카이스트 검색 기사 크롤링 및 자연어 처리 분석 시각화

작업자: 김효준(전산학부), 박기범(산업및시스템공학부), 박규태(전기및전자공학부)

대표연락처: pgb1227@kaist.ac.kr

1. 개요

- 설명 : 카이스트에 대해 보도된 각종 한국어, 영어 기사들을 크롤링하고 이 기사를 바탕으로 NLP(자연어처리) 분석을 진행한다. 최종적으로 html파일 형식으로 직접 유저가 분석 결과를 확인할 수 있는 인터랙티브 워드클라우드 및 워드맵을 제공한다.
- 제공되는 산출물 : 1. 기사 텍스트 자료(xlsx 형식), 2. 시각화 자료(wordcloud 등), 3. 사용 코드(github)

2. 알고리즘

가. Beautifulsoup, Scrapy, Selenium과 같은 라이브러리를 이용하여 'KAIST' 검색 기사를 크롤링하여 .xlsx, .csv 와 같은 형태로 저장

나. ckonlpy의 Twitter 토크나이저를 이용하여 한국어/영어 기사를 토큰화

다. 각종 라이브러리를 통해 토큰화된 기사를 분석

- Countervectiorizer(각 문서에서 고 빈도로 등장한 단어 순으로 정렬)
- TF-IDF(문서당 단어의 가중치를 다르게 줌으로써, 그 가중치가 높은 순으로 단어를 정렬)
- Word2vector(단어를 window를 기반으로 embedding하여, 문장 내에서 비슷한 맥락을 가진 단어들을 가까운 위치에 등장시키는 기법)
- CountCoccur(단어를 기준으로 그 단어가 등장한 경우, window를 만들어서, 그 window내의 단어의 개수를 파악하여, 함께 어떠한 단어가 주로 등장하는지 파악하는 기법)
- LDA (Topic modeling , NMF의 일종으로 주로 문장에서 어떠한 토픽이 등장하는지 분석하는 기법)

라. 분석 결과를 interactive graph로 시각화 하여 .html 파일로 저장.

3. 데이터 설명

- **자료 소스**
    - **한국어 기사** : 한겨레 신문, 경향신문, 동아일보, 중앙일보, 서울신문, 프레시안, 전자신문, 국민일보, 뉴데일리, jtbc

    - **영문 기사** : nature, science, atlas, bizwire, electronics, engineer, eurekalert, korea herald, korea biomed, mirage, pulse

- **획득방법(설명 및 코드)**

****** 코드는 korean_crawling과 english_crawling 폴더에서 확인할 수 있습니다.

    - **한국어 기사** : 각 신문사별로 홈페이지의 html 구조 파악 후 python library인 beautifulsoup4 및 Selenium 을 이용하여 크롤링 진행. 기간을 인풋으로 받을 시에 기간에 맞는 기사를 추출할 수 있도록 함수로 코드 작성.

    - **영어 기사** : 각 신문사별로 홈페이지의 html 구조 파악 후 python library인 Scrapy를 이용하여 크롤링 진행. 기간을 인풋으로 받을 시에 기간에 맞는 기사를 추출할 수 있도록 함수로 코드 작성.

- **자료의 크기(용량, 텍스트 길이 등)**

    - **한국어 기사 :** 2015.01.01~2020.10.23 동안의 'KAIST', '카이스트' 검색 결과로 얻을 수 있는 모든 기사를 크롤링 함.

출처	기사 건수(건)	용량(MB)
서울신문	3251	3.73
경향신문	1194	1.92
한겨레	888	1.77
중앙일보	1959	3.87
동아일보	1417	2.07

국민일보	1142	1.77
프레시안	253	0.420
뉴데일리	1384	2.62
전자신문	3076	3.25
jtbc	1109	0.884
총 합	**15,673**	**26.1**

- **영어 기사 :** 2015.01.01~2020.10.23 동안의 'kaist' 검색 결과로 얻을 수 있는 모든 기사와 저널을(nature, science) 크롤링 함.

출처	기사 건수(건)	용량(KB)
eurekalert	99	365
the korea herald	165	462
nature	629	19,223
atlas	31	97.4
bizwire	84	174.1
electronics	19	44.8
engineer	18	40.3
korea bio med	38	104.9
mirage	106	321.8
pulse	97	204.8
science	51	58.3
총 합	**1,337**	**21,196**

4. 텍스트 분석

- **분석방법(설명 및 코드):**
    - 먼저 위의 기사를 모두 통합하는 과정을 거친다.

    - 한국어 기사는 ckonlpy의 Twitter tokenizer, 영어 기사는 nltk 라이브러리를 거쳐서 문장을 토큰화 하는 작업을 거침. 또한 이 과정속에서 필요없는 토큰들을 제거한다(불용어 제거).

    - 이후 그 토큰들을 detokenize 하고, LDA라는 머신 러닝 기법을 통해서, 카이스트를 검색했을때 기사에서 주로 등장하는지를 분석한다.(visualize 폴더의 topic modeling)

    - 위에서 뽑아냈던 토큰들을 바탕으로, TF-IDF라는 기법 및 빈도수 직접 계산의 방법을 통해서, 카이스트 검색 결과 기사내에서 비중이 높은 단어를 추출한다.

    - 그리고 이 뽑아낸 단어들을 기반으로 word2vec기법을 통해서 유사단어를 추출 및 sliding window기법으로, 함께 자주 등장하는 단어들을 뽑아낸다.

    - 이후 이 정보들을 json 파일로 저장하여 파일을 읽기 쉬운 형태로 만든다.

    - 이 모든 과정은 korean_analyze.py와 english_analyze.py 파일에 들어가있으며 github를 통해서 구현을 확인할 수 있다.

- **시각화 방법(설명 및 코드)**
    - 위의 분석화 방법에서 얻은 결과를 바탕으로 해서 html파일에 렌더링하고, javascript libarary인 d3 cloud를 통해서 원하는 형태로 시각화 하는 것이 가능하다.
    - d3 cloud의 그래프는 다양한 형태의 구현을 가능하게 하나, 현재 있는 정보가 워드 클라우드를 베이스로 하므로, 일단은 기초적인 워드클라우드 분석을 베이스로 하여 구현하였다.

    - 코드의 경우 visualize 폴더 안의 topic_modeling, titleKeyword(제목 기반의 워드클라우드 분석), contentKeyword(내용 기반의 워드클라우드 분석)에 구현되어 있음.

- **시각화 결과:**

1. 워드클라우드 png 파일

원형 모양의 버전과 KAIST 글자 모양의 버전을 png 파일로 저장하였고, 각 신문사마다의 워드클라우드를 확인할 수 있습니다.

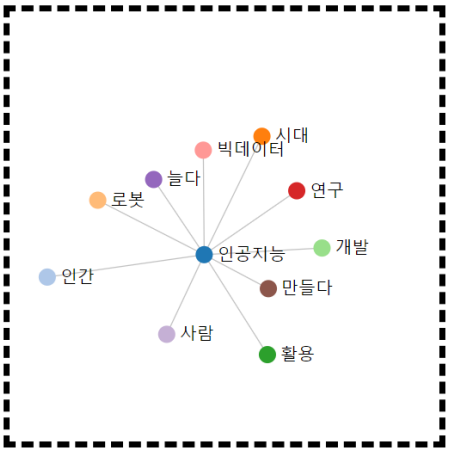
2. 인터랙티브 워드클라우드

카이스트라는 단어를 검색했을 때, 주로 등장하는 단어를 먼저 위에 word cloud형태로 나타냄. 50개의 단어를 기본적으로 나타내고, 상위 10개 단어의 경우 색을 다르게 표현하였습니다.

그리고, 그 단어중 원하는 단어를 누를 경우, 머신러닝 기법을 이용하여 기사 내에서 분석하였을 때 그 단어와 유사한 단어를 추출하고(Similar Detail 그래프), 함께 주로

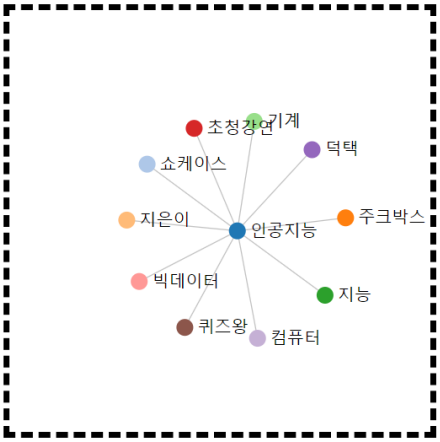
어떤 단어가 등장하는 지를 뽑아냅니다(Cooccurence Detail 그래프). 현재 위의 결과는 '**인공지능**'이라는 단어를 눌렀을 때 , 어떠한 단어와 함께 등장하는지, 그리고 어떠한 단어가 유사하다고 머신러닝이 판단했는지를 보여주는 그림입니다.

Cooccurence Detail



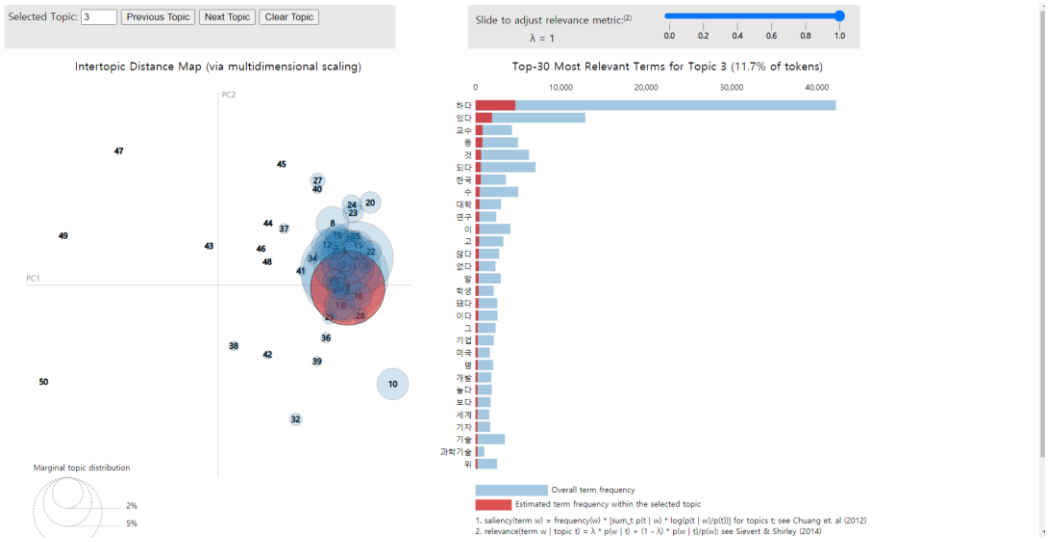
문장내에서 자주 같이 출연하는 단어들의 상위 10개 단어의 네트워크입니다.

Similar Detail



문장 등장 위치에 따른 유사하다고 생각되는 단어 top 10의 네트워크입니다..

3. topic-modeling



- 토픽들이 어디에서 주로 등장하고, 주로 어떤 분포를 보이고 있는지를 뽑아냄, 원을 누르면 그 토픽이 어떠한 토큰들로 이루어져있는지 판단가능하고, 그 토큰이 전체 기사의 토큰중 얼마나 높은 비중을 차지하는지 확인하는 것이 가능합니다.

- 현재 카이스트의 경우, 대부부분의 기사들이 거의 유사한 토픽을 가지고 있다라는 사실을 확인할 수 있습니다.