Számítógépes szintaxis

Dömötör Andrea

Digitális Örökség Nemzeti Laboratórium
ELTE BTK TI Digitális Bölcsészet Tanszék

2022. március 28.

National Laboratory for Digital Heritage

Az előadás témakörei

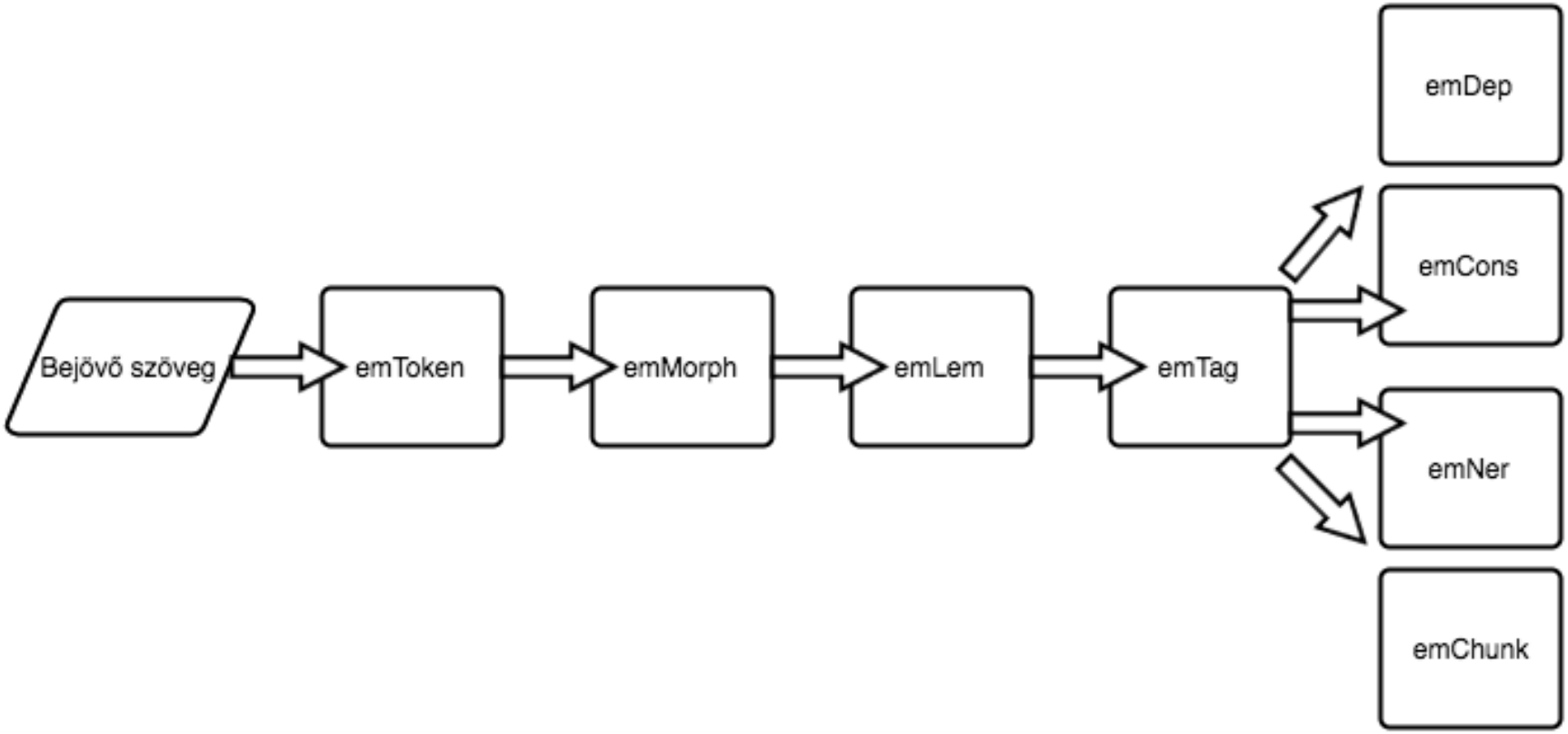
- A (számítógépes) nyelvi elemzés szintjei
- A mondatszerkezet megközelítései a számítógépes nyelvészetben
- Treebankek
- Elemző eszközök

A nyelvi elemzés lépései (szintjei)

- Szegmentálás, tokenizálás
- Szótövesítés (lemmatizálás)
- Morfológiai elemzés
- Szófaji egyértelműsítés
- Szintaktikai elemzés

Az egyes szintek **elemzési lánc**ot alkotnak. Az alacsonyabb szintek kimenete adja a magasabb szintek bemenetét (tehát pl. a szintaktikai elemzésnek feltétele a morfológiai elemzés és szófaji egyértelműsítés).

e-magyar pipeline



Szegmentálás, tokenizálás

A **szegmentálás** során a szöveget bekezdés vagy/és mondat egységekre bontjuk.

A **tokenizálás** az elemzési egységekre bontást jelenti.

- Token: “nyelvtechnológiai értelemben vett” szó
  - Az írásjelek külön tokenek
- Miért fontos ez?
  - A mondatelemzéshez nyilván meg kell határozni a mondatot…
  - Az *újság* és az *újság?* nem két különböző szó
    - Nem akarunk *újság?*-szerű szavakat

Szótövesítés (lemmatizálás)

A **szótövesítés** során a szótőt és a szófajt állapítjuk meg.

Miért fontos ez?

- Intelligens keresés
- A magyarban rengeteg szóalak tartozik egy szóhoz
  - $\rightarrow$ kezelhetetlen szótárméret
- Szófajspecifikus jegyek taníthatósága

Morfológiai elemzés

A **morfológiai elemző** a szavakat szóelemekre bontja, megadja a szóalakok lehetséges elemzéseit, a szó morfológiai jegyeit.

Fontos, hogy ezen a szinten **lehetséges elemzések**et kapunk (az összeset, sőt...), és még nem tudjuk, melyik a jó.

Miért fontos ez?

- Elemzés nélkül túl sok szóalakunk lenne
- Az elemzett korpuszon tanított mesterséges intelligencia képes lehet:
  - ismeretlen szavak szóelemeit is felismerni
  - új szóalakokat generálni
- A szintaktikai elemzéshez szükség van a morfológiai jegyekre

Szófaji egyértelműsítés

A morfológiai elemző minden szóalakhoz megadja az összes lehetséges elemzést. A **szófaji egyértelműsítő** feladata, hogy ezek közül kiválassza a helyeset (a kontextus alapján legvalószínűbbet).

Miért fontos ez?

- Homonímiák feloldása
  - pl. *meg*: kötőszó vagy igekötő?
- Nyelv gazdaságossága $\rightarrow$ sok azonos alakú szó
  - Az emberi nyelvfeldolgozás is a kontextus alapján különbözteti meg őket.

Szintaktikai elemzés

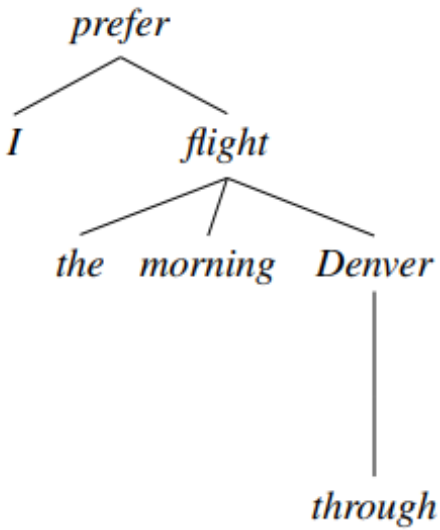
A **szintaktikai elemző** a mondatszerkezetet határozza meg.

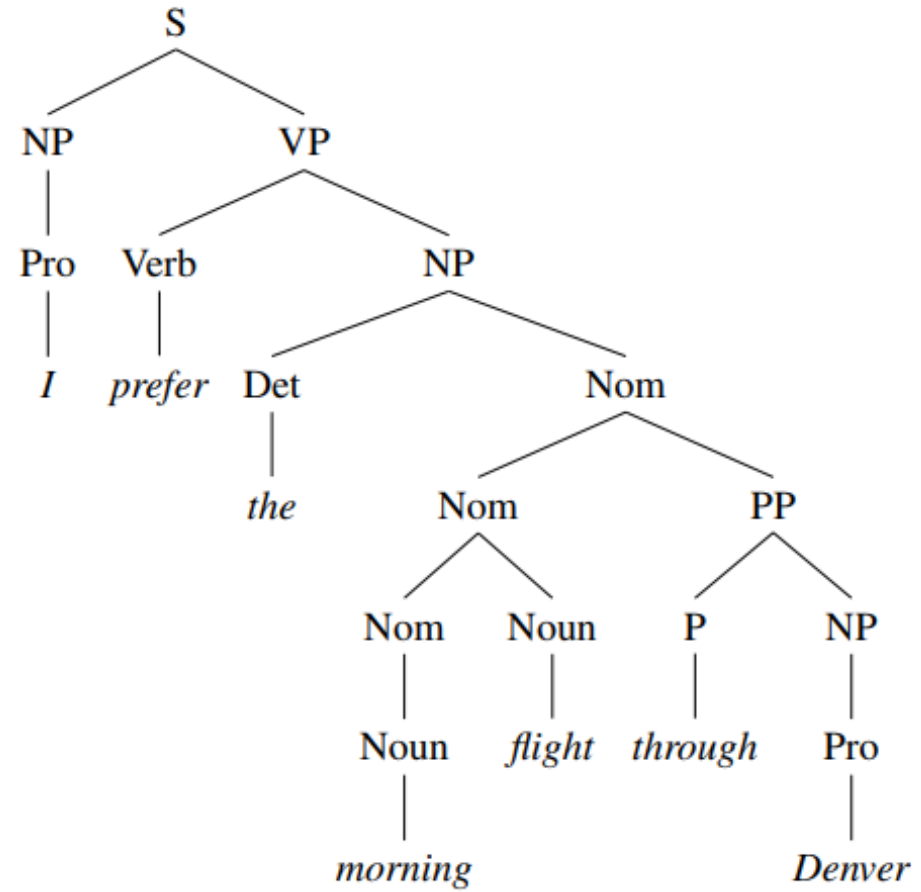
Ennek két fő megközelítése van:

- **Közvetlen összetevős elemzés**
  - A szavak frázisokat alkotnak (NP, VP, stb.)
  - A frázisokból épül fel a mondat
  - A szintaktikai szabályok a mondat és a frázisok lehetséges szerkezeteit írják le
- **Függőségi elemzés**
  - A szavak közötti függőségeket és a relációk típusát adja meg
  - Minden szónak van egy feje
  - A szó és a feje közti viszony írja le a szó szintaktikai szerepét (pl. subj, obj...)
  - A teljes elemzés mindig egy gyökércsomópontból indul ki, és fa struktúrában ábrázolható

Közvetlen összetevős elemzés

I prefer the morning flight to Denver.





Lexikon és szabályok

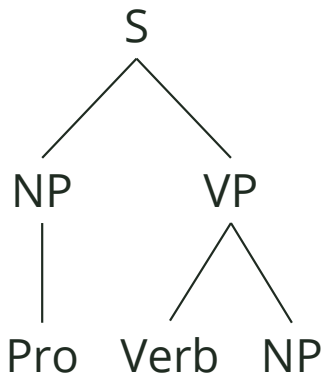
Lexikon	
Pro	*I, Denver*
Verb	*prefer*
Det	*the*
Noun	*morning, flight*
P	*through*

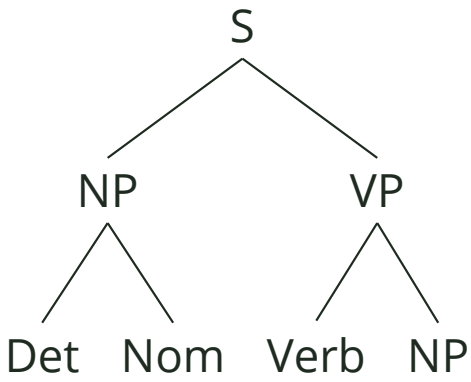
Szabályok
S $\rightarrow$ NP VP
NP $\rightarrow$ Pro
NP $\rightarrow$ Det Nom
VP $\rightarrow$ Verb NP
Nom $\rightarrow$ Nom PP
Nom $\rightarrow$ Nom Noun
Nom $\rightarrow$ Noun
PP $\rightarrow$ P NP

Top-down és bottom-up elemzés

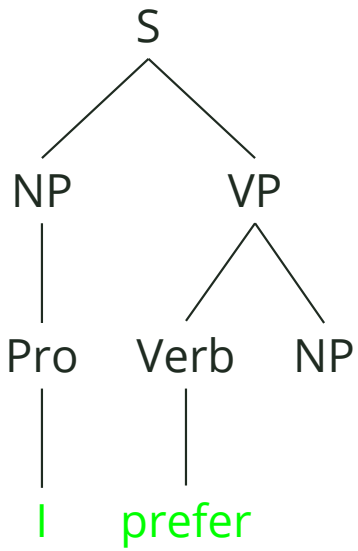
- **Bottom-up elemzés**
  - A levelektől indul, és felfelé haladva építi a fát
  - Ha eljutunk a mondatszimbólumig (S-ig), akkor a mondat grammatikus
  - Előnyei:
    - Nem töltjük az időt a bemenettel nem kompatibilis szabályokkal
    - Minden részszerkezet kompatibilis a bemenet valamelyik részével
- **Top-down elemzés**
  - A mondatszimbólumtól indul, és lefelé haladva építi a fát
  - Ha eljutunk a levelekig, akkor grammatikus a mondat
  - Előnyei:
    - Nem töltjük az időt nem modatszimbólummal végződő részelemzésekkel
    - Minden részszerkezetnek helye lesz a fában

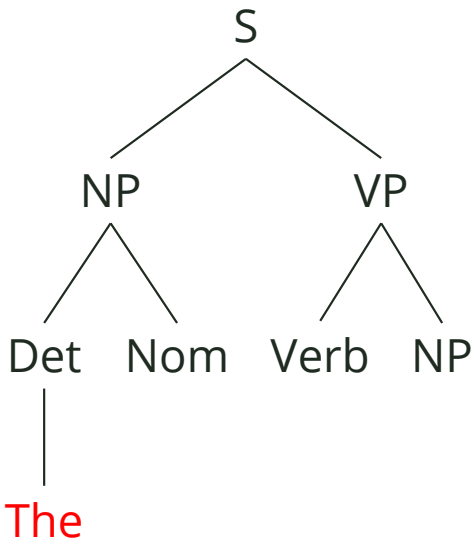
Top-down elemzés – példa





Top-down elemzés – példa

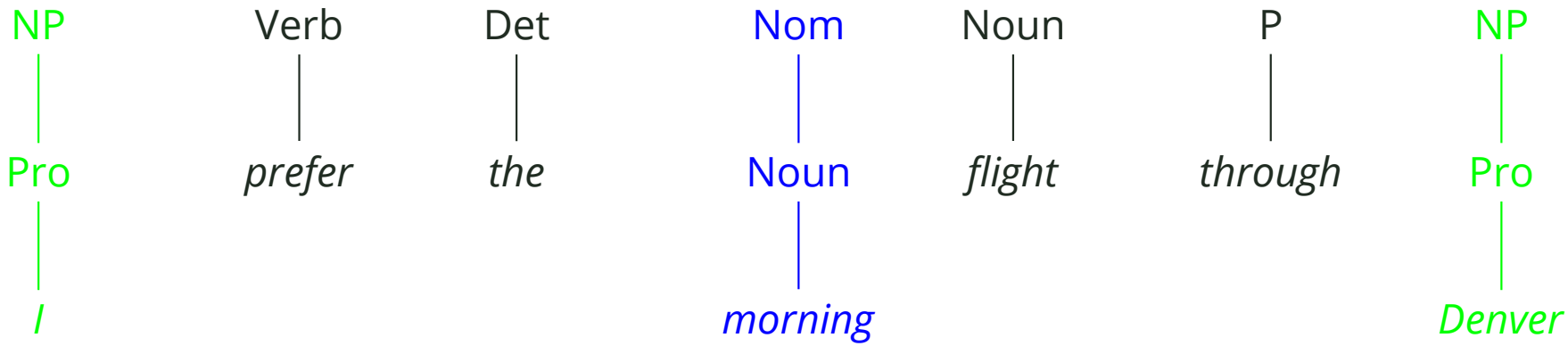




A jobb oldali elemzést elvetjük, mert nem kompatibilis a bemenettel. A bal oldalit folytatjuk az NP-vel az előzőekhez hasonlóan, amíg meg nem kapjuk az összes levelet.

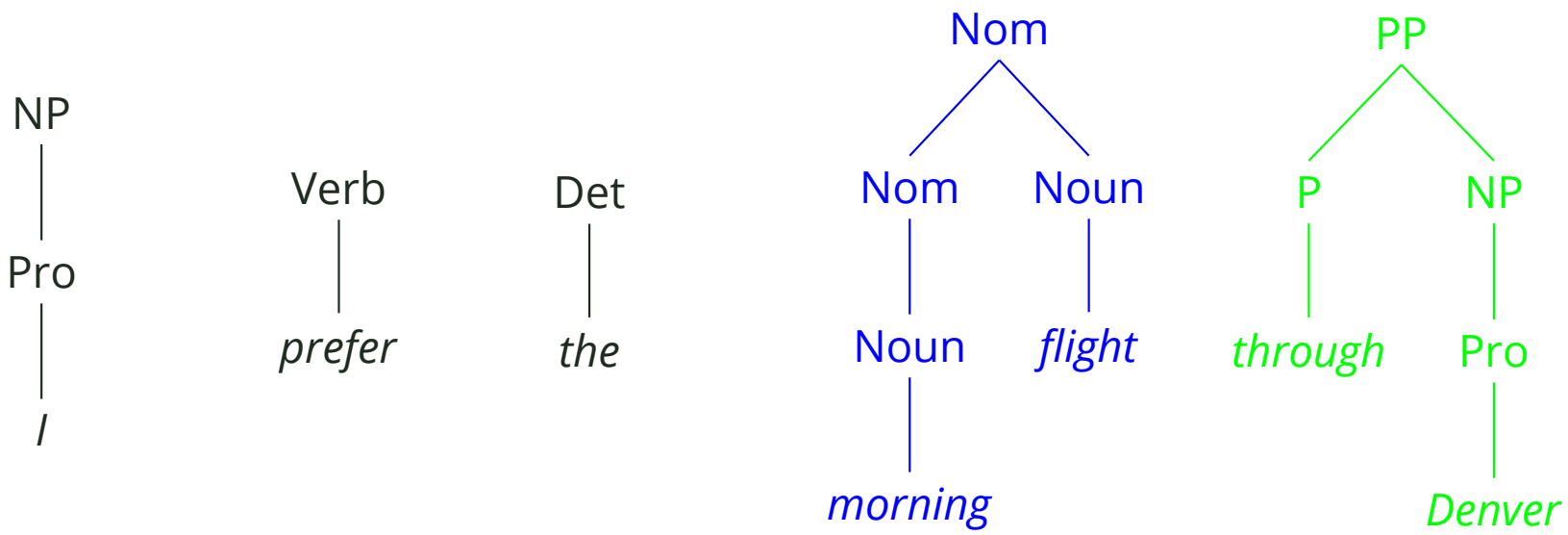
Bottom-up elemzés – példa

- Nom $\rightarrow$ Noun
- NP $\rightarrow$ Pro



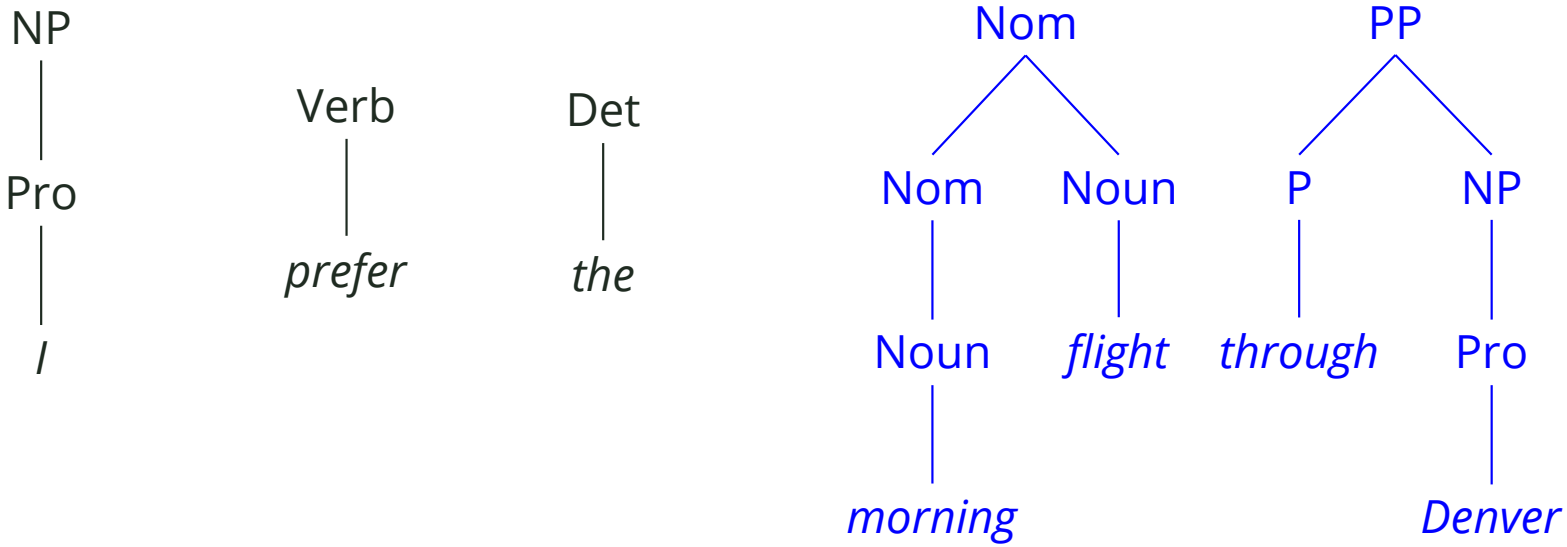
Bottom-up elemzés – példa

- Nom $\rightarrow$ Nom Noun
- PP $\rightarrow$ P NP



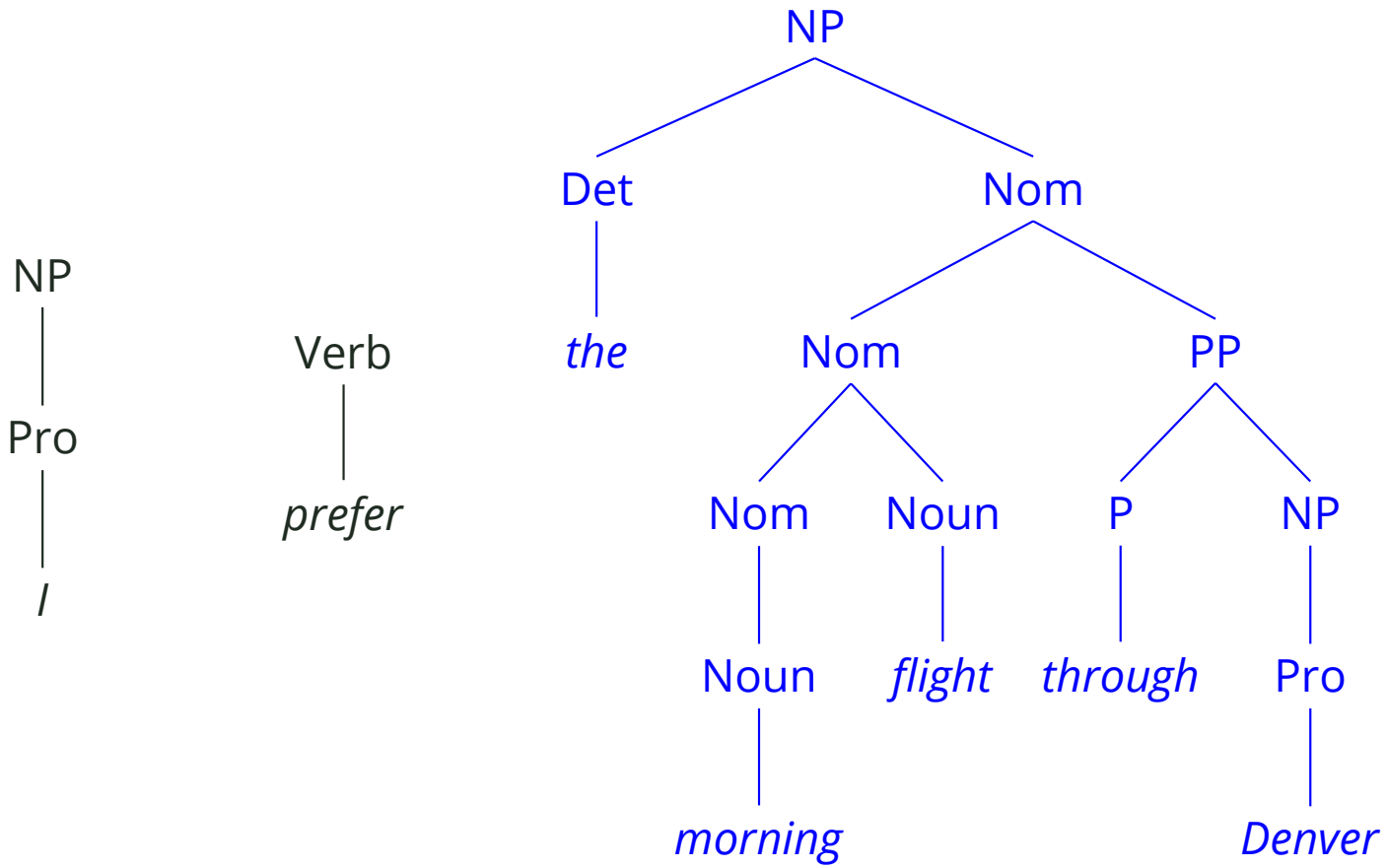
Bottom-up elemzés – példa

- Nom → Nom PP

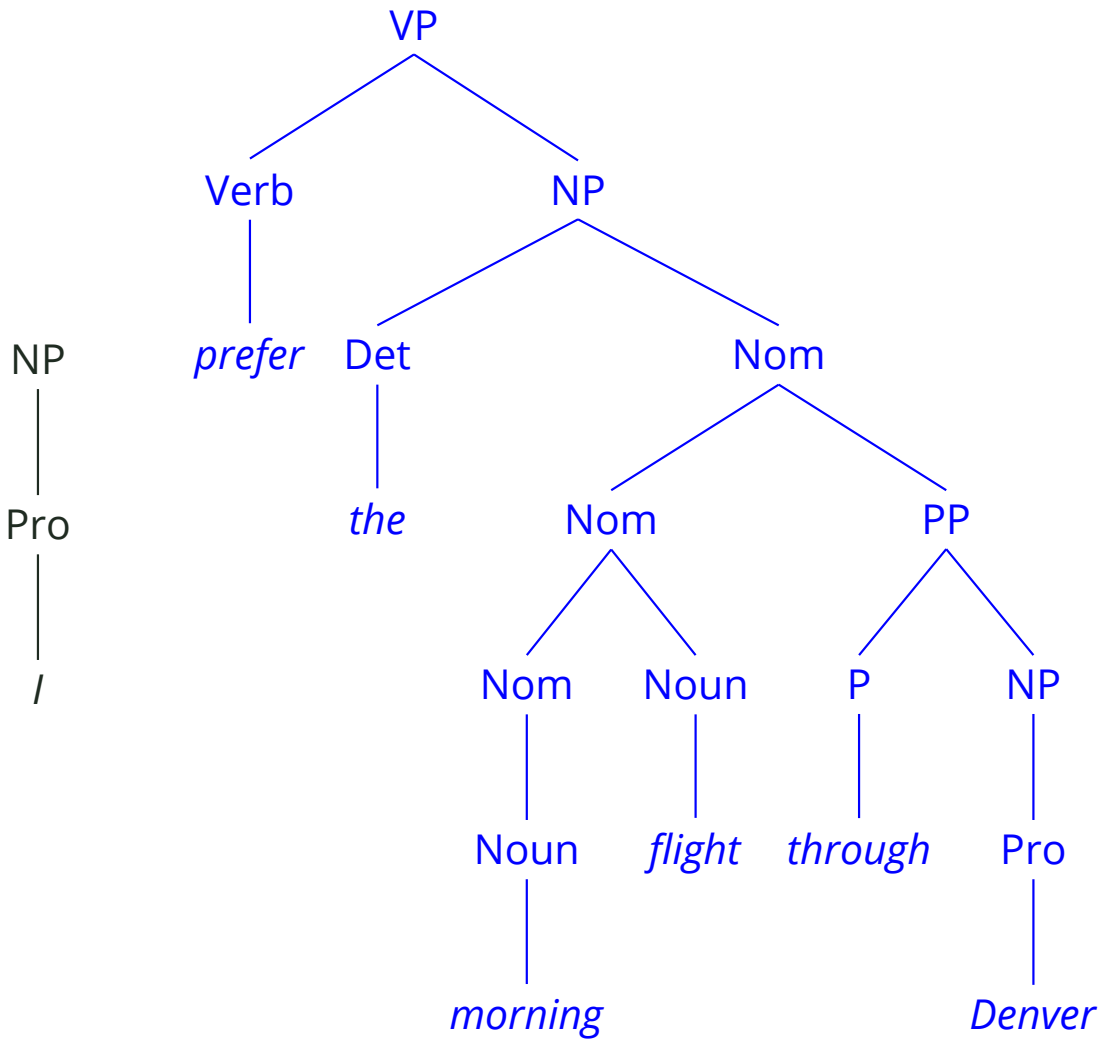


Bottom-up elemzés – példa

- NP → Det Nom

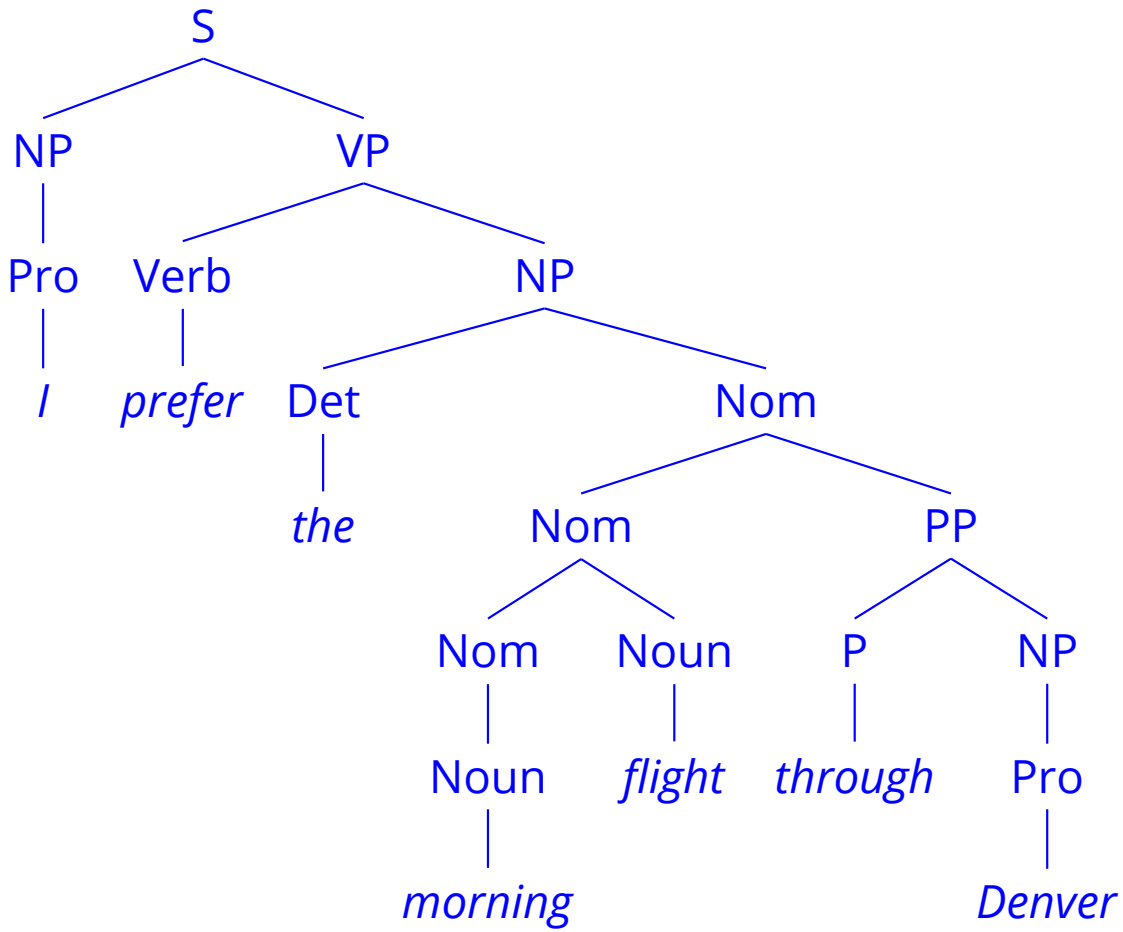


Bottom-up elemzés – példa



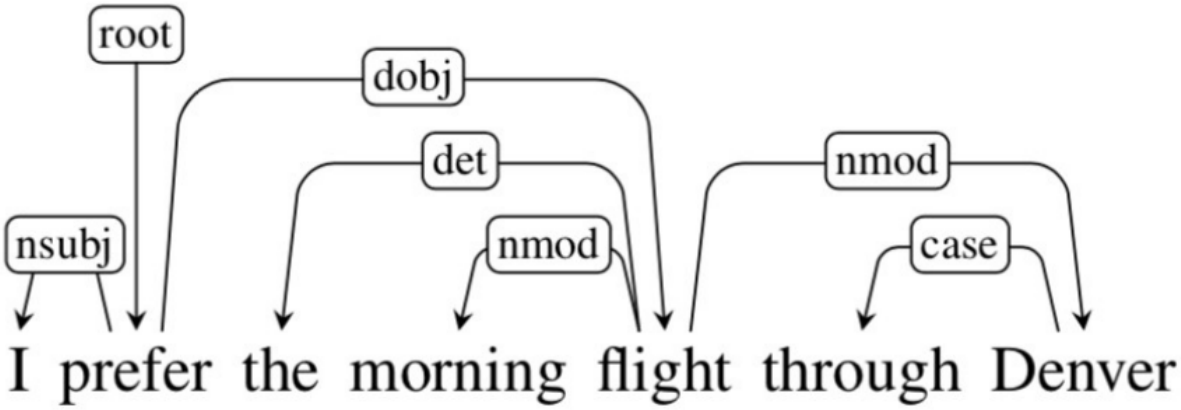
- VP $\rightarrow$ Verb NP

Bottom-up elemzés – példa



- S $\rightarrow$ NP VP

Függőségi elemzés



- Minden szónak pontosan egy feje van
- A egy virtuális gyökércsomópontból (root) indul ki
- A szavak közötti élek címkéi a nyelvtani viszonyt írják le

Relációtípusok – példák

Clausal Argument Relations	Description
NSUBJ	Nominal subject
DOBJ	Direct object
IOBJ	Indirect object
CCOMP	Clausal complement
XCOMP	Open clausal complement
Nominal Modifier Relations	**Description**
NMOD	Nominal modifier
AMOD	Adjectival modifier
NUMMOD	Numeric modifier
APPOS	Appositional modifier
DET	Determiner
CASE	Prepositions, postpositions and other case markers
Other Notable Relations	**Description**
CONJ	Conjunct
CC	Coordinating conjunction

Összetevős vs. függőségi elemzés

Összetevős elemzés	Függőségi elemzés
mi mit előzhet meg	mi mitől függ
szavak → frázisok	szavak → bináris relációk
magasabb szintű kategóriák lehetősége	szintaktikai viszonyok kifejezhetősége
érzékeny a szórendre	nem érzékeny a szórendre

- A gazdag morfológiájú és/vagy szabad szórendű nyelvek számára előnyösebb a függőségi elemzés
- Általában is ez az uralkodó megközelítés

Universal Dependencies

A függőségi annotációt is tartalmazó korpuszokat **treebank**nek nevezzük. Magyar nyelvre jelenleg a legnagyobb ilyen korpusz a Szeged Treebank.

A treebankek annotációjának egységesítésére jött létre a Universal Dependencies projekt.

- Nemzetközi szabvány
- Célja: a különböző nyelvű treebankek annotációja kompatibilis legyen egymással
- $\rightarrow$ univerzális nyelvtant feltételez
- Nyílt, közösségi projekt
  - ~200 résztvevő
  - >100 nyelv (köztük a magyar)

CoNLL-U formátum

CoNLL: Conference on Natural Language Learning

A CoNLL fájlformátumok a CoNLL konferencia ”shared task”-jaihoz tartozó fájlok formátumai.

- Vertikális fájl (tsv)
  - Minden szó (token) külön sorban
  - A szó tulajdonságai külön oszlopokban
- Többféle verziója létezik
- CoNLL-U: a Universal Dependencies által használt formátum

CoNLL-U fájlok mezői

- **ID**: egyedi azonosító
- **Form**: eredeti szóalak (token)
- **Lemma**: szótő
- **Upos**: szófajcímke
- **Xpos**: nyelvspecifikus szófajcímke
- **Feats**: morfológiai jegyek
- **Head**: a szó feje (id)
- **Deprel**: a függőségi viszony típusa
- **Deps**: továbbfejlesztett függőségi gráf
- **Misc**: egyéb annotációk

CoNLL-U példa

Szöveg: Kosztolányi: Esti Kornél

Elemző: UDPipe

```
# sent_id = 5270
# text = Meg volt győződve, hogy maradandó alkotása, s évek múlva is boldogan gondol erre a téli estére, amikor a semmiből létrehívta.
1	Meg	meg	PART	_	_	2	compound:preverb	_	TokenRange=369083:369086
2	volt	van	VERB	_	Definite=Ind|Mood=Ind|Number=Sing|Person=3|Tense=Past|VerbForm=Fin|Voice=Act	0	root	_	TokenRange=369087:369091
3	győződve	győződve	ADV	_	VerbForm=Conv	2	advmod:mode	_	SpaceAfter=No|TokenRange=369092:369100
4	,	,	PUNCT	_	_	2	punct	_	TokenRange=369100:369101
5	hogy	hogy	SCONJ	_	_	7	mark	_	TokenRange=369102:369106
6	maradandó	maradandó	ADJ	_	Case=Nom|Number=Sing|Number[psed]=None|Number[psor]=None|Person[psor]=None|VerbForm=PartFut	7	amod:att	_	TokenRange=369107:369116
7	alkotása	alkotás	NOUN	_	Case=Nom|Number=Sing|Number[psed]=None|Number[psor]=Sing|Person[psor]=3	2	nsubj	_	SpaceAfter=No|TokenRange=369117:369125
8	,	,	PUNCT	_	_	14	punct	_	TokenRange=369125:369126
9	s	s	CCONJ	_	_	14	cc	_	TokenRange=369127:369128
10	évek	év	NOUN	_	Case=Nom|Number=Plur|Number[psed]=None|Number[psor]=None|Person[psor]=None	14	obl	_	TokenRange=369129:369133
11	múlva	múlva	ADP	_	_	10	case	_	TokenRange=369134:369139
12	is	is	CCONJ	_	_	10	cc	_	TokenRange=369140:369142
13	boldogan	boldog	ADJ	_	Case=Ess|Degree=Pos|Number=Sing|Number[psed]=None|Number[psor]=None|Person[psor]=None	14	amod:mode	_	TokenRange=369143:369151
14	gondol	gondol	VERB	_	Definite=Ind|Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin|Voice=Act	2	conj	_	TokenRange=369152:369158
15	erre	ez	PRON	_	Case=Sub|Number=Sing|Number[psed]=None|Number[psor]=None|Person=3|Person[psor]=None|PronType=Dem	18	nmod:obl	_	TokenRange=369159:369163
16	a	a	DET	_	Definite=Def|PronType=Art	18	det	_	TokenRange=369164:369165
17	téli	téli	ADJ	_	Case=Nom|Degree=Pos|Number=Sing|Number[psed]=None|Number[psor]=None|Person[psor]=None	18	amod:att	_	TokenRange=369166:369170
18	estére	est	NOUN	_	Case=Sub|Number=Sing|Number[psed]=None|Number[psor]=Sing|Person[psor]=3	14	nmod:obl	_	SpaceAfter=No|TokenRange=369171:369177
19	,	,	PUNCT	_	_	14	punct	_	TokenRange=369177:369178
20	amikor	amikor	ADV	_	PronType=Rel	23	advmod:tlocy	_	TokenRange=369179:369185
21	a	a	DET	_	Definite=Def|PronType=Art	22	det	_	TokenRange=369186:369187
22	semmiből	semmi	PRON	_	Case=Ela|Number=Sing|Number[psed]=None|Number[psor]=None|Person[psor]=None	23	nmod:obl	_	TokenRange=369188:369196
23	létrehívta	lé+trehív	VERB	_	Definite=Def|Mood=Ind|Number=Sing|Person=3|Tense=Past|VerbForm=Fin|Voice=Act	14	advcl	_	SpaceAfter=No|TokenRange=369197:369207
24	.	.	PUNCT	_	_	2	punct	_	TokenRange=369207:369208
```

Elemző eszközök

e-magyar

- Integrált elemzőlánc
  - Tokenizálás, morfológiai elemzés, összetevős elemzés, függőségi elemzés, főnévicsoport-felismerés, névelem-felismerés
- Webes felület
  - Vagy: emtsv parancssori alkalmazás/Python csomag sok egyéb modullal
- Saját kódkészlet (emMorph)
  - UD-re konvertálható
- tsv, xml kimenet

Elemző eszközök

UDPipe

- Cseh fejlesztés
- Több nyelvre (köztük magyarra) tanított elemzőlánc
  - Tokenizálás, morfológiai elemzés, függőségi elemzés
- Könnyen kezelhető webes felület
- CoNLL-U (tsv) kimenet

Stanza

- Stanford NLP fejlesztés
- Szövegfeldolgozásra alkalmas Python-csomag
- Magyar nyelvre is elérhető

Keresés a függőségi annotációban – demo

CoNLL vonzatkereső **demo** (colab notebook)

Feladatok

1. **Összetevős elemzés** papíros
   - dravida_fak.pdf

2. **Függőségi elemzés** programozós
   Csinálj valamilyen összehasonlító kimutatást a vonzatkereső demo segítségével (pl.: a *kap* ige gyakori tárgyai irodalmi szövegekben és hírportál szövegekben)!
   - Elemezd a szövegeket UDPipe-pal!
   - Futtasd le a demo kódot a szükséges paraméterekkel!
   - Írj egy rövid összefoglalót a kapott eredményekről!

3. **Szorgalmi** extra programozós
   Írj egy függvényt a demo kódba, amely a paraméterben megadott lemma fejeiről készít gyakorisági listát! Csinálj ezzel is valamilyen összehasonlítást (pl. milyen fejek tartoznak az *orosz* és az *ukrán* szavakhoz)!

Köszönöm a figyelmet!

domotor.andrea@btk.elte.hu

https://dh-lab.hu/
https://elte-dh.hu/
https://github.com/elte-dh
https://zenodo.org/communities/elte-dh