졸업과제 착수 보고서

뉴스 중복 필터링 시스템

201724645 오상현

201424542 조현빈

201624422 김동주

목차

1 과제 배경 및 목표

1.1 과제 배경

국내 신문기사를 접할 때, 활자로 된 종이신문으로 읽는 사람들도 있지만 대다수의 사람들은 인터넷을 통해 신문 기사를 접하고 있다. 인터넷의 경우, 많은 언론사에서 보도한 비슷한 제목의 기사들을 한꺼번에 보여줌으로 뉴스 소비자 입장에서는 보았던 기사들도 중복해서 보게 되는 비효율적인 측면이 있다. 또한 자신이 원하지 않는 언론사의 기사들도 보게 됨으로 소비자 입장에서 불편하다. 현재 앱스토어에 여러가지 뉴스 앱들이 있지만 보완할 점이 많이 있어 이를 개선하고자 새로운 뉴스 어플리케이션의 개발이 필요하다.

표 1 - 애플 스토어에 등록된 뉴스 App 기능 비교

기능	NewsDaily	오늘의 헤드라인	뉴썸	다음 뉴스
User Interface	오래됨	최신	최신	최신
속도	느림	빠름	빠름	빠름
언론사 필터링	X	O	O	O
기사 중복제거	O	O	O	O
연관 링크 제공	O	X	X	O
즐겨찾기	O	X	O	O
카테고리	O	O	O	O
댓글	X	O	X	O
Night mode	O	O	O	O
폰트 사이즈 조절	O	X	O	O

1.2 과제 목표

본 졸업 과제는 자연어 처리를 통해 뉴스 유사도를 측정하여 중복된 기사들을 제거하며 사용자의 편의성을 위한 뉴스 서비스 플랫폼 개발에 목표를 둔다.

- IOS 어플
  - 기존 PC Web 으로 구현되어 있는 기능 중 필요한 기능들을 모바일 애플리케이션으로 개발하여 뉴스를 소비하는 사용자들이 스마트폰으로도 뉴스 소비를 가능하게 하여 사용자의 접근성을 높인다.

- 자연어 처리를 통해 뉴스 클러스터링 기능 구현
  - 여러 기사들의 유사도를 바탕으로 분류 및 군집화 후, 여러 필터 값 들로 보다 직관적으로 뉴스를 볼 수 있게 한다.
- 사용자 편의성을 위한 기능 구현
  - 사용자의 기호에 맞는 언론사의 기사들을 볼 수 있는 언론사 필터링 기능 제공
  - UX 를 보다 좋게 만들어, 여러 소비자층을 아우를 수 있게 한다.

2 요구 조건 분석

2.1 언론사별 크롤링

- 주요 언론사 선택(10 가지 정도)
- 인지도가 낮은, 마이너한 언론사의 경우 좋은 기사인데도 불구하고 제외될 수 있는 문제점이 있다.

2.2 데이터 전처리

- 자연어 처리를 위한 데이터 전처리
- 여러 번 시도를 하며 가장 적절한 형태를 찾는 것이 중요.

2.3 자연어 처리

- 유사도를 통한 뉴스 중복 확인
- Clustering
- 2 차 가공을 통한 뉴스 합치기

2.4 서비스 제공

- IOS 어플을 통한 서비스 제공
- 중복 처리된 뉴스 제공 및 검색 지원, 언론사별 filtering 지원.

3 현실적 제약 사항 및 대책

3.1 기사의 저작권

- 기사를 크롤링, 2 차가공, 재배포를 하는 행위는 저작권에 위배가 된다.
  - 당장은 기사 및 어플을 배포하지 않는 방향으로 졸업 과제를 마무리할 예정이다.
  - 만들어진 어플을 실제로 사용할 시에는 각 언로사에 뉴스 저작권료를 지불해야한다.

3.2 기사의 정당성

- 기사를 임의로 2 차가공을 하면 그 기사에는 정당성을 만족하는가?
  - 원본의 기사를 최대한 보존하는 식으로 가공 예정.

3.3 언론사별 성향

- 기사를 합쳤을 때 언론사별 성향이 유지가 되는가?
  - 원본의 기사를 최대한 보존한다 하더라도, 여러 기사를 합칠 경우에는 언론사별 특징이 사라질 수 있다.
  - 언론사별 필터링 기능을 제공하여 원하는 언론사를 볼 수 있게 할 예정.

4 설계 문서

4.1 개발 환경

- 개발 언어
  - Python(자연어 처리, server, crawling), Swift(IOS)

- 개발 도구
  - Xcode(Client), Django REST Framework(Server Framework)
  - KoNLPy(형태소, 구문 분석), khaiii(형태소 분석), Mecab(형태소 분석)
  - TensorFlow(기계 학습)
- 실행 환경
  - AWS(Server), IOS(Client)

- 데이터 베이스
  - Mongo DB

4.2 사용 기술

- 크롤링 – Selenium

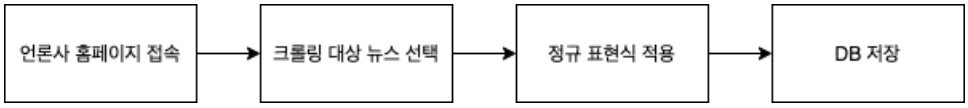


그림 1 – 크롤링 순서도

  - 셀레니움은 본래 웹 크롤링을 위해 만들어지지 않았다. 웹 애플리케이션 자동화 테스트를 위해 만들어진 Software 로 사용자를 대신하여 브라우저 환경에서 작업을 수행하는 데 사용할 수 있다. 이 방식은 code 를 활용해서 정보를 수집하는 방식을 차단한 웹 사이트로 부터 정보를 가져올 수 있게 한다. 또한 javascript 데이터에 추가적인 라이브러리 사용없이 직접 액세스가 가능하다. 또한 준수한 성능을 가지기에 추가적인 작업에 있어서 편리하고 성능 또한 준수한 Selenium 을 크롤링을 위한 라이브러리로 선택하였다.

- Data preprocessing – KoNlpy, khaiii, Mecab

  - KoNLPy 는 단어를 품사 형태로 딕셔너리를 정의하고 이를 이용해 단어를 품사로 분리하는 방법을 사용한다. 총 5 가지의 형태소 분석 방법을 제공하고 이는 Hannanum, Kkma, Komoran, Mecab, Okt 5 가지 클래스로 제공되고 이 중 Mecab 의 성능이 가장 좋다 평가 받는다. Mecab 은 우분투와 MacOS 환경에서 Mecab-python 설치를 통해 사용 가능하다.

  - Khaiii 는 규칙 기반으로 동작하는 것이 아닌 데이터 기반으로 동작하기 때문에 기계학습 알고리즘(딥러닝)을 사용한다. 기계학습을 위해서 신경망 알고리즘들 중 Convolutional Neural Network 를 사용한다. github 에서 소스코드를 받아 설치 후 파이썬에 바인딩하여 사용 가능하다.

- TensorFlow

  - 딥러닝, 머신러닝 등의 기계학습에 사용되는 TensorFlow 는 Python, java, C++등 다양한 언어에 API 를 제공해주며, 특히 python 에 강력한 생산성을 보여주고 있다. 파이썬을 이용해 서버 및, 크롤링 작업을 하기 때문에 기사의 유사도 측정 및 군집화에 TensorFlow 를 쓰는게 가장 강력할거 같아, 본 과제에서는 TensorFlow 를 사용할 계획이다.

4.3 프로세스

4.3.1 Server

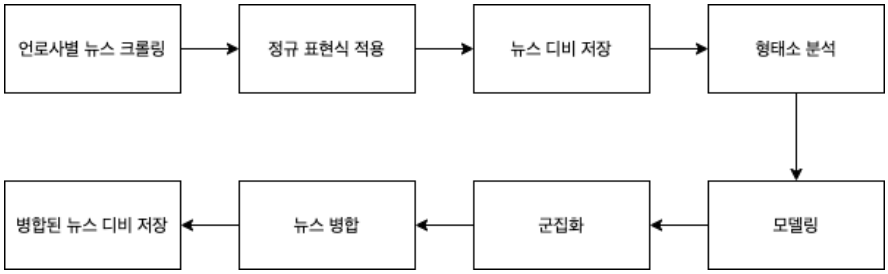


그림 2 - 뉴스 병합 Server 순서도

4.3.2 Client

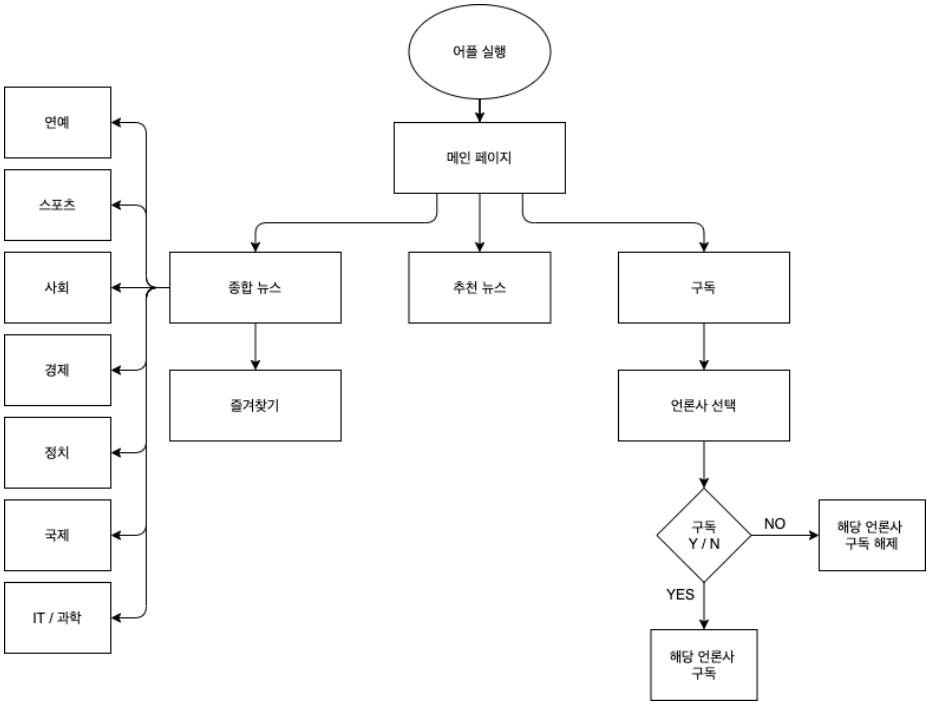
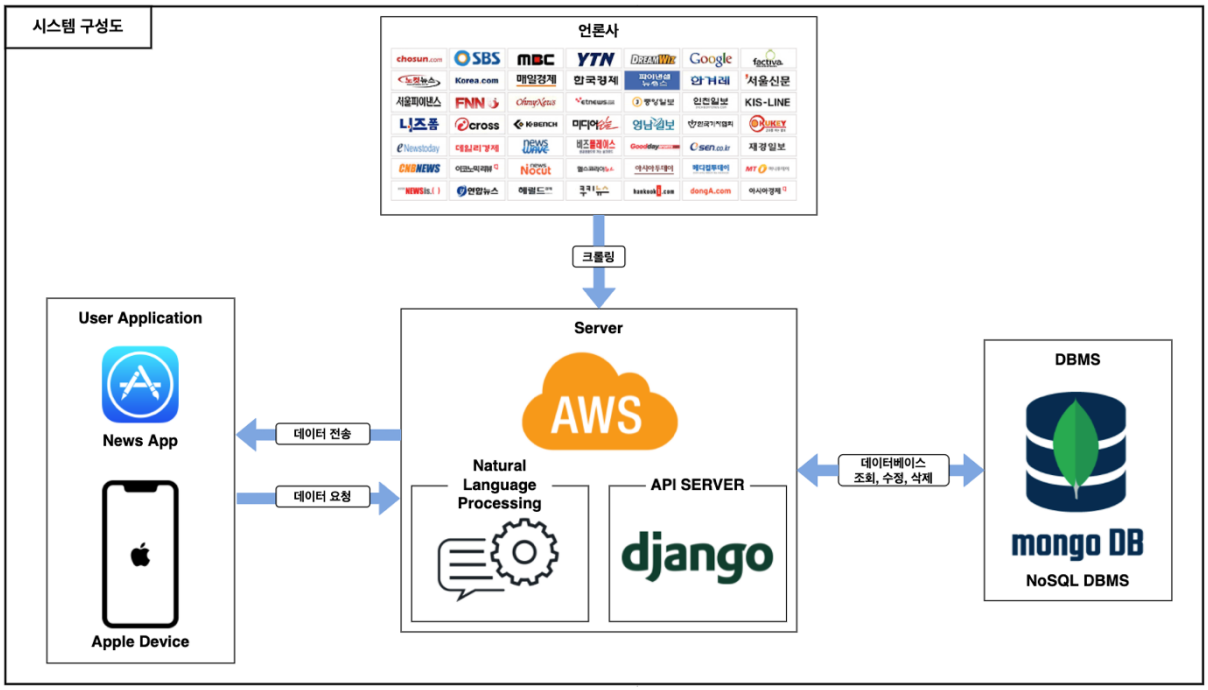


그림 3 - 뉴스 App 순서도

4.4 전체 구상도



5 개발 일정 및 역할 분담

5.1 개발 일정

5월			6월					7월					8월					9월	
3주	4주	5주	1주	2주	3주	4주	5주	1주	2주	3주	4주	5주	1주	2주	3주	4주	5주	1주	2주
착수보고서																			
	API 및 관련기술 공부																		
						서버 환경 구축													
								서버 개발											
								크롤링 모듈 개발											
								데이터 베이스 개발											
											자연어 처리 모듈 개발								
											아이폰용 어플리케이션 개발								
													테스트 및 디버깅						
																	최종 발표/보고서 준비		

5.2 역할 분담

이름	역할 분담
오상현	● 모바일 어플리케이션 개발 ● 웹 크롤링 모듈 개발
조현빈	● 데이터 베이스 개발 ● 웹 크롤링 모듈 개발
김동주	● 모바일 어플리케이션 개발 ● 서버 개발
공통	● 자연어 처리 ● 시스템 테스트 및 성능 평가 ● 보고서 작성 및 발표