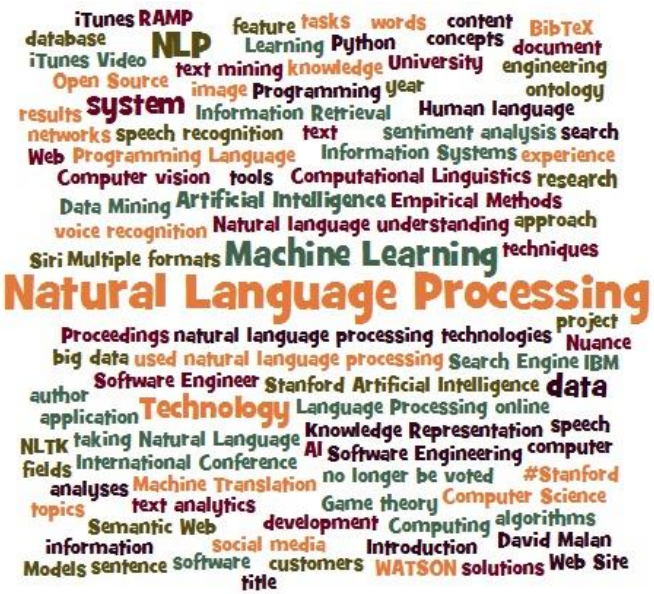
자연어처리 개요

강 승 식
국민대학교 인공지능학부

NLP issues and applications



NLP 핵심 기술

- Morphological analysis(형태소 분석)
  - Word-level
- Syntactic analysis(구문 분석)
  - Sentence-level
- Semantic analysis(의미 분석)
  - Word-sense disambiguation
- Natural Language Generation(자연어 생성)
- Language Resources(언어 자원)
  - 말뭉치, WordNet, 온톨로지 등

NLP 분석 모듈

- Tokenizer, stemmer, word splitter
- 형태소 분석기
- Tagger
  - 품사 태거 (POS tagger)
  - 개체명 태거 (NER tagger)
- 구문 분석기
  - NP chunking

주요 활용분야

- Machine Translation, 1950's-now
- Information Retrieval, 1980's-now
  - Text Classification, Information Extraction
  - Text Summarization
  - Text Mining, Opinion Mining
  - Sentiment Classification(감성 분류)
- Natural Language Understanding, 1960-70, 2000's
  - ELIZA: Doctor, Joseph Weizenbaum, MIT, 1965
  - SHRDLU: Robot arm, Terry Winograd, MIT, 1971
  - LUNAR
  - Ask Jeeves(ask.com), 1996
  - Wolfram alpha, 2009

- Speller and grammar checker

- Spam mail filtering, Spam 문자 filtering
- Sentiment analysis(감성 분석)

- 아이폰 시리, IBM 왓슨, 자동통역 시스템
- 텍스트 마이닝, 빅데이터 분석

NLP Resources and NLTK in Python

NLP resources in http://nlp.stanford.edu/

Core | Projects | Archive

- + Stanford CoreNLP
- + Stanford Parser
- + Stanford POS Tagger
- + Stanford Named Entity Recognizer
- + Stanford RegexNER
- + Stanford Word Segmenter
- + Stanford Classifier
- + Stanford EnglishTokenizer
- + Stanford TokensRegex
- + Stanford Temporal Tagger (SUTime)
- + Stanford Pattern-based Information Extraction and Diagnostics (SPIED)
- + Stanford Relation Extractor

Core | Projects | Archive

- + Stanford Neural Machine Translation
- + Stanford Natural Language Inference Corpus (SNLI)
- + Semantic Parsing with Execution (SEMPRE)

SEMPRE is a toolkit for training semantic parsers, which map natural language utterances to denotations (answers) via intermediate logical forms.

- + Stanford Open Information Extraction

A tool for extracting open domain relation triples; e.g., "cats play with yarn" yields (cats; play with; yarn).

- + GloVe: Global Vectors for Word Representations
- + Deep Learning for Sentiment Analysis
- + Tregex, Tsurgeon, and Semgrex

Tools for matching patterns in linguistic trees (following the tgrep/tgrep2 tradition), a GUI for this, and a tree-transformation utility built on top of this matching language. Also, a similar utility for matching patterns in dependency graphs.

- + Phrasal

A state-of-the-art phrase-based machine translation system.

POS tagging

The strongest rain ever recorded in India shut down the financial hub of Mumbai, snapped communication lines, closed airports and forced thousands of people to sleep in their offices or walk home during the night, officials said today.

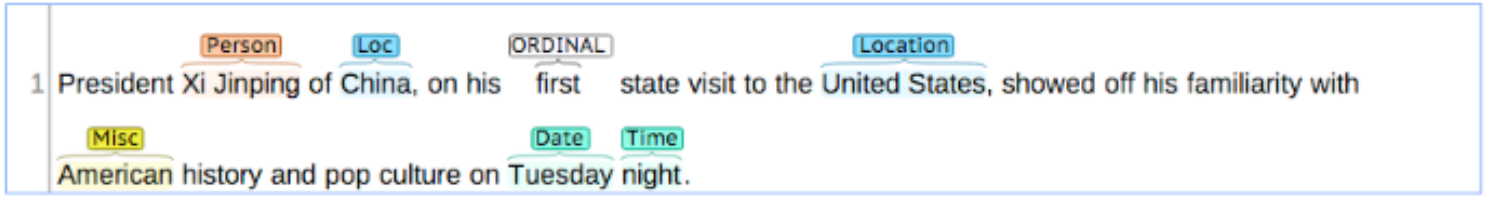
The/DT strongest/JJS rain/NN ever/RB recorded/VBN in/IN India/NNP shut/VBD down/RP the/DT financial/JJ hub/NN of/IN Mumbai/NNP ,/, snapped/VBD communication/NN lines/NNS ,/, closed/VBD airports/NNS and/CC forced/VBD thousands/NNS of/IN people/NNS to/TO sleep/VB in/IN their/PRP$ offices/NNS or/CC walk/VB home/NN during/IN the/DT night/NN ,/, officials/NNS said/VBD today/NN ./.

```
(ROOT
  (S
    (S
      (NP
        (NP (DT The) (JJS strongest) (NN rain))
        (VP
          (ADVP (RB ever))
          (VBN recorded)
          (PP (IN in)
            (NP (NNP India)))))
      (VP
        (VP (VBD shut)
          (PRT (RP down))
          (NP
            (NP (DT the) (JJ financial) (NN hub))
            (PP (IN of)
              (NP (NNP Mumbai)))))
        (, ,)
        (VP (VBD snapped)
          (NP (NN communication) (NNS lines)))
        (, ,)
        (VP (VBD closed)
          (NP (NNS airports)))
        (CC and)
        (VP (VBD forced)
          (NP
            (NP (NNS thousands))
            (PP (IN of)
              (NP (NNS people))))
          (S
            (VP (TO to)
              (VP
                (VP (VB sleep)
                  (PP (IN in)
                    (NP (PRP$ their) (NNS offices))))
                (CC or)
                (VP (VB walk)
                  (NP (NN home))
                  (PP (IN during)
                    (NP (DT the) (NN night)))))))))
    (, ,)
    (NP (NNS officials))
    (VP (VBD said)
      (NP (DT the) (NN today)))
    (. .)))
```

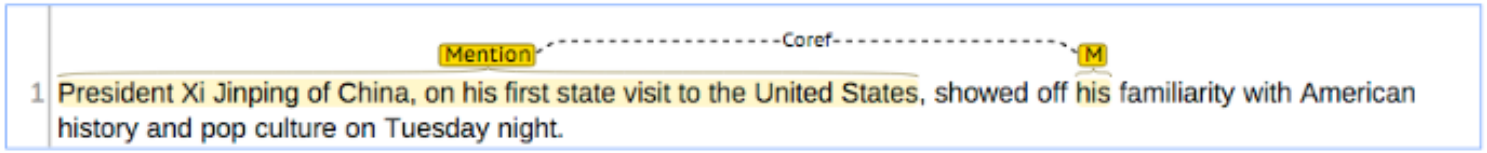
```
det(rain-3, The-1)
amod(rain-3, strongest-2)
nsubj(shut-8, rain-3)
nsubj(snapped-16, rain-3)
nsubj(closed-20, rain-3)
nsubj(forced-23, rain-3)
advmod(recorded-5, ever-4)
partmod(rain-3, recorded-5)
prep_in(recorded-5, India-7)
ccomp(said-40, shut-8)
prt(shut-8, down-9)
det(hub-12, the-10)
amod(hub-12, financial-11)
dobj(shut-8, hub-12)
prep_of(hub-12, Mumbai-14)
conj_and(shut-8, snapped-16)
ccomp(said-40, snapped-16)
nn(lines-18, communication-17)
dobj(snapped-16, lines-18)
conj_and(shut-8, closed-20)
ccomp(said-40, closed-20)
dobj(closed-20, airports-21)
conj_and(shut-8, forced-23)
ccomp(said-40, forced-23)
dobj(forced-23, thousands-24)
prep_of(thousands-24, people-26)
aux(sleep-28, to-27)
xcomp(forced-23, sleep-28)
poss(offices-31, their-30)
prep_in(sleep-28, offices-31)
xcomp(forced-23, walk-33)
conj_or(sleep-28, walk-33)
dobj(walk-33, home-34)
det(night-37, the-36)
prep_during(walk-33, night-37)
nsubj(said-40, officials-39)
root(ROOT-0, said-40)
tmod(said-40, today-41)
```

- This output was generated with the command:
- java -mx200m edu.stanford.nlp.parser.lexparser.LexicalizedParser -retainTMPSubcategories -outputFormat "wordsAndTags,penn,typedDependencies" englishPCFG.ser.gz mumbai.txt

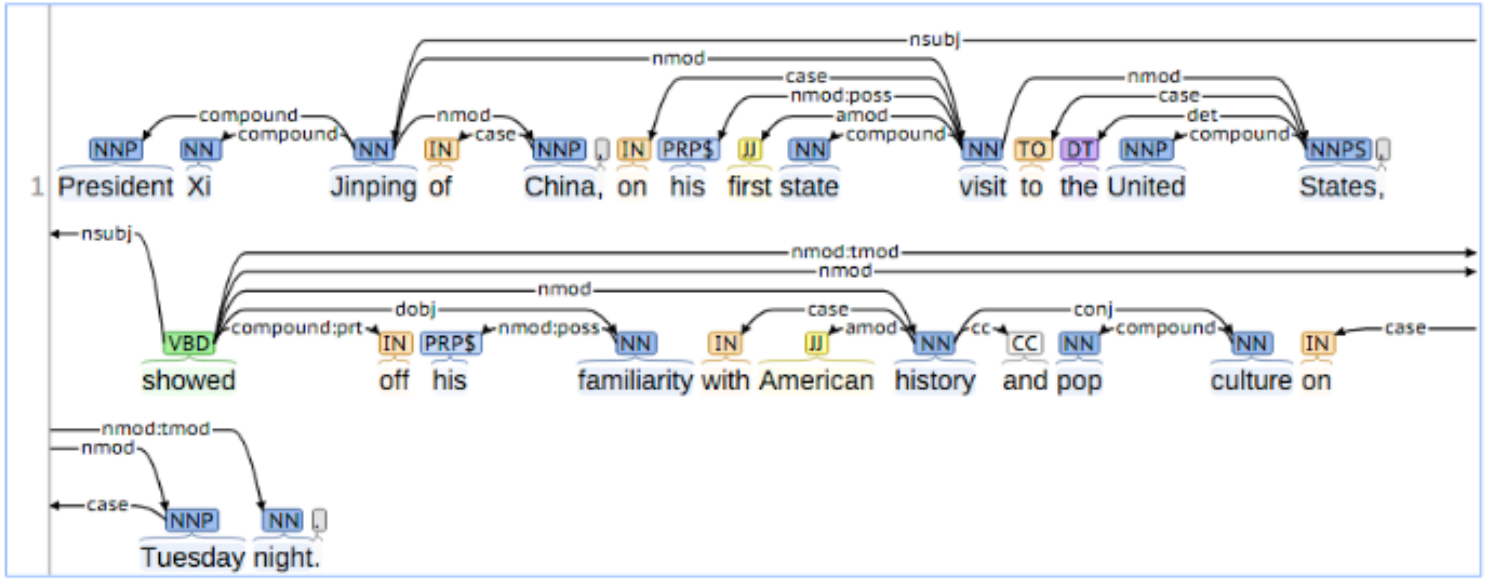
Named Entity Recognition:



Coreference:



Basic Dependencies:



NLTK: NLP Tool Kit

- Natural Language Toolkit
  - http://www.nltk.org/
- Suite of classes for several NLP tasks
  - Parsing, POS tagging, classifiers…
- Easy-to-use interfaces to over 50 corpora and lexical resources
  - http://www.nltk.org/nltk_data/

Installing NLTK

- http://www.nltk.org/install.html

- Mac/Unix
  1. Install Setuptools
  2. Install Pip
  3. Install Numpy(optional)
  4. Install PyYAML and NLTK
  5. Test installation

- Windows
  1. Install Python
  2. Install Numpy(optional)
  3. Install Setuptools
  4. Install Pip
  5. Install PyYAML and NLTK
  6. Test installation

Modules

- The NLTK modules include:
  - nltk.token : processing individual elements of text, such as words or sentences
  - nltk.tagger : tagging tokens with supplemental information, such as POS or wordnet sense tags
  - nltk.parser : high-level interface for parsing texts
  - nltk.classify : classify text into categories
  - nltk.corpus : access (tagged)corpus data

  …….

- http://www.nltk.org/py-modindex.html#

Example: POS tagging

POS_tagging.py - C:₩Users₩jinwoo₩Desktop₩python_ex₩POS_tagging.py

File Edit Format Run Options Windows Help

```
from nltk import pos_tag,word_tokenize
sentent1 = '''
this is a demo that will show you how to
detects parts of speech with little effort using NLTK!'
'''
tokenized_sent = word_tokenize(sentent1)
print pos_tag(tokenized_sent)
```

Ln: 4 Col: 0

Python Shell

File Edit Shell Debug Options Windows Help

```
alicia ambiguity anger bathos blunder boards commoners correspondence
doors english france incubi it james organ oriental ours outspoken
paper perpetration
>>> ================================ RESTART ================================
>>>
[('this', 'DT'), ('is', 'VBZ'), ('a', 'DT'), ('demo', 'NN'), ('that', 'WDT'), ('
will', 'MD'), ('show', 'VB'), ('you', 'PRP'), ('how', 'WRB'), ('to', 'TO'), ('de
tects', 'NNS'), ('parts', 'NNS'), ('of', 'IN'), ('speech', 'NN'), ('with', 'IN')
, ('little', 'JJ'), ('effort', 'NN'), ('using', 'VBG'), ('NLTK', 'NNP'), ('!', '
.'), ("'", "''")]
>>>
```

Ln: 214 Col: 44

Example: Parsing

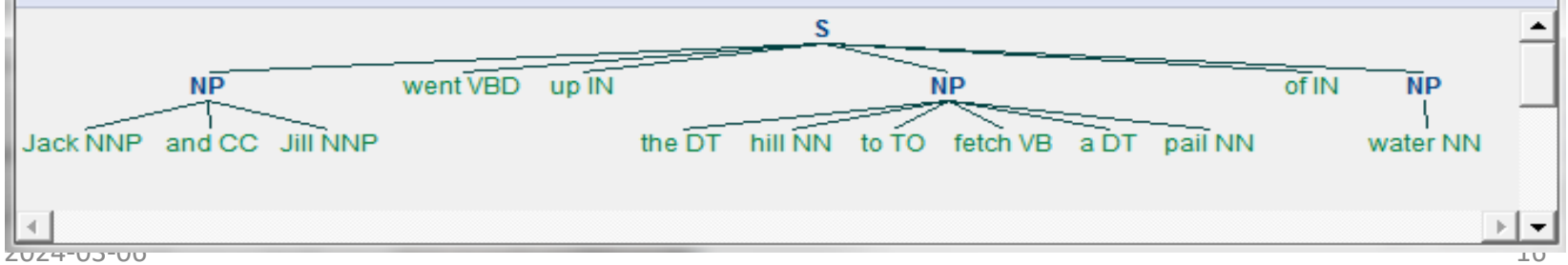
nounphrase_chunker.py - C:₩Users₩jinwoo₩Desktop₩python_ex₩nounphrase_chunker.py

File Edit Format Run Options Windows Help

```
from nltk.chunk import *
from nltk.chunk.util import *
from nltk.chunk.regexp import *
from nltk import word_tokenize
from nltk import pos_tag

text = '''
Jack and Jill went up the hill to fetch a pail of water
'''

tokens = pos_tag(word_tokenize(text))

chunk = ChunkRule("<.*>+", "Chunk all the text")
chink = ChinkRule("<VBD|IN|\.>", "Leave verbs and prepositions out of this")
split = SplitRule("<DT><NN>", "<DT><NN>","Chunk on sequences of determiner+noun phr

chunker = RegexpChunkParser([chunk, chink, split],chunk_node='NP')
chunked = chunker.parse(tokens)
chunked.draw()
```

NLTK

File Zoom



Example: WordNet

similarity.py - C:₩Users₩jinwoo₩Desktop₩python_ex₩similarity.py

File Edit Format Run Options Windows Help

```python
from nltk.corpus import wordnet as wn

Aword = 'language'
Bword = 'barrier'

synsetsA = wn.synsets(Aword)
synsetsB = wn.synsets(Bword)

similars = []

for sseta in synsetsA:
        for ssetb in synsetsB:
                path_similarity = sseta.path_similarity(ssetb)

                if path_similarity is not None:
                        similars.append({
                                'path':path_similarity,
                                'wordA':sseta,
                                'wordB':ssetb,
                                'wordA_definition':sseta.definition,
                                'wordB_definition':ssetb.definition
                        })

similars = sorted(similars, key=lambda item: item['path'],reverse=True)

for item in similars:
        print item['wordA'],"\n",item['wordA_definition']
        print item['wordB'],"\n",item['wordB_definition']
        print 'Path similarity - ',item['path'],"\n"
```

Ln: 30 Col: 0

Python Shell

File Edit Shell Debug Options Windows Help

```
>>> ================================ RESTART ================================
>>>
Synset('linguistic_process.n.02')
the cognitive processes involved in producing and understanding linguistic commu
nication
Synset('barrier.n.02')
any condition that makes it difficult to make progress or to achieve an objectiv
e
Path similarity -  0.111111111111

Synset('language.n.05')
the mental faculty or power of vocal communication
Synset('barrier.n.02')
any condition that makes it difficult to make progress or to achieve an objectiv
e
Path similarity -  0.111111111111

Synset('language.n.01')
a systematic means of communicating by the use of sounds or conventional symbols
Synset('barrier.n.02')
any condition that makes it difficult to make progress or to achieve an objectiv
e
Path similarity -  0.1

Synset('language.n.01')
a systematic means of communicating by the use of sounds or conventional symbols
Synset('barrier.n.03')
anything serving to maintain separation by obstructing vision or access
Path similarity -  0.1

Synset('language.n.01')
a systematic means of communicating by the use of sounds or conventional symbols
Synset('barrier.n.01')
a structure or object that impedes free movement
Path similarity -  0.090909090909091
```

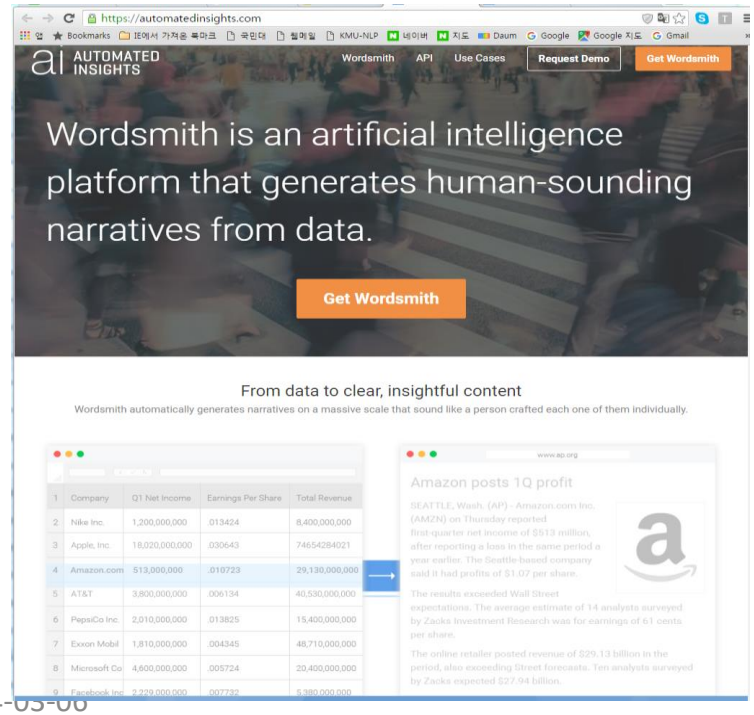
Ln: 431 Col: 35

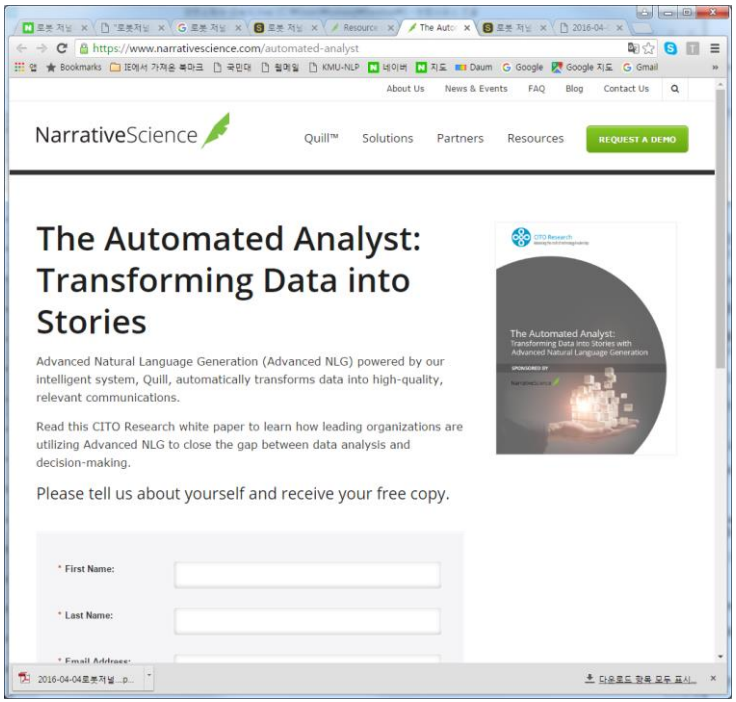
For more details

- NLTK
  - http://www.nltk.org/index.html
- NLTK demo site
  - http://text-processing.com/demo/

NLP Generation

- Robot Journalism: 스포츠, 지진, 교통, 일기예보
  - https://automatedinsights.com/
  - https://www.narrativescience.com/





NLP Generation (cont)

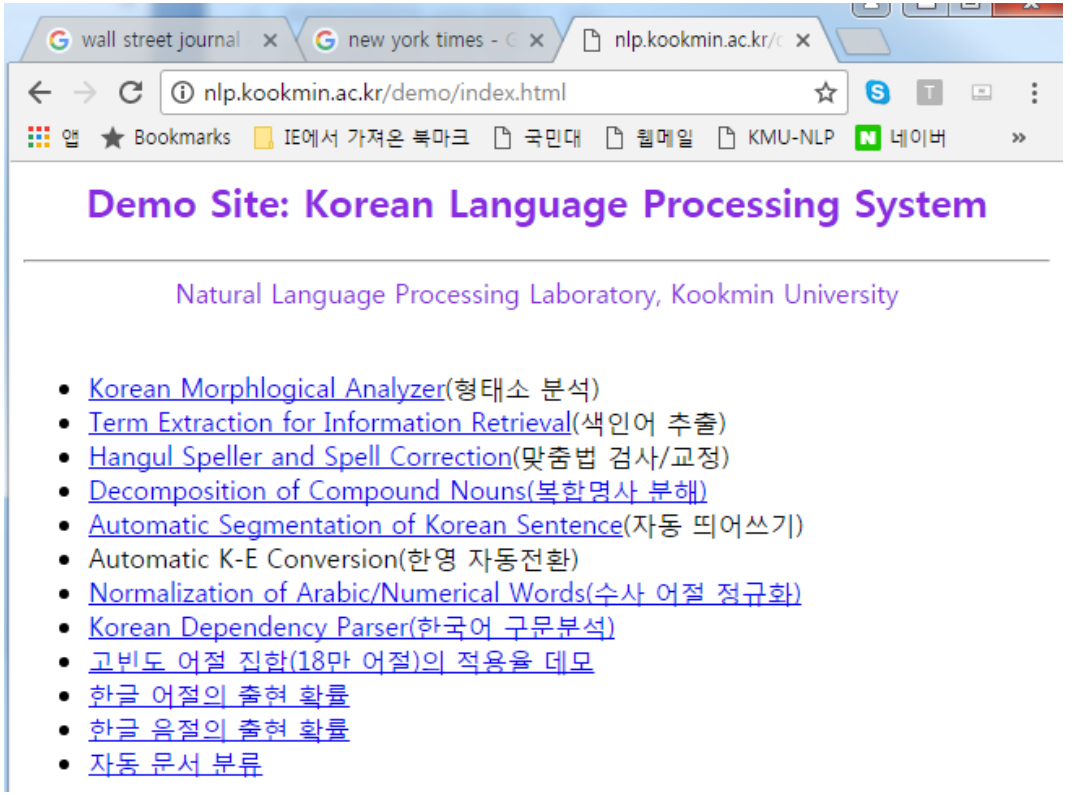
- ChatBot: dialogue analysis and generation
- Pattern match in the new programming languages
  - Scala, Swift, and Wolfram Language

http://nlp.kookmin.ac.kr/
http://cafe.naver.com/nlpkang

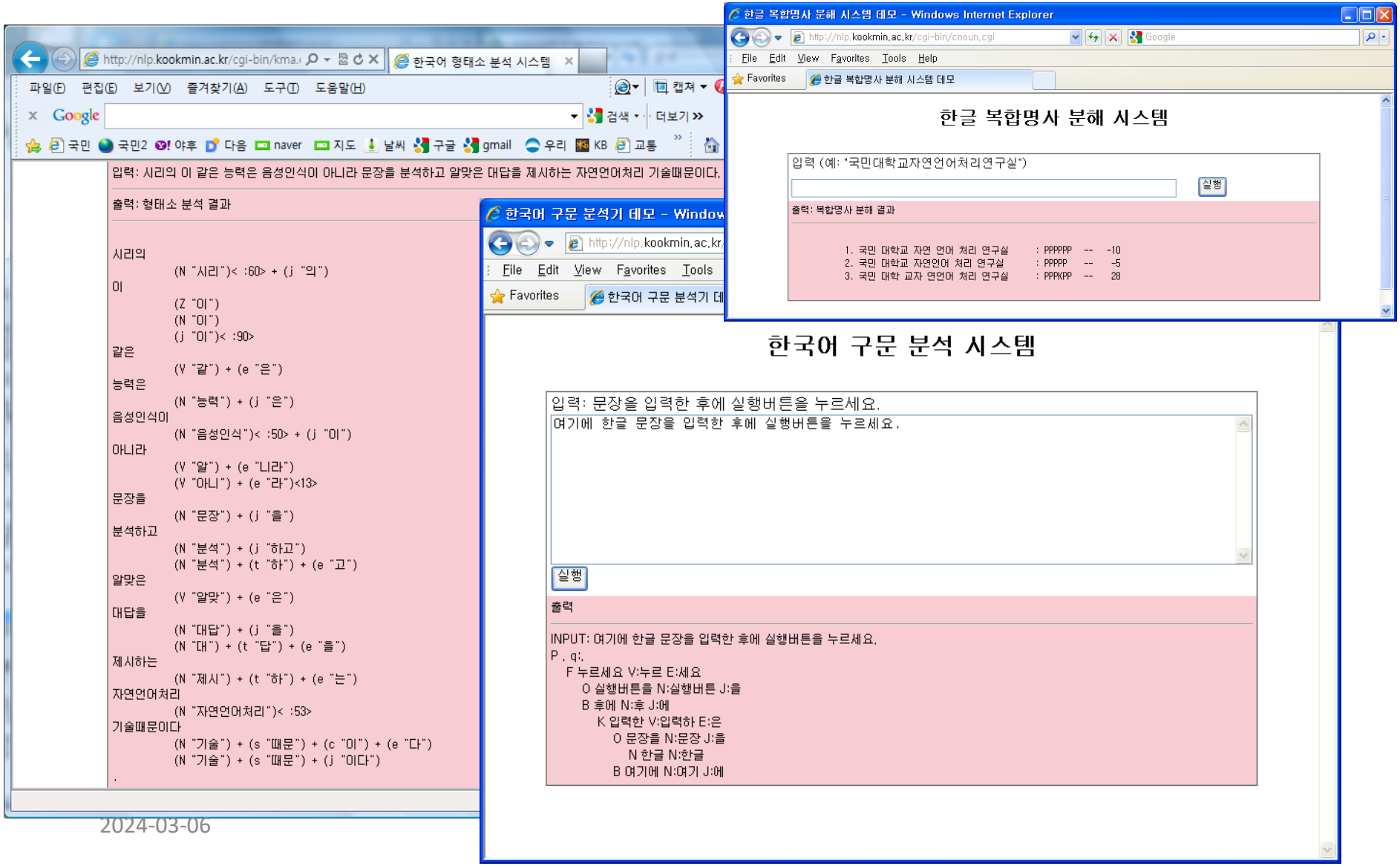
- 한국어 형태소 분석
- 구문 분석
- 색인어 추출 및 가중치 계산

- 복합명사 분해
- 맞춤법 검사 및 교정

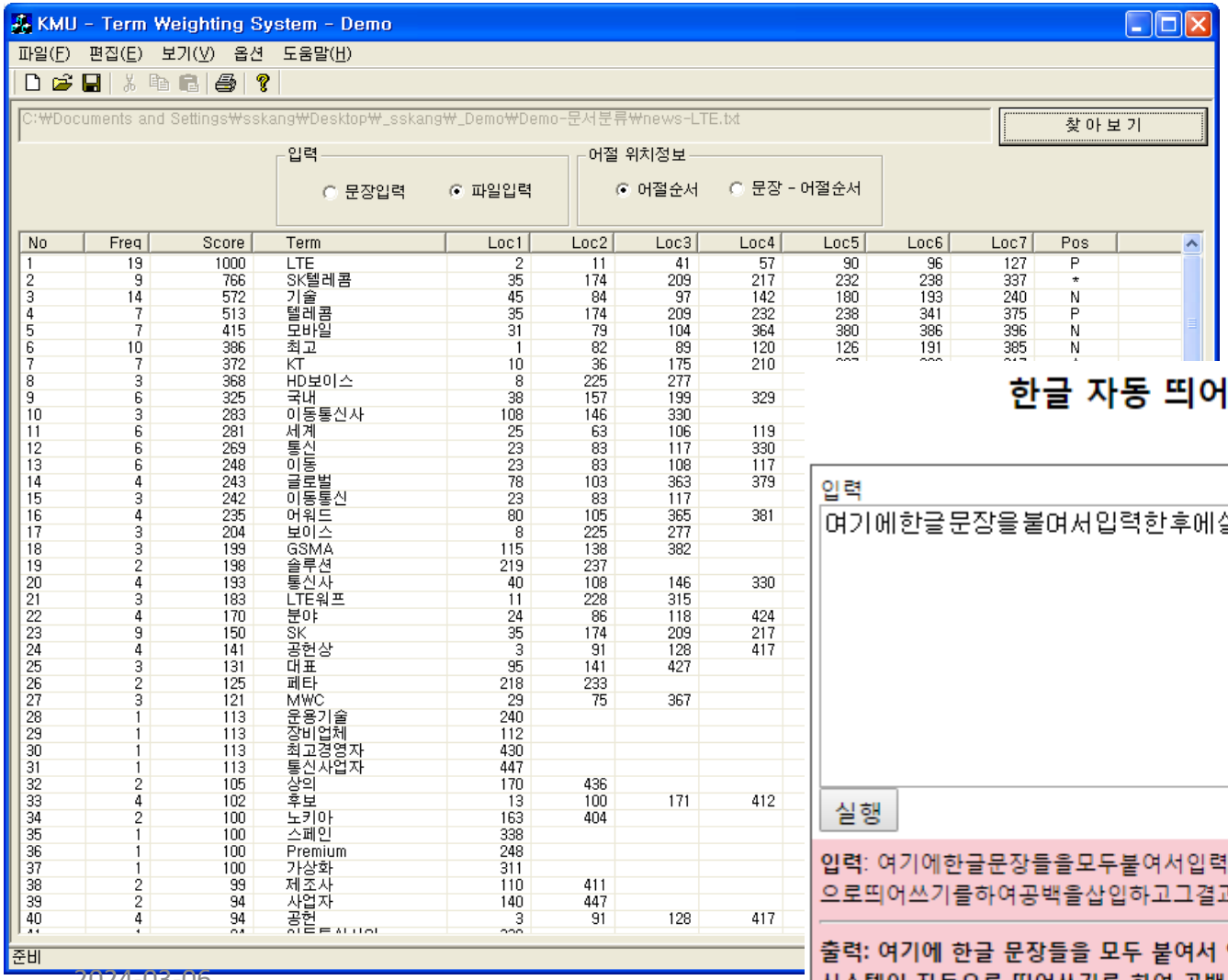
- 자동 문서 분류
- 자동 띄어쓰기 등



형태소 분석과 구문분석


한국어 형태소 분석 시스템
입력: 시리의 이 같은 능력은 음성인식이 아니라 문장을 분석하고 알맞은 대답을 제시하는 자연언어처리 기술때문이다.
출력: 형태소 분석 결과
한글 복합명사 분해 시스템 데모 - Windows Internet Explorer
한글 복합명사 분해 시스템
입력 (예: "국민대학교자연언어처리연구실")
실행
출력: 복합명사 분해 결과
한국어 구문 분석기 데모
한국어 구문 분석 시스템
입력: 문장을 입력한 후에 실행버튼을 누르세요.
여기에 한글 문장을 입력한 후에 실행버튼을 누르세요.
실행
출력

문서에서 키워드 추출



No	Freq	Score	Term	Loc1	Loc2	Loc3	Loc4	Loc5	Loc6	Loc7	Pos
1	19	1000	LTE	2	11	41	57	90	96	127	P
2	9	766	SK텔레콤	35	174	209	217	232	238	337	*
3	14	572	기술	45	84	97	142	180	193	240	N
4	7	513	텔레콤	35	174	209	232	238	341	375	P
5	7	415	모바일	31	79	104	364	380	386	396	N
6	10	386	최고	1	82	89	120	126	191	385	N
7	7	372	KT	10	36	175	210				
8	3	368	HD보이스	8	225	277					
9	6	325	국내	38	157	199	329				
10	3	283	이동통신사	108	146	330					
11	6	281	세계	25	63	106	119				
12	6	269	통신	23	83	117	330				
13	6	248	이동	23	83	108	117				
14	4	243	글로벌	78	103	363	379				
15	3	242	이동통신	23	83	117					
16	4	235	어워드	80	105	365	381				
17	3	204	보이스	8	225	277					
18	3	199	GSMA	115	138	382					
19	2	198	솔루션	219	237						
20	4	193	통신사	40	108	146	330				
21	3	183	LTE워프	11	228	315					
22	4	170	분야	24	86	118	424				
23	9	150	SK	35	174	209	217				
24	4	141	공헌상	3	91	128	417				
25	3	131	대표	95	141	427					
26	2	125	페타	218	233						
27	3	121	MWC	29	75	367					
28	1	113	운용기술	240							
29	1	113	장비업체	112							
30	1	113	최고경영자	430							
31	1	113	통신사업자	447							
32	2	105	상의	170	436						
33	4	102	후보	13	100	171	412				
34	2	100	노키아	163	404						
35	1	100	스페인	338							
36	1	100	Premium	248							
37	1	100	가상화	311							
38	2	99	제조사	110	411						
39	2	94	사업자	140	447						
40	4	94	공헌	3	91	128	417