

한국어분석용 *Package* 설치 *WordCloud, WordNetwork*

12주차

 **KLPLAB** 울산대학교 한국어처리연구실

옥철영

지능형컴퓨터/한국어처리연구실
울산대학교 IT융합학부

❖ 한국언론재단 www.bigkinds.or.kr

BIGKinds Big Data Kinds

기본 검색어를 입력해주세요

10월분 '카드 캐시백' 15... 20건

뉴스분석 기획분석 뉴스보기 빅카인즈 활용 빅카인즈 소개

금일 뉴스 수집 8,507건
전체 뉴스 수집 73,701,218건

오늘의 이슈

뉴스특가

분석대상뉴스 3,177건 · 뉴스클러스터 321건

분석기준 2021.11.15(월) 08:00 ~ 17:00

이슈별

업종사별

나의 관심뉴스

김종인, 윤석열 리브물에 "게기 되면 도울 수도"

56건

코로나19 확진자 추가 발생

56건

고민정 "경희대 수원캠퍼스 분교" 표현에 논란 커져

34건

요소수 거점 주유소 재고 정보, 내일부터 인터넷 공개

29건

경찰, 문답지 배송 등 '수능 경비' 1만여명 투입

27건

전체 뉴스보기 >



매일신문

김종인, 윤 "김박사님 나설 때" 리브물 받아들일 까..." 도와줄 수도" (종합)

野 선대위 인적구성 백지...김종인 원동 가지만 위원 구성은 난항

윤석열-이준석, "김종인 역할할 때" 선대위 참여요청

尹 "김박사님-" 구애에 김종인 "게기가 되면 도울 수도"

윤석열, 김종인에 "공개 구매"...선대위 '밀당' 언제까지?

'이준석은 왜 윤석열에 박수 안 쳤나' 김종인 행사서 포착

尹·李 공개 리브물...김종인 "도와줄 수도"...선대위 구성은 후보...

'쇼 사퇴론' 언급했던 김무성, 윤석열에게 한 조언은

관계도 분석

서울경제

대구일보

중앙일보

SBS

한국경제

서울경제

서울신문

빅데이터 시대의 새로운 뉴스분석서비스

빅카인즈 서비스 안내 >



오늘의 키워드

분석대상 뉴스 6,424건

분석기준 2021.11.15(월) 00:00 ~ 15:00



전체



정치



경제



사회



문화



국제



지역



스포츠



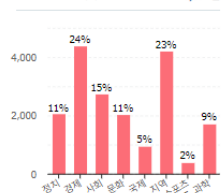
IT 과학



오늘의 뉴스 현황

2021년 11월 15일 (월)

8,507건



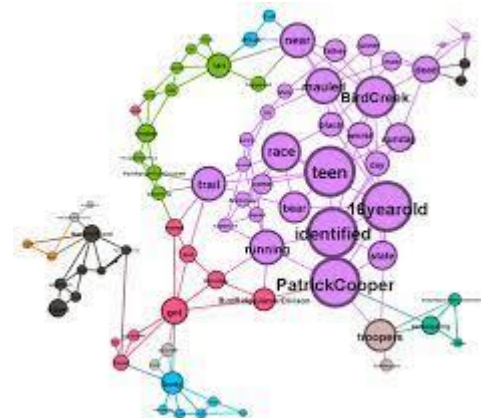
뉴스 분석 방법안내

연도별 뉴스 현황

사용자 매뉴얼 다운로드

- KoNLPy 설치
- UTagger-python 설치
- 형태소분석
- 단어 빈도

- Pytagcloud 설치
- Pytagcloud에서 한글지원
- Wordcloud 설치



- ❖ <https://liveyourit.tistory.com/56> (KoNLPy 윈도우 설치 방법)
- ❖ KoNLPy <http://konlpy.org/ko/v0.5.1/>



KoNLPy

Star 799

KoNLPy is a Python package for Korean natural language processing.

목차

- KoNLPy: 파이썬 한국어 NLP
 - 거인의 어깨 위에 서기
 - 라이선스
 - 참여하기
 - 시작하기
 - 사용하기
 - API

KoNLPy: 파이썬 한국어 NLP

build passing docs passing

KoNLPy("코엔엘파이"라고 읽습니다)는 한국어 정보처리를 위한 파이썬 패키지입니다. 설치 방법은 [이 곳을](#) 참고해주세요.

NLP를 처음 시작하시는 분들은 [시작하기](#) 에서 가볍게 기본 지식을 습득할 수 있으며, KoNLPy의 사용법 가이드는 [사용하기](#), 각 모듈의 상세사항은 [API](#) 문서에서 보실 수 있습니다.

```
>>> from konlpy.tag import Kkma
>>> from konlpy.utils import pprint
>>> kkma = Kkma()
>>> pprint(kkma.sentences(u'네, 안녕하세요. 반갑습니다.'))
[네, 안녕하세요...,
반갑습니다.]
>>> pprint(kkma.nouns(u'질문이나 건의사항은 깃헙 이슈 트래커에 남겨주세요.'))
[질문,
건의,
건의사항,
사항,
깃헙,
이슈,
트래커]
>>> pprint(kkma.pos(u'오류보고는 실행환경, 에러메시지와함께 설명을 최대한상세히!^^'))
[(오류, NNG),
(보고, NNG),
(는, JX),
```

1. MS VisualC++ Build Tools 패키지 설치

- 시각화용
- <https://visualstudio.microsoft.com/ko/downloads/>

Microsoft | Visual Studio 제품 다운로드 구입 지원 구독자 액세스 무료 Visual Studio

Microsoft 전체 검색 Sign in

다운로드



Visual Studio 2022

버전 17.0

Windows에서 .NET 및 C++ 개발자를 위한 최고의 포괄적인 IDE입니다. 소프트웨어 개발의 모든 단계를 향상시키고 개선할 수 있는 다양한 도구와 기능이 완벽하게 포함되어 있습니다.

[릴리스 정보](#) > [버전 비교](#) > [오프라인 설치 방법](#) >

커뮤니티

강력한 IDE. 학생, 오픈 소스 제공자 및 개인을 위해 무료로 제공

무료 다운로드

Professional

소규모 팀에 적합한 Professional IDE

평가판

Enterprise

모든 규모의 팀을 위한 확장성 뛰어난 통합형 솔루션

평가판

미리 보기

아직 기본 릴리스에 없는 최신 기능에 조기에 액세스할 수 있습니다.

[자세한 정보](#) >

[릴리스 정보](#) >

2. Jpype1 설치

- `conda install -c conda-forge jpype1`

```
C:\Users\woky\Anaconda3>conda install -c conda-forge jpype1
Collecting package metadata: done
Solving environment: /
Warning: 8 possible package resolutions (only showing differing packages):
- anaconda::ca-certificates-2018.03.07-0, anaconda::certifi-2018.11.29-py37_0, anaco
.1a-he774522_0
```

- <https://www.lfd.uci.edu/~gohlke/pythonlibs/#jpype>

- OS 및 python version 확인
- 해당 whl 파일 down (anaconda 설치 fold)
- `Pip install Jpype1-1.3.0-cp38-win_amd64.whl`

JPyPe: allows full access to Java class libraries.

[JPyPe1-1.3.0-cp310-cp310-win_amd64.whl](#)

[JPyPe1-1.3.0-cp310-cp310-win32.whl](#)

[JPyPe1-1.3.0-cp39-cp39-win_amd64.whl](#)

[JPyPe1-1.3.0-cp39-cp39-win32.whl](#)

[JPyPe1-1.3.0-cp38-cp38-win_amd64.whl](#)

[JPyPe1-1.3.0-cp38-cp38-win32.whl](#)

[JPyPe1-1.3.0-cp37-cp37m-win_amd64.whl](#)

[JPyPe1-1.3.0-cp37-cp37m-win32.whl](#)

3. Java SDK 다운로드 페이지에서 JDK 다운로드

- <https://www.oracle.com/java/technologies/downloads/#jdk17-windows>

Java SE Development Kit 17.0.1 downloads

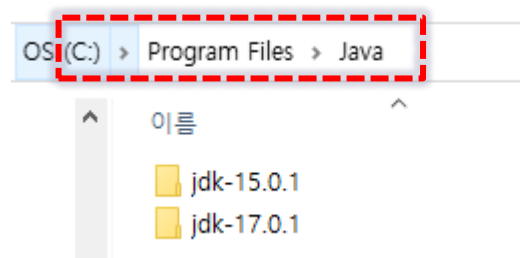
Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building apps and components using the Java programming language.

The JDK includes tools for developing and testing programs written in the Java programming language and running on the Java platform.

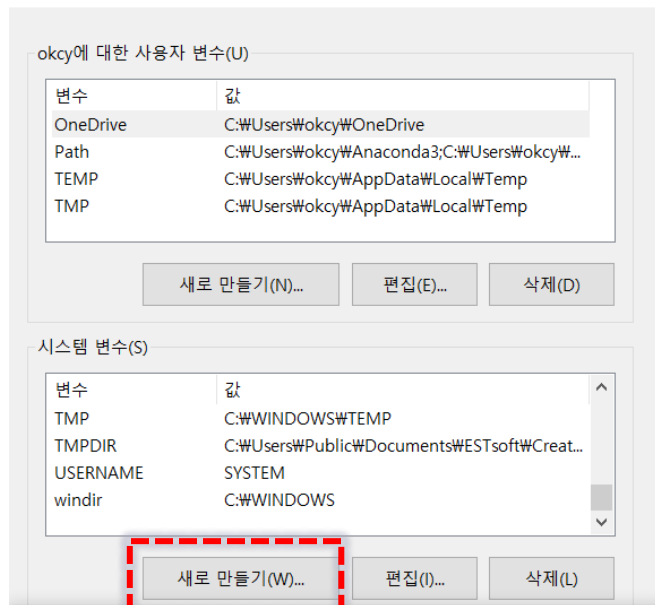
Linux macOS **Windows**

Product/file description	File size	Download
x64 Compressed Archive	170.66 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.zip (sha256 ↗)
x64 Installer	152 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.exe (sha256 ↗)
x64 MSI Installer	150.89 MB	https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi (sha256 ↗)

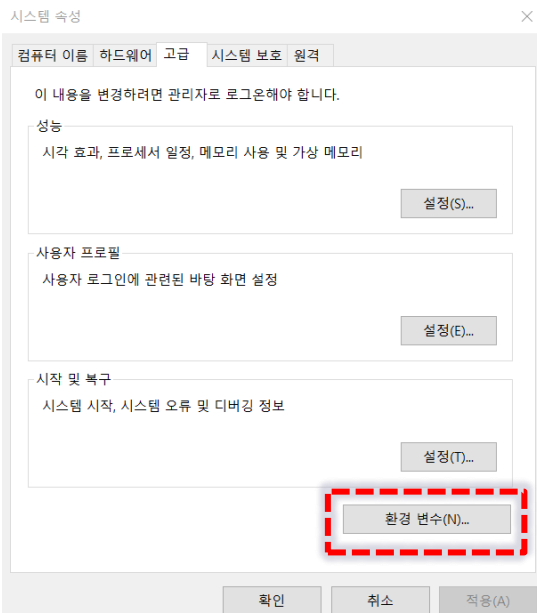
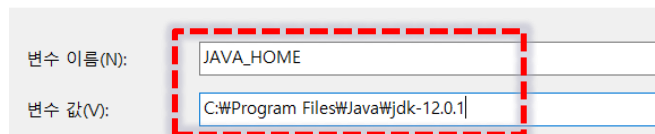
❖ JDK 설치, JAVA_HOME 설정



환경 변수



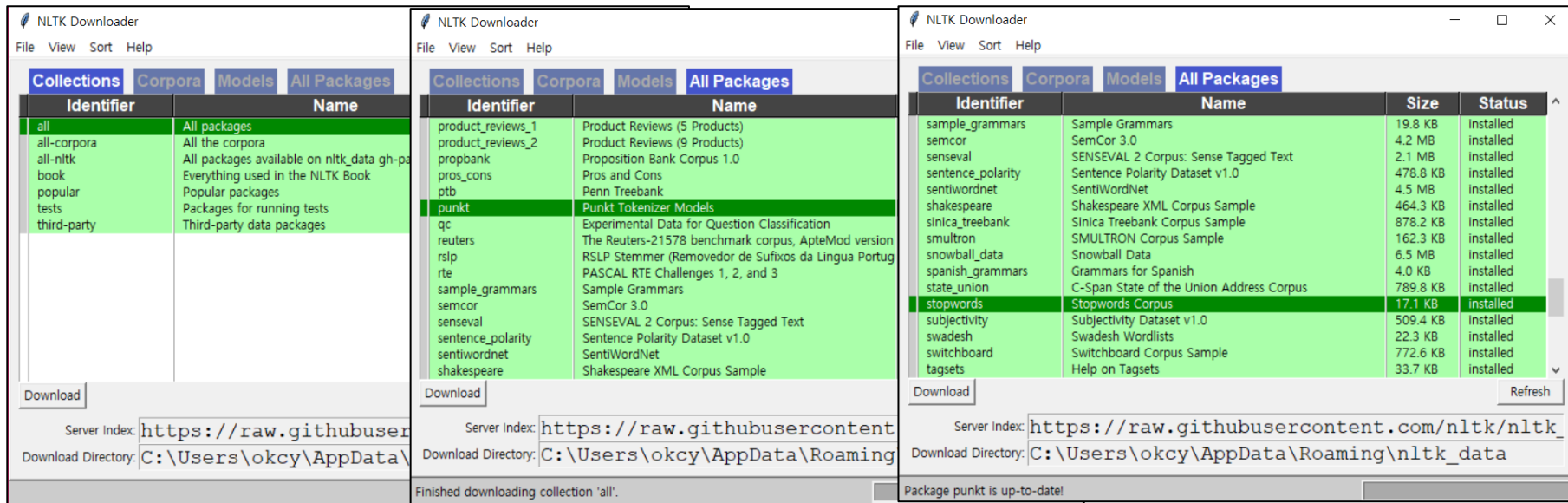
새 시스템 변수



4. NLTK 설치

■ Python 실행 후

- `import nltk`
- `nltk.download()` 후 `stopword`, `punkt` 다운로드



The following table represents the data shown in the 'All Packages' tab of the NLTK Downloader interface:

Identifier	Name	Size	Status
all	All packages		
all-corpora	All the corpora		
all-nltk	All packages available on nltk_data gh-pa		
book	Everything used in the NLTK Book		
popular	Popular packages		
tests	Packages for running tests		
third-party	Third-party data packages		
product_reviews_1	Product Reviews (5 Products)		
product_reviews_2	Product Reviews (9 Products)		
propank	Proposition Bank Corpus 1.0		
pros_cons	Pros and Cons		
ptb	Penn Treebank		
punkt	Punkt Tokenizer Models		
qc	Experimental Data for Question Classification		
reuters	The Reuters-21578 benchmark corpus, AptMod version		
rsdp	RSLP Stemmer (Removedor de Sufixos da Lingua Portug		
rte	PASCAL RTE Challenges 1, 2, and 3		
sample_grammars	Sample Grammars		
semcor	SemCor 3.0		
senseval	SENSEVAL 2 Corpus: Sense Tagged Text		
sentence_polarity	Sentence Polarity Dataset v1.0		
sentiwordnet	SentiWordNet		
shakespeare	Shakespeare XML Corpus Sample		
sample_grammars	Sample Grammars	19.8 KB	installed
semcor	SemCor 3.0	4.2 MB	installed
senseval	SENSEVAL 2 Corpus: Sense Tagged Text	2.1 MB	installed
sentence_polarity	Sentence Polarity Dataset v1.0	478.8 KB	installed
sentiwordnet	SentiWordNet	4.5 MB	installed
shakespeare	Shakespeare XML Corpus Sample	464.3 KB	installed
sinica_treebank	Sinica Treebank Corpus Sample	878.2 KB	installed
smultron	SMULTRON Corpus Sample	162.3 KB	installed
snowball_data	Snowball Data	6.5 MB	installed
spanish_grammars	Grammars for Spanish	4.0 KB	installed
state_union	C-Span State of the Union Address Corpus	789.8 KB	installed
stopwords	Stopwords Corpus	17.1 KB	installed
subjectivity	Subjectivity Dataset v1.0	509.4 KB	installed
swadesh	Swadesh Wordlists	22.3 KB	installed
switchboard	Switchboard Corpus Sample	772.6 KB	installed
tagsets	Help on Tagsets	33.7 KB	installed

Server Index: https://raw.githubusercontent.com/nltk/nltk_data/gh-pages/index.html

Download Directory: C:\Users\okcy\AppData\Roaming\nltk_data

Finished downloading collection 'all'.

Package punkt is up-to-date!

5. KoNLPy 설치

- pip install konlpy

```
C:\Users\Wokcy>pip install konlpy
Requirement already satisfied: konlpy in c:\w\anaconda3\lib\site-packages (0.5.2)
Requirement already satisfied: colorama in c:\w\anaconda3\lib\site-packages (from konlpy) (0.4.4)
Requirement already satisfied: tweepy>=3.7.0 in c:\w\anaconda3\lib\site-packages (from konlpy) (3.9.0)
Collecting beautifulsoup4==4.6.0
  Using cached beautifulsoup4-4.6.0-py3-none-any.whl (86 kB)
Requirement already satisfied: lxml>=4.1.0 in c:\w\anaconda3\lib\site-packages (from konlpy) (4.6.3)
Requirement already satisfied: numpy>=1.6 in c:\w\anaconda3\lib\site-packages (from konlpy) (1.19.5)
Requirement already satisfied: JPype1>=0.7.0 in c:\w\anaconda3\lib\site-packages (from konlpy) (1.3.0)
Requirement already satisfied: requests-oauthlib>=0.7.0 in c:\w\anaconda3\lib\site-packages (from tweepy>=3.7.0->konlpy) (1.3.0)
Requirement already satisfied: six>=1.10.0 in c:\w\anaconda3\lib\site-packages (from tweepy>=3.7.0->konlpy) (1.16.0)
Requirement already satisfied: requests[socks]>=2.11.1 in c:\w\anaconda3\lib\site-packages (from tweepy>=3.7.0->konlpy) (2.25.1)
Requirement already satisfied: oauthlib>=3.0.0 in c:\w\anaconda3\lib\site-packages (from requests-oauthlib>=0.7.0->tweepy>=3.7.0->konlpy) (3.1.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\w\anaconda3\lib\site-packages (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy) (1.26.6)
Requirement already satisfied: idna<3,>=2.5 in c:\w\anaconda3\lib\site-packages (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in c:\w\anaconda3\lib\site-packages (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy) (2021.10.8)
Requirement already satisfied: chardet<5,>=3.0.2 in c:\w\anaconda3\lib\site-packages (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy) (3.0.4)
Requirement already satisfied: PySocks!=1.5.7,>=1.5.6 in c:\w\anaconda3\lib\site-packages (from requests[socks]>=2.11.1->tweepy>=3.7.0->konlpy) (1.7.1)
Installing collected packages: beautifulsoup4
  Attempting uninstall: beautifulsoup4
    Found existing installation: beautifulsoup4 4.10.0
    Uninstalling beautifulsoup4-4.10.0:
      Successfully uninstalled beautifulsoup4-4.10.0
Successfully installed beautifulsoup4-4.6.0
```

- ✓ konlpy 0.5.2 requires beautifulsoup4==4.6.0
- ✓ JPype 버전과 파이썬 버전이 맞지 않는 경우 오류

6. Wordcloud 설치

- pip install wordcloud, conda install -c conda-forge wordcloud
- pip install pytagcloud
- pip install pygame

```
C:\Users\okcy\Anaconda3>pip install wordcloud
Collecting wordcloud
  Using cached https://files.pythonhosted.org/packages/23/4e/1254d26ce5d36facdcbb5820e7e434328aed68e99938c75c9d4e2fee5efb/wordcloud-1.5.0-cp37-cp37m-win_amd64.whl
Requirement already satisfied: numpy>=1.6.1 in c:\users\okcy\anaconda3\lib\site-packages (from wordcloud) (1.15.4)
Requirement already satisfied: pillow in c:\users\okcy\anaconda3\lib\site-packages (from wordcloud) (5.3.0)
Installing collected packages: wordcloud
Successfully installed wordcloud-1.5.0
```

8. Gensim 설치 (word2vector 용)

- Word embedding(Word2vec, doc2vec 용)
- pip install gensim

```
C:\Users\okcy\Anaconda3>pip install gensim
Collecting gensim
  Downloading https://files.pythonhosted.org/packages/81/80/858ef502e80baa6384b75fd5c89f01074b791a13b830487f9e25bdce50ec/gensim-3.7.3.tar.gz (23.4MB)
    | 23.4MB 1.3MB/s
Requirement already satisfied: numpy>=1.11.3 in c:\users\okcy\anaconda3\lib\site-packages (from gensim) (1.15.4)
Requirement already satisfied: scipy>=0.18.1 in c:\users\okcy\anaconda3\lib\site-packages (from gensim) (1.1.0)
Requirement already satisfied: six>=1.5.0 in c:\users\okcy\anaconda3\lib\site-packages (from gensim) (1.12.0)
```

12_1_한국어분석_KoNLPy.ipynb

❖ 울산대학교 한국어정보처리연구실 홈페이지 <http://klplab.ulsan.ac.kr>

다운로드 링크 모음

- 아래 태깅프로그램(윈도우즈용) 및 오픈소스는 교육 및 연구용으로 제한없이 사용할 수 있습니다.
- UTagger 라이브러리(윈도우즈/리눅스, C/C++/C#/JAVA/Python3), UWordMap API(C/C++/C#/JAVA/Python3), 말뭉치 등은 기술이전(연구용 무료, 상업용 유료)이 필요함
- UTagger 2011: 형태소분석및동형이의어태깅
- UTagger 2013: 형태소분석및동형이의어태깅
- UTagger-TCM 2016: 형태소분석및동형이의어태깅
- UTagger-TCM 2018: : 형태소분석및동형이의어태깅(100어절당 1초 딜레이)
- UTagger-TCM 32비트 버전(유지보수 종료됨)
- 유태거 오픈소스 2011 다운로드 페이지로 가기 : <http://203.250.77.242:5900/UTaggerSource/UTaggerSourcePage.aspx>
- UTagger-SR 프로그램: 의존관계및의미역 반자동태깅
- UTagger-NE 프로그램: 개체명 반자동태깅
- UTagger 파이썬 공개 2019-2(윈도우) 100어절당 1초의 추가 딜레이가 주어집니다.
- UTagger 파이썬 공개 2018(윈도우, 우분투, 센토스) 100어절당 1초의 추가 딜레이가 주어집니다.
- UCorpus-HG(형태/의미 말뭉치, 18,000천어절): 학습용/테스트용 분리, 18.10.20
- UCorpus-DP/SR(형태소/동형이의어/의존관계/의미역 부착 한별말뭉치 2,000천어절)
- UPropBank(표준국어대사전용언 의미역 90,100 다의어)
- UTagger 학습사전 버전 18.03.13
- UTagger 학습사전 버전 18.11.03

❖ 다국어 대역어 제공 <http://203.250.77.242:5900/utagger/>

- Android/iPhone app. store “한통이” (Han-Tong-E)

서술형 명사 용언 분석 ☒ 통합 ☐ 분리
한줄 한어절 출력 ☐ 사용안함 ☒ 사용함
의미 매핑 정보 ☐ 출력없음 ☒ 의미분별(한자) ☐ 의미범주(명사어휘망)
한글->한자 ☒ 안함 ☐ 한자 변환 ☐ 한글(한자) 병기
한자->한글 ☒ 사용안함 ☐ 사용함
외국어 대역어(뜻풀이) 출력 : ☐ 출력안함 ☒ 영어.English ☐ 일본.日本語 ☐ 프랑스.Français ☐ 스페인.Español ☐ 아랍.Arabic
☐ 몽골.Монгол хэл ☐ 베트남.Tiếng Việt ☐ 태국.ภาษาไทย ☐ 인도네시아.Indonesia ☐ 러시아.Русский ☐ 중국어.中文
show meaning of particle(ex: 을, 를, 이, 가, 예, 다) ☐ on ☒ off

Enter

눈에 미끄러져서 눈을 다쳤다. Re-enter

눈(snow) 에
미끄러지(slip; skid) 어서
눈(eye) 을
다치(be injured) 었 다
눈__040000/NNG+에/JKB 미끄러지__000101/VV+어서/EC 눈__010001/NNG+을/JKO 다치__010001/VV+었/EP+다/EF+./SF

```
1 2 눈에 눈__040000/NNG + 에/JKB
2 4 미끄러져서 미끄러지__000101/VV + 어서/EC
3 4 눈을 눈__010001/NNG + 을/JKO
4 다쳤다. 다치__010001/VV + 었/EP + 다/EF + ./SF
```

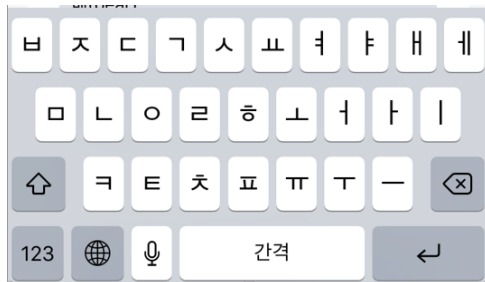
눈에 FWD AF
[1] 눈__01[감각_기관]/NNG+에/JKB --- 0.00174110
[2] 눈__04[얼음의_결정체]/NNG+에/JKB --- 1.74110115
[3] 눈__05[{식물}_초목의_싹]/NNG+에/JKB --- 0.00174110
[4] 눈__03[이어_이른_구멍]/NNG+에/JKB --- 0.00174110

미끄러져서 FWD EF
[1] 미끄러지/VV+어서/EC --- 58.95871446

눈을 FWD AF
[1] 눈__01[감각_기관]/NNG+을/JKO --- 5.22330344
[2] 눈__04[얼음의_결정체]/NNG+을/JKO --- 0.00017411
[3] 눈__05[{식물}_초목의_싹]/NNG+을/JKO --- 0.00017411
[4] 눈__02[눈금]/NNG+을/JKO --- 0.00017411

❖ 다문화 가정을 위한 다국어 어휘대역어 제공 앱

- 부산외국어대학교와 공동연구(연구재단 학제간 융합연구 지원)
- 한통이(HAN-TONG-E): **한**국어를 **통**해서 세계의 모든(**E**very) 언어로



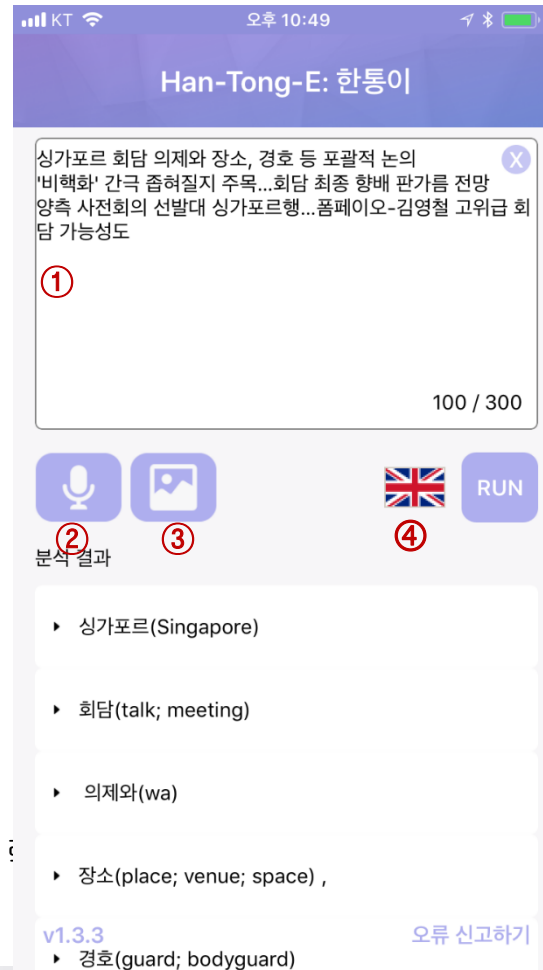
① 자판입력



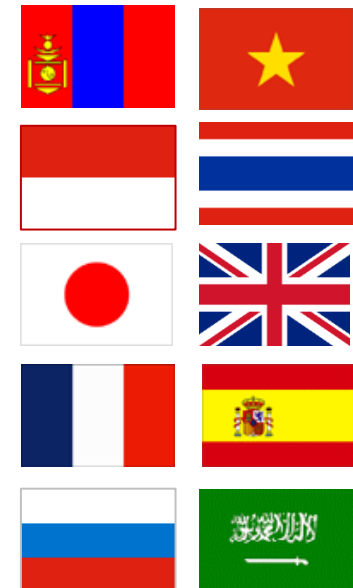
② 음성입력



③ 이미지입력



④ 언어선택



❖ POS(Part-Of-State) TAG

체언	명사	NN	일반명사	NNG
			고유명사	NNP
			의존명사	NNB
	대명사	NP	대명사	NP
	수사	NR	수사	NR
	동사	VV	동사	VV
용언	형용사	VA	형용사	VA
	보조용언	VX	보조용언	VX
	지정사	VC	긍정지정사	VCP
			부정지정사	VCN
수식언	관형사	MM	관형사	MM
	부사	MA	일반부사	MAG
			접속부사	MAJ
독립언	감탄사	IC	감탄사	IC
관계언	격조사	JK	주격조사	JKS
			보격조사	JKC
			관형격조사	JKG
			목적격조사	JKO
			부사격조사	JKB
			호격조사	JKV
			인용격조사	JKQ
	보조사	JX	보조사	JX
	접속조사	JC	접속조사	JC

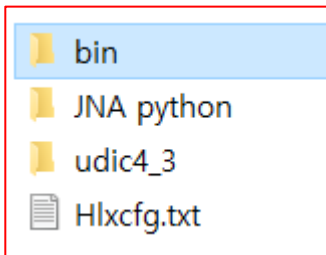
의존형태	어미 E	선어말어미	EP
		종결어미	EF
		연결어미	EC
		명사형전성어미	ETN
		관형형전성어미	ETM
	접두사 XP	체언접두사	XPN
		명사파생접미사	XSN
	접미사 XS	동사파생접미사	XSV
		형용사파생접미사	XSA
	어근 XR	어근	XR

기호	마침표, 물음표, 느낌표	SF
	쉼표, 가운뎃점, 콜로트, 빗금	SP
	따옴표, 괄호표, 줄표	SS
	줄임표	SE
	불임표(물결, 숨김, 빠짐)	SO
	외국어	SL
	한자	SH
	기타기호	SW
	(논리 수학기호, 화폐기호 등)	
	명사추정범주	NF
	용언추정범주	NV
	숫자	SN
분석불능범주	NA	

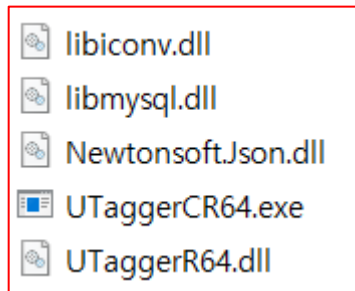


❖ 설치

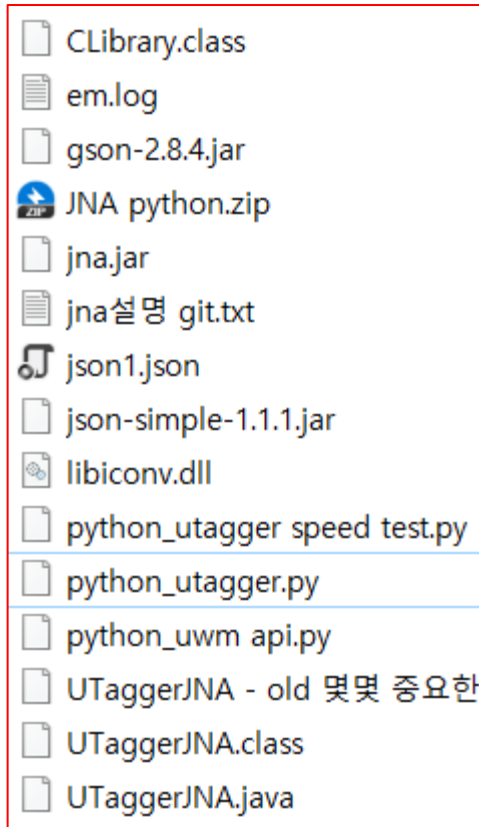
설치 폴더



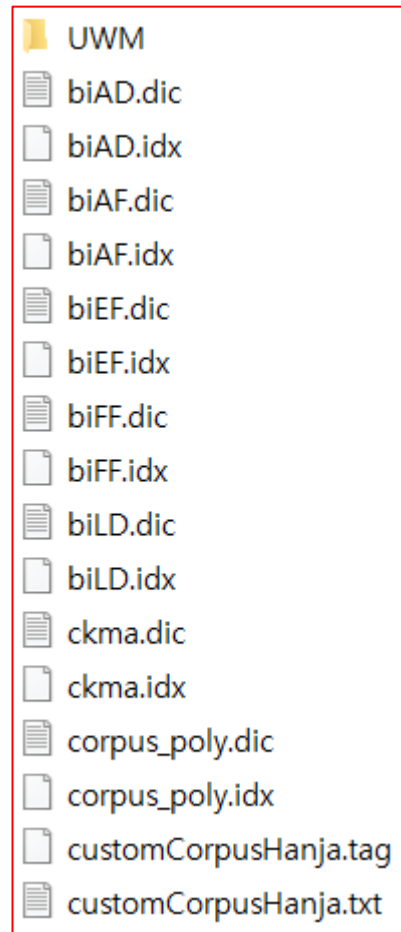
bin 폴더



JNA python 폴더



udic4_3 폴더



❖ python3

❖ 라이브러리는 C++로 작성됨

■ 윈도우/리눅스 지원

```
lib = cdll.LoadLibrary('../bin/UTaggerR64.dll') # dll 로드
...
s = "안녕하세요! 오늘 날씨가 참 좋네요."
rt = tag_line(0, c_wchar_p(s), 0) #분석!
```



❖ python_utagger.py

```
#!/usr/bin/env python
# coding: utf-8 -*-
```

```
import sys
import os
from ctypes import *
from ctypes import cdll
```

#주의! 유태거 dll은 보통 64비트이기 때문에, 파이썬도 64비트를 사용해야 합니다.

```
lib = cdll.LoadLibrary('../bin/UTaggerR64.dll') # dll 로드
```

```
def Init_Utagger():
```

```
    print("python call utagger function")
```

#설정파일의 위치를 절대경로로 구하기. py에서의 현재경로와, dll에서의 현재경로가 다르기 때문.

```
    hl_xpass = os.path.join( os.path.dirname(sys.argv[0]), "../Hlxcfg.txt")
```

```
    hl_xpass = os.path.abspath(hl_xpass)
```

```
    print(hl_xpass)
```

cstr_hlx = c_char_p(hl_xpass.encode('cp949')) #hlxcfg 파일 경로를 cp949로 인코딩.

```
    lib.Global_init2.restype = c_wchar_p #유태거 초기화 함수의 반환자 정의
```

msg = lib.Global_init2(cstr_hlx , 0) # Hlxcfg.txt 위치 지정. 학습 파일 로딩. 오래걸림.

```
    if msg != '': # Hlxcfg.txt와 모든 학습 파일을 읽었는지 확인.
```

```
        print("hlxcfg bug")
```

```
        print(msg)
```

```
        sys.exit(1)
```

```
    lib.newUCMA2.restype = c_wchar_p #ucma 생성 함수의 반환자 정의
    msg = lib.newUCMA2(0) # 유태거의 0번 객체 생성(0~99까지 생성 가능)
```

```
    if msg != '':
        print("newUCMA bug")
        print(msg)
        sys.exit(1)
```

lib.cmaSetNewlineN(0) #유태거 tag_line이 newline을 만들 때 /r/n 이 아니라 /n이 되게 한다.

#lib.cmaSetNewlineN(0) #유태거 tag_line이 newline을 만들 때 /n이 되게 한다.

#dll의 태깅함수 정의. 사용하기 편하게.

```
    tag_line = lib.cma_tag_line_BSP #함수 가져오기
```

```
    tag_line.restype = c_wchar_p #반환자 설정.
```

#uwm api 함수 정의

```
    uwm1_api = lib.DLL_UWM_1
```

```
    uwm1_api.restype = c_wchar_p
```

#대역어 함수 정의

```
    tw_api = lib.cma_tag_target_word_json2
```

```
    tw_api.restype = c_wchar_p
```

```
    arr = [tag_line, uwm1_api, tw_api]
```

```
    return arr
```

```
arr = Init_Utagger()
```

```
tag_line = arr[0]
```

```
uwm1_api = arr[1]
```

```
tw_api = arr[2]
```

print("로딩 성공") #여기까지 오면 유태거 사용가능.



```
# 지역 변수 테스트
s = "나는 사과가 맛있더라."
rt = tag_line(0, c_wchar_p(s), 0) #분석!
print(rt)

# 키보드로 직접 문장을 입력하여 유태거 테스트.
s = ".."
while(len(s)>1):
    s = input('input line = ')
    rt = tag_line(0, c_wchar_p(s), 1) #분석!
    print(rt)

s = ".."
while(len(s)>1):
    s = input('uwm api = ')
    rt = uwm1_api(0, c_wchar_p(s))
    print(rt)

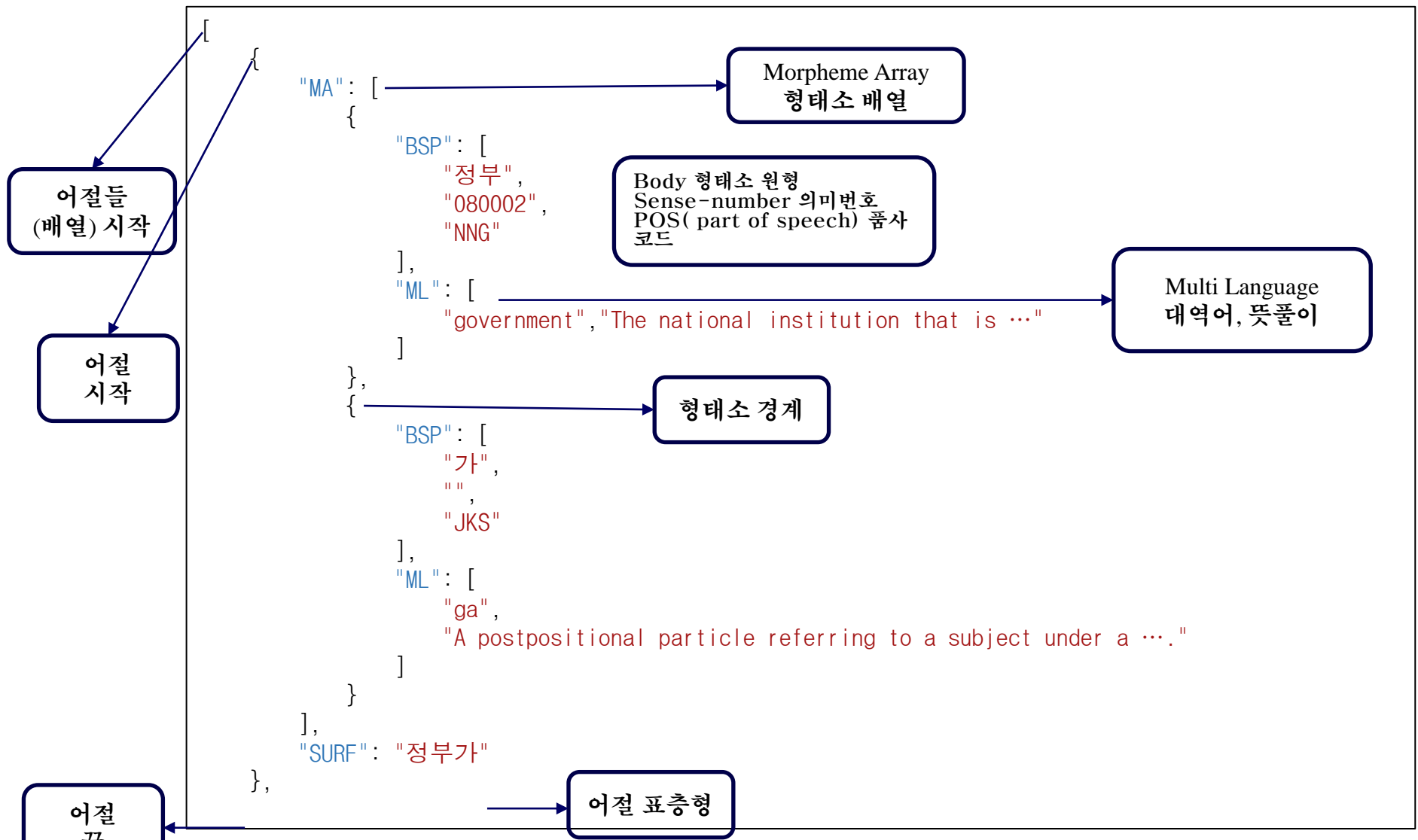
s = ".."
while(len(s)>1):
    s = input('tw json input = ')
    rt = tw_api(0, c_wchar_p(s), 1, 0)
    print(rt)

lib.deleteUCMA(0) # 0번 객체 삭제
lib.Global_release() # 메모리 해제

print("종료")
```



❖ 입력문: “정부가 구성되었다.”



한국어 Word Cloud



❖ pip install pytagcloud

▪ Pygame, simplejson 추가 설치

```
PS F:\ucorpus\perl_programs> pip install pytagcloud
Collecting pytagcloud
  Downloading https://files.pythonhosted.org/packages/a6/36/8626ddead8006c0dbab0a4994737db10463b9c5f837f1fbbd2bbf5bfa27c/pytagcloud-0.3.5.tar.gz (754kB)
    100% |#####| 757kB 2.2MB/s
Building wheels for collected packages: pytagcloud
  Running setup.py bdist_wheel for pytagcloud ... done
  Stored in directory: C:\Users\okcy\AppData\Local\pip\Cache\wheels\08\bf\d4\8899fd82d7f62af5960d17cc59b80edae474380b92828c8c1f
Successfully built pytagcloud
Installing collected packages: pytagcloud
Successfully installed pytagcloud-0.3.5
PS F:\ucorpus\perl_programs>
```

❖ Pytagcloud에서 한글 지원

- Pytagcloud 설치된 폴더로 이동(C:\Anaconda3\Lib\site-packages\pytagcloud\fonts)

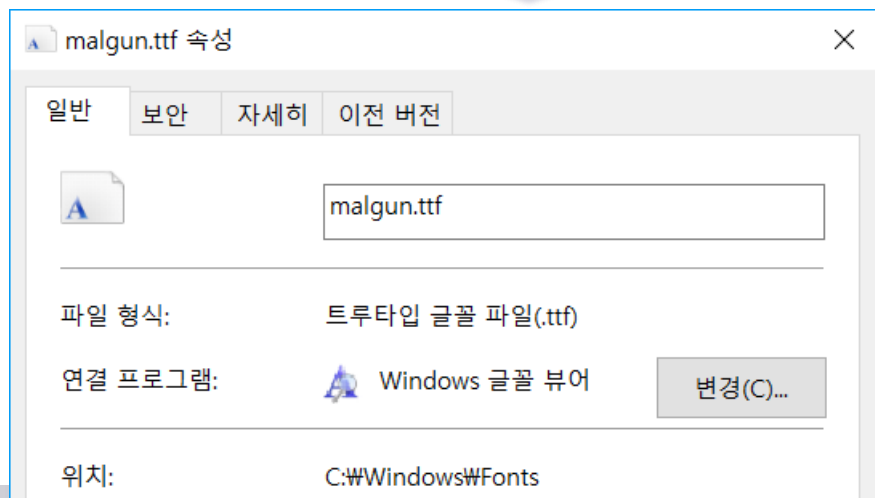
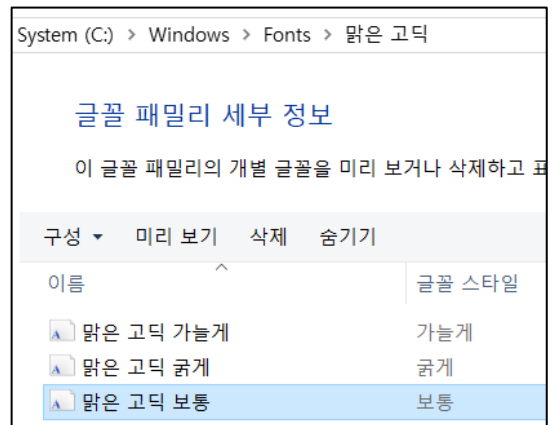
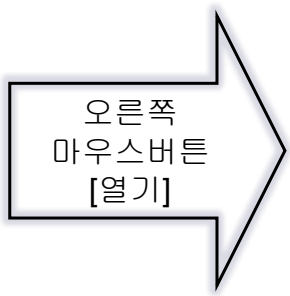
```
[.]          [...]          Cantarell-Regular.ttf
Cardo99s.ttf  Coustard-Regular.ttf  CrimsonText-Roman.ttf
Cuprum.ttf    DroidSans.ttf           fonts.json
IMFeENrm28P.ttf  Inconsolata.ttf        JosefinSansStd-Light.ttf
Lobster.ttf    Molengo-Regular.ttf     Neucha.ttf
Neuton.ttf     nobile.ttf              OldStandard-Regular.ttf
Philosopher.ttf PT_Sans-Regular.ttf     ReenieBeanie.ttf
Tangerine-Regular.ttf Voilkorn-Regular.ttf    YanoneKaffeesatz-Regular.ttf
```

- font.json 파일에 한글 font 정보 추가

```
[↓
{↓
  "name": "korean",↓
  "ttf": "malgun.ttf",↓
  "web": "http://fonts.googleapis.com/css?family=Nobile"↓
},↓
{↓
  "name": "Nobile",↓
  "ttf": "nobile.ttf",↓
  "web": "http://fonts.googleapis.com/css?family=Nobile"↓
},↓
```



Font fold(C:\Windows\Fonts)에 한글 font 파일(malgun.ttf) 복사




```
from matplotlib import font_manager, rc
import matplotlib as plt
import matplotlib.pyplot as plt
```

```
font_location = "c:/Windows/fonts/malgun.ttf "
font_name = font_manager.FontProperties(fname=font_location).get_name()
matplotlib.rc('font', family=font_name)
```

```
plt.plot([1,2,3,4])
plt.xlabel('x축 한글표시')
plt.show()
```



❖ 한국어 분석

- Facebook에서 crawling한 문서 대상
- 문장단위로 형태소분석
- 명사/대명사 빈도

❖ 분석 결과 출력

- 상위 n 개의 단어 막대그래프
- 상위 $n * 3$ 의 단어 대상으로 Word Cloud



Word Network