Computational morphological analysis of Czech

Pavel Šmerk

Natural Language Processing Centre
Faculty of Informatics
Masaryk University

http://nlp.fi.muni.cz/ma, aurora:/nlp/projekty/ajka
these slides: http://www.fi.muni.cz/~smerk/majka

2. 10. 2019

Morphological analysis

- basic level of text processing
  - (word forms are obvious in Czech, except *gen.*, *byl-li*, *oč/očs* etc.)
- for a word form, the morphological analysis should return
  - *lemma*, the dictionary form of the word
  - possible grammatical meanings („tags“) — values of relevant grammatical categories like part of speech, case, number, person
  - e.g., for word form *stroj* we expect
    - *stroj*: noun, masculine inanimate, singular, nominative/accusative
    - *strojit*: verb, 2nd person singular, imperative mood, imperfective
- + synthesis, lemmatization (returns only the lemma), . . .
  - (it’s not decomposition to morphemes, as one could guess)
- the talk has three parts
  - what information we want to catch and describe (s. 5–6)
  - how can we organize that data (s. 7–9, 11–18)
  - how to implement the analysis itself (s. 10, 19–22)

Tags

- strings representing grammatical information
- positional tagset: a tag consists of values only
  - the corresponding category is determined by the position in the tag
  - Prague system — 16 positions: part of speech, detailed PoS, gender, number, case, . . . , negation, . . .
  - `NNIS4-----A-----`
    - noun, general, masc. inanim., singular, accusative, affirmative
  - full description
- attributive tagset: attribute–value pairs, the order does not matter
  - Brno system — similar categories and values like in Prague
    - e.g., attribute `c` for case with values `1` to `7`
  - `k1gInSc4` = noun, masc. inanim., singular, accusative
    - no detailed PoS and negation
  - advantages: shorter, easier to read and extend, simpler RE
  - https://nlp.fi.muni.cz/raslan/2011/paper05.pdf

Tags

- „heterogeneous“ tagset (Bratislava)
  - like the positional system, but without empty positions
  - the first character denotes the part of speech, the following symbols correspond to attribute–value pairs
  - the order is fixed, although each symbol is used in only one „sense“
  - `SSis4`
    - noun, noun declension, masc. inanim., singular, accusative
  - pros: the shortest tags
  - cons: hard to extend — limited set of ASCII symbols
    - $\Rightarrow$ two character values like `:r` for proper names
  - `https://korpus.sk/morpho_en.html`
- different type of language, different solution: BNC tagset
  - a fixed set of few dozens tags, for example:
    - `AJ0` Adjective (general or positive) (e.g. good, old, beautiful)
    - `AJC` Comparative adjective (e.g. better, older)
    - `AJS` Superlative adjective (e.g. best, oldest)
    - `PNX` Reflexive pronoun (e.g. myself, yourself, itself, ourselves)
  - `http://www.natcorp.ox.ac.uk/docs/c5spec.html`

What we want to describe

- seems obvious at the first sight, taught at grammar school
- but disputes are both practical and theoretical (linguists)
- choices in lemmatization
  - take into account word derivation?
    - otcova $\Rightarrow$ otcův/otec, učený $\Rightarrow$ učený/učit, učení $\Rightarrow$ učení/učit
    - nejstaršího $\Rightarrow$ starý/nejstarší (searching: [věk] . . . člověk)
      - (+ „starší paní“ can be younger than „stará paní“)
    - nebral $\Rightarrow$ brát/nebrat (úplatky); nemalý $\Rightarrow$ malý/nemalý
    - bachelor thesis from VŠMIE: in online marketing singular and plural of nouns are treated as different key words
  - what about equivalents (*dublety*)?
    - mysli $\Rightarrow$ myslet/myslit
    - kapitalismem $\Rightarrow$ kapitalismus/kapitalizmus

What we want to describe

- selection of grammatical categories and their values
  - parts of speech: abbreviations, interpunction, numbers, contractions (*cos*, *oč*, *kdyby*)
  - categories: subdivision of pronouns, numerals, adverbs, case for prepositions, animateness *koho*/*čeho*
  - values: dual number (pes se 4 nohama), subdivision of pronouns
- a bigger problem is to set possible tags for a word form
  - e.g., which parts of speech can be attributed to *a*, *ani*, *ať*, *až*, ...
- the most problematic is to set rules for analysing a word form in a particular sentence context
  - if a word form can have tags A or B, it should be clear which one to select in a particular context (interannotators aggreement)
  - it's hard to learn computer to decide between A and B if even the native speakers do not agree

Morphological analyser ajka

- „original“ solution (Osolsobě 1996, Sedláček 1999+2005)
- organization of data
  - (which forms belong to the same lemma is known a priori)
  - word forms ⇒ “stem” (longest common left substring) + “endings”
  - lemmata with the same ending set belong to the same paradigm
    - *kluk* is like *vlk*, but not like *pes* or *slon*

nom. sg.	vl-k	p-es	slon-0
gen. sg.	vl-ka	p-sa	slon-a
dat. sg.	vl-ku	p-su	slon-u
dat. sg.	vl-kovi	p-sovi	slon-ovi
	...		
nom. pl.	vl-ci	p-si	slon-i
	...		

  - technical solution: “intersegment” between the stem and the ending
    - vl-k-0, p-es-0, slon-0-0; ... vl-c-i, p-s-i, slon-0-i; ...
    - smaller data, the principle is the same

Dictionary and paradigm files format

- dictionary file
  - format `lemma:paradigm`, `!` has negation, `%` reflexiva tantum, notes

```
hanbit:barvit!%|793.1,167.1
zelený:nový!|148.1
osel:orel|180.1
...
```

- paradigm file: paradigm definition
  - paradigm lemma + <intersegment> + list of ending sets

```
+barvit
        <i> NEWES717, NEWES744, konc44
        <en> NEWES710
        <il> NEWES705, NEWES778
        <ě> NEWES757
        <íc> NEWES759
        ...
```

Dictionary and paradigm files format

- paradigm file: ending set definition
  - set of ending + tag pairs
  - (the names are arbitrary, generated)

```
=NEWES717
        {t, k5aImF}
=NEWES705
        {y, k5aImAgFnP}
        {i, k5aImAgMnP}
        {a, k5aImAgFnS}
        ...
```

- interpretation
  - get the stem by deleting the first intersegment and the first ending from the end of the lemma, then add intersegments and endings
  - `hanbit` = `hanb` + `-i` + `-t`
  - ⇒ `hanb-i-t k5aImF`, ..., `hanb-il-i k5aImAgMnP`, ...

Principle of the analysis

- analysed word form $w_1 w_2 \ldots w_n = S + I + E$
- any part, stem S, intersegment I, or ending E, can be empty
  - e.g. `slon-0-0` or `0-člověk-0`, `0-lid-é`
- $\Rightarrow$ possible stems: ϵ, w_1, ..., $w_1 \ldots w_n$
- for each possible stem $S = w_1 \ldots w_i$ in the list of stems it tries to find candidates for $w_{i+1} \ldots w_n = I + E$ in the paradigm of the stem
- the result are the tags corresponding to the found triplets $S + I + E$
- in fact it's a bit more complicated, as the analysis works also with possible prefixes `nej` and `ne` and postfixes like `s` in `Byls tam`?

Disadvantages of the old data format

- the basic principles are the same in both Brno and Prague
  - dictionary of stems + set of paradigms, i.e., endings with tags
  - stems belong to paradigms; by joining the stem with its paradigm endings one obtains word forms with tags
  - both stems and endings are strings, which are only joined together
- the main disadvantage follow from that: redundancy
  - *Luděk/Luďka*, *Staněk/Staňka*, *vrah/vraha*, *medvídek/medvídka*, etc., are declined in a very similar way but need separate paradigms (or exceptions in Prague system)
- in a long term, redundancy leads to inconsistency
  - e.g.: adding of a colloquial gen. sg. *-a*: *muža* for masc. anim.
    - 217 paradigms $\Rightarrow$ needs to be automated: Gsg *-e* $\rightarrow$ *-a*
      - but ca 10 paradigms had *-ě* instead of expected *-e*
      - *strašpytel* and *neumětel* already had *-a*
    - it's hard or even impossible to check the results

New data format

- dictionary and paradigm files remains
  - the goal is to separate the regular and the irregular
  - dictionary: what is specific for particular lemmata
    - what a language user has to remember
  - paradigms + program: “language system”
    - endings and their regular behaviour, phonological rules
- stems are in the dictionary: `slon:pán`
- endings forms the paradigms:

```
pán k1gM
        nSc1    0
        nSc2    a
        nSc3    u, ovi
        ...
```

- the stems are joined with the endings: `slon-0`, `slon-a`, `...`
- the corresponding tags are concatenations of the paragigm part and the ending-specific part: `k1gMnSc1`, `k1gMnSc2`, `...`

New data format

- some simple rules transform the strings (`slon-0`) to word forms
  - obviously, we have to remove all `-` and `0`
  - `ňe` → `ně`: `tuleň-e` → `tuleňe` → `tuleně`
    - or `ň-e` → `ně`: `tuleň-e` → `tulen-ě` → `tuleně`
    - the first intermediate form corresponds to what is read
  - *Ábel* × *ďábel* ⇒ `Ábel` × `ďáb.el`: `.eC-0` → `eC-0`, `.eC-V` → `C-V`
    - (the phonological context is the same ⇒ it's dictionary information)
  - `vlk-i` → `vlc-i` (and also `pán-i` → `páň-i` → `páňi` → `páni`)
- the use of endings can be restricted according to end of the stem
  - e.g. `nPc6` `ech, ích/[ghk]|ch` (in a paradigm)
- even only these few improvements allows us to unify description of many (in the old format) distinct paradigms
  - `Luď.ek-0` → `Luďek-0` → `Luďek` → `Luděk`
  - `pejs.ek-ích` → `pejsk-ích` → `pejsc-ích` → `pejscích`

New data format

- some other enhancements
- paradigm inheritance:
  `soudce:muž`
  `nSc1 e`
  `nSc5 e`
  - by default, the endings for given tags are replaced
    - `+nSc5 e` would add the ending
  - restricted inheritance: `despota:pán_nP` + singular endings
- partial paradigms for some specific endings:
  `-ové k1gM`
  `nPc1 ové`
- using multiple paradigms:
  `filozof:pán,-ové`
  `dřevokaz:pán,+muž`

New data format — technical details in Czech :-)

- dále
  - hovorové tvary: Npl (a Vpl) *?učitelové*, ale **pokrytcé*
    - obecně: 1) ne/lze *-é*; 2) které z koncovek *-i* a *-ové* jsou spisovné
    - `filozof:pán,<-ové`; `občan:pán,<-é`; `akrobat:pán,<-i,+-é`
    - (bez < bych musel substandardní koncovky definovat ve vzorech `-é`)
  - více slovních základů, nepravidelné tvary (tedy slovník)

    ```
    přítel:muž,<-é
            <přátel:muž_nP,<-é
            <přátel-0 nPc2
    ```

    - `wH` tvary dokládá Google, jen spisovné tvary by byly bez <
    - pořadí ovlivňuje výsledek (dosud data neuspořádaná)
    - vyjadřuje, co je základní a co specifické (dosud tvary rovnocenné)
    - (Google: *přítelů* < *přátelů* < *přátel*, podobně i pro nepřítele)
    - `pejs.ek` je ve „struktuře“ vždy stejný, ale lze i

      ```
      pejsk:pán
              pejsek-0 / pejsek / pejsek:pán nSc1
      ```

    - ovšem zde nelze <, nemluvě o tom, že by to komplikovalo data

New data format — technical details in Czech :-)

- dále
  - příklad rozdílné interpretace téhož výsledku *g* ⇒ Npl jen *g-ové*
    - `nPc1 i/[^g], ové/` — tvary typu **mázi* systémově nemožné
    - `mág:filozof` — shodou okolností takové slovo aktuálně neexistuje
  - zachycení rozdílů mezi zápisem a výslovností
    `Smith[t:pán,-ové`
    `+Smith[s:muž,-ové`
- dosavadní umožňuje popis pomocí tradičních mluvnických vzorů, případně s upřesněními, bez nichž se ale neobejdou ani mluvnice
- ztotožňování shodných koncovek
  - falešný vzor `$shoda`

`c1`	`c5`
`k1gMnS\Kc3`	`c6`

  - `Marcel:pán,<-ové,muž_nSc5` ⇒ *Marceli* i *Marcelu*
  - `despot:žena_nS,-ovi,pán_nP gM`
  - `gigol:město_nS,+-ovi,pán_nP gM (ě/!gM)`
- (skládání značky, implicitní značka, implicitní vzor, ...)

New data format — from paradigms to features

- native speakers do not remember paradigms for all words but decline words according to some semantic, structural, or phonological features
  - for proper names `-ové` is preferred over `-i`
  - words derived with suffix `an` are `pán,<-é`
  - masculine animates that end with *d* have “hard” declension
- implicit rules: typical, regular behaviour controlled by
  - phonological features of the stem end or
  - semantic features described by a tag in the dictionary

```
$k1gM
        \Ko      město_nS,+-ovi,pán_nP,muž_nP/$M|i,-ové
        s/qJO    muž,<pán_nPc[67],+pán_nPc4
```

- then in the dictionary

```
gigolo   k1gM
Klaus    k1gMqJOP
```

New data format — example of results

- masculine animates: 19975 lemmata k1gM
- the most common types of word descriptions in the dictionary

# lemmat	% z celku	příklad
13871	69.17	`gaučo k1gM`
2207	11.01	`Ionesc[ko k1gMqJOP`
1654	8.25	`Severo+evrop=an`
683	3.41	`Mario k1gMqJO`
440	2.19	`kok.eš:-ové k1gM`
321	1.60	`sob.ěk:-i k1gM`
146	0.73	`uniat:-é k1gM`

- for >90% we need to know only (a part of) the tag
- redundancy is reduced
- the description is more linguistically acceptable

New morfological analyser majka

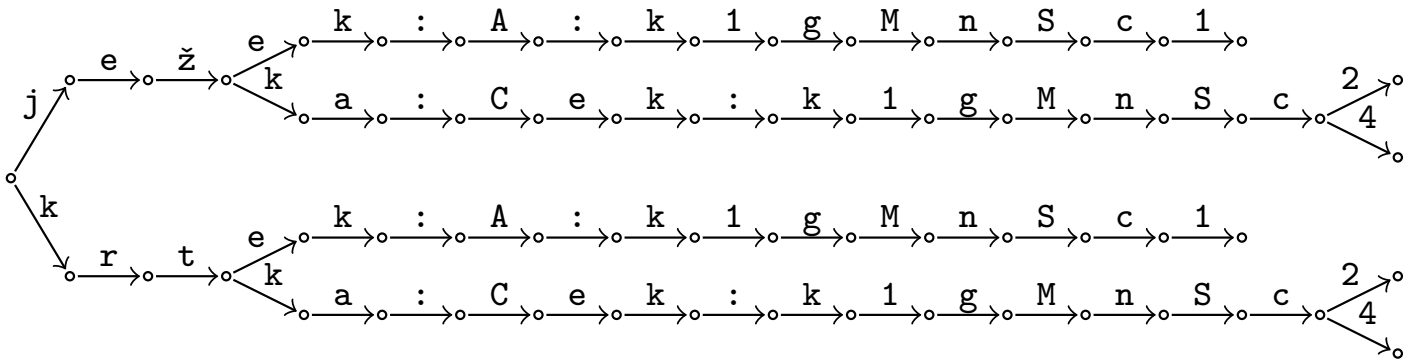
- ajka was quite fast, but too complex ⇒ unmaintainable
- we employed an approach from Jan Daciuk's dissertation thesis
  - the analysis is only searching the word form in WLT list
  - in fact, the data is a list query:response with the following format

```
ježek:A:k1gMnSc1      ← ježek:ježek:k1gMnSc1
ježka:Cek:k1gMnSc2    ← ježka:ježek:k1gMnSc2
ježka:Cek:k1gMnSc4    ← ježka:ježek:k1gMnSc4
krtek:A:k1gMnSc1      ← krtek:krtek:k1gMnSc1
krtka:Cek:k1gMnSc2    ← krtka:krtek:k1gMnSc2
krtka:Cek:k1gMnSc4    ← krtka:krtek:k1gMnSc4
```

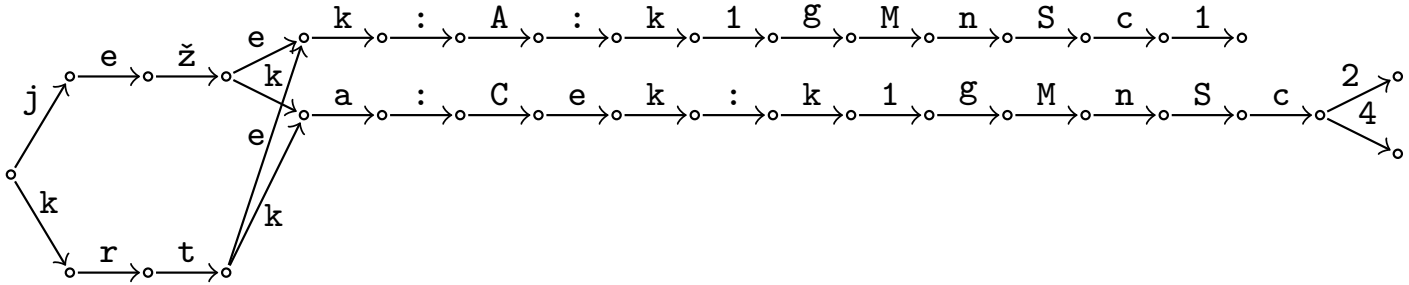
  - the list is a finite language ⇒ there is a DAFSA for it
    - encoding the lemma allows the neccessary minimalization
    - Daciuk offers incremental construction that preserves minimality
- (NB: this part is independent: WLT can be generated from the old format, and data for ajka can be generated from the new format)

New morfological analyser majka

- non-minimalized deterministic automaton for the example data



- minimalized deterministic automaton for the same data



- “analysis” is just fast and simple passing through the FSA
  - deterministic for the “query” + all the “responses”

New morfological analyser majka

- in a similar way data for lemmatization, generation, etc.:
- lemmatization: `krtek:A`, `krtka:Cek`
- generation: `krtek:A:k1gMnSc1`, `krtek:Cka:k1gMnSc2`
  - or from lemma and tag: `krtek:k1gMnSc2:Cka`
- "deep" structure: `krtek:C.ek-0`, `mužova:D=%ov-a`
  - or after application of some rules: `krtek:Cek-0`, `krtka:Ck-a`
- prefixes: `nemalý:CA:k2*, malý:Ane:A:k2*/malý:ACneA:k2*`

New morfological analyser majka

- statistical information about (some) dictionaries

dictionary	lines	source MB	dictionary MB	bytes/line
w	13,609,590	186	3.3	0.240
w → l	14,101,767	240	4.0	0.287
w → l+t	80,303,929	2,478	4.4	0.054
w → w	957,464,060	19,993	6.1	0.006

- comparison with morphological analyser ajka

	data size		time in seconds		
	ajka	majka	ajka	majka	ration
analysis	3.1	4.4	**18.22**	**2.88**	**6.3x**
lemmatization		4.0	16.76	1.57	10.7x
word forms		6.1	55.33	8.42	6.6x
diacritics		3.3	8698.80	1.61	5403x

- analysis is $\sim$4.6$\times$ faster than Prague analyser Morfo
- majka is used in, e.g., Seznam.cz or IS MU projects