캡스톤디자인I 중간보고서

<table>
<tr><td rowspan="2">제 목</td><td colspan="2">국문</td><td colspan="3">키워드 기반 콘텐츠 큐레이션</td></tr>
<tr><td colspan="2">영문</td><td colspan="3">Keyword based contents curation</td></tr>
<tr><td rowspan="2">진 행
상 황</td><td>중요마일스톤</td><td colspan="4">기능 요구사항에 대한 중요 마일스톤
1) 추출 - 일부 사이트 크롤링 소스코드 작성
2) 불용어 데이터 전처리 - 라이브러리를 사용한 소스코드 작성
3) 저장 - MongoDB 소스코드 작성
4) 멀티 키워드로 인한 Intersection 선택 - TF-IDF를 기반한 알고리즘 코드 작성
5) 변환 - 기존 NLP 모델 Ko-BERT 및 Ko-BART를 활용한 clustering 및 데이터 생성 코드 작성

성능 요구사항에 대한 중요 마일스톤
1) 추출 과정 중 글 개수 - 언어의 범위에 따른 기사 및 SNS 등 글 크롤링 시간 측정
2) clustering - 글 개수 및 주제의 개수에 따른 clustering 정도 측정 및 시간 측정
3) transform - 요약문 생성의 성능 및 시간 측정</td></tr>
<tr><td>진행상황</td><td colspan="4">· 전반적인 시스템 구성도 작성 완료
· 크롤링 범위를 확장, 추가적으로 소스코드 작성 중
· 추출된 데이터 전처리 코드 작성 중
· 데이터들 간의 TF-IDF 측정 코드 작성 중
· mongoDB에 저장될 DB 모델 구성 완료
· 데모버전 UI 구상 완료
· Vue.js를 통한 시각화를 위해 프론트엔드 소스코드 작성 중
· 다양한 글을 활용한 vector화 시키는 소스코드 작성 중
· clustering 기반 주제별 글 작성 모델 추가 예정</td></tr>
<tr><td>산출물</td><td colspan="5">요구사항 정의서(별첨 1), 중간보고서(별첨 2)</td></tr>
<tr><td rowspan="4">팀
구성원</td><td>학년</td><td>학 번</td><td>이 름</td><td colspan="2">연락처(전화번호/이메일)</td></tr>
<tr><td>4</td><td>20181617</td><td>박재필</td><td colspan="2">010-2936-8255/20181617@edu.hanbat.ac.kr</td></tr>
<tr><td>4</td><td>20181622</td><td>송재민</td><td colspan="2">010-5118-4832/20181622@edu.hanbat.ac.kr</td></tr>
<tr><td>4</td><td>20181630</td><td>이용경</td><td colspan="2">010-5784-0928/20181630@edu.hanbat.ac.kr</td></tr>
<tr><td colspan="6">컴퓨터공학과의 프로젝트 관리규정에 따라 다음과 같이 요구사항 정의서와 중간보고서를 제출합니다

2023년 04월 28일

책임자 : 송 재 민 (인)
지도교수 : 장 한 얼 (인)</td></tr>
</table>

[별첨1]

프로젝트명 : 키워드 기반 콘텐츠 큐레이션

소프트웨어 요구사항 정의서

Version 1.0

개발 팀원 명(팀리더):송재민
박재필
이용경
대표 연락처:010-5118-4832
e-mail: 20181622@edu.hanbat.ac.kr

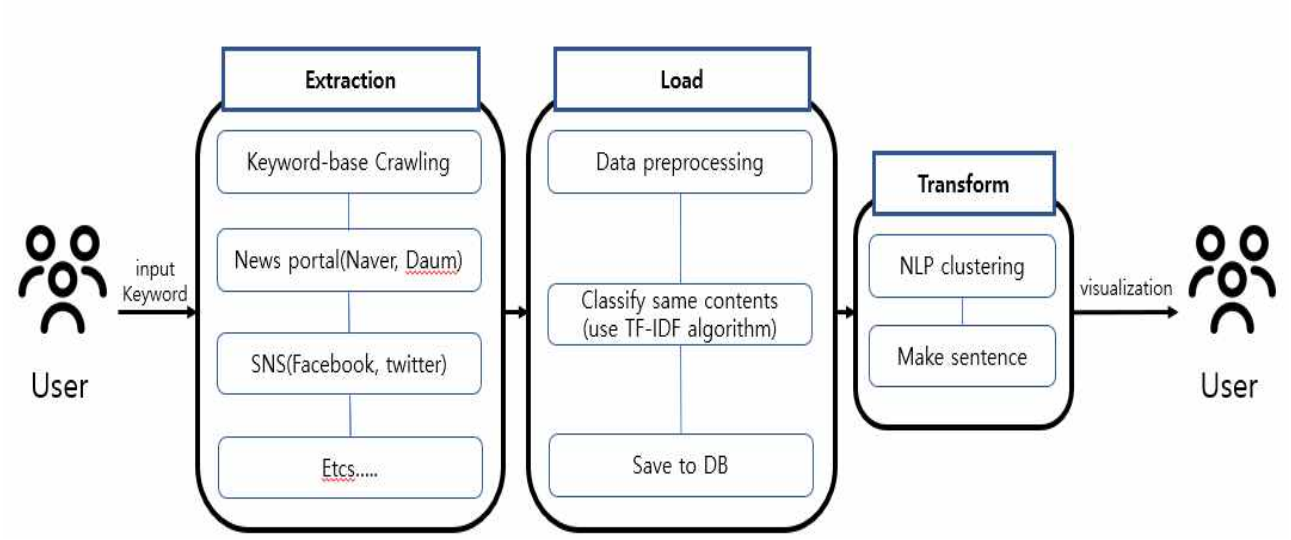
목차

1. 시스템 개요

1) 목표

현재 재테크에 대한 관심이 뜨거워지는 가운데, 이에 대한 여러 정보들을 얻기 위해 커뮤니티 게시글이나, 지인들로부터 정보를 얻던 과거와 달리 현재는 뉴스, 신문 등 관련 기사를 통해 보다 더 전문적인 재테크 정보를 얻으려는 모습이 나타나고 있다. 이에 대하여, 바쁜 현대사회에서 모든 경제 뉴스를 전부 보는 것은 어려운 일이기에, 용이하게 재테크 관련 정보를 얻기 위해서 인기 키워드로 연결되어있는, 재테크에 관한 콘텐츠들을 요약하여 보여주는 것을 목표로 한다.

2) 시스템 구성도



2. 시스템 장비 구성요구사항

요구사항 고유번호		ECR-001		
요구사항 명칭		자연어 처리 서버		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 자연어 처리 서버		
	세부 내용	- 장비 품목 : 서버 그래픽카드 - 장비 수량 : 1식 - 장비 기능 : 데이터 처리 - 장비 성능 및 특징 : cuda를 이용한 병령 컴퓨팅 서비스 지원한다.		

요구사항 고유번호		ECR-002		
요구사항 명칭		DB 관리 소프트웨어		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세 설명	정의	- DB 저장 소프트웨어		
	세부 내용	- 장비 품목 : mongoDB - 장비 수량 : 1식 - 장비 기능 : DB관리 - 장비 성능 및 특징 : BSON 구조로 데이터를 직관적으로 이해 가능하고 읽고 쓰기 편하며 저장 형식이 유연하여 데이터 저장에 용이하다.		

요구사항 고유번호		ECR-003		
요구사항 명칭		프론트엔드 구성 요구사항		
요구사항 분류		시스템 장비구성 요구사항	응락수준	필수
요구사항 상세 설명	정의	프론트엔드 파트		
	세부 내용	User가 모바일과 PC에서 시각적으로 확인 가능한 UI 제작 - 장비 품목 : Vue.js - 장비 기능 : JS기반 웹 UI 개발 프레임워크, 가상 DOM 지원, 빠른 UI 랜더링		

3. 기능 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 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 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 시켜 distance를 계산하여 군집화시킨다.		

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.		

4. 성능 요구사항

요구사항 고유번호		PER-001		
요구사항 명칭		크롤링 시간에 따른 딜레이		
요구사항 분류		성능 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 정보 크롤링 시간		
	세부 내용	- 서버의 접근 금지를 막기 위해, 그리고 동적 크롤링을 위해 필수적으로 timesleep()을 사용한다. - 리퀙토링 등 여러 방안을 모색해 최선의 시간 효과를 내는 방법을 선택해야 한다.		

요구사항 고유번호		PER-002		
요구사항 명칭		데이터 미흡 오류 응답 시간		
요구사항 분류		성능 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 데이터가 미흡할 경우 오류 처리		
	세부 내용	- 입력한 키워드의 데이터가 미존재하거나 미흡할 경우 생기는 오류는 입력 후 5초 이내에 입력 오류 메시지를 알려야 한다.		

요구사항 고유번호		PER-003		
요구사항 명칭		clustering에 발생하는 성능		
요구사항 분류		성능 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 벡터화 중 발생하는 지연시간 평가 - 벡터의 정확도 평가		
	세부 내용	- 학습된 KO-BERT 모델을 사용함에 encoding 화 되며 필요한 시간 측정한다. - 주제에 따라 벡터화된 distance를 측정하여 벡터의 정확도 측정하여야 한다.		

요구사항 고유번호		PER-004		
요구사항 명칭		자연어 생성에 발생하는 성능		
요구사항 분류		성능 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 자연어 생성에 필요한 지연시간 평가 - 자연어 생성의 정확도 평가		
	세부 내용	- 자연어 생성에 글의 개수에 따른 지연시간을 측정한다. - 자연어 생성에 필요한 정보가 나열되어 있는지에 대해 정확도 평가한다.		

5. 인터페이스 요구사항

요구사항 고유번호		SIR-001		
요구사항 명칭		웹 디자인		
요구사항 분류		인터페이스 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 편의성 높은 직관적인 웹 UI		
	세부 내용	- 어떠한 환경(PC, Mobile)에서도 접근 가능하게 반응형 웹으로 디자인 - 직관적으로 파알할 수 있는 단어로 눈에 띄게 배치한다. - 아이콘을 사용해 기능 서비스의 용도를 쉽게 파악할 수 있다.		

요구사항 고유번호		SIR-002		
요구사항 명칭		메인 페이지 인터페이스		
요구사항 분류		인터페이스 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 메인 페이지 인터페이스 요구사항		
	세부 내용	- 시스템명 중앙에 배치한다. - 그 시스템명 아래 keyword를 입력할 검색상자를 생성한다.		

요구사항 고유번호		SIR-003		
요구사항 명칭		로딩 페이지 인터페이스		
요구사항 분류		인터페이스 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 로딩페이지 요구사항		
	세부 내용	- Task를 실행하기 위한 시간이 필요하다. - Task가 완료되는지 시간이 얼마나 필요한지 직관적으로 알 수 있는 Loadingbar를 사용한다.		

요구사항 고유번호		SIR-004		
요구사항 명칭		결과 출력 인터페이스		
요구사항 분류		인터페이스 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 결과 출력 인터페이스		
	세부 내용	- 멀티 키워드들 간의 교집합으로 연관되어 있는 Contents를 추려서 보여준다. - 최종적으로 작업을 거쳐 완성된 글을 출력한다.		

6. 데이터 요구사항

요구사항 고유번호		DAR-001		
요구사항 명칭		다양한 데이터 포맷		
요구사항 분류		데이터 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- 핀테크 관련 텍스트, 이미지, 동영상 등 키워드와 관련된 다양한 포맷의 데이터를 추출한다.		

요구사항 고유번호		DAR-002		
요구사항 명칭		전처리 데이터 저장		
요구사항 분류		데이터 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- 추출한 데이터를 불필요한 요소들을 제거하여 사용가능한 문장으로 변환한다. - 전처리된 데이터를 관려도 있는 데이터들만 DB에 저장시킨다.		

요구사항 고유번호		DAR-003		
요구사항 명칭		학습 데이터		
요구사항 분류		데이터 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- 기존의 모델을 사용하되 우리가 사용하는 글의 형식에 맞추어 추가 학습하기 위한 글 데이터셋을 구축한다.		

요구사항 고유번호		DAR-004		
요구사항 명칭		데이터 형식		
요구사항 분류		데이터 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- 여러 가지 글을 사용하되 일정한 형식에 맞춰 날짜, 출처 등 다양한 글에 대한 정보가 필요하다.		

7. 테스트 요구사항

요구사항 고유번호		TER-001		
요구사항 명칭		테스트 방안		
요구사항 분류		테스트 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- 전체적인 프로세스는 Extraction, Load, Transform에서 일어나므로 각각의 task에서 test 진행한다. - Extraction 단계에서 크롤링을 진행하여 키워드에 맞는 결과를 도출하는지 평가한다. - Load 단계에서 크롤링된 데이터들을 전처리 후 이를 키워드의 교집합된 데이터를 채택하는지 평가를 진행한다. - Transform 단계에서 글들을 clustering, generation 된 결과를 확인한다. - 위 단계를 각각 확인 후 이를 통합하여 전체적인 process를 평가		

요구사항 고유번호		TER-002		
요구사항 명칭		단위 테스트		
요구사항 분류		테스트 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- 기능 요구사항에 기술된 요구사항에 대한 각각의 테스트를 수행한다. - 데이터 추출, 저장, 출력이 정상적으로 작동하는지 테스트한다.		

요구사항 고유번호		TER-003		
요구사항 명칭		통합 테스트		
요구사항 분류		테스트 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- 테스트가 완료된 최소 단위의 기능들을 통합하여 기능들이 정상적으로 연결되어 오류없이 수행되는지 테스트를 진행. - 최초 입력부터 최종 입력까지의 수행 결과가 정상적으로 출력되는지 테스트를 진행		

8. 보안 요구사항

요구사항 고유번호		SER-001		
요구사항 명칭		출처에 대한 보호		
요구사항 분류		보안 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- 인터넷에 올라와 있는 글들을 crawling 하는 과정이므로 robots.txt.를 확인하여 crawling이 가능/불가능한지 판단하는 과정이 필요하다.		

9. 품질 요구사항

요구사항 고유번호		QUR-001		
요구사항 명칭		오류 발생률		
요구사항 분류		품질 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 오류 관리		
	세부 내용	- 자연어 처리 특성상 데이터의 길이가 매우 가변적이며 이에 따라 입력된 글의 길이에 따라 오류를 발생할 수 있으므로 load 과정에서 일부 데이터는 길이에 대한 관리가 필요하다.		

요구사항 고유번호		QUR-002		
요구사항 명칭		변경상황에 대한 업데이트		
요구사항 분류		품질 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 크롤링 대상 웹사이트 변경상황 관리		
	세부 내용	- 크롤링 대상 사이트의 페이지 소스 변경 시 크롤러 소스코드를 업데이트하지 않으면, 전체적인 로직이 정상적으로 등장하지 못하기 때문에 지속적인 모니터링을 통한 변경사항 업데이트가 필요하다.		

요구사항 고유번호		QUR-003		
요구사항 명칭		정확도 관리		
요구사항 분류		품질 요구사항	응락수준	필수
요구사항 상세 설명	정의	- 키워드 관련 데이터들의 사용 자격		
	세부 내용	- 추출한 데이터들의 유사도가 20% 미만으로 불일치 시 해당 키워드는 사용 불가를 알리고 새로운 키워드를 요청한다. - 추출한 데이터들의 수량이 일정 수준 미달 시 새로운 키워드를 요청한다.		

10. 제약 사항

요구사항 고유번호		COR-001		
요구사항 명칭		크롤링 범위		
요구사항 분류		제약 사항	응락수준	필수
요구사항 상세 설명	정의	- 크롤링할 사이트 지정		
	세부 내용	- 크롤링할 데이터를 가져올 데이터풀을 뉴스기사, SNS로 지정한다.		

요구사항 고유번호		COR-002		
요구사항 명칭		전처리 및 필터링		
요구사항 분류		제약 사항	응락수준	필수
요구사항 상세 설명	정의	- 한글 데이터 전처리		
	세부 내용	- 사용하지 않는 문장, 단어, 기호 등 불필요한 요소들은 처리가 필요하다. - 용어를 관련도 측정 작업에 불필요한 것들의 처리가 필요하다.		

11. 프로젝트 관리 요구사항

요구사항 고유번호		PMR-001		
요구사항 명칭		프로젝트 수행 조직		
요구사항 분류		프로젝트 관리 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- 이용경 : Vue.js를 이용한 웹 페이지 구축 및 프론트 엔드의 전반적인 부분 담당, 전반적인 크롤링 작업 수행한다. - 송재민 : 추출한 데이터들의 전처리 및 데이터 간 유사도 측정, 분석을 수행해 사용할 데이터들을 mongoDB를 이용해 데이터를 Load 하는 작업을 수행한다. - 박재필 : DB에 저장된 데이터를 Kobert를 통해 벡터화 시켜 이를 주제별로 군집화 한 후 주제별로 분류된 글들을 Kobart 기반의 생성 AI로 글을 작성함.		

요구사항 고유번호		PMR-002		
요구사항 명칭		프로젝트 일정 계획		
요구사항 분류		프로젝트 관리 요구사항	응락수준	필수
요구사항 상세 설명	세부 내용	- ~ 5월 : 추가적인 크롤링, 전처리, 후처리 과정 - 6월 : 프로젝트 프로토타입 완성(모바일 웹 사이트) - 7 ~ 8월 : 유지보수과정 - ~ 10월 : 정확도 향상을 위한 자연어 모델 개선 및 UI 개선		

[별첨2]

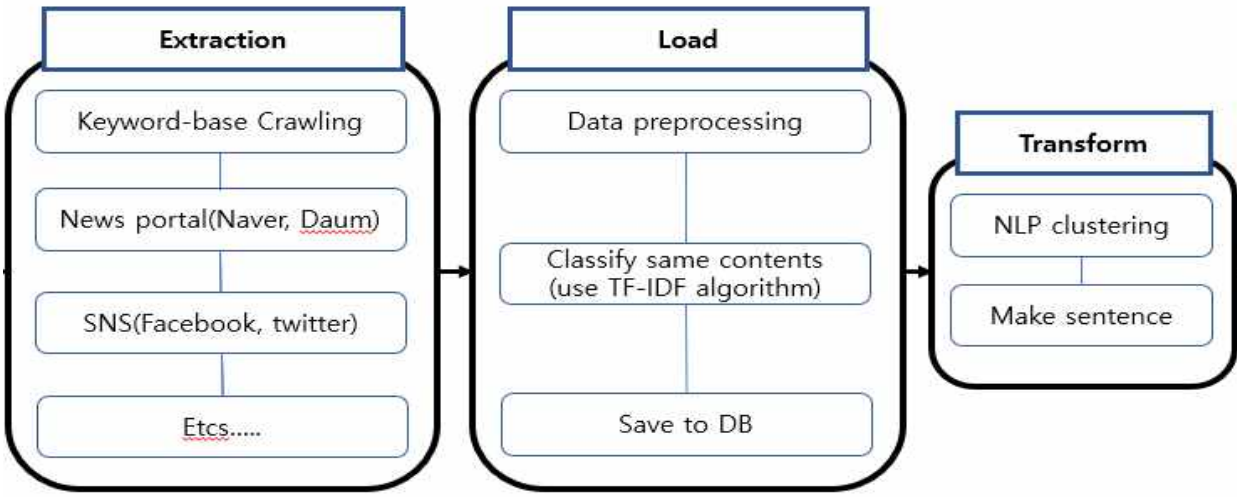
중간보고서

1. 요구사항 정의서에 명세된 기능에 대하여 현재까지 분석, 설계, 구현(소스코드 작성) 및 테스팅한 내용을 기술하시오.

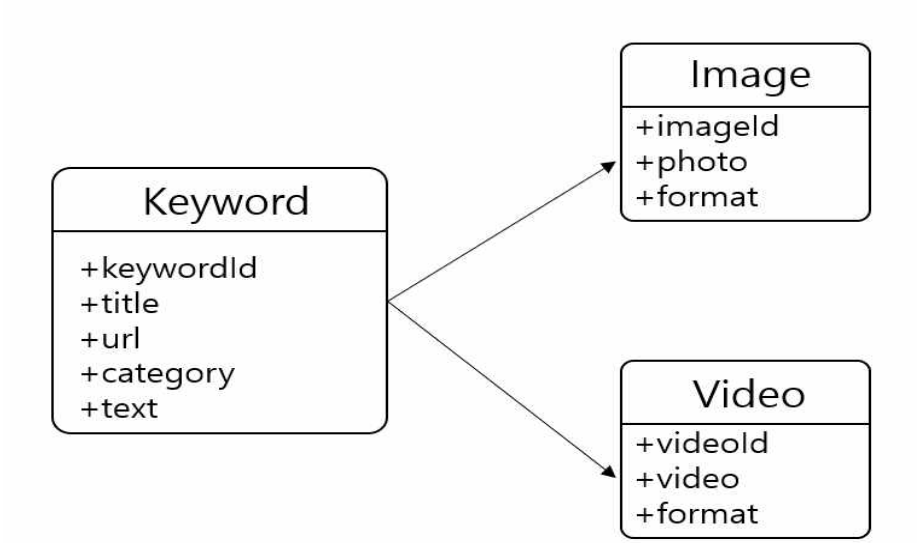
- 분석

위의 목차에서 시스템 개요는 키워드 기반 콘텐츠 큐레이션의 대략적인 순서도이다. 사용자가 키워드들을 입력하면 뉴스기사, SNS 등에서 추출할 수 있는 각각의 키워드에 대한 데이터를 정제되지 않은 많은 양의 데이터들을 추출한다. 추출된 데이터들에서 불필요한 문장, 단어, 기호들을 제거하는 전처리 과정을 거쳐 정형화된 데이터로 전처리한다. 키워드들의 전처리된 데이터에서 교집합이 일어나는 즉, 서로 관련된 데이터들만 선별하고 기준이 충족된 데이터들만 DB에서 저장시킨다. DB의 저장된 데이터들을 군집화를 통해 주제별로 분류하고 분류된 글들 중 키워드와 관련된 내용을 NLP모델을 통해 콘텐츠 큐레이션을 수행해 다양한 형태의 콘텐츠를 생성하는 것을 목표로 하고 있다.

- 설계



위의 그림과 같이 ELT 프로세스를 이용해 Extraction 부분에서는 웹 크롤링을 이용해 키워드에 관련된 데이터들을 뉴스기사나 SNS 등에서 추출을 하고 Load 부분에서 데이터의 전처리를 거쳐 사용 가능한 데이터로 변환한 뒤 아래의 DB 설계 모델과 같이 Load를 진행한다.



NOSQL DB를 사용하기 때문에 스키마 등의 제약을 받지 않아 자유롭게 데이터 저장이 가능하여 Keyword별 Collection을 생성하여 제목, url, 추출된 사이트 종류, 본문을 저장하고 추가로 관련 이미지나 영상 또한 저장이 가능하다. 마지막으로 Transport에서는 각 글에대한 encoding된 벡터를 구하여, 이 벡터들은 Collection된 Keyword들 내에서 한번 더 clustering을 통한 주제별 분류가 이루어지며, 이 clustering된 글들은 각각 주제별로 분류되어, 각 주제별 글 내에서 KoBART를 기반으로 한 Generation 모델을 사용하여 글을 작성하며 사용자에게 제공된다.

- 현재까지의 구현 및 테스트
  현재 웹 크롤링을 통한 데이터 추출까지 완료가 되어 있다.

소스코드

라이브러리

```
importrequests
frombs4 importBeautifulSoup
importre
importpandas aspd
fromdatetime importdatetime
fromtqdm importtqdm

fromselenium importwebdriver
fromselenium.webdriver.common.keys importKeys
fromselenium.webdriver.common.by importBy
fromselenium.webdriver.chrome.service importService
fromwebdriver_manager.chrome importChromeDriverManager
importtime
```

본문 크롤링 함수

```
def crawling_article(url,company):

content_lists = list()

res = requests.get(url,headers=headers)
soup = BeautifulSoup(res.content, 'html.parser')
title_data = soup.select_one('div.head_view > h3')
content_data = soup.select('#mArticle > div.news_view.fs_type1 > div.article_view
> section > p')
    forindex incontent_data:
content_lists.append(index.get_text())
content_lists = " ".join(content_lists)

category = soup.find_all('h2')[1].get_text()
    ifcategory == "경제":
content_lists = pre_processing(company, content_lists)
isEconomy = 1
        print("경제")
    else:
isEconomy = 0
        print("X")

    returntitle_data.get_text(), content_lists, isEconomy
```

메인 함수

```
keyword = input('검색할 키워드를 입력하시오.  : ')
print("본 프로그램은 \"경제\" 카테고리 뉴스만 크롤링하므로, 탐색뉴스의 수와 실제 크롤링
된 기사의 개수는 다를수도 있습니다.")
news_num = int(input('언론사별 탐색 뉴스의 수(숫자만 입력) : '))
sort_sequence = int(input("정확도순은 1, 최신순은 2, 오래된순은 3  : "))
ifsort_sequence == 1:
sort_sequence = "accuracy"
elifsort_sequence == 2:
sort_sequence = "recency"
elifsort_sequence == 3:
sort_sequence = "old"
sort_day = int(input("전체는 1, 최근 1일은 2, 최근 1주는 3, 최근 1개월은 4, 최근 6개월
은 5, 최근 1년은 6 : "))
url = "https://search.daum.net/search?nil_suggest=btn&w=news&DA=STC&q="+ str(keywo
rd) +"&sort="+ str(sort_sequence) +"&p=1"
print('\n'+ '='* 100+ '\n')
print("크롤링을 위한 크롤링 브라우저를 실행합니다.")
chromedriver = 'C:/dev_python/Webdriver/chromedriver.exe'
driver = webdriver.Chrome(service=Service(chromedriver))
driver.get(url)
time.sleep(sleep_sec)
elem = driver.find_element(By.XPATH, "//*[@id='newsColl']/div[1]/div[2]/div/div[1]
")
elem.click()
time.sleep(sleep_sec)
elem = driver.find_element(By.XPATH, "//*[@id='newsColl']/div[1]/div[2]/div/div[1]
/div/ul/li["+ str(sort_day) + "]/a")
elem.click()
time.sleep(sleep_sec)
print('\n'+ '='* 100+ '\n')
print("총 10개의 언론사별로 "+ str(news_num) + "개씩 총 "+ str(news_num * 10) + "개
기사를 탐색합니다.")

news_search_lists=['매일경제', '뉴시스',  '연합뉴스',  '한국경제',  'kbs', '중앙일보
',  '조선일보',  '국민일보', '아시아경제', '조선비즈']
forarticle_company intqdm(news_search_lists):
    print(article_company + " 크롤링을 시작 합니다.")
elem = driver.find_element(By.XPATH,"//*[@id='newsColl']/div[1]/div[2]/div/div[2]"
)
elem.click()
time.sleep(sleep_sec)
elem = driver.find_element(By.XPATH, "//button[@class = 'btn_flex btn_cp']")
elem.click()
time.sleep(sleep_sec)
elem = driver.find_element(By.XPATH, "//*[@id='newsCollCpInp']")
elem.clear()
elem.send_keys(article_company)
elem.send_keys(Keys.RETURN)
time.sleep(2)
url = driver.current_url
    print(url)
```

```
news_index = news_num // 10
page_index = news_num % 10

    while(True):
        ifnews_index == 0 andpage_index == 0:
            break;
index_url = driver.current_url
res = requests.get(index_url, headers=headers)
soup = BeautifulSoup(res.content, 'html.parser')
        fori in range(10):
            try:
article_url = soup.select_one('#newsColl > div.cont_divider > ul > li:nth-child('+
 str(i+1) +') > div.wrap_cont > span.cont_info > a:nth-child(3)')['href']
            except:
news_index == 0
page_index == 0

            ifnews_index == 0 andpage_index == 0:
                break;

article_title, article_content, isEconomy = crawling_article(article_url, article_
company)
time.sleep(2)
            ifisEconomy == 1:
                try:
                    if(title_lists[-1] != article_title):
title_lists.append(article_title)
company_lists.append(article_company)
url_lists.append(article_url)
content_lists.append(article_content)
count +=1
                except:
title_lists.append(article_title)
company_lists.append(article_company)
url_lists.append(article_url)
content_lists.append(article_content)
count +=1
            ifnews_index == 0 andpage_index != 0:
page_index -= 1
        try:
elem = driver.find_element(By.XPATH, "//*[@id='newsColl']/div[3]/span/span[3]/a")
elem.click()
        except:
            pass
        ifnews_index != 0:
news_index -= 1
time.sleep(sleep_sec)

result_tocsv(count, keyword, title_lists, company_lists, url_lists, content_lists)
time.sleep(3)
print("모든 크롤링이 완료되었고, 크롬 브라우저를 종료합니다.")
driver.quit()
```

결과

제목	언론사명	링크	내용
부동산 분	매일경제	http://v.da	2020년 처음 선보인 '리치고'는 빅데이터로 부동산의 가치와 시세를 분석하고 여
부동산 하	매일경제	http://v.da	부동산 시장은 불과 1년 새 분위기가 완전히 바뀌었다. 자동차로 치면 빠르게 달
부동산 값	매일경제	http://v.da	"금리가 떨어진다고 부동산이 오르는 것은 아닙니다. 금리가 떨어진다는 것은 경
변동·고정	매일경제	http://v.da	'부동산발 은행 위기 오나...미국, 사무실 공실률 최고' '미국 부동산 투자 늘어난
국회예정처	매일경제	http://v.da	국회예산정책처는 24일 '부동산 보유세 과세가격의 이슈 및 시사점' 보고서에서
[부동산 리	매일경제	http://v.da	"원래대로라면 시간조정 원칙이지만 경매는 그런 게 없으니까요. 세를 끼고 투

2. 프로젝트 수행을 위해 적용된 추진전략, 수행 방법의 결과를 작성하고, 만일 적용과정에서 문제점이 도출되었다면 그 문제를 분석하고 해결방안을 기술하시오.

- 추진전략 및 수행 방법

캡스톤 주제인 키워드 기반 콘텐츠 큐레이션을 역할 분담을 크게 데이터 수집 및 전처리를 2명, 인공지능 모델 구현 담당을 1명으로 구성했다. 먼저 크롤링은 다양한 언론사의 웹 환경을 조사하고, 이를 토대로 자신이 원하는 검색어에 대한 결과를 수집한다. 이에 맞춰 자연어 처리는 다양한 가사를 요약할 수 있는 모델을 이용하여 환경을 조성한 후, 모델의 파라미터를 조정하여 더욱 더 의미있는 텍스트를 생성하여 고객에게 제공하는 것을 목표로 하여 수행하였다.

- 결과 및 문제점 분석

역할을 분담하여 수행을 해본 결과, 데이터 수집에 두 명이나 수행하는 점이나 UI 등의 처리해야할 작업들의 역할 분담의 손실, 부실 등의 문제점이 생겨났다. 그리고 목표에 대한 팀의 이해도가 부족해 추진해야할 목표의 방향을 잡기가 어려웠다.

- 문제점 해결 방안

목표에 대한 좀 더 구체적인 요구사항과 캡스톤 주제에 대한 구체적인 피드백을 위해 멘토분과의 회의를 통해 이 주제에 대한 구제적인 요구사항과 피드백을 받아 ELT 프로세스 기반으로 역할 분담을 확실하게 하였고 캡스톤 주제에 대한 이해도가 증가하여 방향성을 확실히 정할 수 있게 되었다.

캡스톤 디자인 I 중간보고서 채점표

평가도구	평 가 항 목	평 가 점 수				
		1	2	3	4	5
중간 보고서 및 실행 결과	1. 요구사항 정의서(기능, 성능, 인터페이스 등)가 구체적으로 작성되었는가?					
	2. 요구분석, 설계 산출물(모델, 프로토타입 등)의 내용이 충실한가?					
	3. 설계 및 구현 문제를 위해 적용한 이론, 문제해결 방법이 제시되었으며 그 적용이 적합한가?					
	4. 구현된 소프트웨어(또는 이와 동등한 하드웨어 시스템)가 버그 없이 실행되었는가?					
	5. 구현된 소프트웨어(또는 이와 동등한 하드웨어 시스템)의 성능 요구사항은 충족되었는가?					
도구활용	6. 설계 및 구현을 위해 도구가 적절히 활용되었는가?					
	7. 도구의 활용수준(능숙도)은 프로젝트 수행에 적합한가?					
팀원의 업무 및 역할	8. 팀원의 업무분담에 따른 역할 및 협력이 충실히 이루어졌는가? (평가자에 의한 질의)					
	9. 프로젝트 중간 진척상황에 대해 팀원이 충분히 인지하고 있는가?(평가자에 의한 질의)					
합계						

***검토 의견(최종완료 때까지 보완해야할 점에 대해 작성 요망)**

심사위원(소속): **(이름)** **(인)**