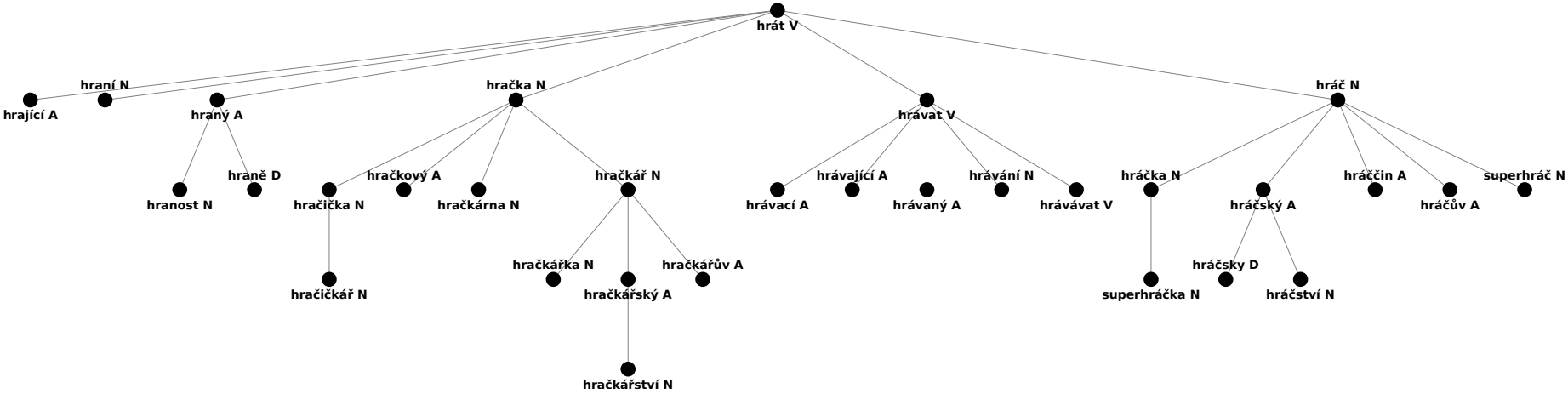
DeriNet: Lexikální databáze českých derivátů

Magda Ševčíková, Zdeněk Žabokrtský

Univerzita Karlova v Praze
Ústav formální a aplikované lingvistiky

Praha, 15. prosince 2014


hrát V
hraní N
hrající A
hraný A
hranost N
hraně D
hračka N
hračičkový A
hračička N
hračičkář N
hračkárna N
hračkář N
hračkářka N
hračkářský A
hračkářův A
hračkářství N
hrávat V
hrávající A
hrávací A
hrávaný A
hrávání N
hrávávat V
hráč N
hráčka N
superhráčka N
hráčský A
hráčsky D
hráčství N
hráčkin A
hráčův A
superhráč N

Technická část - osnova

1. Motivace
2. Technické prostředky pro vývoj DeriNetu
3. Pracovní postup
4. Kvantitativní vlastnosti v. 0.9
5. Závěrečné poznámky, otevřené otázky, práce do budoucna

Motivace

Lingvistická motivace I

- standardní: existence dat usnadní lingvistický výzkum derivační morfologie
- z hlediska experimentů podložených elektronickými daty zatím pro češtinu poměrně neprobádaná oblast
- neexistuje žádná derivačně označkovaná obdoba ČNK

Lingvistická motivace II – zkoumání vývoje jazyka

- slovotvorba jako archeologické okénko do historie jazyka
- vývoj nelze přesně rekonstruovat ani přesně predikovat
  - extrémní množství nahodilých mimojazykových vlivů
  - ultramultiagentní systém
- zkoumáním dynamické rovnováhy ale můžeme poznávat hlavní jazykové samoorganizační mechanismy, jako jsou
  - konkurence prostředků (v tlaku jazykové ekonomie)
  - záporná zpětná vazba
  - kladná zpětná vazba

Aplikační motivace

- řada úkolů v NLP se dotýká jazykového významu
  - MT, IR, postojová analýza ...
- problém:
  - slovní forma = kořenový morfém + inflexní morfémy + derivační morfémy
- jádro významu – kořenový morfém, ale jak se k němu dostat?
  1. očištění formy od inflexních morfémů (lemmatizace)
  2. očištění formy od derivačních morfémů (jak tomu nazývat: nesting? rooting? niching? hnízdování?)
- sémantická souvislost se v řetězcové podobnosti manifestuje velmi nepřímočaře
- kde pomáhá lemmatizace, mohl by pomoci i nesting

Nešlo by to nějak jednoduše?

- alternativa k derivační databázi: naivnější řešení jako např. sada regulárních výrazů pro časté záměny přípon a časté hláskové změny
- nevyhnutelně povede ke značnému nadgerování, příklady:
  - ovarium - ovar
  - pohádka - pohádaný
  - potkan - potkání
  - sepse - sepsání
  - sesle - seslání
  - skrutátor - skrotum
  - sněhule - sněhulák
  - sobec - sob
  - sušárna - suši
  - svinutí - svině
  - vůle - vůl
  - štěnice - štěně
  - ženista - žena

Technické prostředky pro vývoj DeriNetu

Implementační rozhodnutí

- maximální recyklace existujících zdrojů
- minimální objem ručních anotací - v nevyhnutelných případech postačí ad-hoc textový anotační miniformát
- objektové API
- sestavení DeriNetu spustitelné kdykoli znovu od začátku
- se souborovým formátem se raději držme při zemi (tsv, dump)
- v počáteční fázi důraz spíše na bezchybnost než na pokrytí (očekávání: pokrytí později doženeme strojovým učením)

Jednoduchá datová struktura

- lexém
  - uzel grafu
- derivační vztah mezi základním a odvozeným lexémem
  - hrana grafu
- slovotvorné hnízdo (slovní čeled, derivational nest)
  - kořenový strom
- derivační databáze
  - les

Aplikační rozhraní pro DeriNet

- inspirace z Treexu:
  - objektové rozhraní v Perl+Moose, zatím v namespace `Treex::Tool::DerivMorpho`
  - `procedura sestavení derinetu jako sekvence bloků`
  - `scénáře bloků spustitelné z příkazové řádky`

Ukázka scénáře v Makefile

```
assemble:
        derimor CreateEmpty \
        CS::AddLexemesFromList file=sorted_lemmas_from_syn.tsv  \
        CS::AddOstLexemesFromCNC \
        CS::AddOstLexemesByRules \
        CS::AddAdj2AdvByRules \
        CS::AddManuallyConfirmedAutorules \
        CS::Prefixes \
        CS::AddDerivationsFromLemmaSuffices \
        CS::AddManuallyConfirmedAutorules2 \
        CS::AddConfirmedMluvCandidatesMonosource \
        CS::RevertDerivationDirection \
        CS::RestructureClusters \
        CS::AddOrDeleteLinksInClusters \
        CS::AddDerivationsFromList file=otaznickovi.tsv \
        Save file=derinet09.tsv
```

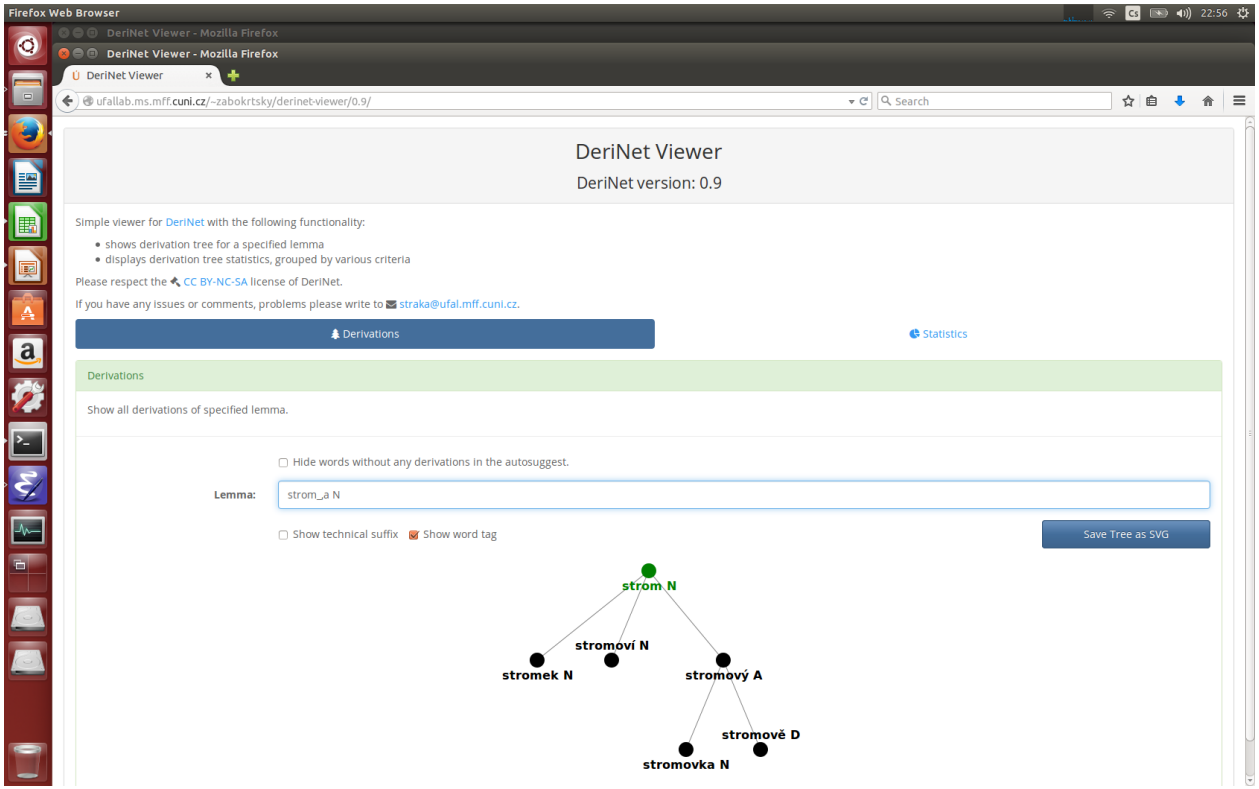
Ukázka jednoho z anotačních miniformátů

```
TRYING TO APPLY RULE V-it --> N-ba
  chodit --> chodba
  dražit --> dražba
  družit --> družba
*  holit --> holba
  honit --> honba
  hradit --> hradba
  hrozit --> hrozba
*  klátit --> klatba (CHANGE: á -> a)
  léčit --> léčba
*  mámit --> mamba (CHANGE: á -> a)
*  nadstavit --> nadstavba
*  nastavit --> nástavba (CHANGE: a -> á)
*  pažit --> pažba
  platit --> platba
*  podvolit --> podvolba
  prosit --> prosba
```

Primární souborový formát - ukázka

```
117580  cukroví         cukroví              N   117581  A2N   CS::AddConfirmedMluvCandidatesMonosource
117581  cukrový         cukrový              A   117547  N2A   CS::AddManuallyConfirmedAutorules
117582  cukrově         cukrově_^(*1ý)       D   117581  A2D   CS::AddAdj2AdvByRules
117583  cukrující       cukrující_^(*5ovat)  A   117568  V2A   CS::AddDerivationsFromLemmaSuffices
117584  cukrárenský     cukrárenský          A
117585  cukrárenství    cukrárenství         N   117584  A2N   CS::AddManuallyConfirmedAutorules
117586  cukrárna        cukrárna             N   117547  N2N   CS::AddDerivationsFromList
117587  cukrárnička     cukrárnička          N   117586  N2N   CS::AddConfirmedMluvCandidatesMonosource
117588  cukrátko        cukrátko             N
117589  cukrář          cukrář               N
117590  cukrářka        cukrářka_^(*2)       N   117589  N2N   CS::AddDerivationsFromLemmaSuffices
```

Webový prohlížeč DeriNetu, autor Milan Straka



Sestavení DeriNetu 0.9 krok za krokem

- `CS::AddLexemesFromList`
- `CS::AddOstLexemesFromCNC`
- `CS::AddOstLexemesByRules`
- `CS::AddAdj2AdvByRules`
- `CS::AddManuallyConfirmedAutorules`
- `CS::Prefixes`
- `CS::AddDerivationsFromLemmaSuffices -`
- `CS::AddManuallyConfirmedAutorules2`
- `CS::AddConfirmedMluvCandidatesMonosource`
- `CS::RevertDerivationDirection`
- `CS::RestructureClusters`
- `CS::AddOrDeleteLinksInClusters`
- `CS::AddDerivationsFromList`

Překvapivý problém: sporná homonymie

- "číslice za pomlčkou" na výstupu morfologické analýzy překvapivě často neoznačuje homonymii, ale polysémii
- v současnosti skoro 7000 n-tic lemmat rozlišených indexem, ale majících identickou flexi (kredit: Milan Straka)
- příklady:
  - *zpaličkovatelný-1 zpaličkovatelný-2*
  - *zvěcňovávatelnost-1 zvěcňovávatelnost-2 zvěcňovávatelnost-3 zvěcňovávatelnost-4*
  - *skuhrat-1 skuhrat-2 skuhrat-3*
  - *zlít-1 zlít-2*
  - *navrčet-1 navrčet-2*

DeriNet 0.9: vybrané kvantitativní charakteristiky

Velikost sítě

- počet lexémů: 305 781
- počet derivací: 117 327
- průměrná velikost hnízda: 1,6 lexému
- největší velikost hnízda: 31 lexémů (*hrát*, *řezat*)
- největší hloubka hnízda: 7 úrovní (*vědět-věda-vědec-vědátor-vědátorský-vědátorství-pseudovědátorství*)

Zastoupení slovních druhů

Lexémy podle slovního druhu:

- N: 55 % (z toho cca polovina propria)
- A: 32 % (z toho cca pětina přivlastňovací tvary proprií)
- V: 8 %
- D: 5 %

Podle slovního druhu základového a derivovaného lexému:

- N2A: 28 %
- V2A: 21 %
- V2N: 15 %
- A2D: 12 %
- N2N: 11 %
- A2N: 10 %
- V2V: 2 %
- . . .

Měření kvality dat

- 500 náhodně vybraných lexémů z DeriNetu 0.9, ručně označený základový lexém
- z toho 4 % negramatických lexémů (*mapuče*, *německýˇ*, *vracovat* . . .)
- porovnání ruční anotace s DeriNetem (zbývajících 480 lexémů): 196 rozdílů
  - precision: 0.983
  - recall: 0.650
  - f-measure: 0.783

Chybová analýza

Hlavní příčiny chybějících derivací:

- prefixace
- kompozita
- přechýlování
- zdrobněliny
- vidové protějšky

Závěrečné poznámky

Poučení z derinetového vývoje

Aneb co bych dnes dělal jinak:

- podcenil jsem užitečnost vizualizace, vizualizátor pomohl odhalit chyby dříve
- nevýhoda zvoleného přístupu:
  - je těžší udržet pořádek (více specializovaných anotací)
  - měl jsem psát víc testů (riziko interference pravidel a anotací)
- měl jsem se víc soustředit na hláskové změny, je to všeprostupující problém
- chybovou analýzu jsme měli provést dříve

Otevřené otázky (1)

- Není náš model příliš zjednodušený?
- Zjevný nedostatek: nelze reprezentovat kompozici
- Riziko falešné dichotomie - nebyla by vhodnější přechodná škála?
  - Příklad: *skok-doskok* vs. *pád-nápad*

Otevřené otázky (2)

- Netranzitivita významové podobnosti: přestože jednotlivé derivace ve stromu se mohou jevit nesporné, vzdálenější uzly se (přinejmenším z aplikačního hlediska) mohou jevit jako sémanticky naprosto nesouvisející.
  - Příklad: *řezbářství - řeznice*
- Co s homonymií a pravopisnými variantami?
- "Fantomové lexémy" (přeskakování v derivačním modelu) - někdy uvnitř derivačního stromu pocitově "chybí" lexém, je myslitelný a někdy i vyslovitelný, ale čeština takové slovo nemá.
  - Příklad: *plyn-plynárna-plynárenství, mlýn-*mlynárna-mlynárenství*

Otevřené otázky (3)

- Analogie k problémům anotace závislostní syntaxe:
  - nejistá orientace hrany, příklad: *brumla-brumlat*
  - systematicky vznikající neorientované cykly, příklad: *chemie-biochemie-chemik-biochemik*
  - nutnost příliš specifických rozhodnutí bez intuitivní opory
    - přináležitost lexému do clusteru může být jasná, přesný předek ne
    - příklad: *popěvovat*, *vandalství*, *kravinec*

Možné směry budoucího rozvoje DeriNetu

- rozšířit množství lexémů, překvapivě složitý problém
  - co lze považovat za "dobré české slovo"? (frekvence nestačí)
  - jaké výběrové kritérium zvolit? (frekvence nestačí)
  - co s proprii?
- doplnit derivace snadno extrahovatelné z dostupných zdrojů (např. z Vallexu nebo překladových slovníků)
- doplnit chybějící derivační vztahy s využitím strojového učení
- upravit logickou strukturu tak, aby bylo možné zachytit i kompozita
- doplnit k derivacím sémantickou informaci (např. ve smyslu lexikálních funkcí)
- využití derivací na tektogramatické rovině

Děkuji za pozornost!