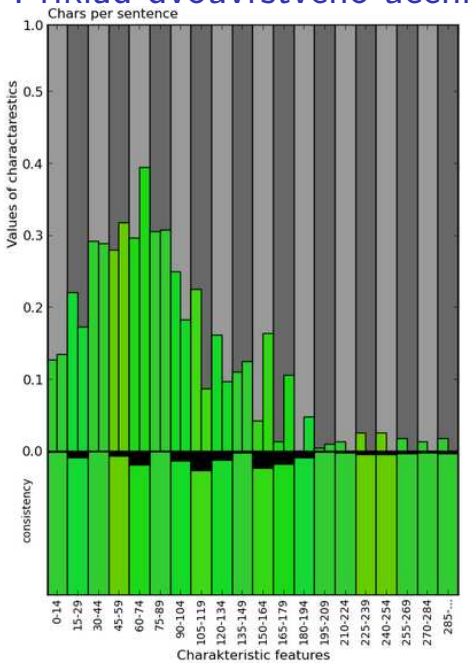
- Vždy, když budeme chtít srovnávat dva dokumenty, extrahujeme jejich autorské charakteristiky a předložíme je jako n-tici modelu. Ten vrátí odpověď.

Porovnání dvou dokumentů – strojové učení

Volby:

- Který přístup ke strojovému učení použít?
  - SVM: nejlepší výsledky, podporuje pravděpodobnost, pomalé
  - Naive Bayes: dobré pro testování hypotéz, rychlé a podporuje pravděpodobnost
- Strojové učení pracuje s vektorem atributů, ne s vektorem vektorů
  - Jednovrstvé učení:
    vytvořit jeden vektor, pokud budou charakteristiky uspořádané
    – pomalé
    – příliš mnoho atributů
  - Dvouvrstvé učení:
    rozdělit strojové učení do dvou vrstev
    v 1. vrstvě se pro každou kategorii vytvoří model
    v 2. vrstvě se pracuje pouze s jednou pravděpodobností za kategorii, tj. s jednotkami hodnot
    + flexibilní přístup, rychlejší
    – nelze hledat souvislosti mezi atributy z různých kategorií

Příklad dvouvrstvého učení



Partial morph. tags	0.5924
Full morph. tags	0.4601
Wordclasses	0.2118
Wordclass bigrams	0.2421
Punctuation	0.9231
Chars per sentence	0.9426
Syntactic analysis	0.6214
Upgraded syntactic analysis	0.6884
Stopword	0.9523
Word repetition	0.3996
Vocabulary richness Simpson	0.4778
Typography	0.5885
Emoticons	0.4062
Capital letters stats	0.2704
global	0.5

Porovnání dvou množin dokumentů

1. Spočítáme pravděpodobnost shody autorství pro každou dvojici dokumentů
   - z 1. množiny (konzistence 1. autora), $C1$
   - z 2. množiny (konzistence 2. autora), $C2$
   - takovou, že jsou z různých množin (podobnost množin), Sim
2. Pro $C1, C2, Sim$ převedeme množiny hodnot podobnosti dvojic dokumentů na jedno číslo jako vážený průměr:

$$weight(p) = 100 \cdot [0.5 + |p - 0.5|]^2$$

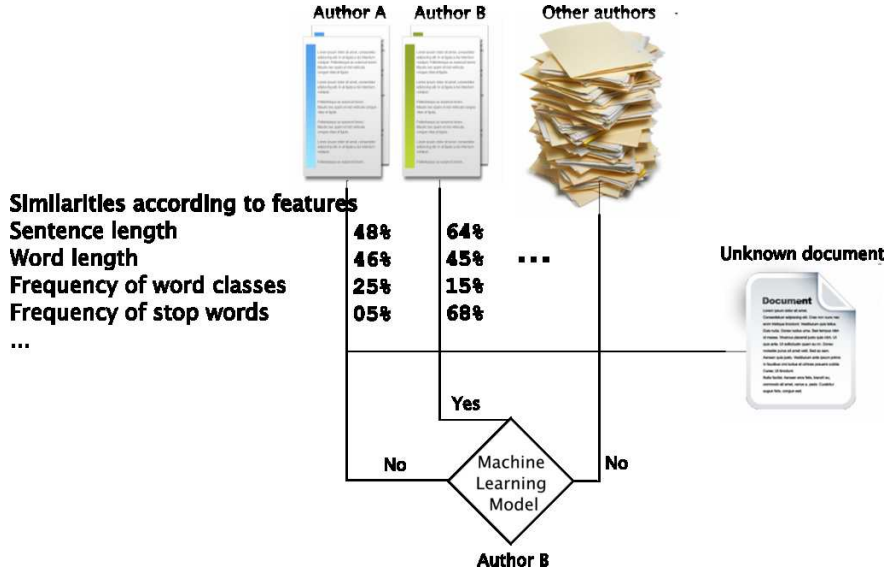
3. Pokud je podobnost sim podobná hodnotám $C1$ a $C2$ či vyšší, autory množin považujeme za 1 autora.
   Pokud je podobnost sim řádově nižší, autorství je různé.

Příklad:

- 1. avg sim: 0.89, autor Less
- 2. avg sim: 0.78, autor Fairyfire
- border value: $\frac{0.78}{0.89} \cdot 0.78 = 0.68$
- distance: $0.62 \rightarrow$ Less ! = Fairyfire

Rozšíření na přiřazování autorství

Vybereme toho kandidáta, který je nejpodobnější. Kandidát však musí opět překročit min. mez, pokud ne, nikdo z kandidátů text nenapsal.



Rozšíření na přiřazování autorství – nahrazení podobnosti pořadím

Knihy:

dlouhý konzistentní text

Blogy, Fóra:

středně dlouhý text

E-maily, Tweety, Diskuze:

krátký zašuměný text

- Jako atribut nemusíme brát podobnost dokumentů, ale pořadí podobnosti ve srovnání s ostatními dokumenty.
- Docílíme tím univerzálních modelů strojového učení, protože každá skupina textů má jiné hranice podobnosti pro shodu autorství.

Úspěšnost závisí na typu dokumentu

Verifikace:

- knihy, eseje: 95 % → 99 %
- blogy, články: 70 % → 90 %

Přiřazování (záleží na počtu kandidátů, př. z blogů):

- 4 kandidáti: 80 % → 95 %
- 100 kandidátů: 40 % → 60 %

Shlukování:

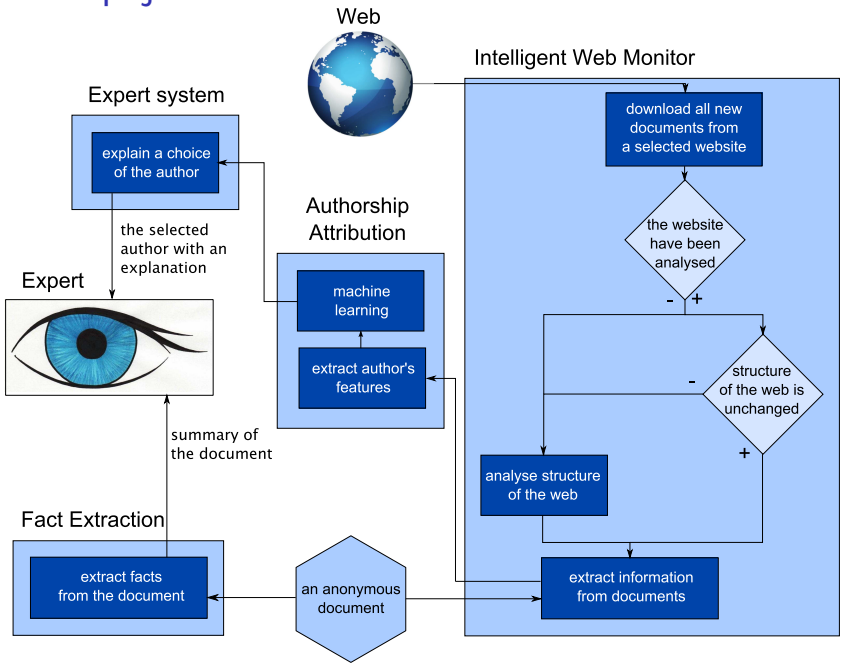
- vyhodnocení záleží případ od případu, není metrika

Přidání Internetu

Co potřebujeme:

- Znát weby, kde jsou texty autorů.
- Detekovat strukturu webů.
- Pravidelně stahovat nové dokumenty z webů (nutná struktura).
  - Odhalit změnu struktury webu a aktualizovat informace.
- Dokumenty spravovat v databázi.
  - Vyhledávání (stovky tisíc a více dokumentů)
  - Hledání dle času, kategorií, autora, . . .
- Předzpracovávat si dokumenty.
- Ukládat si drahé mezivýsledky (nepočítat např. frekvence slovních druhů vícekrát pro jeden dokument).

Schéma zapojení Internetu



Shrnutí

- Projekt se dokončuje tento rok, ale práce na jeho rozšiřování budou pokračovat nadále
- Pokud vás něco zaujalo, je pravý čas se přidat
  - lingvistika nebo statistika (1 b)
    Vytvářet **nové charakteristiky autora** (analýza chyb, nářečí, počet vět, formátování textu, či nejlépe vymyslet vlastní)
  - grafika nebo analýza dat (1 d)
    Vymyslet kreativní přístupy k **vizualizaci výstupních dat**, případně k sumarizaci výsledků programu, aby jim rozuměl školený uživatel
  - programování a struktura webu (1 d, 1 b)
    Navrhnout nové metody pro **automatickou detekci struktury webu**, přihlašování se ke zdrojům vyžadujícím autentizaci, vyhledávání odkazů na dokumenty v doméně
  - strojové učení a analýza
    Hrát si s různými metodami **strojového učení** a frameworky
  - vše ostatní
    A mnoho dalšího, stačí se domluvit, jsou potřeba **lingvisti, programátoři, grafici, právníci,** ...

Syntaktická analýza přirozeného jazyka

Syntaktická analýza:

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

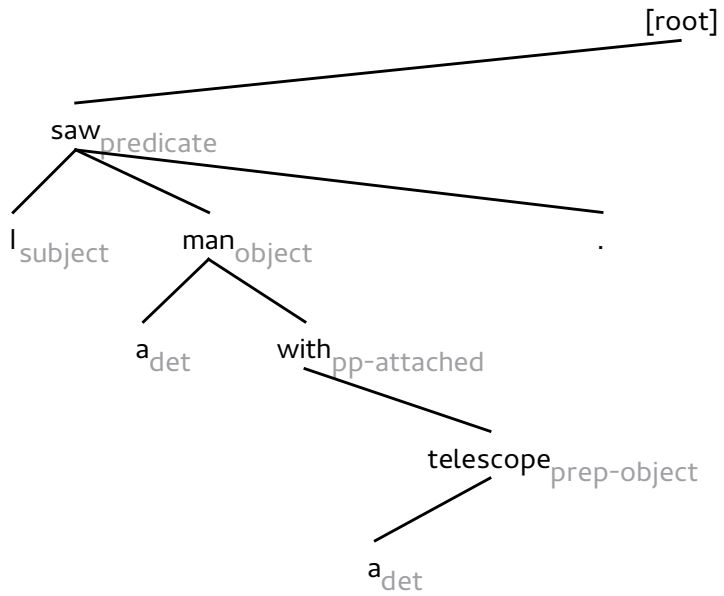
Závislostní formalismus:

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

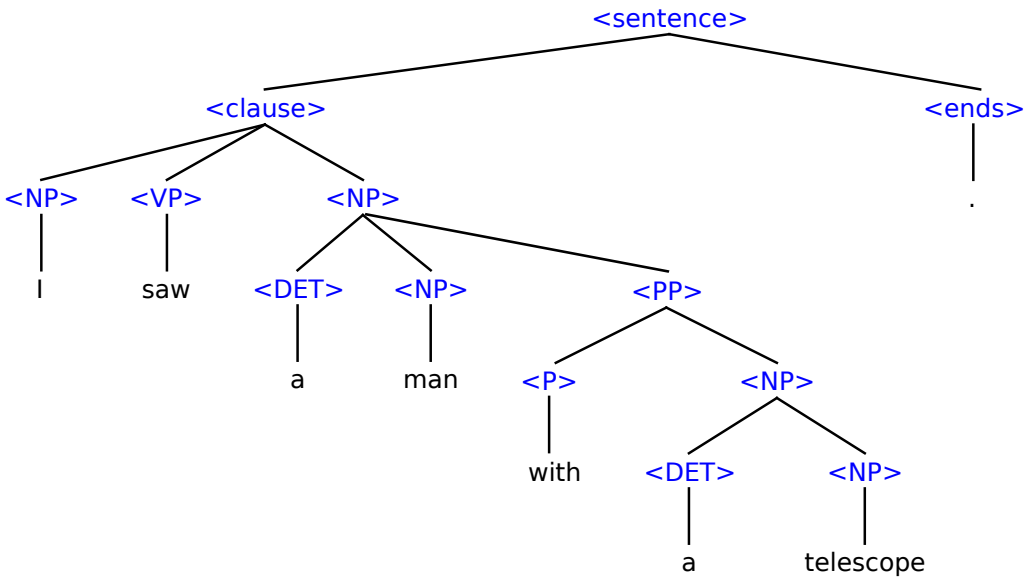
Složkový formalismus:

- 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 → logické konstrukce
- odvozování z textu
- extrakce informací
- opravy jazykových chyb
- ...

Syntaktický analyzátor SET

„Syntactic Engineering Tool"

- jednoduchost v návrhu i v použití
- → snadné úpravy pro použití v různých aplikacích
- některé syntaktické jevy jsou lépe rozpoznatelné než jiné
- nejprve určíme snadnější vztahy, dále pokračujeme složitějšími

Principy:

- využití principů parciální analýzy pro analýzu úplnou
- 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
- a morfologické shody
- pravděpodobnostní ohodnocení nalezených vzorků – délka, pravděpodobnost pravidla

Příklady pravidel:

```
prep ... noun       AGREE 0 2 c MARK 2 DEP 0 PROB 500
noun ... noun2      MARK 2 DEP 0

[tag k1] ... [tag k1c2]       MARK 2 DEP 0

verb ... comma conj ... verb ... bound       MARK 2 7 <relclause>
```

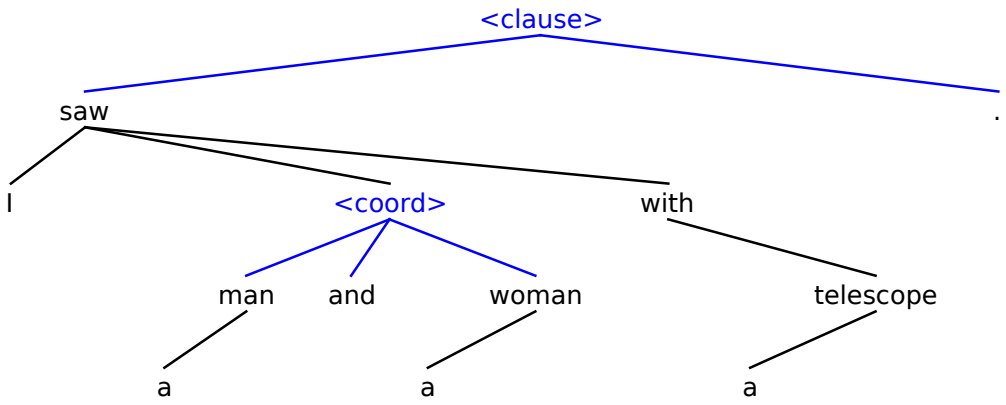

Výstup analýzy

Tzv. **hybridní stromy** – kombinují závislostní a složkové prvky

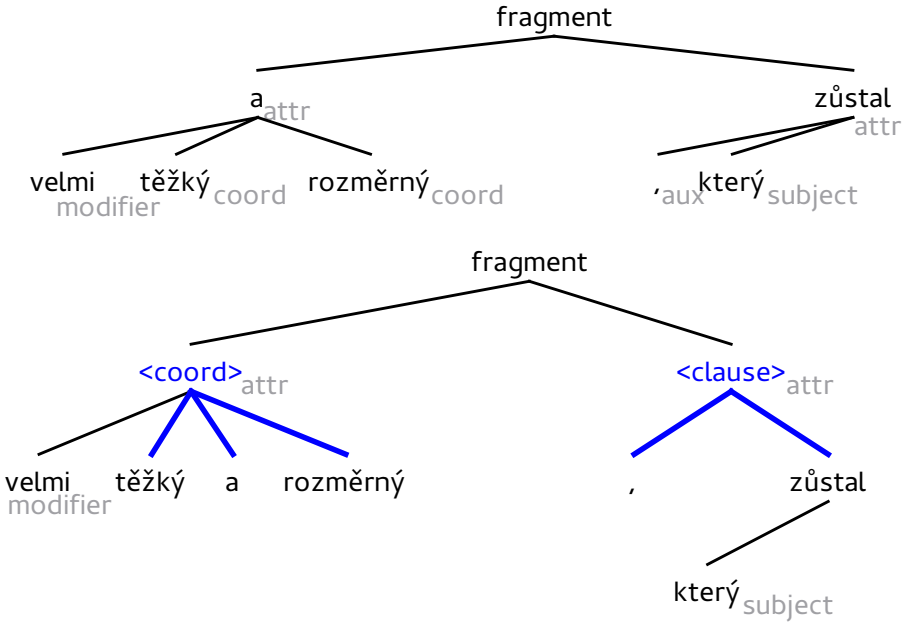
- čitelnější pro člověka
- rozlišování složkových a závislostních jevů je výhodou při analýze
- možnost převodu do čistě závislostního i čistě složkového formátu

Na výstupu analýzy je vždy **jediný strom**, možnost výpisu **všech nalezených vzorků** – zachycení možné víceznačnosti

Hybridní strom – příklad



Hybridní a závislostní strom



Implementace

Technické detaily

- implementace v jazyce Python
- objektový model věty, pravidel a syntaktických vztahů
- ucelený soubor pravidel pro analýzu syntaxe češtiny
- gramatiky pro angličtinu, slovenštinu, slovesné fráze ...
- 3000 řádků kódu, 70 pravidel

Funkce:

- analýza morfologicky označkovaného textu
- výstup ve formě různých typů stromů, frází a kolokací
- reprezentace víceznačnosti
- grafická vizualizace výstupu

Přesnost a rychlost

Rychlost:

- asymptoticky $O(R\,N^2\,log(R\,N^2))$
- v praxi 0.14 sekundy na větu

Přesnost závislostního výstupu (vzhledem k PDT, SET v0.3):

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

Aplikace

Poslední vývoj:

- metodologie vyhodnocování proti anotovaným datům je kontraproduktivní
- → zaměření na využití v aplikacích

Aplikace:

- verifikace autorství
- extrakce informací
- automatické odvozování z textu
- automatické opravy chyb
- skloňování českých frází
- rozpoznávání anafor
- automatické odpovídání na otázky
- ...

Shrnutí

Syntaktický analyzátor SET:

- 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 a procesu analýzy
- nezávislost na anotovaných datech
- → lepší využitelnost v praktických aplikacích

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