Korpuszannotáció, annotációs szintek

A számítógépes nyelvészet alapjai – 2022/23 tavasz
6. óra

Simon Eszter

2023. április 17.

Tartalom

Korpuszannotáció

A nyelvi annotáció

a sztenderd szövegfeldolgozó lépések a modern korpuszoknál nagyjából ugyanazok:

- szegmentálás (tokenizálás, mondatra bontás)
- morfológiai elemzés
- morfoszintaktikai egyértelműsítés

Mi kell az annotációhoz?

- annotációs séma
  - elméleti nyelvészeti alapok lefektetése (pl. mi a tulajdonnév?)
  - címkekészlet
  - az annotáció formátuma (inline vagy standoff)
- annotációs eszköz
- az annotátorok száma $\rightarrow$ annotátorok közötti egyetértés mérése
- annotációs útmutató
- az annotáció minőségének ellenőrzése

Annotációs útmutató

- az útmutatónak egyszerre kell kellően kidolgozottnak és egyszerűnek lennie, hogy az annotátorok számára követhető legyen → ha nem így van, akkor az annotátorok magas hibaszázalékkal fognak dolgozni
- tartalmaznia kell az annotációs feladat leírását, az annotálandó nyelvi elemek felsorolását és példákat arra, hogy mit kell és mit nem kell annotálni
- minél magasabb nyelvi szintre megyünk, minél több szemantika van benne, annál képlékenyebb a feladat → bizonyos nyelvi jelenségek nehezen megfoghatók/formalizálhatók
- ha az útmutató nem elég egzakt, akkor az annotátorok elkezdik követni az intuíciójukat → a nem teljesen egyértelmű esetekben ez problémákat okozhat

NER annotációs útmutatók

- MUC-7 Named Entity Task Definition (Chinchor, 1997)
- ACE (Automatic Content Extraction) English Annotation Guidelines for Entities (Linguistic Data Consortium, 2008)
- Hunner project proposal és útmutató
- NYTK-NerKor útmutatók

Az annotáció formátuma

inline (XML)

```
<s><w>Ez</w> <w>egy</w> <w>mondat</w> <c> .</c>
<s><w>Meg</w> <w>a</w> <w>második</w> <c> .</c>
```

standoff

Ez
egy
mondat
.

Meg
a
második
.

XML

EXtensible Markup Language
egyfajta jelölőnyelv (markup language) → vannak más hasonlók:
YAML, JSON, MD

Előnyei:

- mind ember, mind gép számára olvasható formátum
- támogatja a Unicode-ot
- szabványos és platformfüggetlen
- képes a legtöbb általános számítástudományi adatstruktúra ábrázolására

Hátrányai:

- szintaxisa elég bőbeszédű és részben redundáns
- nagyobb tárolási költség
- nincs lehetőség a dokumentum egyes részeinek közvetlen elérésére
- átfedő adatstruktúrák modellezése nehéz/lehetetlen

Standoff annotáció

- az eredeti dokumentumok sima szöveg fájlok maradnak
- az annotációk nem szövegközi tagek, hanem egy külső fájlban jelöljük úgy, hogy megadjuk, hogy az eredeti szöveg melyik karaktertartományára vonatkozik a címkézés, és hogy milyen címkét kap a szövegrészlet
- az annotálást teljesen különválasztjuk a használt feldolgozó eszközöktől, és közben minden információt megtartunk
- az átfedő és beágyazott annotáció is könnyen kezelhető

Beágyazott és átfedő annotáció

Beágyazott annotáció

<LOC><PERSON>Kossuth Lajos</PERSON>utca</LOC>

Átfedő annotáció

a Kossuth Lajos és a Petőfi Sándor utca sarkán

BIE1/BIOES/BILOU

[...] közölte Wolf László, az OTP Bank vezérigazgató-helyettese az MTI érdeklődésére.

közölte	O	O	O
Wolf	B-PER	B-PER	B-PER
László	E-PER	E-PER	L-PER
,	O	O	O
az	O	O	O
OTP	B-ORG	B-ORG	B-ORG
Bank	E-ORG	E-ORG	L-ORG
vezérigazgató-helyettese	O	O	O
az	O	O	O
MTI	1-ORG	S-ORG	U-ORG
érdeklődésére	O	O	O
.	O	O	O

BIE1/BIOES/BILOU

A szállásunk egy Balaton melletti kis üdülőfaluban, Zamárdiban volt.

A	B-NP	B-NP	B-NP
szállásunk	E-NP	E-NP	L-NP
egy	B-NP	B-NP	B-NP
Balaton	I-NP	I-NP	I-NP
melletti	I-NP	I-NP	I-NP
kis	I-NP	I-NP	I-NP
üdülőfaluban	I-NP	I-NP	I-NP
,	I-NP	I-NP	I-NP
Zamárdiban	E-NP	E-NP	L-NP
volt	O	O	O
.	O	O	O

A kézi annotáció minősége

Az annotáció minősége

- a kézzel annotált korpuszokat tanító- vagy kiértékelőanyagként használják felügyelt gépi tanulással működő eszközök számára
- felügyelt gépi tanulással működő rendszerek sikeressége a tanítóanyag minőségén múlik
- csak olyan feladatokat lehet felügyelt gépi tanulással megoldani, amelyeket az ember is képes elvégezni
- csak olyan nyelvi jelenségekhez tudunk kézi annotációt készíteni, amelyeket eléggé megértettünk ahhoz, hogy pontosan le tudjuk írni őket
- megbízható az annotáció, ha a jelenségek leírását több annotátor is hasonlóképpen megértette és ez alapján hasonlóképpen kódolják az egyes jelenségeket
- a feladatleírásnak tehát érthetőnek kell lennie az annotátorok számára, akik ideális esetben egyetértenek az egyes jelenségek címkézésében

Az annotátorok közötti egyetértés

- a cél a minél magasabb annotátorok közötti egyetértés
- minél egyszerűbben leírható nyelvi jelenség annotálásáról van szó, annál könnyebb magas annotátorok közötti egyetértést elérni, a nyelvi jelenség összetettségével az egyetértés mértéke is könnyen csökken
- mitől lehet alacsony?
  - a feladat megfogalmazása nem egyértelmű vagy nem teljes
  - az annotátoroknak túl sok kategóriát kell kezelniük
  - átláthatatlan felületen kell dolgozniuk

Az annotátorok közötti egyetértés

- az annotátorok (vagy kódolók), amikor kategóriákat rendelnek egyes elemekhez, szubjektív döntéseket hoznak
- ha az annotátorok egyetértenek az egyes elemekhez rendelt kategóriákban, akkor az adat megbízható, és ha a kódolók következetesen hasonló eredményt produkálnak, akkor hasonlóképpen értették meg a feladatot és az annotálási útmutatót, ezért a továbbiakban is hasonló eredményeket várhatunk tőlük
- *megfigyelt egyetértés*: azt mutatja meg, hogy az esetek hány százalékában értett egyet a két kódoló
- DE! nem elég, ha két kódoló egyetért, hiszen mindketten tévedhetnek is
- a címkék számának csökkentésével növekszik a megfigyelt egyetértés, ráadásul nem érzékeny az egyes címkék eltérő gyakoriságára
- megoldás: valószínűség-korrigált együtthatók, amelyek számolnak a véletlen eseményekkel is

Az annotátorok közötti egyetértés: koefficiensek

Különböző mérőszámok az egyetértésre:

- megfigyelt egyetértés
- S (Bennett, Alpert és Goldstein 1954): minden kategória ugyanolyan valószínű, a kategóriák között egyenletes eloszlást feltételez
- π (Scott, 1955): kategóriánként eltérő, de kódolók között megegyező eloszlás
- κ (Cohen, 1960): kategóriánként és kódolónként eltérő eloszlás, ez már kezeli az elfogultságot
- α (Krippendorff, 1980): nem csak az egyetértést vizsgálja, hanem az egyet nem értés különböző fokozatait

Az egyetértés erőssége

Landis and Koch (1977)

κ	strength of agreement
<0.00	poor
0.00 – 0.20	slight
0.21 – 0.40	fair
0.41 – 0.60	moderate
0.61 – 0.80	substantial
0.81 – 1.00	almost perfect

Példák

Tulajdonnév-felismerés

hunNERwiki korpusz (Simon and Nemeskey, 2012):

- $\kappa = 0,967$
- F-mérték: 92,94%

Szeged NER korpusz (Szarvas et al., 2006):

- egyetértési arány: 99,6%

Metaforikus kifejezések felismerése (Babarczy et al., 2010)

egyetértési arány:

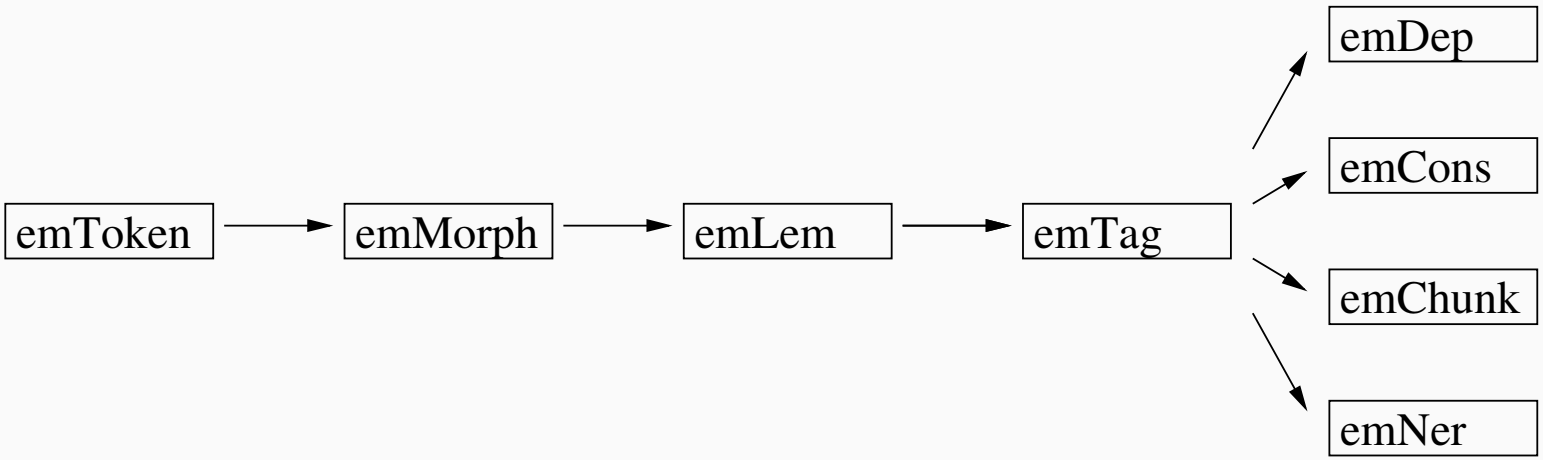
- 1. körben: 17%
- 2. körben: 48%

Szövegfeldolgozási és annotációs szintek

Alapszintű szövegfeldolgozási szintek

- mondatrabontás és tokenizálás
- morfológiai elemzés
- sekély szintaktikai elemzés
- mély szintaktikai elemzés
- tulajdonnév-felismerés
- ...

Egymásra épülő annotációk



Mondatszegmentálás

Mittelholcz (2017)

- Minden mondat.
- Mondathatárok azonosítása.
- Pontos problémák:
  - Rövidítések (*du. 5-kor*).
  - Római számok (*V. László*).
  - Sorszámok (*10. éve, hogy …*).
- Egyéb nehézségek:
  - Idézeten belüli mondatok.
  - Zárójelen belüli mondatok.

Tokenizálás

- Detokenizálhatóság és elválasztás (és az *-e* partikula).
- Szóalkotó karakterek, szónemalkotó karakterek, és amik köztük vannak:
  - Zárójelek, idézőjelek, aposztrófok kezelése.
  - Rövidítések végén lévő pont vs. mondatvégi pont.
- Számok (space-szel tagolt számok, mértékegységek, képletek, dátumok).
- Informatikai kifejezések (URL, elérési út, emailcím).
- Smiley-k és emoji-k.

Morfológiai elemzés

tokenszintű elemzés → nem lát se előre, se hátra → no kontextus → többértelműség

kerekesszék

kerek/ADJ+esszé/NOUN<PLUR>

kerekes/ADJ+szék/NOUN

kerék/NOUN[ATTRIB]/ADJ+szék/NOUN

kerek/ADJ[ATTRIB]/ADJ+szék/NOUN

kerék/NOUN[ATTRIB]/ADJ+szék/NOUN

kerek/ADJ[ATTRIB]/ADJ+szék/NOUN

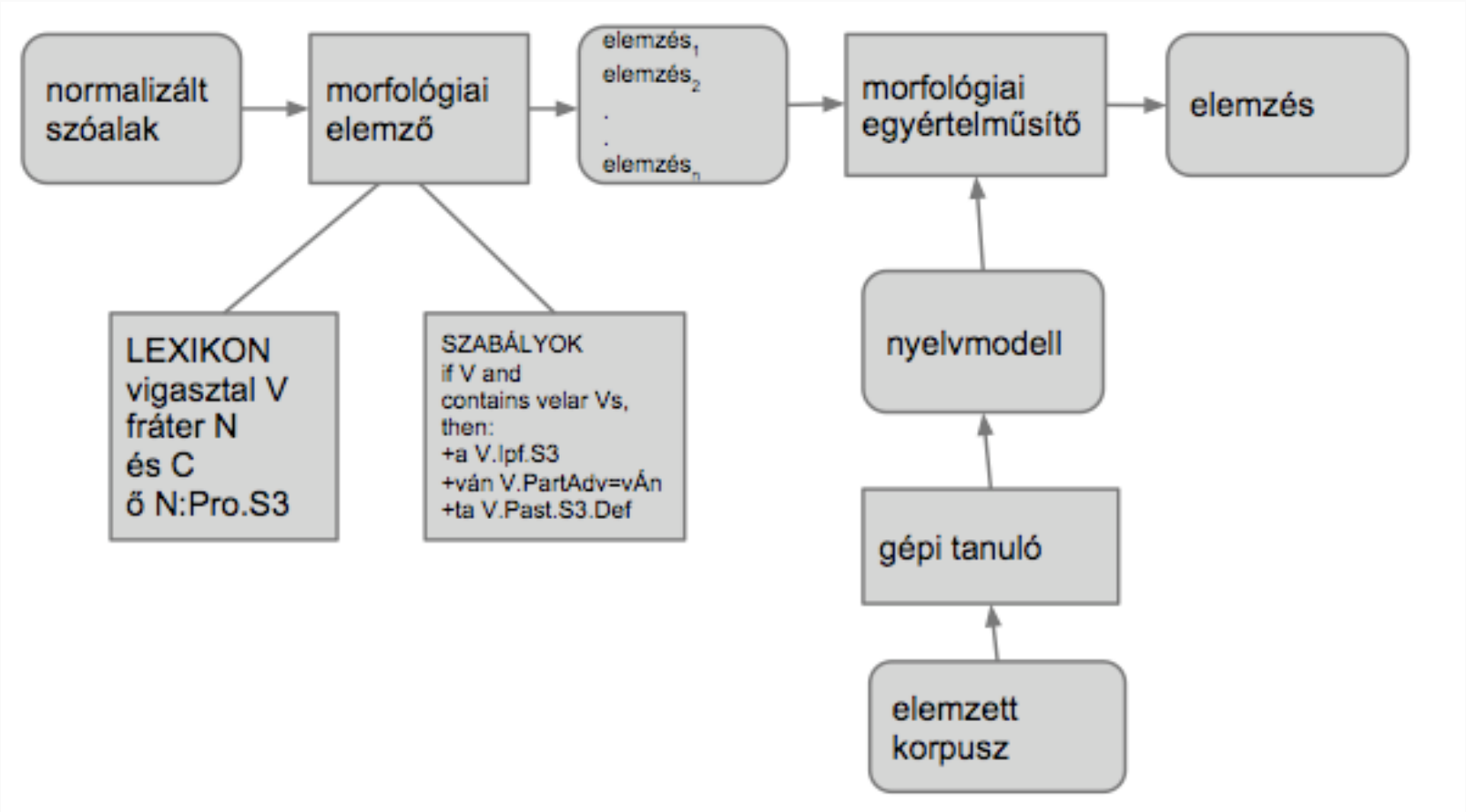
falucska

```
fa [/N] + luc[/N] + ska[/N] + [Nom]
fa[/N] + lucsok[/N]=lucsk + a[Poss.3Sg] + [Nom]
falu[/N] + cska[_Dim:cskA/N] + [Nom]
falucsok[/N]=falucsk + a[Poss.3Sg] + [Nom]
falucska[/N] + [Nom]
```

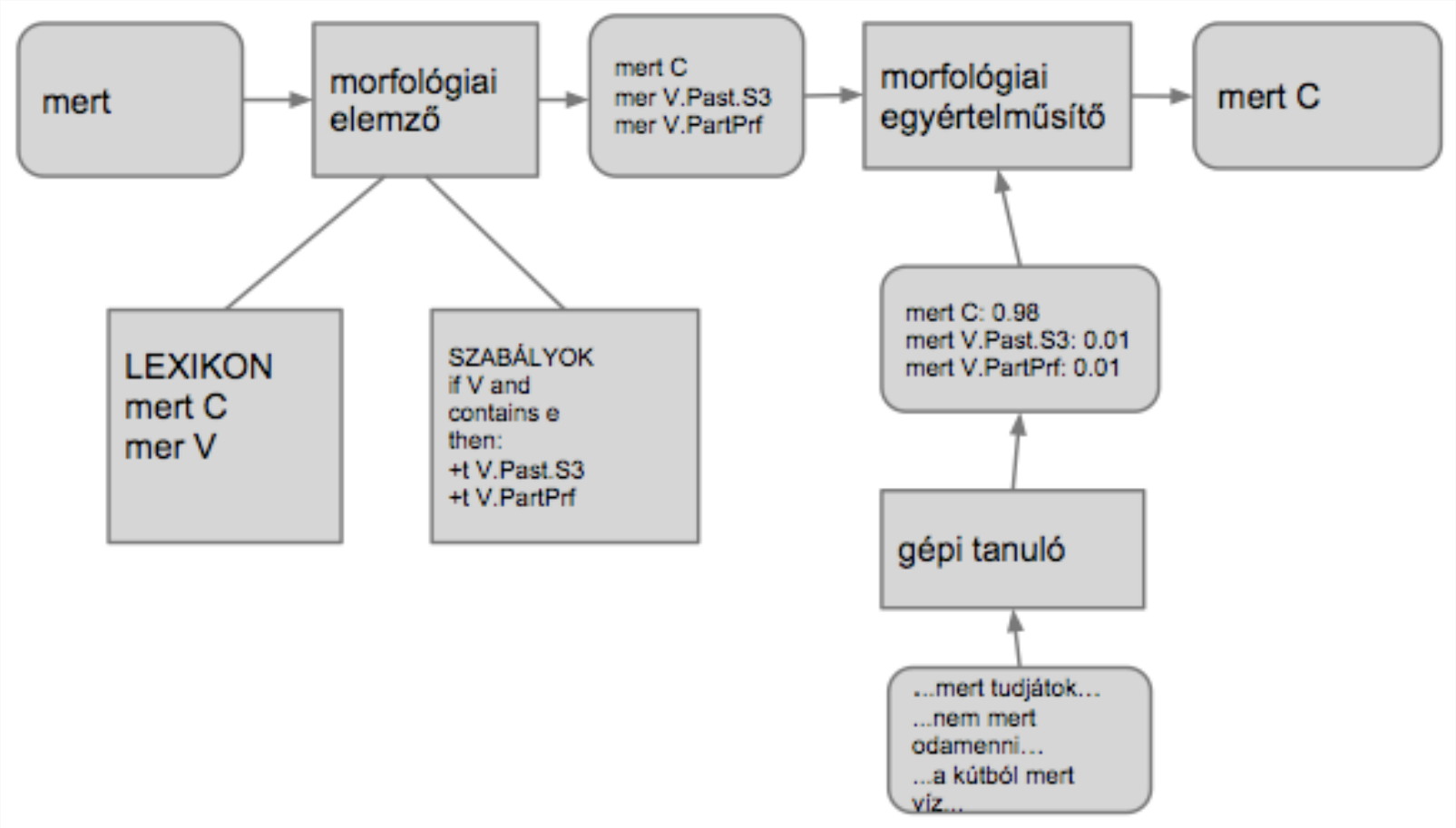
Mit tartalmazhat a kimenet?

- morfoszintaktikai információk
- jelentésre vonatkozó információk
- hangalakra vonatkozó információk (allomorfia)
- szófajkód
- lemma
- morfológiai szegmentumok

Morfológiai egyértelműsítés 1.



Morfológiai egyértelműsítés 2.



Nézzük meg az e-magyart!

Szekvenciális címkézési feladatok – Tulajdonnév-felismerés

Named Entity Recognition (NER)

2 lépésből áll:

1. a nevek lokalizálása strukturálatlan szövegben
2. a megtalált elemek besorolása előre definiált névosztályokba
   - *Person*, *Location*, *Organization*, *Date*, *Time*, *Money*, *Percent*, *Measure* (MUC)
   - *Person*, *Location*, *Organization*, *Miscellaneous* (CoNLL)

Kérdések, nehézségek

- a tulajdonnevek definiálása problémás
- egymásba ágyazott nevek és kompozicionalitás
- van-e a tulajdonnévnek jelentése?
- a tulajdonnevek a szintaxis szempontjából oszthatatlan nyelvi egységek
- nem lehet belülről módosítani őket
- a ragok mindig az NP-t alkotó tulajdonnév végére kerülnek
- a tulajdonnevek alaki sérthetetlenségének elve
- metonimikusan viselkedő tulajdonnevek
- eltérő annotációs sémák $\rightarrow$ még a statisztikai alapú rendszereket is nehéz átvinni egyik korpuszról a másikra, vagy egyik műfajról a másikra

Szekvenciális címkézési feladatok – Sekély szintaktikai elemzés

chunking

[Immár] [negyedik éve] [a Manchester United]
[a világ leggazdagabb csapata] [bevétel szerint].

1. minden frázis megtalálása egy mondatban
2. maximális NP-k megtalálása
3. alap NP-k megtalálása

Összetevős és függőségi elemzés

Összetevős elemzés

A mondatok összetevős szerkezeti elemzése azt tárja fel, hogy a szavak egymással kombinálódva milyen kifejezéseket alkotnak, illetve hogyan állnak össze egy mondattá.

Függőségi elemzés

A függőségi elemzés a mondatok szerkezeti egységei közötti függőségi viszonyokat (pl. alany, tárgy, jelző) tárja fel.

Hogyan működik az elemző?

Morfológiai elemzés – A szókincs modellezése

Morfológiai elemzés

1. a szóalakot elemi morfémáira bontja
2. meghatározza a morfémák lexikális alakját
3. meghatározza a morfémák morfoszintaktikai tulajdonságait (esetleg egyéb nyelvtani tulajdonságokat)

pl. *többértelműségeket*

- `sok[/Num]=tö+bb[_Comp/Num]=bb+értelem[/N]=értelm+ű[_Adjz:Ú/Adj]=ű+ség[_Nz_Abstr/N]=ség+ek[Pl]=ek+et[Acc]=et`
- `többértelmű[/Adj]=többértelmű+ség[_Nz_Abstr/N]=ség+ek[Pl]=ek+et[Acc]=et`
- `többértelműség[/N]=többértelműség+ek[Pl]=ek+et[Acc]=et`

A szókincs modellezése

Az összes lehetséges szóalak felsorolása helyett:

- a morfémák szerepelnek a szótárban
- szabályokkal írjuk le a szóalakok felépítését

$\rightarrow$ formális nyelvtan

Nehézség:

- túlgenerálás a produktív szabályok által
- a jelentésfüggő morfológiai jelenségek kezelése

Szótárak és szabályok

morfoszintaktikai szabályok: az egyes morfémák hogyan (milyen sorrendben és milyen feltételek mellett) következhetnek egymás után egy szóalakban

- a morfémák szótárban (morfématárban, lexikonban) vannak
- metainformációkkal, különféle osztályokba rendezve
- a tőmorfémák külön lexikonban, szófajkódokkal
- affixumok is külön (a szóalakban pre- vagy szuffixumok)
- külön lexikonba mehetnek az előtagok, a nem feltétlenül szóvégi szuffixumok (képzők)

Különböző szabálymegadási modellek

- kétszintű morfológiák
- folytatási osztályok
  - minden morféma mellett jelöli, hogy milyen morféma követheti (pl. minden tőhöz, hogy milyen toldalék)
  - *labda [főnév] (+ t [tárgyrag], + val [eszközhatározó rag], + nak [birtokosrag], …); + k [többesszám jele] (+ at [tárgyrag], + nak [birtokosrag], + val [eszközhatározó rag], + ból [helyhatározó rag] stb.)*
  - a morfémák osztályai az egyes folytatási osztályok
- unifikációs modellek
  - minden morféma (mindkét oldalán) összetett adatstruktúra a morféma morfoszintaktikai és morfofonológiai tulajdonságaival
  - az illeszkedési pontoknál meg kell vizsgálni (unifikáció), hogy a jegyszerkezetek (feature structure) passzolnak-e

Szótár a számítógépen

- a szótárnak elfogadható méretűnek kell lennie
- a benne való keresésnek elég gyorsnak kell lennie

$\rightarrow$ nem elég egy lineárisan kereshető lista

megoldás: állapotátmenetes modell (állapotgép, automata), ahol a természetes nyelv teljes szókincsét egy véges halmazzal közelítjük

panmorph: morfológiai címkekészletek

panmorph

- összegyűjtöttük és közzétettük a magyarra alkalmazott morfológiai annotációs sémákkal és címkekészletekkel kapcsolatos elérhető információkat
- konvertereket írtunk a címkekészletek között
- https://github.com/nytud/panmorph

Morfológiai címkekészlet:

- **informativitás**: pontosság és teljesség
- **adekvátság**: nyelvészetileg megalapozott kategóriák
- **egyszerűség**: kézi és automatikus feldolgozhatóság

Kimeneti formalizmusok 1.

MSD (Erjavec, 2004)

- pozícióalapú
- az első pozíció mindig a szófaji kategóriáé, a többi pedig további morfoszintaktikai infókat kódol
- *`Vmis2s---y`*: kijelentő módú, múlt idejű, egyes szám második személyű, tárgyas ragozású főige
- lemma külön
- nincs szegmentálás, nincs deriváció, nincsenek jelölve az allomorfok, csak morfoszintaktikai kódok vannak
- nem hierarchikus, és nem tükrözi a morfológiai jelöltséget
- sok nyelvre
- Szeged Korpusz és Treebank
- magyarlanc 2.0 elemzőlánc kimenete

Kimeneti formalizmusok 2.

Universal Dependencies and Morphology

- univerzális szófajkódok fix halmaza és nyelvspecifikus elemekkel bővíthető feature–érték párok halmaza
- meg van adva, hogy milyen feature milyen értékeket vehet fel
- hierarchikus jegy–érték struktúra (Attribute–Value Structure, AVS) (Trón, 2002)
- ez sem tükrözi a morfológiai jelöltséget
- lemma külön
- nincs szegmentálás, nincs deriváció, nincsenek jelölve az allomorfok, csak morfoszintaktikai kódok vannak
- *hozzád*:
  `Case=All|Number=Sing|Person=2|PronType=Prs`
- Szeged Treebank egy részkorpusza
- magyarlanc 3.0 elemzőlánc kimenete, *`emmorph2ud`*, HuSpacy, UDPipe, Stanza

Kimeneti formalizmusok 3.

KR (Rebrus et al., 2012)

- hierarchikus: irányított körmentes gráf (fa)
- a gyökércsomópont a szófaj
- bináris morfoszintaktikai jegyek és ezek pozitív és negatív értékei
- lemma külön
- nincs szegmentálás, nincs deriváció, nincsenek jelölve az allomorfok, csak morfoszintaktikai kódok vannak
- *fotelben*: `fotel/NOUN<CAS<INE>>`,
  fotelban: `fotel/NOUN<CAS<INE>>`
- `hun*` eszközlánc

Kimeneti formalizmusok 4.

emMorph (Novák et al., 2017)

- van szegmentálás, jelölve vannak a derivációk, az allomorfok, van lemma, van morfoszintaktikai annotáció
- mint a glosszázás:

harmad napon halottaiból feltámada

```
három[/Num]=harm + ad[_Frac/Num] + [Nom]
nap[/N] + on[Supe]
halott[/N] + ai[Pl.Poss.3Sg] + ból[Ela]
fel[/Prev] + támad[/V] + a[Pst.NDef.3Sg]
```

harmal	napon	halottay bool	felthamata
harmad	nap-on	halott-a-i-ból	fel-támad-a
third	day-SUP	dead-POSS-PL-ELA	up-rise-PST.3SG

‘on the third day he is risen from the dead’ (Müncheni emlék 114v)

panmorph konverterek

Közvetlen leképezés az egyik címkekészletről a másikra:

- *emmorph2msd*
- *emmorph2conll*
- *emmorph2ud*

Miért az emMorph címkét konvertáljuk?

- az emMorph címkekészlet a legfinomabb, legrészletesebb
- az egyik bevett elemzőlánc, az e-magyar bocsátja ki, ezért jól beilleszthetők a konverterek a szövegfeldolgozási folyamatba

Konverzió címkekészletek között

- szerencsés, ha egy-az-egyhez megfelelés áll fenn a bemenet és a kimenet között
- sok esetben kellett aleseteket és kivételeket kezelni
- néha a lemmára vagy a token felszíni alakjára is támaszkodni kellett
- zárt szóosztályok (pl. kötőszavak, névmások) esetén felsorolhatóak az alesetek
- igekötők, tulajdonnevek kezelése eltér
- bizonyos címkék soha nem jelennek meg a kimenetben, noha a kimeneti készletek tartalmaznak címkéket a jelenségekre: -nAk ragos névszók, segédigék

Elemzőláncok magyarra

e-magyar/emtsv i

- moduláris, bármelyik lépésnél ki-be lehet szállni, új modulokat könnyen lehet integrálni
- lassabb: dependenciáig REST API-val: 310 token/s
- python, docker
- modulok:
  - tokenizálás, mondatrabontás
  - morfológiai elemzés és egyértelműsítés
  - szintaktikai elemzés: függőségi és összetevős
  - tulajdonnév-felismerés
  - főnévi frázisok felismerése
  - szótáralapú kifejezésfelismerő
  - zérónévmás-beszúró
  - konverterek
  - kiértékelő

e-magyar/emtsv ii

- nyílt forráskódú, szabadon felhasználható
- SOTA teljesítmény

https://github.com/dlt-rilmta/emtsv

https://e-magyar.hu/hu/

Stanza

- a Stanford NLP Group Python-alapú szövegfeldolgozó eszköze
- a teljes pipeline neurális eszközökön alapul
- elemzési szintek:
  - tokenizálás, mondatrabontás
  - morfológiai elemzés és egyértelműsítés: csak UD POS tageket és morfológiai feature-öket ad ki, de nem lemmatizál
  - szintaktikai elemzés: függőségi
  - tulajdonnév-felismerés
- nyílt forráskódú, szabadon használható
- kevésbé jó teljesítmény

https://stanfordnlp.github.io/stanza/

UDPipe

- elemzési szintek:
  - tokenizálás, mondatrabontás
  - morfológiai elemzés és egyértelműsítés
  - szintaktikai elemzés: függőségi elemzés
- kevésbé jó teljesítmény
- gyors: dependenciáig: 3.300 token/s
- az egyes lépéseknél ki-be lehet szállni az egységes formátumnak köszönhetően, de új modulokat nem lehet integrálni
- nyílt forráskódú, szabadon használható
- neurális módszereken alapul minden modulja

http://ufal.mff.cuni.cz/udpipe

HuSpacy

- elemzési szintek:
  - tokenizálás, mondatrabontás
  - morfológiai elemzés és egyértelműsítés
  - szintaktikai elemzés: függőségi
  - tulajdonnév-felismerés
  - szóvektorok
- a leggyorsabb: dependenciáig: 15.000 token/s
- nyílt forráskódú, szabadon használható
- python
- elég jó teljesítmény
- bármelyik lépésnél ki-be lehet szállni, új modulokat könnyen lehet integrálni

https://github.com/huspacy/huspacy

https://huggingface.co/spaces/huspacy/demo

Magyarlánc

- a Szegedi Tudományegyetemen fejlesztett eszközlánc
- nem moduláris, csak egyben lehet futtatni az elejétől a végéig, új modulokat nem lehet integrálni
- gyorsabb: dependenciáig: 450
- java
- elemzési szintek:
  - tokenizálás, mondatrabontás
  - morfológiai elemzés és egyértelműsítés
  - szintaktikai elemzés: függőségi és összetevős
- letölthető, szabadon felhasználható
- SOTA teljesítmény

https://rgai.inf.u-szeged.hu/magyarlanc

Házi feladat

Házi feladat (dupla)

- ketten-hárman összefogni,
- specifikálni egy annotálási alfeladatot,
- annotációs sémát és útmutatót gyártani hozzá,
- kiválasztani egy szöveget, tokenizálni és mondatra bontani egy szabadon választott elemzőlánccal,
- ketten-hárman leannotálni, gold standardban megállapodni (NE dobjatok ki belőle mondatokat!),
- annotátorok közötti egyetértést számolni (akár több körben is),
- a szöveget leelemeztetni a választott elemzőlánc megfelelő moduljával,
- a fent létrehozott gold standard annotációval összevetve kiértékelni az elemző teljesítményét az adott feladaton

Az annotálási alfeladat

- legyen olyan, amit a választott eszköz lefed,
- ne legyen túl bonyolult,
- a gold standard szöveg tartalmazzon legalább 100 adatpontot

Választható alfeladatok:

1. tulajdonnevek felismerése (PER, LOC, ORG, MISC)
2. maximális NP-k felismerése
3. tárgyesetű főnevet vonzó igék felismerése
4. ...

Beküldés

Mit kell elküldeni a végén?

- a bemenő szöveg
- annotációs séma és útmutató
- az annotátorok annotációit tartalmazó tsv fájl (token TAB annotáció1 TAB annotáció2 TAB gold)
- az annotátorok közötti egyetértés számítása ((leírás vagy szkript) és eredmények)
- az elemző kimenete (tsv)
- kiértékelés ((leírás vagy szkript) és eredmények)

Határidő

- ápr. 24.: a terv beküldése
- máj. 15.: a kész anyag beküldése

Irodalom

Hivatkozások

Babarczy, A., Bencze, I., Fekete, I., and Simon, E. (2010). The Automatic Identification of Conceptual Metaphors in Hungarian Texts: A Corpus-based Analysis. In Bel, N., Daille, B., and Vasiljevs, A., editors, *Proceedings of the LREC 2010 Workshop on Methods for the automatic acquisition of Language Resources and their evaluation methods*, pages 31–36, Malta.

Erjavec, T. (2004). *MULTEXT-East Morphosyntactic Specifications. Version 3.0*. http://nl.ijs.si/ME/Vault/V3/msd/html/.

Landis, J. R. and Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174.

Irodalom ii

Mittelholcz, I. (2017). *emToken*: Unicode-képes tokenizáló magyar nyelvre. In *XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017)*, pages 61–69, Szeged.

Novák, A., Rebrus, P., and Ludányi, Zs. (2017). Az *emMorph* morfológiai elemző annotációs formalizmusa. In *XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017)*, pages 70–78, Szeged.

Rebrus, P., Kornai, A., and Varga, D. (2012). Egy általános célú morfológiai annotáció. *Általános Nyelvészeti Tanulmányok*, XXIV.:47–80.

Simon, E. and Nemeskey, D. M. (2012). Automatically generated NE tagged corpora for English and Hungarian. In *Proceedings of the 4th Named Entity Workshop (NEWS) 2012*, pages 38–46, Jeju, Korea. Association for Computational Linguistics.

Irodalom iii

Szarvas, Gy., Farkas, R., Felföldi, L., Kocsor, A., and Csirik, J. (2006). A highly accurate Named Entity corpus for Hungarian. In *Electronic Proceedings of the 5th International Conference on Language Resources and Evaluation*.

Trón, V. (2002). Attribútum–érték struktúrák. In Kálmán, L., Trón, V., and Varasdi, K., editors, *Lexikalista elméletek a nyelvészetben*, volume XIII. of *Segédkönyvek a nyelvészet tanulmányozásához*, pages 333–344. Tinta Könyvkiadó, Budapest.