캡스톤 디자인 I 최종결과 보고서

프로젝트 제목(국문) : 키워드 기반 경제 콘텐츠 큐레이션
프로젝트 제목(영문) : Keyword-based economic contents curation

프로젝트 팀장 : 학번 : 20181622 이름: 송재민
프로젝트 팀원 : 학번 : 20181617 이름: 박재필
프로젝트 팀원 : 학번 : 20181630 이름: 이용경

1. 중간보고서의 검토결과 심사위원의 '수정 및 개선 의견'과 그러한 검토의견을 반영하여 개선한 부분을 명시하시오.

- 없음

2. 기능, 성능 및 품질 요구사항을 충족하기 위해 본 개발 프로젝트에서 적용한 주요 알고리즘, 설계방법 등을 기술하시오.

- 프레임워크 Django를 이용한 MVT(Model-View-Template) 형태의 웹 개발

- Selenium, Requests, BeautifulSoup 과 같은 여러 라이브러리를 통한 동적+정적 크롤링

- RegEx를 통한 문장에 들어있는 불필요한 요소 텍스트 전처리

- TF-IDF 알고리즘 : 문장에 사용되는 모든 단어에 점수를 부여하는 알고리즘으로 전체 문서 내에서 비중있는 단어의 가중치를 계산한다. 이를 이용해서 크롤링한 전체 뉴스기사의 비중있는 단어의 가중치를 계산

- KMeans 알고리즘 : 비지도 학습의 일종으로 주어진 데이터를 k개의 클러스터로 묶는 알고리즘으로, 각 클러스터와 거리 차이의 분산을 최소화하는 방식으로 동작하는 알고리즘인 , TF-IDF를 통해 계산한 가중치을 기반으로 군집화

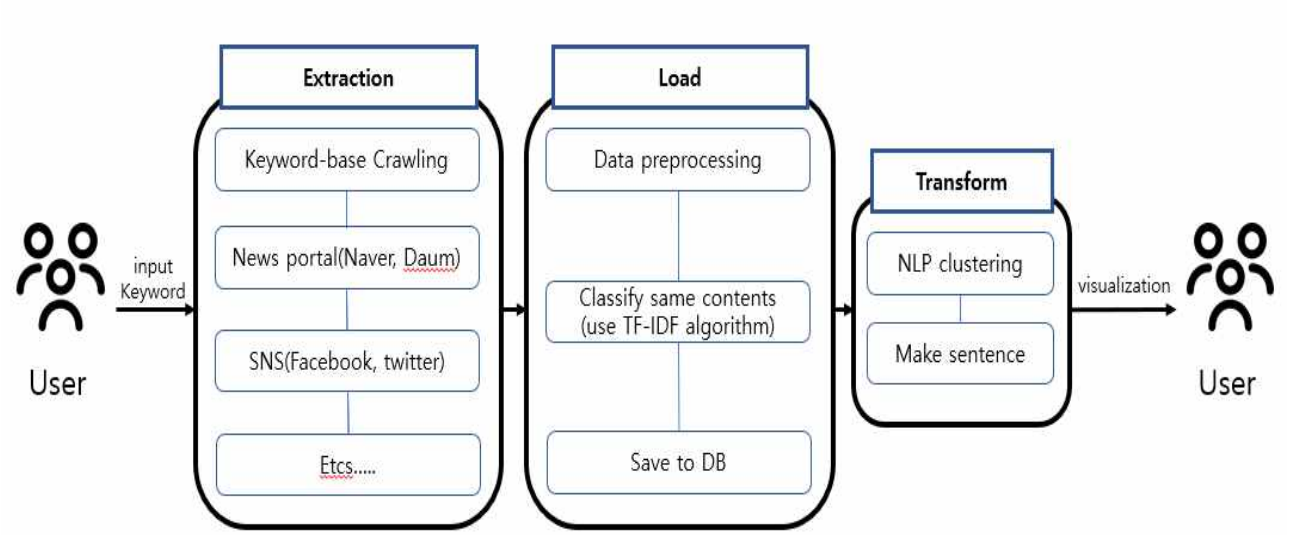
- 코사인 유사도 : 내적공간의 두 벡터간 각도의 코사인값을 이용하여 측정된 벡터간의 유사한 정도를 의미한다. 각도가 0°일 때의 코사인값은 1이며, 다른 모든 각도의 코사인값은 1보다 작다. 1에 가까울수록 유사도가 높음

- konlpy 라이브러리의 Okt를 사용하여 기사 내용의 전체를 명사별로 분류하여 문장들을 벡터화 시키기 위해서 사용

- 문장별 코사인 유사도 측정을 통해 비슷한 내용의 문장들을 군집화하여 반복되는 문장들 제거

- 군집화된 문장들을 모아 점수 계산(대표성, 반복성, 함축성)하여 계산한 후 높은 순으로 정렬

3. 요구사항 정의서에 명세된 기능 및 품질 요구사항에 대하여 최종 완료된 결과를 기술하시오.



3-1. 기능별 상세 요구사항

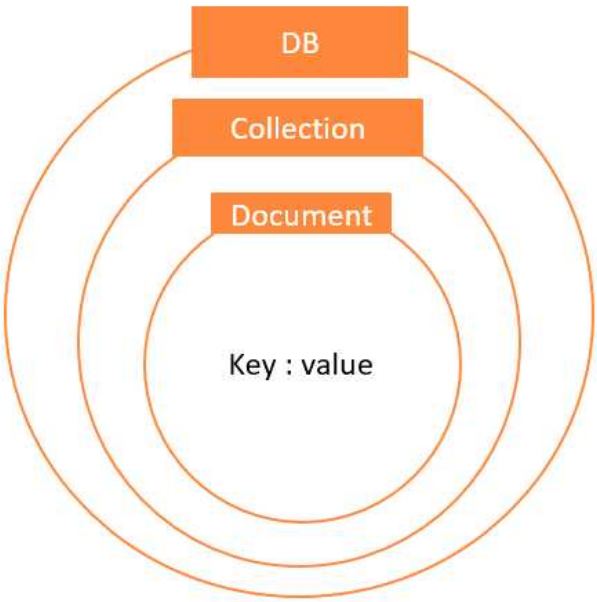
요구사항 고유번호		SFR-001		
요구사항 명칭		Data Crawling		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 크롤링을 통한 데이터 수집		
	세부 내용	- 뉴스 포털 사이트 등 여러 사이트에서 정보를 수집한다. - Selenium, BeautifulSoup, Requests 등의 라이브러리 사용한다.		
요구사항 진행상황	세부 내용	- 네이버 뉴스 포털 사이트에서 뉴스기사를 수집한다.		

요구사항 고유번호		SFR-002		
요구사항 명칭		데이터 간 유사도 분석 및 저장		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 추출한 데이터 간 유사도 분석 후 높은 유사도 데이터 저장		
	세부 내용	- 추출한 데이터 간의 유사도를 TF-IDF 통계를 통해 분석한다. - 유사도가 높은 데이터들로 DB에 저장할 데이터들을 선정한다.		
요구사항 진행상황	세부 내용	- 추출한 데이터 간의 군집을 형성한다. - 군집들에서 키워드 관련 데이터 중 TF-IDF 통계가 높은 데이터를 추출한다.		

요구사항 고유번호		SFR-003		
요구사항 명칭		NLP clustring		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 자연어 처리 - 주제끼리 분류		
	세부 내용	- 교집합된 데이터 내에서 주제별로 분류한다. - 벡터화 시커 distance를 계산하여 군집화시킨다.		
요구사항 진행상황	세부 내용	- 통계적인 방법으로 주제끼리 군집화 진행. - cosine similarity를 계산하여 비슷한 문장 끼리 군집화 실행		

요구사항 고유번호		SFR-004		
요구사항 명칭		NLP Generation		
요구사항 분류		기능 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 자연어 처리 - 전체 내용을 요약하여 정해진 형식으로 출력		
	세부 내용	- Generation 모델을 사용하여 여러 개로 나누어진 내용을 하나의 글로 작성한다.		
요구사항 진행상황	세부 내용	- 인공지능을 활용하여 구현하려 했으나 인공지능 모델의 사이즈 한계로 인해 통계적 알고리즘으로 구현 - 대표성, 간결성, 반복성으로 score를 내어 구현		

- DB 설계 구조



- DB명 : keyword명으로 지정
- Collection : 크롤링을 시행한 날짜
- Document : 하나의 뉴스기사의 데이터

4. 구현하지 못한 기능 요구사항이 있다면 그 이유와 해결방안을 기술하시오,

최초 요구사항	구현 여부(미구현, 수정, 삭제 등)	이유(일정부족, 프로젝트 관리미비, 팀원변동, 기술적 문제 등)
작업 기록하는 페이지 작성	미구현	일정 부족
웹 에러 발생시 예외 처리	미구현	일정 부족
웹 배포 과정	미구현	일정 부족
DB 데이터 읽어드리기	미구현	일정 부족
다양한 데이터 포맷	미구현	일정 부족
데이터 간 유사도 분석 및 저장	수정	코사인 유사도 뿐 아니라 KMean 알고리즘 사용 추가

5. 요구사항을 충족시키지 못한 성능, 품질 요구사항이 있다면 그 이유와 해결방안을 기술하시오.

분류(성능, 속도 등) 및 최초 요구사항	충족 여부(현재 측정결과 제시)	이유(일정부족, 프로젝트 관리미비, 팀원변동, 기술적 문제 등)
텍스트 전처리 정확도 향상	약 80%	여러 기사의 형태. 그에 따라 꾸준히 정규표현식을 이용한 텍스트 전처리과정 필요.
크롤링 시간에 따른 딜레이	실행 시간 소요 시간 약 1시간 정도 소요	실행 시간 축소에 필요한 기술적 지식 부족
다중문서 요약	알고리즘식으로 구현 및 인공지능 형식으로 수정 예정	비 군집화로 인해 대량의 데이터가 input으로 들어와 기술적인 문제로 인공지능을 사용하지 못했지만 군집화로 인해 문서의 개수가 기하급수적으로 적어져 인공지능 방식으로 수정 예정

6. 최종 완성된 프로젝트 결과물(소프트웨어, 하드웨어 등)을 설치하여 사용하기 위한 사용자 매뉴얼을 작성하시오.

프로젝트 파일 명 : HDAM_Django_Project
프로젝트의 위치 : C:\\HDAM_Django_Project 로 가정

1. 사용자 실행환경 구축
    - 터미널 프로그램 cmder을 다운로드 (https://cmder.app/)
    - 파이썬 3.11ver(권장) 다운로드,
    - Google Chrome
    - 패키지 설치
    pip install
    Django, feedparser, pandas, tqdm, selenium, webdriver_manager, konlpy, scikit-learn, bs4, requests, openpyxl, fsspec

2. cmder을 이용하여 프로젝트가 있는 장소로 이동
cd c:\\HDAM_Django_Project

λ piplist

Package	Version
asgiref	3.7.2
async-generator	1.10
attrs	23.1.0
beautifulsoup4	4.12.2
bs4	0.0.1
certifi	2023.5.7
cffi	1.15.1
charset-normalizer	3.1.0
colorama	0.4.6
Django	4.2.1
et-xmlfile	1.1.0
exceptiongroup	1.1.1
feedparser	6.0.10
fsspec	2023.5.0
h11	0.14.0
idna	3.4
joblib	1.2.0
JPype1	1.4.1
konlpy	0.6.0
lxml	4.9.2
numpy	1.24.3
openpyxl	3.1.2
outcome	1.2.0
packaging	23.1
pip	22.3.1
pycparser	2.21
PySocks	1.7.1
python-dateutil	2.8.2
python-dotenv	1.0.0
pytz	2023.3
requests	2.31.0
scikit-learn	1.2.2
scipy	1.10.1
selenium	4.9.1
setuptools	65.5.0
sgmllib3k	1.0.0
six	1.16.0
sniffio	1.3.0
sortedcontainers	2.4.0
soupsieve	2.4.1
sqlparse	0.4.4
threadpoolctl	3.1.0
tqdm	4.65.0
trio	0.22.0
trio-websocket	0.10.2
tzdata	2023.3
urllib3	2.0.2
webdriver-manager	3.8.6
wsproto	1.2.0

3. 장고 프로젝트 가상환경(venv) 활성화
venv\Scripts\activate.bat (가상환경 종료 명령 => venv\Scripts\deactivate)

4. 장고 서버 실행 후 웹 페이지 접속
python manage.py runserver
-> http://127.0.0.1:8000/로 접속

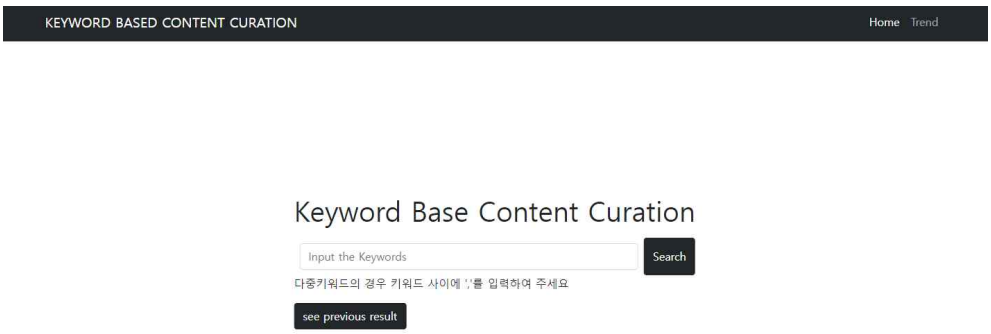


그림 1. 데모버전 메인 페이지(http://127.0.0.1:8000/)

키워드를 입력한 후 Search 버튼을 누르게 되면 키워드에 대한 크롤링, 전처리, 후처리 과정이 이루어 지며 해당 과정의 내용들을 볼 수 있는 작업을 실행함.

See previous result 버튼을 누르게 되면, 이전의 작업에 대한 결과를 다시 보여줌.

Trend를 누르면 구글 트렌드의 RSS를 읽어와 최신 인기 검색어를 볼 수 있어, 사용자의 입력할 키워드의 선정에 도움을 줄 수 있음

7. 캡스톤디자인 결과의 활용방안

다중 키워드를 사용하여 뉴스들을 검색하고, 검색된 기사들을 크롤링하여 다량의 뉴스기사를 수집한다. 그리고 이를 요약함에 있어서 여러 검증을 거치며 다중 키워드와 최대로 관련된 기사를 수집하기에, 사용자에게 친화적이며 다중문서를 요약하여 출력을 하기에 사용자로 하여금 시간 절약에 큰 힘을 실어줄 수 있다. 특히 경제 관련 기사를 위주로 기사를 가져오기에 투자, 부동산, 금융 관련 등 경제 관련 다양한 분야에서 도움을 줄 수 있다