DMoG: A Data-Based Morphological Guesser aka já buřtgulám, ty buřtguláš, všichni chceme buřtgulat

Vojtěch Kovář, Pavel Rychlý

Natural Language Processing Centre
Faculty of Informatics, Masaryk University

Lexical Computing

{xkovar3,pary}@fi.muni.cz

RASLAN 2021

Morphological guesser use cases

- Lemmatizing out-of-vocabulary (OOV) words
  - buřtguláš, online, komorbidita, flash, groupe, knedlo, nVidia
- Bootstrapping lemmatization of a new language
  - manually annotate part of the word list
  - learn patterns
  - automatically annotate next part
  - manually fix the annotations
  - learn better patterns
  - repeat until happy

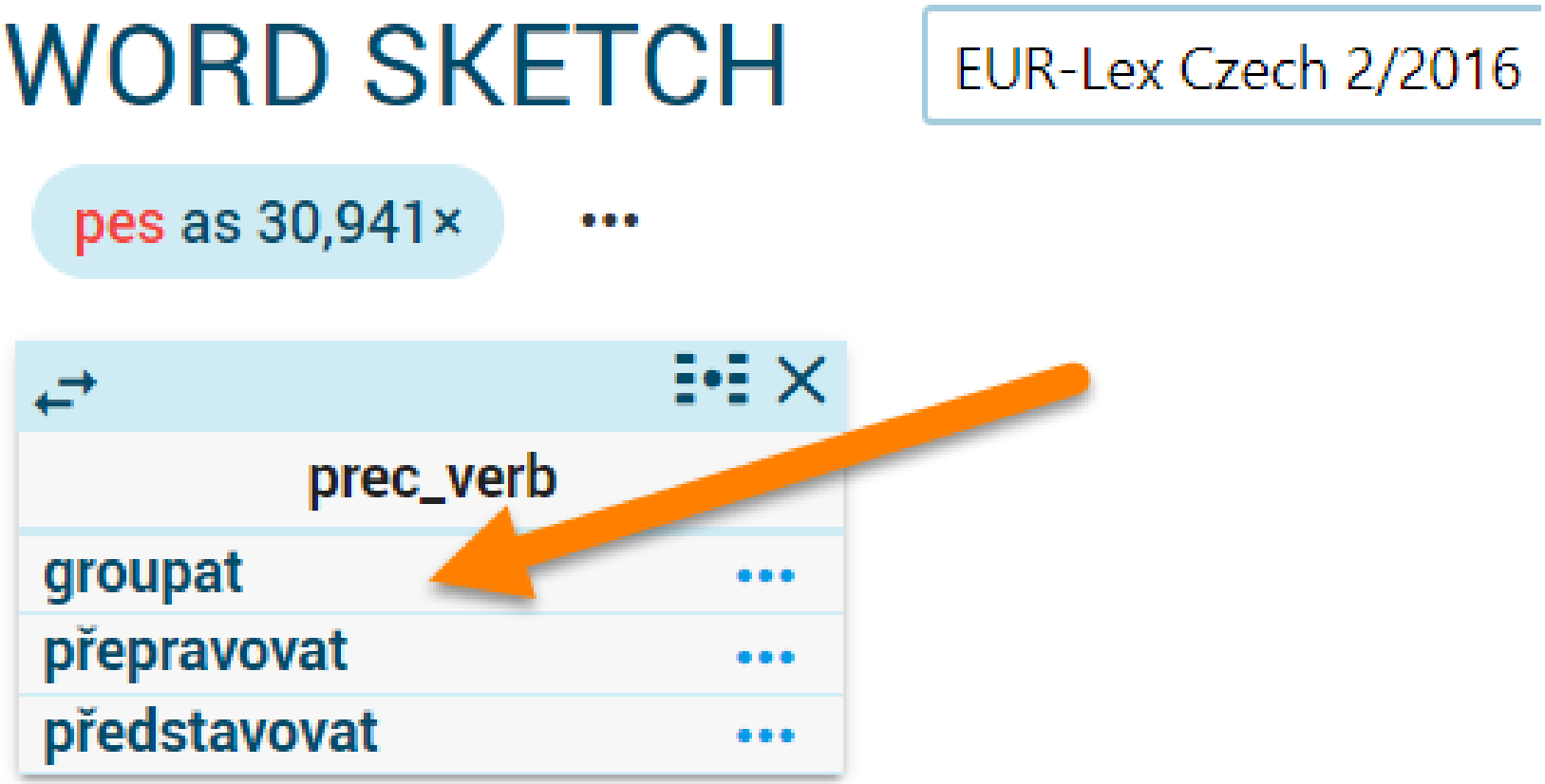
Existing approaches

- CSTlemma, Šmerk 2008 (desamb)
  - based on matching affixes
  - the longer match, the better
  - guessing one OOV word at a time

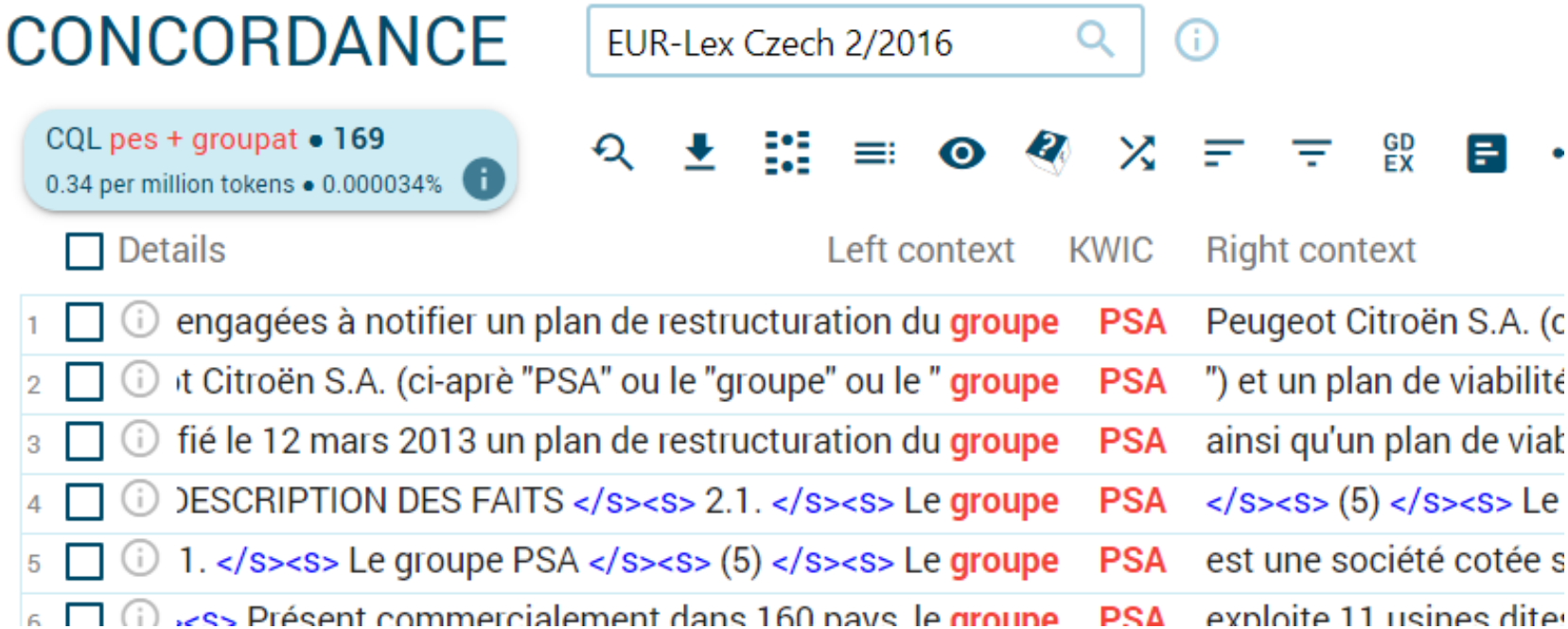
Existing approaches (II)

- Funny effects (Czech)
  - buřtguláš → buřtgulat
  - knedlo → knednout
  - flash → flasha
  - nVidia → nVidium
  - online → onlinout
  - komorbiditou → komorbiditý (adj)
  - knedlo → knednout

Existing approaches (III)



Existing approaches (III)



- groupe → groupat
- *A kde je babička? — Ále, zase groupe psa.*

Our approach

- Organize affixes into groups/patterns
- For each candidate word-form $\rightarrow$ lemma
  - generate all forms predicted by the pattern
  - check how many of them are present in the corpus (word list)
  - pattern with most predictions present in the corpus is the best
- I.e. process the whole corpus word list as the input
  - instead of isolated word forms

Example: Buřtguláš

- Buřtguláš → buřtgulat (verb) ? (like *děláš*)
  - predicts buřtgulám, buřtguláš, buřtgulá, buřtguláme, ...
  - only buřtguláš is present in the corpus
- Buřtguláš → buřtguláš (noun) ? (like *mariáš*)
  - predicts buřtguláše, buřtgulášem, buřtguláši, buřtgulášů, ...
  - some of them will be present in the corpus
  - better candidate

Implementation

- Prototype implementation in Python
  - only suffixes
  - no connection to PoS categories
  - simple: two scripts (train, evaluate), <120 lines of code
  - can be extended easily
- Pattern
  - set of suffix pairs
  - {(-ám, -at), (-áš, -at), (-á, -at), (-ál, -at), (-l, -t), (-jí, -t), ...}

Evaluation

- Extremely preliminary :)
- 40 most frequent OOV words from csTenTen17 web corpus
- Patterns trained on DESAM manually curated corpus
- Results
  - correct: 36, incorrect: 4, accuracy: 90 %
- Compared with Šmerk (2008)
  - correct: 26, incorrect: 14, accuracy: 65 %

Conclusions

- New method of lemmatization for OOV words
- Promising initial results, more work needed