Semantic Regularity of Derivational Relations

Pavel Šmerk

Natural Language Processing Centre
Faculty of Informatics
Masaryk University

5. 12. 2015

Introduction

- inflection
  - different forms of the same word/base form/lemma/lexeme
  - quite regular and complete
    - all Czech nouns in plural have forms for 7 grammatical cases — and e. g. the accussative case has always the same grammatical meaning
  - "morphological" analyzers
- derivation
  - relations between the words
  - irregular and incomplete
    - meaning need not to be trasparent, compositional
    - deriv. affix often cannot be attached to all words of the same class
  - $\Rightarrow$ much more difficult to process

TSD: Derivancze × DeriNet

- "Derivancze — *Deriv*ational *An*alyzer of *Cz*ech"
  - Karel Pala and Pavel Šmerk, TSD 2015
  - 16 relations, >250k pairs (>125k in CzTenTen, >80k in SYN)
  - examples of derivational relations
    - `k1f`: feminines from general masculines
      - *doktor–doktorka* (*doctor*$_{\mathrm{MASC}}$–*doctor*$_{\mathrm{FEM}}$)
    - `k1obyv`: area or city → inhabitant name relation
      - *Kanada–Kanaďan* (*Canada–Canadian*)
    - `k6a`: adjective → adverb relation
      - *dobrý–dobře* (*good–well*)
- DeriNet
  - v. 0.9: almost 120,000 word-formation relations on a set of lexemes whose existence was supported by corpus (SYN) evidence
  - Magda Sevčíková and Zdeněk Žabokrtský, ÚFAL, Charles University
  - `http://ufal.mff.cuni.cz/derinet`, presented at LREC 2014
  - new (Oct. 2015) v. 1.0: 965,535 unique lemmas, 715,729 links
    - "lexemes in DeriNet 1.0 are sampled from the MorfFlex dictionary"

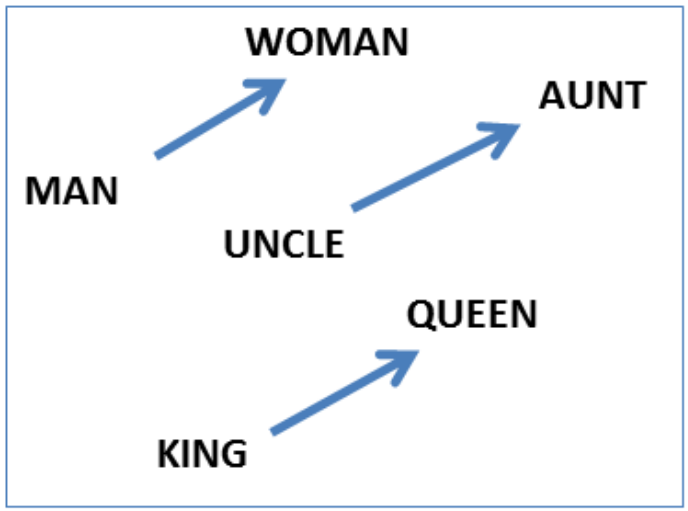
Semantically labelled relations

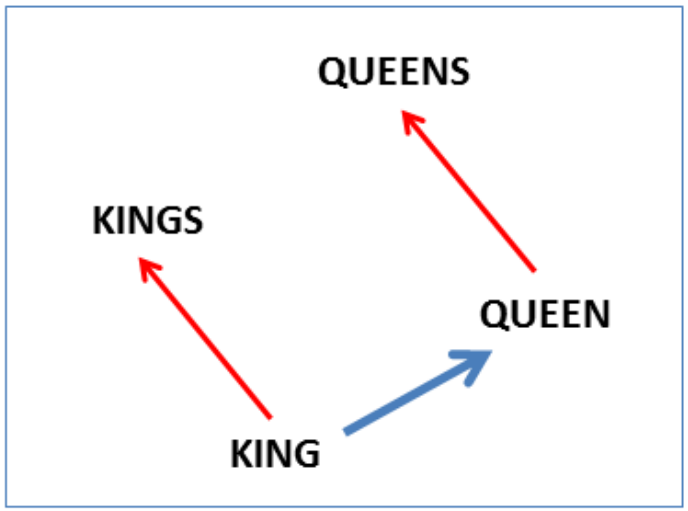
- DeriNet: relations have no explicit labels, words labelled with PoS
  - words are connected when there is a formal derivational relation between them (*černý–černucha*, *mdlý–mdloba*)
- Derivancze: only semantically transparent relations
  - e. g. not *komunismus, rusismus, revmatismus* ← *komuna, Rus, revma*
    - (*communism*, *russism*, *rheumatism*, *commune*, *Russian*, and *rheuma*)
    - formal derivational process is regular, but semantics differ
  - we are ready to ignore the direction of the formal derivational process
    - *Vietnam–Vietnamec*, *Polsko–Polák* (*Poland*), *Rusko–Rus* (*Russia*)
  - we even add derivational "suppletives", for the sake of completness:
    - masculine → feminine changes are mostly expressed by a suffix
    - *učitel* → *učitelka* (*teacher*) or *dělník* → *dělnice* (*worker*)
    - *vnuk* (*grand-son*), *medvěd* (*bear*) → *vnučka*, *medvědice*
    - *syn* (*son*), *kůň* (*horse*) → *dcera*, *kobyla*

Evaluation?

- how to evaluate correctness and usefulness of our data?
- usefulness: we need applications :-)
- correctness (semantic regularity)
  - manual evaluation: expensive, may be not reliable
  - automatic evaluation: word2vec computed on lemmatized 5.3G corpus
- benchmark of word2vec vectors: complementing triplets like
  - Greece Athens Norway ?
  - Kazakhstan Astana Zimbabwe ?
  - ...
  - $\sim$ "what is the word that is similar to Norway in the same sense as Athens is similar to Greece?"
- we are not interested in some abstract similarity of *Greece* and *Athens*
  - these words are only particular representants of some common question "what is the capital city of ..."
  - an average of more such examples seems to be a better representation

word2vec





- Tomáš Mikolov, NIPS Deep Learning Workshop 2013 slides

"Averaged Concept"

- experiment: the first 50 most frequent k1f pairs except *muž/žena*
- "questions" like otec matka muž ? or $average_{m \to f}$ muž ?

	120M		1G		5.3G	
avg	0.617013		0.744693		0.775647	
min	0.434191	občan/občanka	0.509271	občan/občanka	0.541018	občan/občanka
max	0.729597	otec/matka	0.857678	táta/máma	0.889231	táta/máma
1	**0.799218**	žena	**0.877726**	žena	**0.890839**	žena
2	0.648836	dívka	0.754475	dívka	0.762988	dívka
3	0.503029	ženský	0.610383	ženský	0.623256	chlapec
4	0.471445	otrokyně	0.595448	chlapec	0.621017	ženský
5	0.467925	matka	0.573623	matka	0.600329	mužův
6	0.467344	chlapec	0.564603	děvče	0.595389	dívenka
7	0.467089	milenec	0.555519	mladík	0.591686	děvče
8	0.466913	mladík	0.550952	mužův	0.582920	matka
9	0.462479	tmavovlasý	0.546331	partnerka	0.575861	partnerka
10	0.461867	chlap	0.541432	dívenka	0.572923	družka

Evaluation of Derivational Relations

- we should find concept for each relation and check all pairs against it
- experiment: concept from TOP50, evaluation of TOP150, refinement

	TOP10	TOP1	rank	distance	TOP10	TOP1	rank	distance
k1ag	53	6	4.42	0.595907	47	3	3.81	0.613398
k1dem	88	39	1.57	0.709737	85	40	1.59	0.718124
k1f	123	95	0.59	0.783001	123	98	0.56	0.786888
k1obyv	133	85	1.06	0.699675	133	86	1.00	0.703276
k1prop	88	30	1.69	0.654496	89	35	1.74	0.659931
k1verb	130	59	1.63	0.708456	127	56	1.83	0.712785
k2pas	134	99	0.59	0.828821	131	78	0.81	0.792059
k2pos	102	61	1.10	0.744492	95	51	1.36	0.740843
k2proc	119	79	1.14	0.738209	118	76	1.14	0.738124
k2rakt	44	3	3.75	0.563547	47	3	3.83	0.566550
k2rel	122	62	1.12	0.717147	123	56	1.57	0.713637
k2rpas	127	38	1.98	0.719895	124	40	2.06	0.719619
k2ucel	50	1	4.78	0.632999	52	1	4.71	0.636890
k6a	118	53	1.59	0.627307	116	65	1.31	0.635010

Conclusion and Future Work

- the results show that vectors computed by word2vec are useful for checking the semantic consistency of the derivational relations
- number of pairs in TOP10 or even TOP1 is consistently rather high
- many of "unsuccessful" pairs were wrong or irregular
  - *plat* (*salary*), *plátek* (*slice*) for `k1dem` (*plátek* is from *plát*)
  - *věřit* (*believe*), *věřitel* (*creditor*) for `k1ag`
- we can enrich the derivational pairs with a distance from concept

- better lemmatization
  - *vira* sám o *se* představovat značný *nebezpeč* ($\Rightarrow$ vir, sebe, nebezpečí)
- try preserve negation and may be degree ($\uparrow \Rightarrow$ nepředstavovat)
  - ten samozřejmě být žádný katastrofa ($\Rightarrow$ nebýt)
  - sedm léto starý vir by mít představovat velký nebezpečí ($\Rightarrow$ nemít)
- fine tune parameters of word2vec, evaluate the distance instead of rank
- manually clean data, perhaps better refinement