인문학을 위한 한국어처리

경희대학교

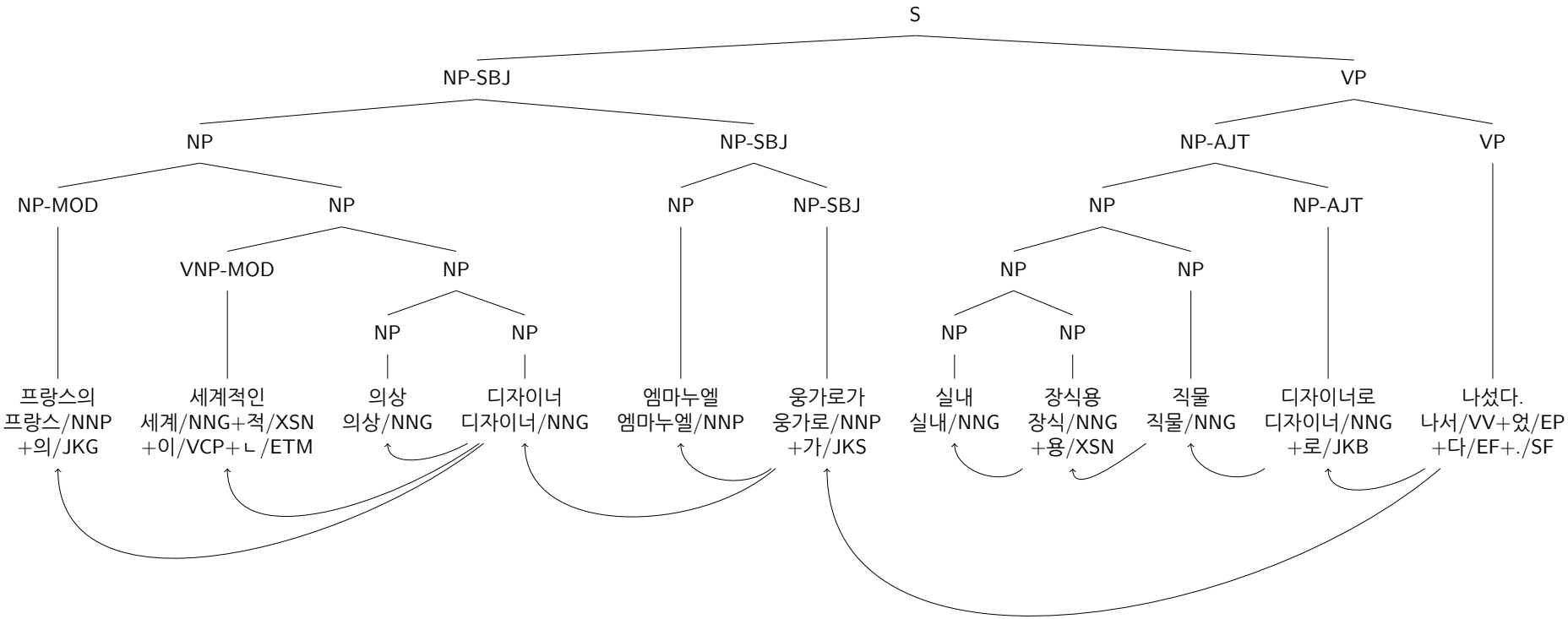
박정열

Department of Linguistics
University at Buffalo

jungyeul@buffalo.edu

2019년 7월4일

Overview


S
NP-SBJ
VP
NP
NP-SBJ
NP-AJT
VP
NP-MOD
NP
NP
NP-SBJ
NP
NP-AJT
VNP-MOD
NP
NP
NP
NP
NP
NP
NP
프랑스의
프랑스/NNP
+의/JKG
세계적인
세계/NNG+적/XSN
+이/VCP+ㄴ/ETM
의상
의상/NNG
디자이너
디자이너/NNG
엠마누엘
엠마누엘/NNP
웅가로가
웅가로/NNP
+가/JKS
실내
실내/NNG
장식용
장식/NNG
+용/XSN
직물
직물/NNG
디자이너로
디자이너/NNG
+로/JKB
나섰다.
나서/VV+었/EP
+다/EF+./SF

1. 형태소 분석 및 품사 태깅: Park J, Tyers F. A New Annotation Scheme for the Sejong Part-of-speech Tagged Corpus. In: *Proceedings of the 13th Linguistic Annotation Workshop (The LAW XIII)*. ; 2019 (August 1, 2019).
2. 구구조 및 의존구조 분석: Park J, Hong J-P, Cha J-W. Korean Language Resources for Everyone. In: *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation: Oral Papers (PACLIC 30)*. Seoul, Korea: Pacific Asia Conference on Language, Information and Computation; 2016:49-58. `http://aclweb.org/anthology/Y/Y16/Y16-2002.pdf`.
3. 실습파일: `https://github.com/jungyeul/july2019-kyunghee`
4. 강의파일: `https://www.overleaf.com/read/ghwqrpktvctr`

형태소 분석 및 품사 태깅

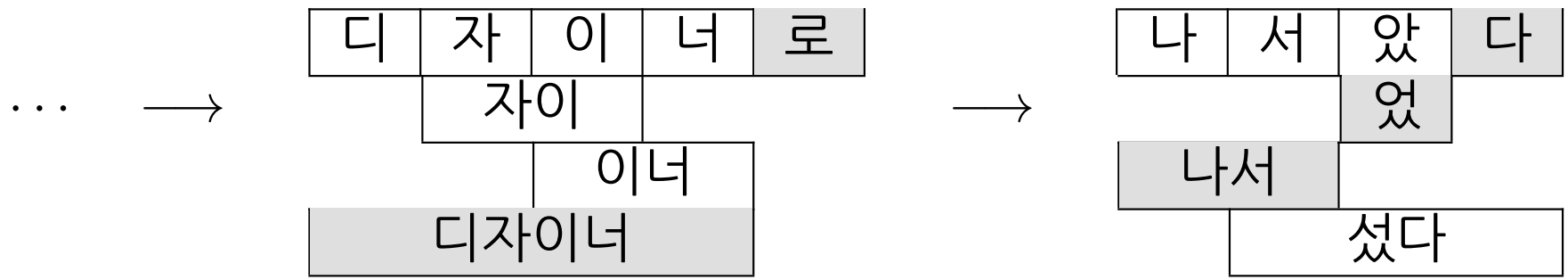
형태소 분석 및 품사 태깅의 입력과 출력

입력:
프랑스의 세계적인 의상 디자이너 엠마누엘 웅가로가 실내 장식용 직물 디자이너로 나섰다.

출력:

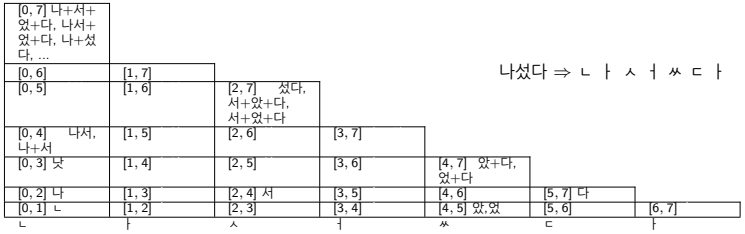
프랑스의	BOS	프랑스/NNP+의/JKG
세계적인		세계/NNG+적/XSN+이/VCP+ㄴ/ETM
의상		의상/NNG
디자이너		디자이너/NNG
엠마누엘		엠마누엘/NNP
웅가로가		웅가로/NNP+가/JKS
실내		실내/NNG
장식용		장식/NNG+용/XSN
직물		직물/NNG
디자이너로		디자이너/NNG+로/JKB
나섰다.	EOS	나서/VV+었/EP+다/EF+./SF

```
<source>
<date>
BTAA0001-00000001	1993/06/08	1993/SN + //SP + 06/SN + //SP + 08/SN
</date>
<page>
BTAA0001-00000002	19	19/SN
</page>
</source>
<head>
BTAA0001-00000003	엠마누엘	엠마누엘/NNP
BTAA0001-00000004	웅가로	웅가로/NNP
BTAA0001-00000005	/	//SP
BTAA0001-00000006	의상서	의상/NNG + 서/JKB
BTAA0001-00000007	실내	실내/NNG
BTAA0001-00000008	장식품으로...	장식품/NNG + 으로/JKB + .../SE
BTAA0001-00000009	디자인	디자인/NNG
BTAA0001-00000010	세계	세계/NNG
BTAA0001-00000011	넓혀	넓히/VV + 어/EC
</head>
<p>
BTAA0001-00000012	프랑스의	프랑스/NNP + 의/JKG
BTAA0001-00000013	세계적인	세계/NNG + 적/XSN + 이/VCP + ㄴ/ETM
BTAA0001-00000014	의상	의상/NNG
BTAA0001-00000015	디자이너	디자이너/NNG
BTAA0001-00000016	엠마누엘	엠마누엘/NNP
BTAA0001-00000017	웅가로가	웅가로/NNP + 가/JKS
BTAA0001-00000018	실내	실내/NNG
BTAA0001-00000019	장식용	장식/NNG + 용/XSN
BTAA0001-00000020	직물	직물/NNG
BTAA0001-00000021	디자이너로	디자이너/NNG + 로/JKB
BTAA0001-00000022	나섰다.	나서/VV + 었/EP + 다/EF + ./SF
</p>
<p>
...
```


· · ·
디 자 이 너 로
자이
이너
디자이너
나 서 았 다
었
나서
섰다

[0, 7] 나+서+었+다, 나서+었+다, 나+섰다, ...						
[0, 6]	[1, 7]					
[0, 5]	[1, 6]	[2, 7] 섰다, 서+았+다, 서+었+다				
[0, 4] 나서, 나+서	[1, 5]	[2, 6]	[3, 7]			
[0, 3] 낫	[1, 4]	[2, 5]	[3, 6]	[4, 7] 았+다, 었+다		
[0, 2] 나	[1, 3]	[2, 4] 서	[3, 5]	[4, 6]	[5, 7] 다	
[0, 1] ㄴ	[1, 2]	[2, 3]	[3, 4]	[4, 5] 았,었	[5, 6]	[6, 7]
ㄴ	ㅏ	ㅅ	ㅓ	ㅆ	ㄷ	ㅏ

나섰다 ⇒ ㄴ ㅏ ㅅ ㅓ ㅆ ㄷ ㅏ



나섰다 ⇒ ㄴ ㅏ ㅅ ㅓ ㅆ ㄷ ㅏ

function CKY-PARSE(*words, grammar*) **returns** *table*

for $j \leftarrow$ **from** 1 **to** LENGTH(*words*) **do**
  for all $\{A \mid A \rightarrow words[j] \in grammar\}$
    $table[j-1, j] \leftarrow table[j-1, j] \cup A$
  for $i \leftarrow$ **from** $j-2$ **downto** 0 **do**
    for $k \leftarrow i+1$ **to** $j-1$ **do**
      for all $\{A \mid A \rightarrow BC \in grammar$ **and** $B \in table[i,k]$ **and** $C \in table[k,j]\}$
        $table[i,j] \leftarrow table[i,j] \cup A$

[2, 7] 섰다, 서+았+다, 서+었+다:

$$
\begin{array}{lll}
[2,7] & = [2,3] + [3,7] & = \emptyset + \emptyset \\
 & = [2,4] + [4,7] & = 서 + (았|었+다) \\
 & = [2,5] + [5,7] & = \emptyset + 다 \\
 & = [2,6] + [6,7] & = \emptyset + \emptyset \\
 & \underline{\qquad\qquad} & \underline{\qquad\qquad} \\
 & = [2,7] & = 섰다 \text{ (modified cky)}
\end{array}
$$

형태소 분석 및 품사 태깅 자원

1. UDPipe (Straka and Straková, 2017)[1]를 사용한 한국어 형태소 분석 및 품사 태깅 (Park and Tyers, 2019)[2]
2. ExpressoK[3] (Park et al., 2016)[4]

[1] http://ufal.mff.cuni.cz/udpipe
[2] https://github.com/jungyeul/sjmorph
[3] http://air.changwon.ac.kr/~airdemo/kg_tagger/
[4] http://doi.org/10.5281/zenodo.884606

형태소 분석 및 품사 태깅 실습

1. `udpipe --input=horizontal --tokenizer=presegmented --tag sjmorph.model input.txt > udpipe-output.txt`
2. `java -classpath kyunghee-v01.jar UDPipe2Espresso udpipe-output.txt > sejong-output.txt`

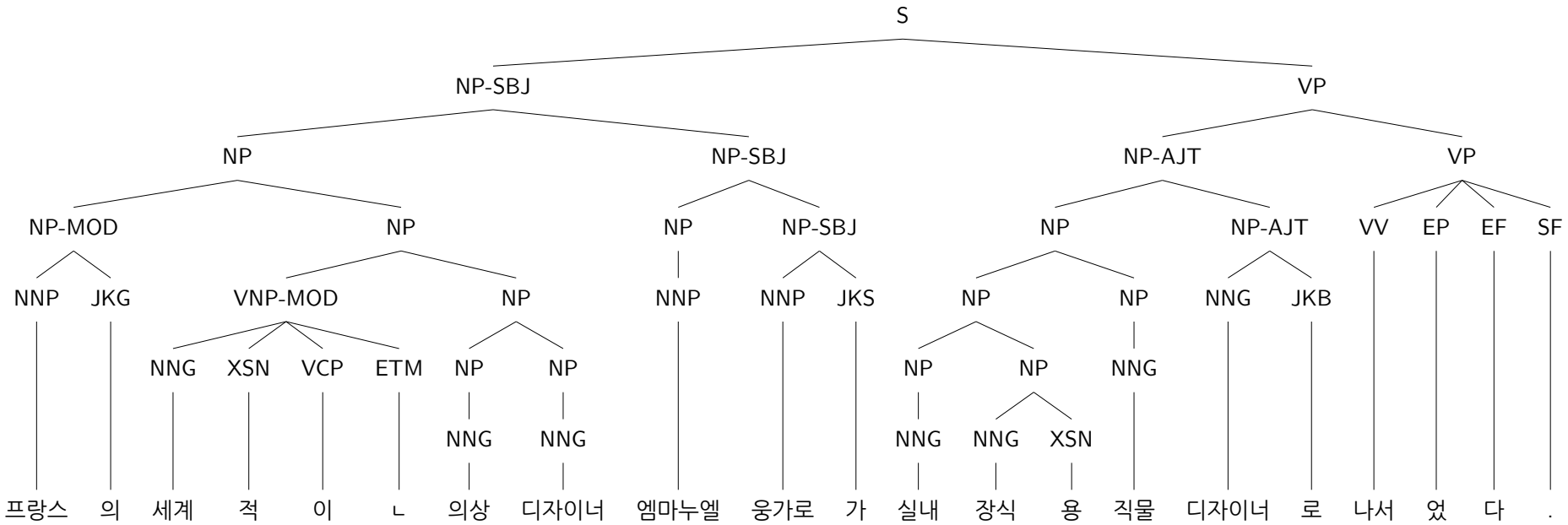
```
# newdoc id = input.txt
# newpar
# sent_id = 1
# text = 프랑스의 세계적인 의상 디자이너 엠마누엘 웅가로가 실내 장식용 직물 디자이너로 나섰다.
1-2 프랑스의 _ _ _ _ _ _ _ _ _
1 프랑스 프랑스 PROPN NNP _ _ _ _ _
2 의 의 ADP JKG _ _ _ _ _
3-6 세계적인 _ _ _ _ _ _ _ _ _
3 세계 세계 NOUN NNG _ _ _ _ _
4 적 적 PRT XSN _ _ _ _ _
5 이 이 VERB VCP _ _ _ _ _
6 ㄴ ㄴ PRT ETM _ _ _ _ _
7 의상 의상 NOUN NNG _ _ _ _ _
8 디자이너 디자이너 NOUN NNG _ _ _ _ _
9 엠마누엘 엠마누엘 PROPN NNP _ _ _ _ _
10-11 웅가로가 _ _ _ _ _ _ _ _ _
10 웅가로 웅가로 PROPN NNP _ _ _ _ _
11 가 가 ADP JKS _ _ _ _ _
12 실내 실내 NOUN NNG _ _ _ _ _
13-14 장식용 _ _ _ _ _ _ _ _ _
13 장식 장식 NOUN NNG _ _ _ _ _
14 용 용 PRT XSN _ _ _ _ _
15 직물 직물 NOUN NNG _ _ _ _ _
16-17 디자이너로 _ _ _ _ _ _ _ _ _
16 디자이너 디자이너 NOUN NNG _ _ _ _ _
17 로 로 ADP JKB _ _ _ _ _
18-20 나섰다 _ _ _ _ _ _ _ SpaceAfter=No
18 나서 나서 VERB VV _ _ _ _ _
19 었 었 PRT EP _ _ _ _ _
20 다 다 PRT EF _ _ _ _ _
21 . . PUNCT SF _ _ _ _ SpacesAfter=\n
```

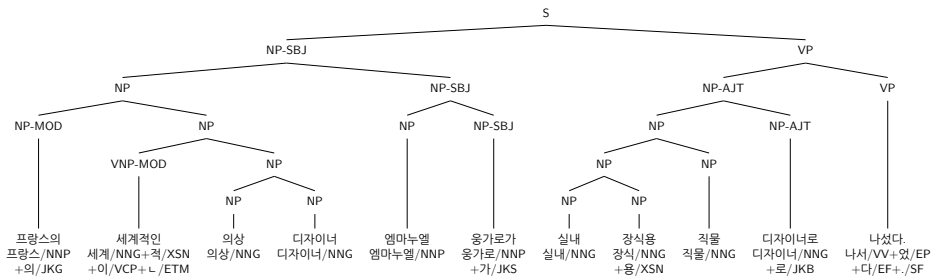
구구조 분석

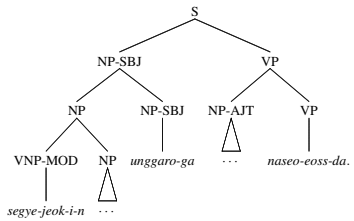
구구조 분석 입력과 출력

입력:
프랑스 의 세계 적 이 ㄴ 의상 디자이너 엠마누엘 웅가로 가 실내 장식 용 직물 디자이너 로 나서 었 다 .

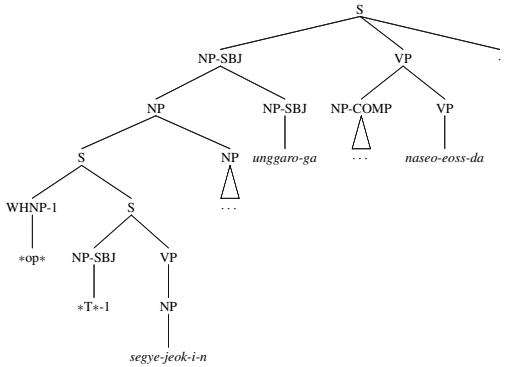
출력:
```
(S (NP-SBJ (NP (NP-MOD (NNP 프랑스) (JKG 의)
                (NP (VNP-MOD (NNG 세계) (XSN 적) (VCP 이) (ETM ㄴ))
                    (NP (NP (NNG 의상))
                        (NP (NNG 디자이너)))))
                    (NP-SBJ (NP (NNP 엠마누엘))
                            (NP-SBJ (NNP 웅가로) (JKS 가))))
    (VP (NP-AJT (NP (NP (NP (NNG 실내))
                        (NP (NNG 장식) (XSN 용)))
                    (NP (NNG 직물)))
                (NP-AJT (NNG 디자이너) (JKB 로)))
        (VP (VV 나서) (EP 었) (EF 다) (SF .))))
```


S
NP-SBJ
VP
NP
NP-SBJ
NP-AJT
VP
NP-MOD
NP
NP
NP-SBJ
NP
NP-AJT
VV
EP
EF
SF
NNP
JKG
VNP-MOD
NP
NNP
NNP
JKS
NP
NP
NNG
JKB
NNG
XSN
VCP
ETM
NP
NP
NP
NP
NNG
NNG
NNG
NNG
NNG
XSN
프랑스
의
세계
적
이
ㄴ
의상
디자이너
엠마누엘
웅가로
가
실내
장식
용
직물
디자이너
로
나서
었
다
.

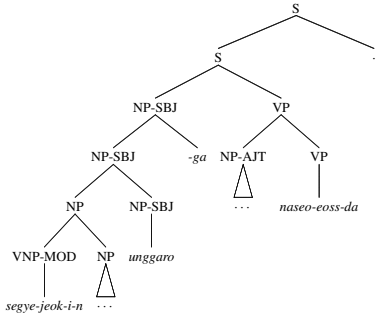

S
NP-SBJ
VP
NP
NP-SBJ
NP-AJT
VP
NP-MOD
NP
NP
NP-SBJ
NP
NP-AJT
VNP-MOD
NP
NP
NP
NP
NP
NP
NP
프랑스의
프랑스/NNP
+의/JKG
세계적인
세계/NNG+적/XSN
+이/VCP+ㄴ/ETM
의상
의상/NNG
디자이너
디자이너/NNG
엠마누엘
엠마누엘/NNP
웅가로가
웅가로/NNP
+가/JKS
실내
실내/NNG
장식용
장식/NNG
+용/XSN
직물
직물/NNG
디자이너로
디자이너/NNG
+로/JKB
나섰다.
나서/VV+었/EP
+다/EF+./SF



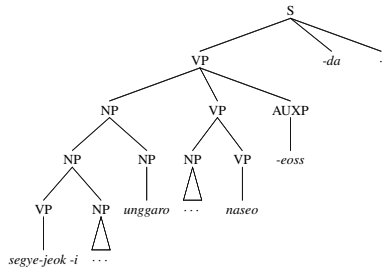
(a) Sejong treebank



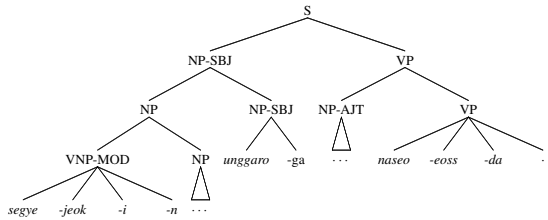
(b) Penn Korean treebank



(c) Korean tree-adjoining grammar

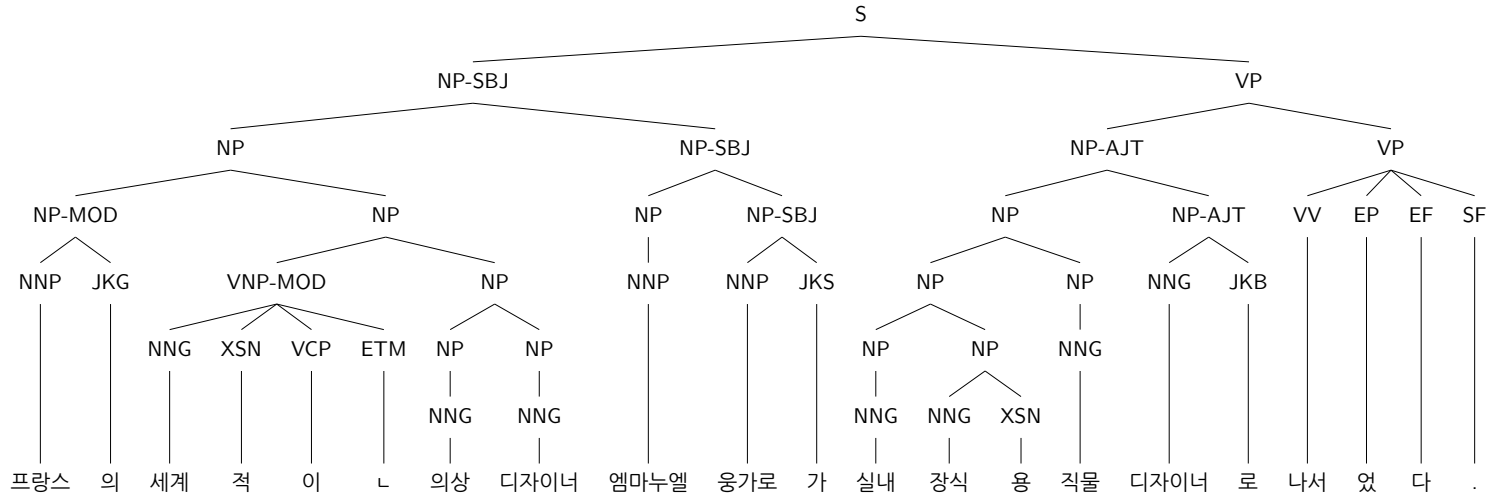


(d) KAIST treebank



(e) Phrase structure parsing for Korean

figures from Park (2018)



- CFG 규칙추출: $V \rightarrow (V \cup \Sigma)^*$
- CFG를 CNF로 변경후 CKY 알고리즘 적용: $A \rightarrow BC$ or $A \rightarrow c$ where $A, B, C \in V$ and $c \in \Sigma$

CFG	CNF
NP → NP-MOD NP	NP → NP-MOD NP
NP-MOD → NNP JKG	NP-MOD → NNP JKG
NP → VNP-MOD NP	NP → VNP-MOD NP
VNP-MOD → NNG XSN VCP ETM	VNP-MOD → NNG X1
...	X1 → XSN X2
	X2 → VCP ETM
	...

구구조 분석 자원

1. Berkeley (Petrov et al., 2006)파서[5]를 사용한 한국어 구구조 분석 (Park, 2017a)[6]
   - 한국어 구구조 분석 연구 (Choi et al., 2012; Park et al., 2016)
   - 한국어 구구조 분석 오류 분석 (Park and Kim, 2019)

[5] https://github.com/slavpetrov/berkeleyparser
[6] http://doi.org/10.5281/zenodo.891267

구구조 분석 실습

1. 토큰만 사용한 구구조 분석
   1. `java -classpath kyunghee-v01.jar UDPipe2tok udpipe-output.txt > udpipe-tokenized.txt`
   2. `java -jar BerkeleyParser-1.7.jar -gr berkeley.sjtree.model < udpipe-tokenized.txt > berkeley-token-output.txt`
2. 토큰 및 품사를 사용한 구구조 분석
   1. `java -classpath kyunghee-v01.jar MakeBerkeleyTestWithPOSIn sejong-output.txt > berkeley-pos-input.txt`
   2. `java -jar BerkeleyParser-1.7.jar -useGoldPOS -gr berkeley.sjtree.model < berkeley-pos-input.txt > berkeley-pos-output.txt`

의존구조 분석

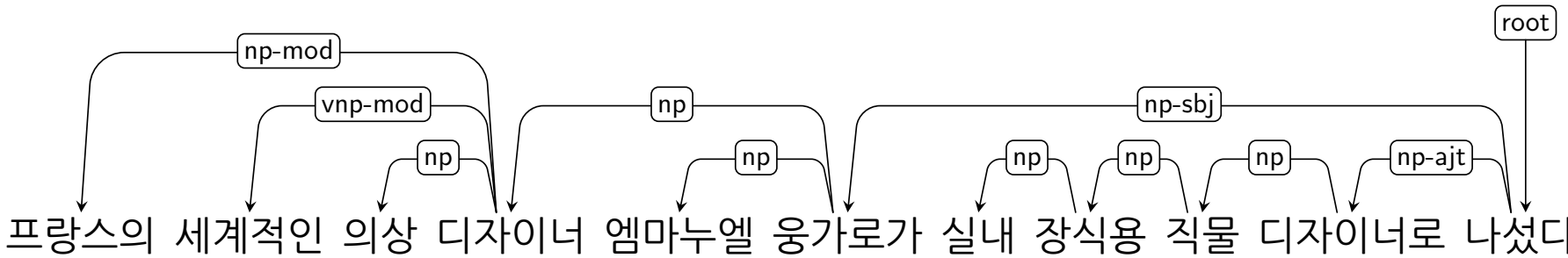
의존구조 분석 입력과 출력

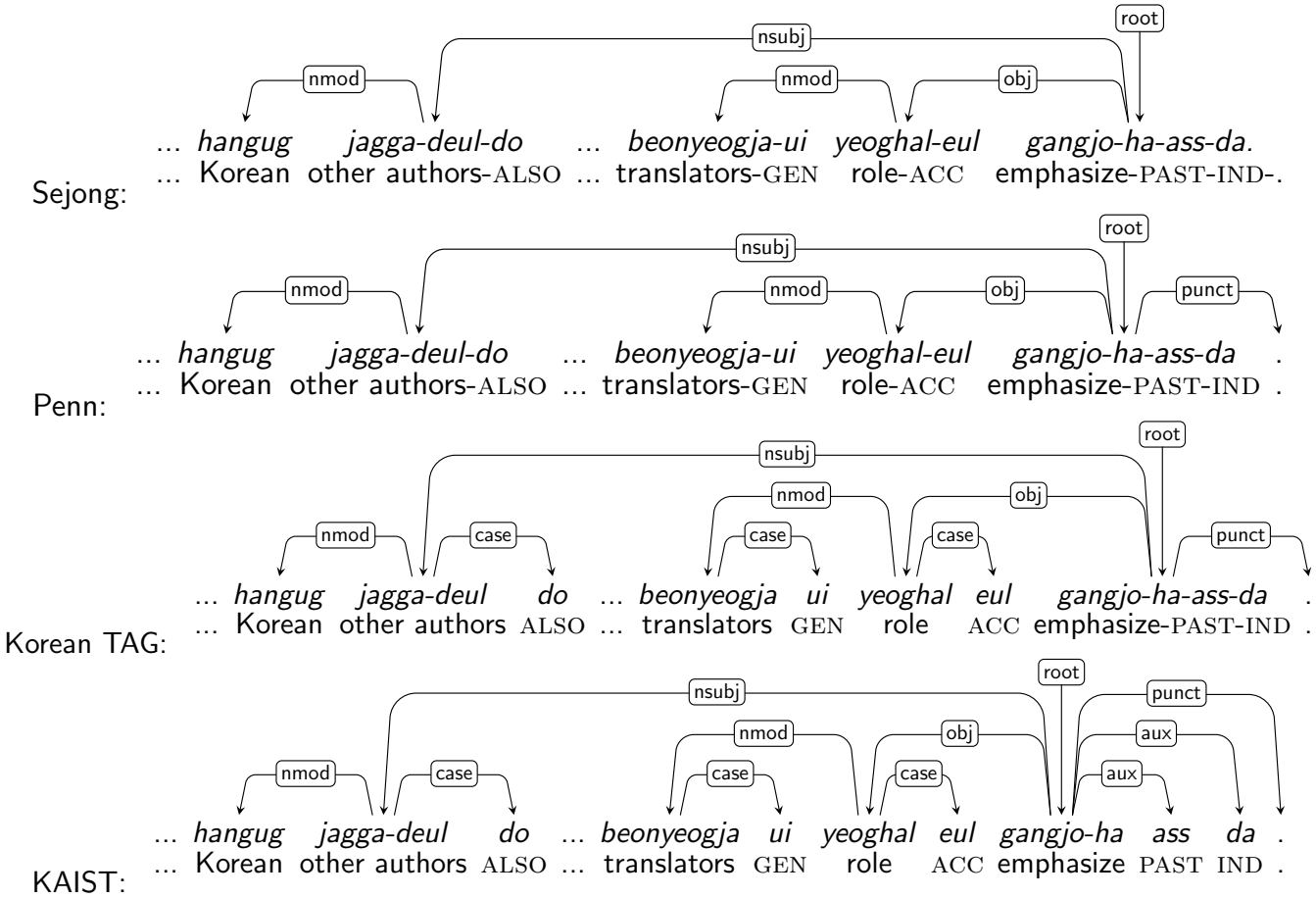
입력:

1	프랑스의	프랑스	NNP	NNP+JKG	JKG
2	세계적인	세계적이	NNG+XSN+VCP	NNG+XSN+VCP+ETM	ETM
3	의상	의상	NNG	NNG	_
4	디자이너	디자이너	NNG	NNG	_
5	엠마누엘	엠마누엘	NNP	NNP	_
6	웅가로가	웅가로	NNG	NNG+JKS	JKS
7	실내	실내	NNG NNG	_	
8	장식용	장식용	NNG	NNG	_
9	직물	직물	NNG	NNG	_
10	디자이너로	디자이너	NNG	NNG+JKB JKB	
11	나섰다.	나서	VV	VV+EP+EF+SF	EP|EF|SF

출력:

1	프랑스의	프랑스	NNP	NNP+JKG	JKG	4	NP-MOD	_ _
2	세계적인	세계적이	NNG+XSN+VCP	NNG+XSN+VCP+ETM	ETM	4	VNP-MOD	_ _
3	의상	의상	NNG	NNG	_	4	NP	_ _
4	디자이너	디자이너	NNG	NNG	_	6	NP	_ _
5	엠마누엘	엠마누엘	NNP	NNP	_	6	NP	_ _
6	웅가로가	웅가로	NNG	NNG+JKS	JKS	11	NP-SBJ	_ _
7	실내	실내	NNG	NNG	_	8	NP	_ _
8	장식용	장식용	NNG	NNG	_	9	NP	_ _
9	직물	직물	NNG	NNG	_	10	NP	_ _
10	디자이너로	디자이너	NNG	NNG+JKB	JKB	11	NP-AJT	_ _
11	나섰다.	나서	VV	VV+EP+EF+SF	EP|EF|SF	0	ROOT	_ _


root
np-mod
vnp-mod
np
np
np
np-sbj
np
np
np
np-ajt
프랑스의 세계적인 의상 디자이너 엠마누엘 웅가로가 실내 장식용 직물 디자이너로 나섰다.



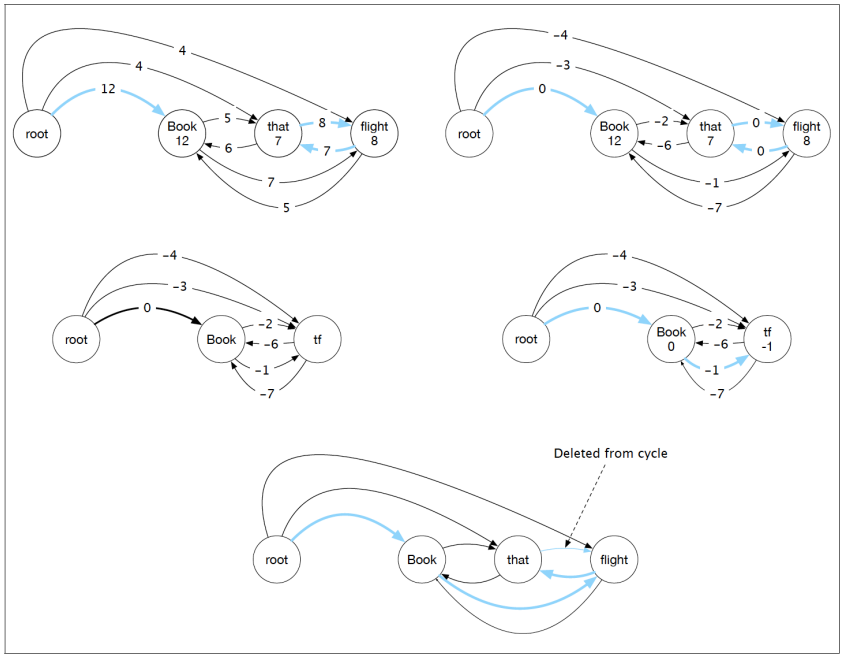
figures from Park (2017c)

전이 기반

Step	Stack	Word List	Action	Relation Added
0	[root]	[book, the, flight, through, houston]	RIGHTARC	(root → book)
1	[root, book]	[the, flight, through, houston]	SHIFT	
2	[root, book, the]	[flight, through, houston]	LEFTARC	(the ← flight)
3	[root, book]	[flight, through, houston]	RIGHTARC	(book → flight)
4	[root, book, flight]	[through, houston]	SHIFT	
5	[root, book, flight, through]	[houston]	LEFTARC	(through ← houston)
6	[root, book, flight]	[houston]	RIGHTARC	(flight → houston)
7	[root, book, flight, houston]	[]	REDUCE	
8	[root, book, flight]	[]	REDUCE	
9	[root, book]	[]	REDUCE	
10	[root]	[]	Done	

Figure 13.10 A processing trace of *Book the flight through Houston* using the arc-eager transition operators.

그래프 기반



figures from Jurafsky and Martin (2018)

- *Advanced Computational linguistics* (CSE/LIN667, Fall 2018) on dependency parsing and FrameNet (University at Buffalo)[7]

[7] https://sites.google.com/view/aclfall2018/

의존구조 분석 자원

1. MaltParser (Nivre et al., 2006)[8]를 사용한 한국어 의존구조 분석 (Park, 2017b)[9]
   - 한국어 의존구조 분석 연구 (Park et al., 2013, 2016; Park, 2017c)

[8] http://www.maltparser.org

[9] http://doi.org/10.5281/zenodo.891274

의존구조 분석 실습

1. `java -classpath kyunghee-v01.jar MakeMaltTestIn sejong-output.txt > malt-input.txt`
2. `java -jar maltparser-1.9.2.jar -c sejong-malt -i malt-input.txt -o malt-output.txt -m parse`

끝

Choi, D., Park, J., and Choi, K.-S. (2012). Korean Treebank Transformation for Parser Training. In *Proceedings of the ACL 2012 Joint Workshop on Statistical Parsing and Semantic Processing of Morphologically Rich Languages*, pages 78–88, Jeju, Republic of Korea. Association for Computational Linguistics.

Jurafsky, D. and Martin, J. H. (2018). *Speech and Language Processing (3rd ed. draft)*. http://www.web.stanford.edu/~jurafsky/slp3/, third edition.

Nivre, J., Hall, J., and Nilsson, J. (2006). MaltParser: A Data-Driven Parser-Generator for Dependency Parsing. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*.

Park, J. (2017a). Berkeley parser model for Korean: Sejong treebank. *10.5281/zenodo.891267*.

Park, J. (2017b). MaltParser model for Korean: Sejong treebank. *10.5281/zenodo.891273*.

Park, J. (2017c). Segmentation Granularity in Dependency Representations for Korean. In *Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017)*, pages 187–196, Pisa, Italy. Association for Computational Linguistics.

Park, J. (2018). Word Granularity in Korean. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP), (Under review)*.

Park, J., Hong, J.-P., and Cha, J.-W. (2016). Korean Language Resources for Everyone. In *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation: Oral Papers (PACLIC 30)*, pages 49–58, Seoul, Korea. Pacific Asia Conference on Language, Information and Computation.

Park, J., Kawahara, D., Kurohashi, S., and Choi, K.-S. (2013). Towards Fully Lexicalized Dependency Parsing for Korean. In *Proceedings of The 13th International Conference on Parsing Technologies (IWPT 2013)*, Nara, Japan. International Conference on Parsing Technologies (IWPT 2013).

Park, J. and Kim, M. (2019). A Note on Constituent Parsing for Korean. *Natural Language Engineering, Cambridge University Press (Under review)*.

Park, J. and Tyers, F. (2019). A New Annotation Scheme for the Sejong Part-of-speech Tagged Corpus. In *Proceedings of the 13th Linguistic Annotation Workshop (The LAW XIII)*.

Petrov, S., Barrett, L., Thibaux, R., and Klein, D. (2006). Learning Accurate, Compact, and Interpretable Tree Annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 433–440, Sydney, Australia. Association for Computational Linguistics.

Straka, M. and Straková, J. (2017). Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada. Association for Computational Linguistics.