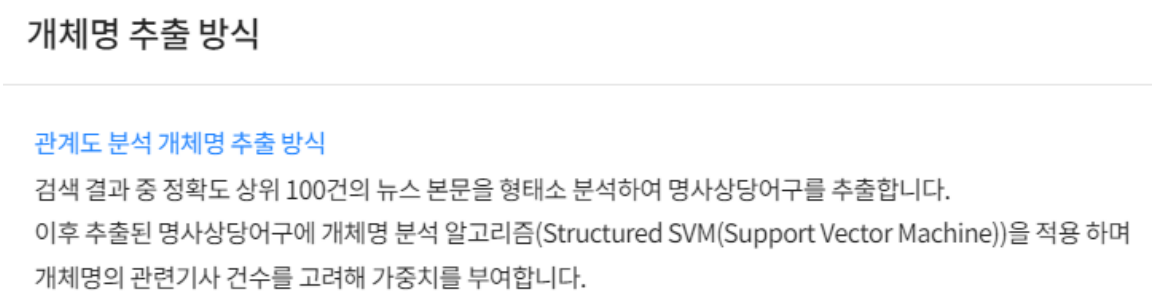
⁉️

# 어떤 알고리즘을 써야할까?

| | |
|---|---|
| Category | |
| DATE | @2022/03/10 |
| Complete | ☐ |
| etc | |
| 열 | |

## 1. 키워드 추출

▼ 개체명 추출방식이란




### 1. 공공인공지능 - 언어 분석 기술 - 형태소 분석 API

https://aiopen.etri.re.kr/guide_wiseNLU.php

- 언어 분석을 위한 6종의 API는 HTTP 기반의 REST API 인터페이스로 JSON 포맷 기반의 입력 및 출력을 지원하며 ETRI에서 제공하는 API Key 인증을 통해 사용할 수 있는 Open API입니다.
- 5,000건/일 (1회 사용시 입력은 1만글자 이하)

#### 자바 예제

```
// 형태소들 중 명사들에 대해서 많이 노출된 순으로 출력 ( 최대 5개 )
            morphemes
```

```
                .stream()
                .filter(morpheme -> {
                    return morpheme.type.equals("NNG") ||
                            morpheme.type.equals("NNP") ||
                            morpheme.type.equals("NNB");
                })
                .limit(5)
                .forEach(morpheme -> {
                    System.out.println("[명사] " + morpheme.text + " ("+morpheme.count+")" );
                });
```

- NNG: 일반명사, NNP: 고유명사, NNB: 의존명사
- NN : 명사

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/42cbfe5e-421f-428e-9905-1f2900137c3f/001.형태소분석_가이드라인.pdf

5페이지

### 5.1. 태그 세트 분류 체계

| 대분류 | 소분류 | 세분류 |
|---|---|---|
| (1) 체언 | 명사(NN) | 일반명사(NNG) |
| | | 고유명사(NNP) |
| | | 의존명사(NNB) |
| | 대명사(NP) | 대명사(NP) |
| | 수사(NR) | 수사(NR) |
| (2) 용언 | 동사(VV) | 동사(VV) |
| | 형용사(VA) | 형용사(VA) |
| | 보조용언(VX) | 보조용언(VX) |
| | 지정사(VC) | 긍정지정사(VCP) |
| | | 부정지정사(VCN) |
| (3) 수식언 | 관형사(MM) | 성상 관형사(MMA) |
| | | 지시 관형사(MMD) |
| | | 수 관형사(MMN) |
| | 부사(MA) | 일반부사(MAG) |
| | | 접속부사(MAJ) |
| (4) 독립언 | 감탄사(IC) | 감탄사(IC) |
| (5) 관계언 | 격조사(JK) | 주격조사(JKS) |
| | | 보격조사(JKC) |
| | | 관형격조사(JKG) |

## 2. 파이썬 오픈소스 라이브러리 - KoNLPy

https://datascienceschool.net/03 machine learning/03.01.02 KoNLPy 한국어 처리 패키지.html

KoNLPy(코엔엘파이라고 읽는다)는 한국어 정보처리를 위한 파이썬 패키지이다.

```
import warnings
warnings.simplefilter("ignore")

import konlpy
konlpy.__version__
```

# 한국어 말뭉치

KoNLPy에서는 대한민국 헌법 말뭉치인 `kolaw`와 국회법안 말뭉치인 `kobill`을 제공한다. 각 말뭉치가 포함하는 파일의 이름은 `fields` 메서드로 알 수 있고 `open` 메서드로 해당 파일의 텍스트를 읽어들인다.

```
from konlpy.corpus import kolaw
kolaw.fileids()
```

```
['constitution.txt']
```

```
c = kolaw.open('constitution.txt').read()
print(c[:40])
```

```
대한민국헌법

유구한 역사와 전통에 빛나는 우리 대한국민은 3·1운동으로
```

```
from konlpy.corpus import kobill
kobill.fileids()
```

```
['1809895.txt',
 '1809890.txt',
 '1809899.txt',
 '1809898.txt',
 '1809891.txt',
 '1809892.txt',
 '1809894.txt',
 '1809893.txt',
 '1809896.txt',
 '1809897.txt']
```

```
d = kobill.open('1809890.txt').read()
print(d[:40])
```

```
지방공무원법 일부개정법률안

(정의화의원 대표발의 )

 의 안
 번 호
```

### 형태소 분석

KoNLPy는 다음과 같은 다양한 형태소 분석, 태깅 라이브러리를 파이썬에서 쉽게 사용할 수 있도록 모아놓았다.

- Hannanum: 한나눔. KAIST Semantic Web Research Center 개발.
  - http://semanticweb.kaist.ac.kr/hannanum/
- Kkma: 꼬꼬마. 서울대학교 IDS(Intelligent Data Systems) 연구실 개발.
  - http://kkma.snu.ac.kr/
- Open Korean Text: 오픈 소스 한국어 분석기. 과거 트위터 형태소 분석기.

- https://github.com/open-korean-text/open-korean-text

```
from konlpy.tag import *

hannanum = Hannanum()
kkma = Kkma()
komoran = Komoran()
mecab = Mecab()
okt = Okt()
```

이 클래스들은 다음과 같은 메서드를 공통적으로 제공한다.

- `nouns` : 명사 추출
- `morphs` : 형태소 추출
- `pos` : 품사 부착

### 명사 추출

문자열에서 명사만 추출하려면 noun 명령을 사용한다.

```
hannanum.nouns(c[:40])
```

```
['대한민국헌법', '유구', '역사', '전통', '빛', '우리', '대한국민', '3·1운동']
```

```
kkma.nouns(c[:40])
```

```
['대한',
 '대한민국',
 '대한민국헌법',
 '민국',
 '헌법',
 '유구',
 '역사',
 '전통',
 '우리',
 '국민',
 '3',
 '1',
 '1운동',
 '운동']
```

```
# komoran은 빈줄이 있으면 에러가 남
komoran.nouns("\n".join([s for s in c[:40].split("\n") if s]))
```

```
['대한민국', '헌법', '역사', '전통', '국민', '운동']
```

```
mecab.nouns(c[:40])
```

```
['대한민국', '헌법', '역사', '전통', '우리', '국민', '운동']
```

```
okt.nouns(c[:40])
```

```
['대한민국', '헌법', '유구', '역사', '전통', '우리', '국민', '운동']
```

https://liveyourit.tistory.com/57
https://livedata.tistory.com/19

# 2. 랭킹

## 특성추출이란??

뉴스에 등장하는 명사 중에서 해당 뉴스에서 중요하다고 판단하여 추출한 키워드입니다. 이 때 중요도는 '텍스트 랭크(Text Rank)' 알고리즘으로 판단합니다.

텍스트 랭크 알고리즘 : 단어 간 연결망을 그려서 중심이 되는 단어를 찾는 알고리즘입니다. 특정 문서에서 같이 사용된 단어들 간에 연결망을 그리고, 그 연결망에서 다른 단어들과 많이 연결

될수록 중요한 단어라고 판단합니다.

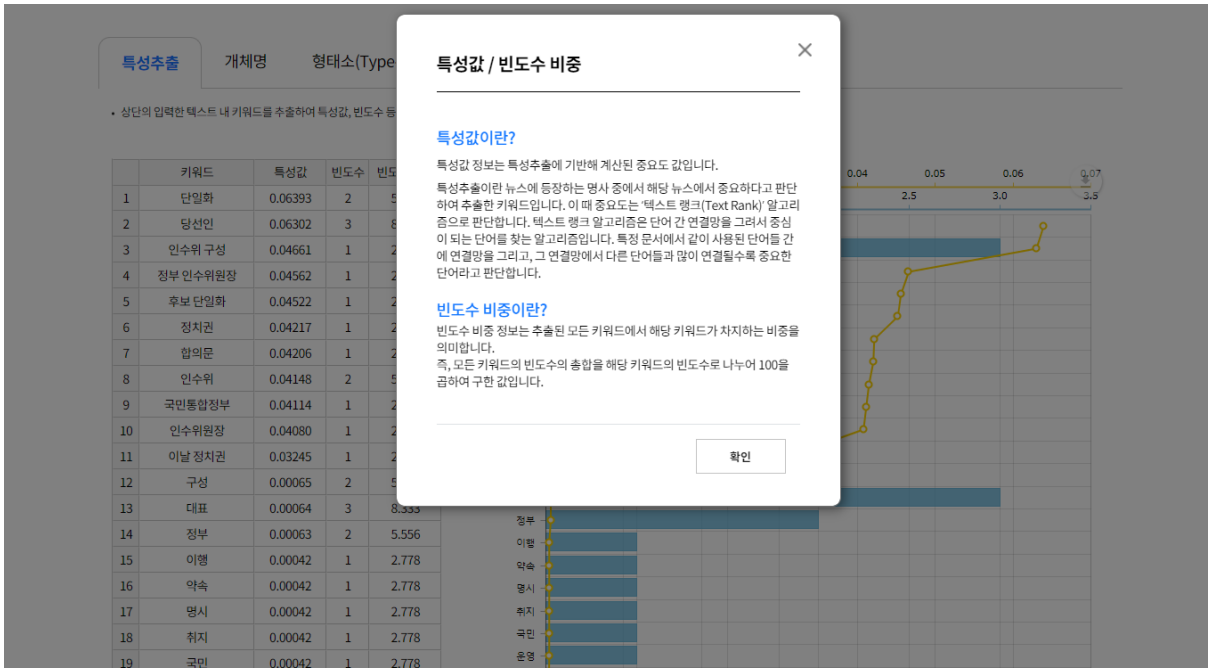
**특성값이란?**

특성값 정보는 특성추출에 기반해 계산된 중요도 값입니다.

**빈도수 비중이란?**

빈도수 비중 정보는 추출된 모든 키워드에서 해당 키워드가 차지하는 비중을 의미합니다.즉, 모든 키워드의 빈도수의 총합을 해당 키워드의 빈도수로 나누어 100을 곱하여 구한 값입니다.

https://www.bigkinds.or.kr/v2/analysis/featureExtraction.do#




# 3. 추천 알고리즘

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/17f3f088-14a5-429b-93bb-1ad4093c0132/빅카인즈_사용자매뉴얼.pdf

4-5 페이지


| 번호 | 구분 | 설명 |
|---|---|---|
| 1 | 분석 대상 뉴스 | 자동으로 분석된 뉴스의 개수가 표시됩니다. |
| 2 | 뉴스 클러스터[1] | 뉴스 클러스터의 개수가 표시됩니다. |
| 3 | 클러스터 내 대표 이슈 | 각 뉴스 클러스터의 대표 이슈가 표시됩니다. 각 이슈를 선택한 이후 관계도 분석 버튼을 클릭하면 관계도 분석 화면으로 이동합니다. 관계도 분석 화면에 대한 자세한 내용은 **"5.1.2 관계도 분석하기"**에서 확인할 수 있습니다. |
| 4 | 상위 클러스터 분류 | 각 뉴스 클러스터에 포함된 뉴스가 표시됩니다 |

위의 **오늘의 이슈** 페이지는 토픽 랭크 알고리즘을 통하여 분석되고 코사인 유사도에 따라 뉴스 클러스터로 재구성됩니다.

- 뉴스 클러스터: 뉴스를 주제별로 분류한 후, 특정 이슈(키워드)별로 묶은 뉴스 그룹을 말합니다.

### 구현 순서

1. 중요 키워드 추출
2. 뉴스 클러스터 생성
3. 뉴스 클러스터 내 대표 키워드 추출
4. 뉴스 클러스터 제목 구성

## 뉴스 클러스터링이란?

- 형태소 분석과 문서 간의 유사도를 계산해 자동으로 뉴스를 모아주는 기술입니다.
- 분석 대상이 되는 뉴스를 1건씩 형태소 분석하여 '명사 상당어구'를 추출

- ‘명사 상당어구’에서 토픽랭크(TopicRank) 알고리즘을 사용하여 뉴스 내 중요 키워드를 선정한 후 키워드를 벡터(Vector)로 구성
- 문서 벡터간 코사인 거리(Cosine Distance) 계산하여 문서 간의 유사도(Similarity)가 임계치(0.2) 이상인 뉴스를 그룹핑
- 모든 문서가 클러스터링 될 때까지 문서 간 그룹핑 반복 수행

작업이 완료 된 뉴스 클러스터링 품질 정확률 : 평균 82.43%