4 차 4 기 실무 프로젝트

# News_ight

뉴스 자동분류 및 대표기사 추출 서비스

**2019 년 05 월 24 일**

**딥러닝 기반의 업종별 빅데이터 예측 전문가**

Team Newsight

김현호

노성문

안성윤

백승현

# 목차

# 1. 프로젝트 개요

## 1.1. 팀 구성원 및 역할 소개

| 이름 | 전공 | 역할 | 구현 부분 |
|---|---|---|---|
| 노성문 | 행정학과 | 팀장 | 프로젝트 관리<br>웹 개발 |
| 김현호 | 수학과 | 팀원 | 알고리즘 구현<br>기능 모듈 개발 |
| 백승현 | 건축공학과 | 팀원 | 워크플로우 구현<br>기능 모듈 개발 |
| 안성윤 | 건축학과 | 팀원 | 데이터 크롤링<br>웹 개발 |

**표 1 팀 구성원 및 역할**

## 1.2. 기획 배경 및 목표

이력서를 작성할 때 가장 많은 시간을 할애하는 작업 중 하나는 기업 조사다. 일반적으로 기업 관련 기사나 기업공시를 참조해 기업에 대한 이슈를 정리한다. 특히 기사를 보고 이슈를 파악하는 작업은 많은 시간이 소요된다.

기존 뉴스 검색 엔진은 관련도, 최신순을 기준으로 뉴스를 출력한다. 이와 같은 검색 엔진의 문제는 1) 대표 이슈를 한눈에 알아보기 힘들며, 2) 비슷한 내용의 기사를 모두 출력한다는 점이다. 예컨대, 2019년 4월 26일 13시 기준 검색 포털 '네이버'에서 '삼성전자' 뉴스를 검색하면 236만건 이상의 기사가 출력된다. 이 중 비슷한 기사를 확인하고, 대표 이슈를 확인하는 일은 상당한 시간이 필요한 작업이다.

이에 검색된 기사의 1) 대표 이슈를 추출하고 2) 시간순으로 이슈에 대한 대표 키워드를 시각화하며 3) 각 이슈에 대한 대표 기사를 추천해주는 서비스를 만든다면 위와 같은 어려움을 해결할 수 있을 것이라 예상한다.

## 1.3. 추진 계획

| 구분 | 기간 | 활동 | 비고 |
|---|---|---|---|
| 사전<br>기획 | 4/26~4/30 | 프로젝트 기획 및 팀 구성 | |
| | 4/26 | PJT주제 선정, 팀(PM/팀원) 구성 | |
| | 4/29 | 프로젝트 멘토링 | 현업 멘토 참여 |
| PJT<br>수행<br>/<br>완료 | 4/30 ~ 5/2 | 데이터 크롤링 | |
| | 5/2~5/9 | 데이터 탐색적 분석<br>전처리 | |
| | 5/9~5/19 | 알고리즘 선정 및<br>프로토타입 구현 | |
| | 5/19~5/23 | DB 구축 및 웹 개발 | |
| | 5/23 | 최종 발표 | |

**표 2 추진 계획**

# 2. 프로젝트 현황

## 2.1. 유사 서비스 분석

유사 서비스 분석을 위해 뉴스 큐레이션 및 분석 사이트와 구직 사이트를 조사했다.

| 서비스명 | 기본 기업 정보 | 뉴스 분석 서비스 | 서비스 구분 |
|---|---|---|---|
| 잡코리아 | 있음 | 제한된 언론사, 단순 큐레이션 | 구직 |
| 사람인 | 있음 | 없음 | 구직 |
| 인크루트 | 있음 | 없음 | 구직 |
| 네이버 뉴스 | 없음 | 군집화 | 뉴스 |
| 네이버 증권 | 있음 | 군집화 | 뉴스 |
| 빅카인즈 | 없음 | 군집화 | 뉴스 |

**표 3 유사 서비스**

일반적으로 구직사이트는 연혁, 월매출, 인재상등 기본정보를 제공하지만 뉴스 분석 서비스는 미비한 것으로 드러났다. '잡코리아'는 'NICE 평가정보'를 통해 '매일경제'와 '한국경제' 뉴스를 제공한다. 그러나 언론사가 2개밖에 없고 별도의 필터링 알고리즘을 적용하지 않아 이슈 요약, 중복 제거 기능은 활용할 수 없다. '사람인', '인크루트'는 뉴스 분석 서비스를 제공하지 않는다.

검색포털 '네이버'는 관련도순, 최신순 정렬기능을 제공하며 비슷한 기사를 군집화해 보여준다. 다만, 카테고리 기준으로 군집화하지 않으며 시간순으로 정렬돼있지 않아 이슈 흐름을 읽기 어렵다. 네이버 증권 분석 서비스 '네이버 증권'은 인공지능(AI) 기반 뉴스 군집화 알고리즘을 적용해 비교적 준수한 뉴스 군집 서비스를 제공한다. 그러나 눈으로 확인한 결과 중복된 기사가 많이 존재했다.

본 프로젝트의 지향점과 가장 유사한 서비스는 한국언론진흥재단 빅데이터 분석 서비스 '빅카인즈(BigKinds)'다. 빅카인즈는 인물관계도, 시간대별 기사빈도 분석 등 다양한 뉴스 분석 서비스를 제공한다. 특히 검색어에 따라 시간별 주요 뉴스를 정렬해 보여주는 '시간대별 주요 이슈' 서비스도 제공한다.

다만 뉴스 군집화가 세부적이지 않아 모든 카테고리의 뉴스를 섞어서 보여준다는 단점이 있다.

예컨대 '승리'에 대한 시간대별 주요 이슈 서비스는 연예인 승리, 스포츠팀의 승리 등 관련 없는 기사를 한꺼번에 출력하는 경우도 있다. 뉴스 군집화가 진행됐지만 세분화되지 않은 것으로 보인다.

### 2.2. 차별화 핵심전략

유사 제품군은 기본적인 기업정보를 제공하지만 뉴스를 통해 기업의 이슈를 정리해주지는 않았다. 빅카인즈 시간대별 주요 이슈 서비스는 분류가 세분화되지 않아 한눈에 이슈를 파악하기 어렵다.

Newsight의 차별화 핵심전략은 다음과 같다.

뉴스 분류를 2단계로 자동분류한다.

1) 토픽별 대표 기사를 추출해 중복 기사를 제거한다.
2) 토픽을 워드클라우드로 출력해 빠르게 정보를 제공한다.
3) 대표 기사 빈도를 시간순으로 출력해 기사 중요도를 제공한다.

## 3. 프로젝트 결과

Newsight는 기사를 2레벨(**카테고리**, **토픽**)로 자동분류하는 것이 가장 큰 특징이다. 또한, **토픽별 대표기사**를 추출해 중복기사를 제거함으로써 사용자가 원하는 기업에 대한 정보를 빠르게 제공한다.

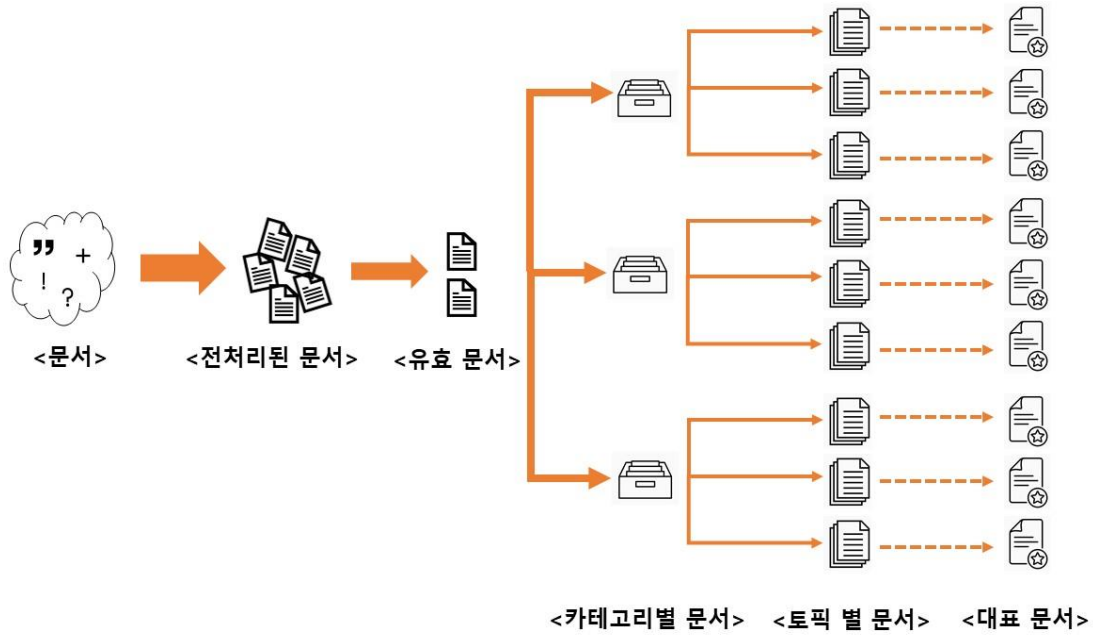



**그림 1 처리 단계별 문서 형태**

## 3.1. 주요 기능 명세

Newsight는 1) 전처리 2) 기사 자동 분류 3) 대표 기사 추출 4) 시각화 순으로 동작한다. 주요 기능 명세와 이에 따른 설명은 다음과 같다.

| 코드 | 주요기능 명 | 설명 |
|---|---|---|
| MDB | 전처리 | 불용어, 유의어 등을 처리하고 유효문서를 추출한다. |
| MB1 | 기사 자동 분류 | 기사 말뭉치(Corpus)를 2단계(카테고리, 토픽)로 분류한다. |
| MB2 | 대표 기사 추출 | 토픽별 대표 기사를 추출한다. |
| MB3 | 시각화 | 군집화 결과를 시각화한다. |

**표 4 주요 기능 명세**

## 3.2. 산출물 명세

| 코드 | 산출물 명 | 형식 |
|---|---|---|
| R1 | 전처리 모듈 | ipynb, py |
| R2 | 기사 자동분류 및 대표기사 추출 모듈 | ipynb, py |
| R3 | 시각화 모듈 | ipynb, py |
| R4 | 기능 모듈 | py |
| R5 | 웹페이지 | html |

**표 5 산출물 명세**

## 3.3. 주요 동작 명세

| 코드 | 주요 동작 명 | 알고리즘 | 설명 |
|---|---|---|---|
| MDB-S1 | 유의어, 불용어, 유효명사 처리 | - | 유효명사를 추가하고 불용어,유의어 처리한다. |
| MDB-S2 | 유효 문서 추출 | 키워드 | 검색어(Query)를 통해 검색된 문서 중 유효 문서를 추출한다. |
| MB1-S1 | 기사별 카테고리 할당 | Top N | 기사별 카테고리(정치,사회,IT_과학,경제,지역,스포츠)를 할당한다. |
| MB1-S2 | 카테고리내 기사별 토픽 할당 | OPTICS | 비지도학습 알고리즘으로 카테고리내 문서를 분류하고, 각 군집을 하나의 토픽으로 정의한다. |
| MB2-S1 | 토픽별 키워드 추출 | Top N | 토픽별 키워드를 추출한다. |
| MB2-S2 | 토픽별 대표기사 추출 | BM25 | 토픽별 대표 기사를 추출한다. |
| MB3-S1 | 카테고리 워드클라우드 출력 | - | 카테고리를 워드클라우드로 출력한다. |
| MB3-S2 | 카테고리 내 기사 빈도 시계열 그래프 출력 | - | 카테고리 내 기사 빈도를 시계열 그래프로 출력한다. |
| MB3-S3 | 토픽 워드클라우드 | - | 토픽 내 주요 키워드를 워드 클라우드로 출력한다. |

**표 6 주요 동작 명세**

## 3.4. 주요 동작 상세

주요 동작 순서는 다음과 같다.

**[Web, Server]**

1) 사용자가 검색창에 검색어를 입력한다.
2) 검색어에 대한 데이터를 가져온다.
3) 유효 문서를 추출한다.
4) 카테고리를 할당한다.
5) 카테고리별 토픽을 할당한다.
6) 토픽별 대표 기사를 추출한다.
7) 분석 결과를 시각화해서 웹페이지에 출력한다.

**[Database]**

1) 뉴스데이터를 실시간 크롤링한다. (미구현)
2) 유효명사, 불용어, 유의어 처리한다
3) 본문을 토큰화(Tokenize)한 후 데이터베이스에 저장한다.

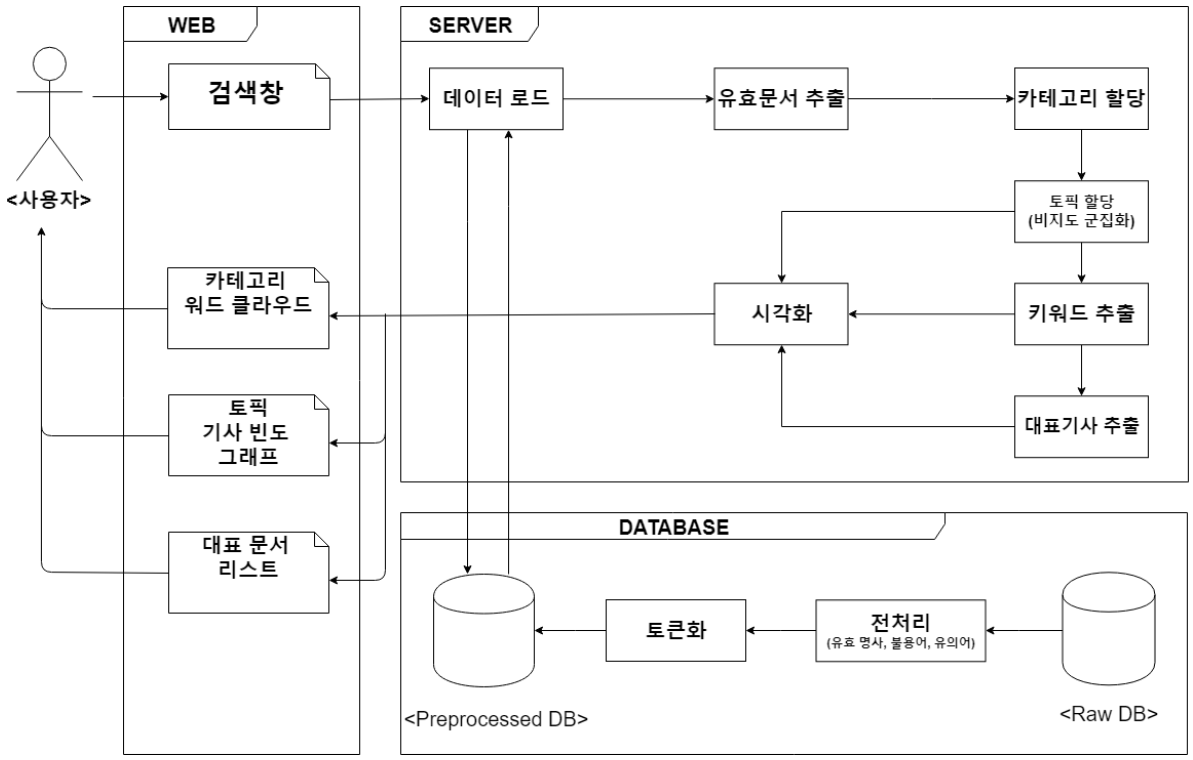



**그림 2 시스템 흐름도**

처리 단계별 데이터 스키마는 다음과 같다.

| 단계 | Features | 자료형 |
|---|---|---|
| 원본<br>(Raw DB) | Date | String |
| | Title | String |
| | Content | String |
| | Category | List_String |
| 전처리 후<br>(Preprocessed DB) | Date | String |
| | Title | String |
| | Content | String |
| | Tokenized Content | List_String |
| 자동 분류 후 | Date | String |
| | Title | String |
| | Content | String |
| | Tokenized Content | List_String |
| | Selected Category | String |
| | Topic | Integer |

**표 7 데이터 스키마**

### 3.4.1. 문서 분류 및 대표 기사 추출

#### 3.4.1.1. 전처리

전처리 과정은 불용어처리, 유의어처리, 유효명사처리, 토큰화로 구성된다. 불용어는 특수문자 등의 '일반불용어'와 검색어에 따른 '특수불용어'로 구분했다. 일반불용어는 '기자 이름', '이메일', '알파벳 소문자', '한자', '특수문자', '날짜', '시간', '숫자'로 정의했다.

```
def __init__(self):
        # 기자
        self.reg_reporter = re.compile('[가-힣]+\s[가-힣]*기자')
        # 이메일
        self.reg_email = re.compile('[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\.[a-zA-Z0-9-.]+$')
        # 알파벳 소문자
        self.reg_eng = re.compile('[a-z]+')
        # 한자
        self.reg_chi = re.compile("[\u4e00-\u9fff]+")
        # 특수문자
        self.reg_sc = re.compile("·|...|◆+|◇+|▶+|●+|▲+|"|"|'|'|\"|\'|\(|\)|\W+")
        # 날짜,시간,숫자
        self.reg_date = re.compile('\d+일|\d+월|\d+년|\d+시|\d+분|\(현지시간\)|\(현지시각\)|\d+')
```

**특수불용어**는 검색결과 **최빈 단어**와 **의미가 약한 단어**를 의미한다. 예컨대 '미래에셋대우'로 검색하면 모든 문서에 '미래에셋대우'가 출현한다. 문서를 벡터화할 때 차원이 너무 커지는 것을 방지하기 위해 미리 제거한다.

**의미가 약한 단어**란 '최근', '그러나', '한편', '이상' 처럼 다른 문장과 접하면 의미를 줄 수 있으나 그 자체로는 의미가 약한 단어를 말한다. 문서 벡터화시 최소 DF(Document Frequency)를 조정하면 제거할 수 있으나, 그 기준점을 일괄적으로 적용할 수 없어 미리 제거했다.

**유효명사**란 특수불용어와 반대되는 개념으로, 주제를 분류할때 반드시 필요하지만 토큰화 모듈이 명사로 인식하지 못하는 단어를 말한다. 한국어 자연어처리 모듈 KoNLPy의 Twitter 클래스(Class)는 문장에서 보통명사를 추출하는 기능을 제공한다. 그러나 특정 사람,사물의 이름 등 고유명사를 명사로 인식하지 못하는 문제가 있다. 예컨대 '문재인'은 ['문재','인'] 으로, '미래에셋대우'는 [ '미래에셋','대우']로 인식하는 등 고유명사를 명사의 조합으로 판단하는 경우가 있다. 기사는 특정 도메인에 기반한 고유명사를 많이 포함하고 있어 이를 유효명사로 추가해주는 작업이 필요하다.

**토큰화**는 문장을 단어로 분절하는 작업을 말한다. 한국어는 조사를 결합해 명사를 활용하기 때문에 단순히 띄어쓰기 기준으로 토큰화를 할 경우 문서 벡터의 차원이 커질뿐만 아니라, 같은 의미의 단어를 다른 단어로 정의하게된다. 또한 교착어의 특성상 어근화(Stemming)를 하지 않으면 문서 벡터의 차원이 커지는 문제가 발생한다. 예컨대 '미래에셋대우는', '미래에셋대우가', '미래에셋대우를'은 모두 '미래에셋대우'를 의미하지만 띄어쓰기로 토큰화할 경우 다른 단어로 정의된다.

이에 품사 정보를 담고있는 KoNLPy Twitter 클래스를 활용해 문장에서 명사만 추출했다. 명사만

추출하면 서술어를 어근화할 필요가 없으며, 조사를 제거해 단어 벡터 차원을 최소화할 수 있다.

토큰화작업은 비교적 시간이 오래걸린다는 단점이 있다. '미래에셋대우'의 1049건 기사에 대해 토큰화를 테스트해본 결과 실행시간은 약 14초로 전체 처리단계 소요시간의 약 90%에 달한다. 실제 서버를 구축한다면 데이터베이스 서버에서 별도로 토큰화작업을 진행해야 할 것으로 보인다.

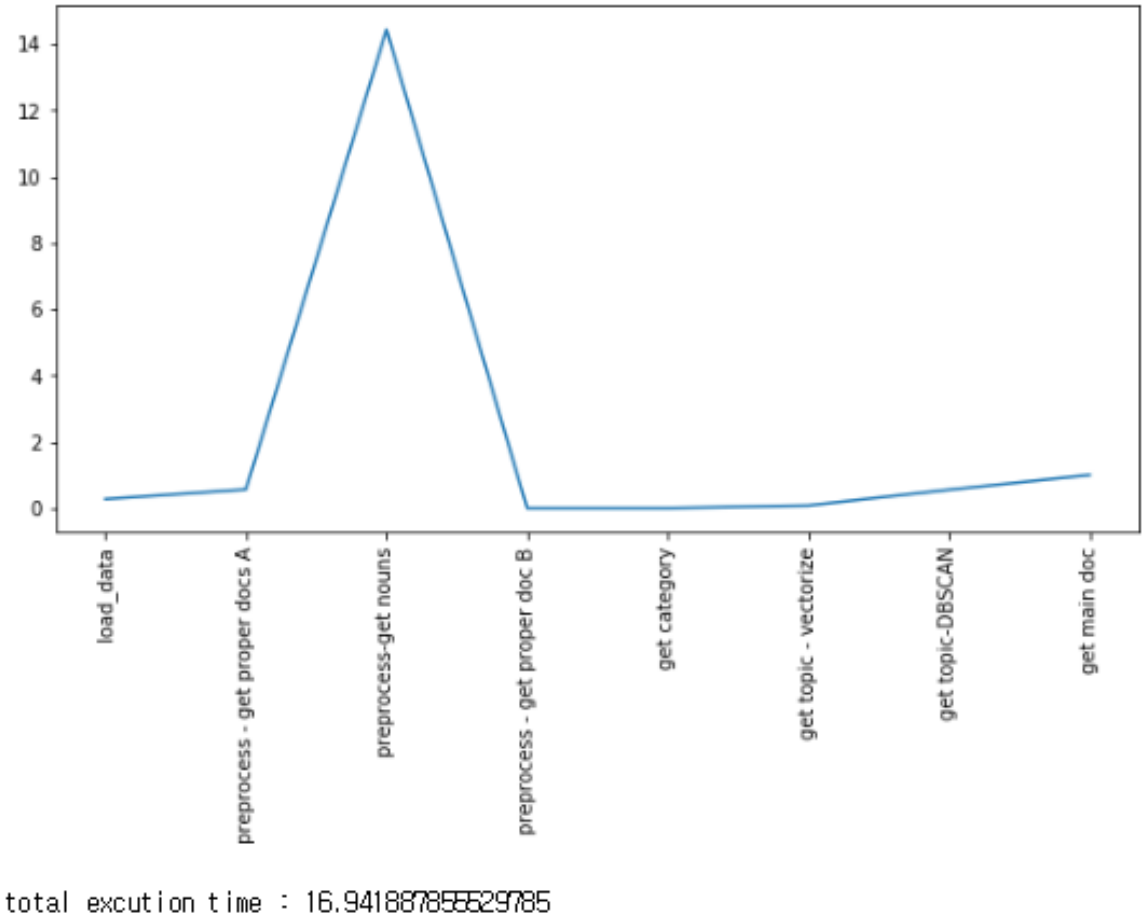



**그림 3 처리 단계별 소요 시간**

3.4.1.2. 유효문서 추출

**유효문서**란 '제목에 검색어를 포함하면서 본문에 검색어가 3회 이상 출현한 문서'로 정의한다. 본문에 검색어 출현 횟수를 3회 이상으로 제한한 이유는 해당 검색어에 관한 기사라면 서론, 본론, 결론에 검색어를 한번 이상 포함할 것으로 예상한 까닭이다. 한국언론진흥재단 빅데이터 분석 서비스 '빅카인즈(Big Kinds)' API는 본문에 검색어를 포함한 모든 문서를 가져온다. 검색어와 관련없는 주제의 문서를 제거하기 위해 유효문서를 추출했다.

3.4.1.3. 문서별 카테고리 할당

**카테고리**는 '정치','사회','경제' 등 일반적으로 뉴스를 분류할 때 사용하는 대분류를 말한다. 한국언론진흥재단에서 지정한 뉴스 통합 분류체계 값을 카테고리로 활용했다. 한국언론진흥재단은 카테

고리의 하위 분류 체계인 사건/사고 분류체계도 제공하지만 결측치가 대부분이라 활용할 수 없었다.

한국언론진흥재단에서 제공하는 카테고리와 문서는 일대일대응 관계가 아니다. 문서 하나당 최대 3개의 카테고리를 가지며, 2단계로 구성돼(정치>북한) 전처리가 필요하다. 문서 하나당 하나의 카테고리를 할당하기 위해 다음과 같은 방법을 적용했다.

1) *세부 카테고리를 정규표현식으로 제거한다.*
2) *빈도가 가장 높은 카테고리를 major category로 지정하고, 나머지를 minor category로 지정한다.*
3) *카테고리를 할당한다.*
    - *3-1) 카테고리 리스트의 길이가 1일 경우 첫번째 원소를 선택한다.*
    - *3-2) 카테고리 리스트의 길이가 2이상일 경우*
        - *3-2-1) major category 만 있으면 major category를 선택한다.*
        - *3-2-2) minor category 가 하나라도 있으면 minor category 중 전체 출현빈도가 가장 높은 카테고리를 선택한다.*

3.4.1.4. 카테고리 내 문서별 토픽 할당

토픽은 카테고리 하위 개념을 말한다. 예컨대 '문재인'으로 검색한 '정치'라는 카테고리에'남북 정상회담', '패스트트랙 법안 상정' 등의 토픽이 있을 때 정치는 일반어, 토픽은 특수어에 대응한다.

뉴사이트 토픽 분류 알고리즘은 세가지 조건을 만족해야 한다. 먼저 카테고리별 토픽 수를 모르기 때문에 군집 수를 지정하면 안된다. 또한 실시간 서비스를 위해 처리속도가 빨라야 하며, 검색어간 분류성능이 균일해야하기 때문에 파라메터가 적어야 한다. 이에 밀도 기반 비지도 군집화 알고리즘 OPTICS를 활용했다.

OPTICS는 엡실론(Epsilon)을 군집 밀도에 따라 차등 적용한다는것이 가장 큰 특징이다. 일괄적으로 엡실론을 적용하는 DBSCAN과는 다르게 군집 당 지역 엡실론(Local Epsilon)을 산출해 밀도가 다른 군집도 군집으로 처리할 수 있다.

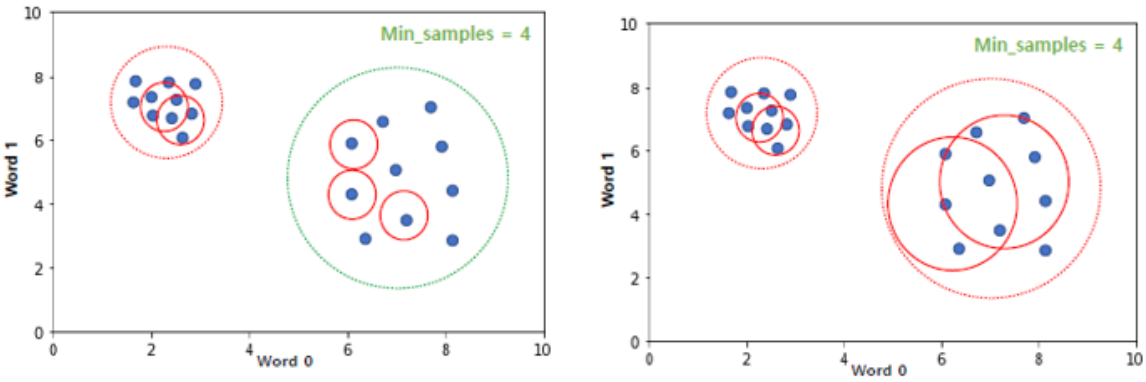
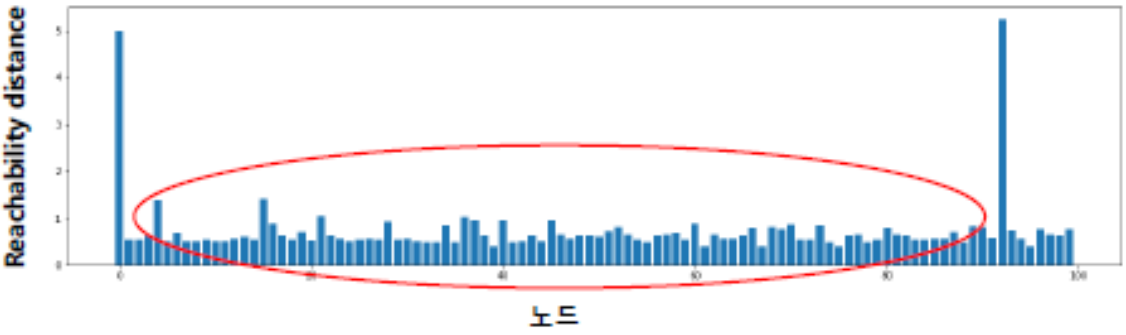



**그림 4 DBSCAN 군집화(좌), OPTICS 군집화(우) 모식도**

OPTICS는 Reachability distance, Core distance를 이용해 군집화하고, reachability-distance plot에서 골짜기 부분을 군집으로 정의한다.




.

**그림 5 OPTICS Reachability distance**

OPTICS는 다음 4가지 조건을 만족하는 구간을 군집으로 정의한다.

*1. 군집의 시작점은 '가파르게 내려가는 영역'의 한 원소다.*

*2. 군집의 끝점은 '가파르게 올라가는 영역'의 한 원소다.*

*3. 시작점과 끝점의 거리는 min_samples보다 커야하며, 원소들의 reachability-distance는 시작점의 reachability-distance와 끝점의 reachability-distance보다 작다.*

*4. '가파르게 내려가는 영역'의 시작점과 '가파르게 올라가는 영역'의 끝점을 비교하여, reachability-distance가 낮은 값을 기준으로 cluster의 시작점과 끝점을 결정한다*

일반적으로 OPTICS는 DBSCAN보다 1.6배 느린것으로 알려졌다. 문서 1000건으로 구동시간을 측정한 결과 OPTIC는 3.4초, DBSCAN은 0.8초를 기록했다. 처리속도가 느리지만 최소 군집수만 정해도 비지도 분류를 할 수 있다는 점에서 OPTICS를 채택했다.

3.4.1.5. 토픽별 대표기사 추출

토픽 내 **대표기사**는 **토픽 키워드**를 많이 포함하면서 길이가 짧은 문서로 정의했다. 길이가 짧을수록 효율적으로 정보를 전달하는 문서라고 가정했다. **토픽 키워드**란 토픽 내 말뭉치에서 가장 많이 출현한 단어 10개를 말한다.

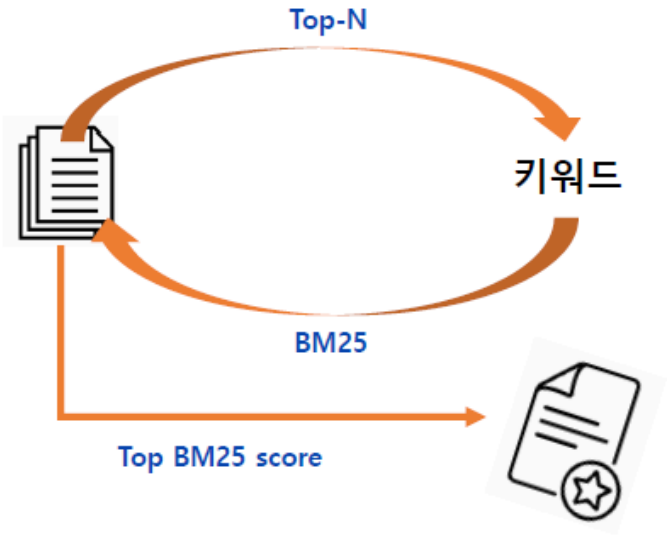



**그림 6 대표기사 추출 과정 모식도**

이에 오픈소스 검색엔진 엘라스틱서치(Elastic Search)의 검색 알고리즘 BM25를 사용했다. BM25는 문서 랭킹 방법중 하나로, 키워드와 문서의 상관성을 계산해 점수를 산출한다. 토픽 키워드에 대해 BM25 점수가 가장 높은 문서를 대표문서로 선정하는 방식이다. 키워드에 대한 문서의 BM25 점수를 계산하는 식은 다음과 같다.

$$\text{score}(D,Q) = f(x) = \sum_{i=1}^{n} IDF(q_i)\frac{f(q_i,D)\cdot(k+1)}{f(q_i,D)+k\cdot(1-b+b\cdot\frac{D_l}{avg(D_l)})}$$

D : 문서
Q : 키워드 리스트
$q_i$ : i번째 키워드
n : 전체 문서의 수
k1, b : 파라메터

$IDF(q_i)$ : 전체 문서집합에 대한 $q_i$의 IDF
$avg(D_l)$: 문서 길이 평균
$D_l$ : 문서 D의 길이
$f(q_i,D)$ : D에서 $q_i$ 가 출현한 횟수

BM25 점수는 키워드가 문서에 많이 등장할수록(TF가 높을 수록), 문서의 길이가 짧을수록 높게 산출된다. BM25는 토픽 군집내 이질적인 문서를 걸러내는 역할도 한다. 토픽내 대부분의 기사와 주제가 다른 문서의 BM25점수는 TF값이 작기에 낮게 산출되는 까닭이다. 만약 군집화가 잘돼있다면 점수를 산출할 때 TF의 영향력이 줄어들고, 길이가 짧은 문서를 대표기사로 선정한다.

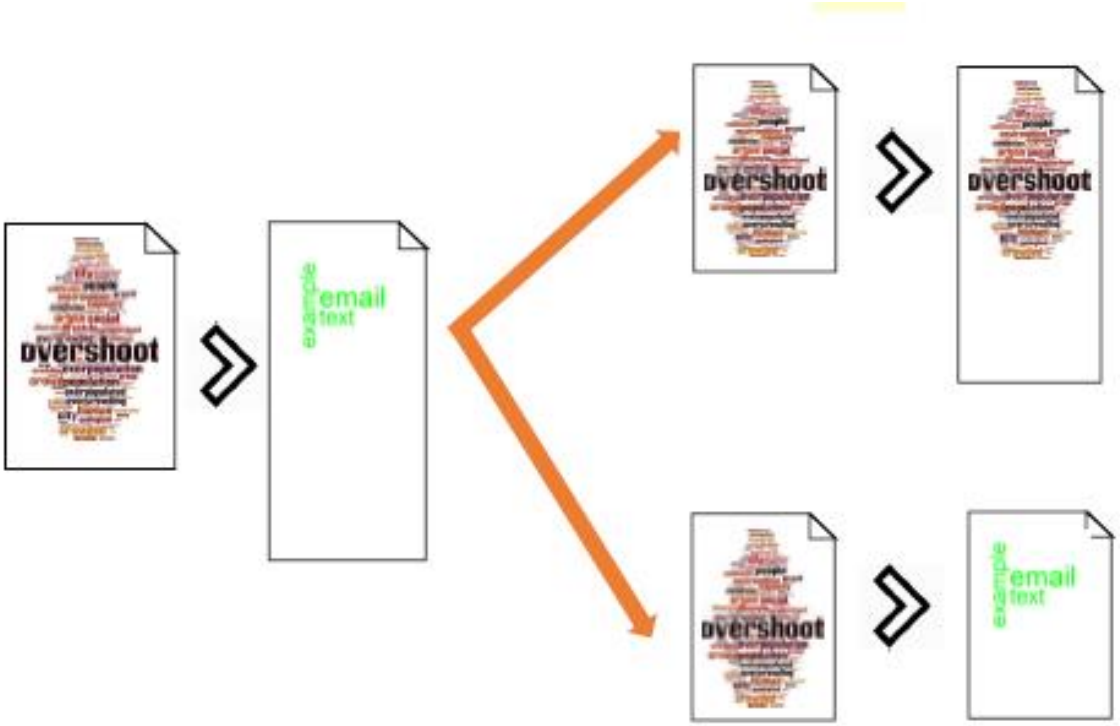



**그림 7 BM25 대표기사 선정방식**

## 3.5. 웹 구현 및 시각화

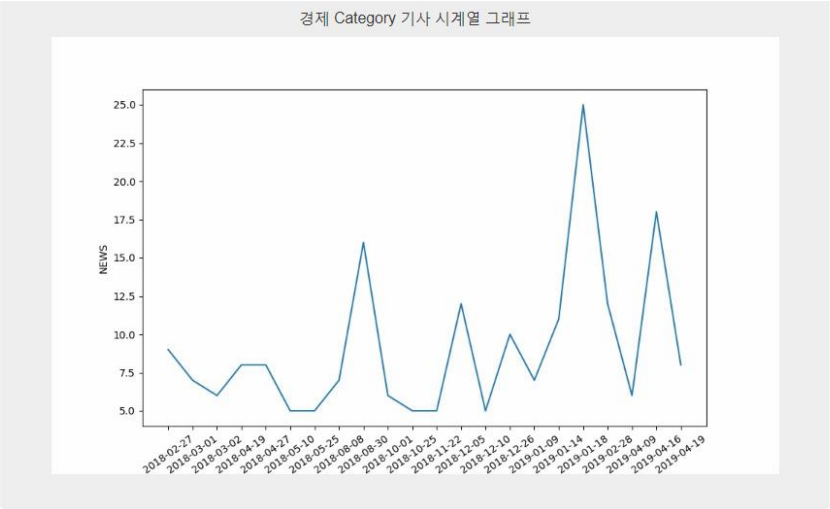
MariaDB로 데이터베이스를 구축하고 Django를 활용해 웹페이지를 구현했다.


**그림 8 메인 페이지**

검색어를 입력하면 카테고리 워드클라우드와 검색어에 대한 뉴스 빈도 시계열 그래프를 출력한다.


**그림 9 '미래에셋대우' 검색 결과. 카테고리 워드클라우드(위), 전체 기사 빈도 시계열 그래프(아래)**

원하는 카테고리를 클릭하면 토픽별 기사 빈도 시계열 그래프와 대표기사 제목을 출력한다.




Topic별 대표 기사

| Topic | Date | Title |
|---|---|---|
| 1 | 2018-02-27 | 미래에셋대우, 7000억 유상증자 마무리…자기자본 8조 달성 |
| 2 | 2018-03-01 | 미래에셋대우, 인도현지법인 개업 "인도시장 본격 공략" |
| 3 | 2018-03-02 | 미래에셋대우 유상증자 호반건설, 500억원 참여 |
| 4 | 2018-04-19 | [Hot-Line] "미래에셋대우, 자사주 소각 결정 긍정적…중장기적 관심" |
| 5 | 2018-04-27 | 5조원대 홍콩빌딩 인수에 미래에셋대우 3200억 투자 |
| 6 | 2018-05-10 | 미래에셋대우 11년 만에 '분기 순익 2000억원' 돌파 |
| 7 | 2018-05-25 | 미래에셋대우, 올해 300명 채용 |
| 8 | 2018-06-10 | 미래에셋대우 과천사옥, 재개발 사업 '난기류' |

**그림 10 토픽별 기사 빈도 시계열 그래프(위), 토픽별 대표기사(아래)**

대표기사를 클릭하면 해당 토픽의 워드클라우드와 본문을 출력한다.

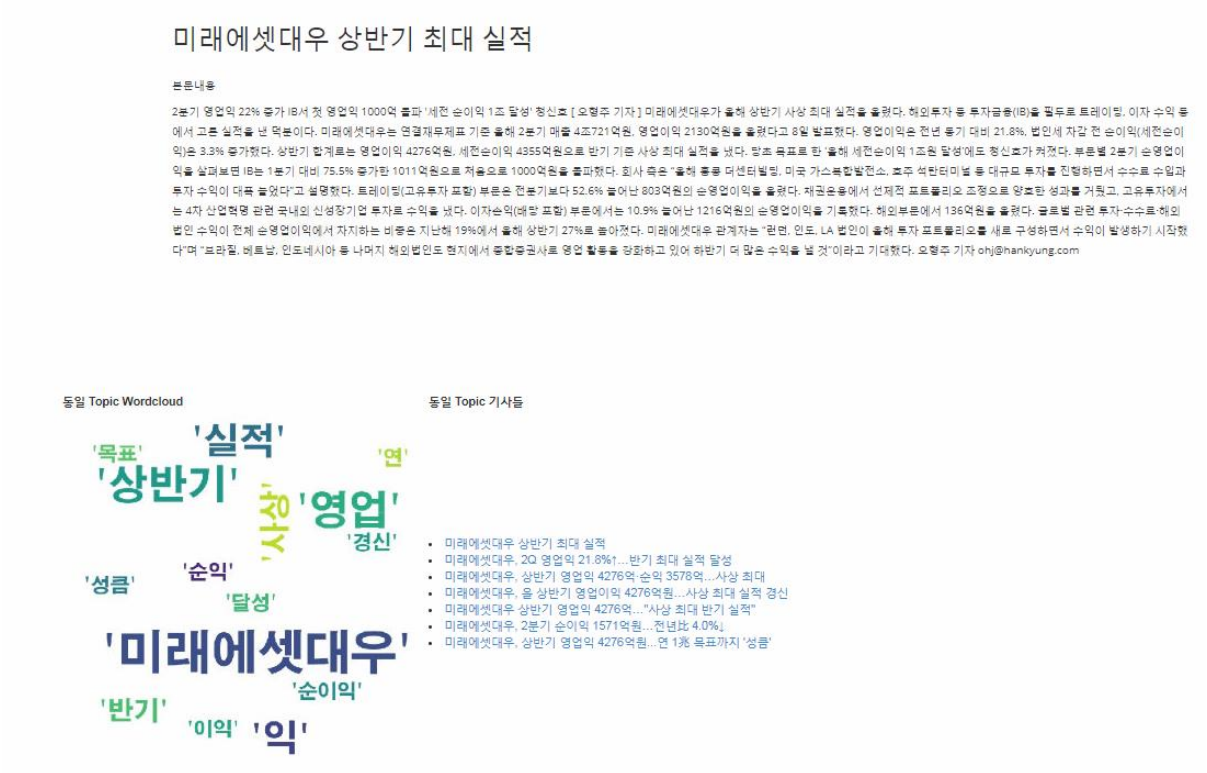



**그림 11 토픽 워드클라우드, 기사 본문(위)**

## 3.6. 단계별 알고리즘 후보 및 단점

### 3.6.1. 비지도 군집화

비지도 군집화 알고리즘은 1) 파라메터 2) 실행 시간 3) 분류 성능의 3가지 기준으로 선정했다. 실시간 비지도 자동 문서 분류를 목적으로 하기에 파라메터 수가 최소화돼야하며 빠르고 분류 성능이 좋아야 한다. 프로젝트 개발 중 테스트해본 비지도 군집화 알고리즘과 특징은 다음과 같다.

| 알고리즘 | Metric | Parameter | 시간복잡도 |
|---|---|---|---|
| LSA | Cosine | 주제 벡터 수 | $O(\min(mn^2, nm^2))$ |
| LDA | Probability | 군집 수 | $O(nm^2) \sim O(n^3)$ |
| KMeans | Euclidean | 군집 수 | $O(nmki)$ |
| Spherical KMeans | Cosine | 군집 수 | $O(nmdi)$ |
| Community Detection | Modularity | 엣지(Edge) 생성 기준값 | $NP - Class, over\ O(n^2)$ |

| | | | |
|---|---|---|---|
| DBSCAN | Cosine | 엡실론, 최소 군집 크기 | $O(nlogn)\sim O(n^2)$ |
| OPTICS | Cpsine | 최소 군집 크기 | $O(nlogn)\sim O(n^2)$ |

표 8 비지도 학습 알고리즘 (n : 데이터 수, m : 차원 수, k : 군집 수, i : 반복횟수)

3.6.1.1. LSA

LSA는 수직관계의 주제 벡터만 반환하며, 방향이 일정하지 않다. (180도로 회전할 수 있음) 이에 다양한 주제를 포함한 말뭉치를 분류하기에 적합하지 않을뿐더러, 코사인 유사도 기반으로 군집화 할 경우 높은 분류 성능을 기대할 수 없다.

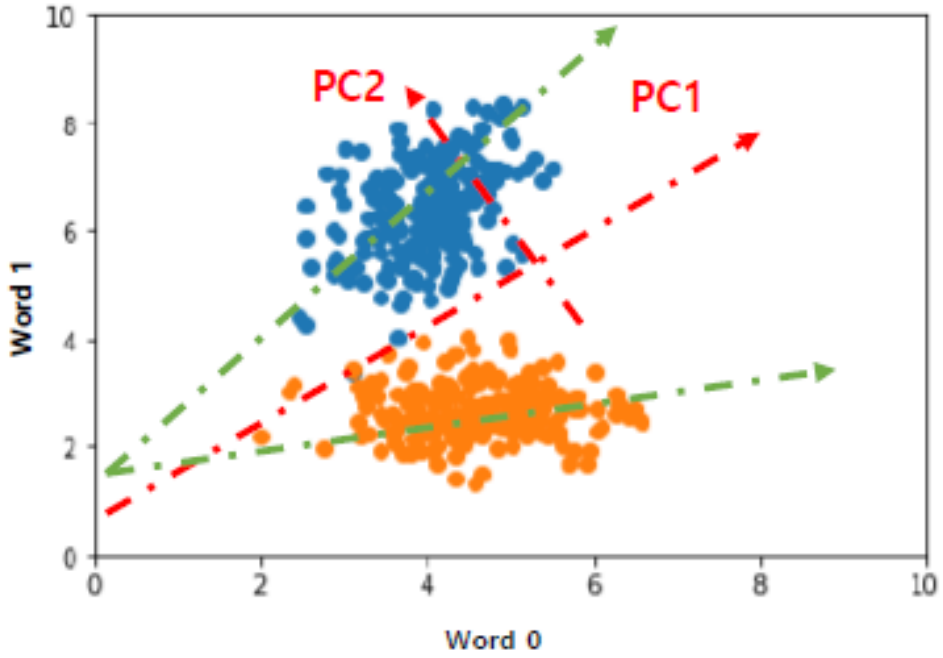



**그림 12 LSA로 추출된 대표 벡터와 예상 대표 벡터**

3.6.1.2. LDA

LDA도 역시 군집 수를 지정해야한다. 시간복잡도는 $O(nm^2)$로 실시간 서비스에 적용하기는 어렵다고 판단했다. 눈으로 확인한 결과 분류성능 역시 좋지 않았다.

3.6.1.3. KMeans, Spherical KMeans

Kmeans와 Spherical KMeans을 활용하려면 군집 수를 지정해야 한다. KMeans는 유클리드 거리(Euclidean Distance)로 군집을 결정하기 때문에 군집 크기가 커질 수록 방향이 다른 문서벡터를 하나의 군집에 할당한다. Spherical Kmeans는 Kmeans의 한 갈래로 코사인 유사도를 기준으로 군집화한다. 눈으로 확인한 결과 두 알고리즘 모두 분류성능이 좋지 않았다.

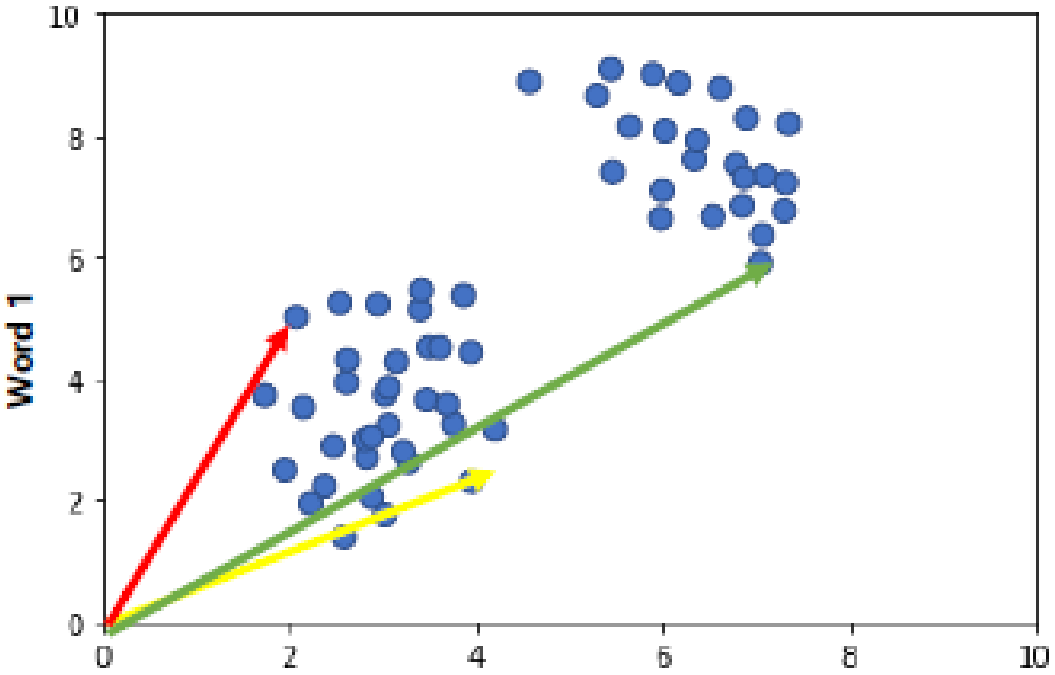



**그림 13 KMeans 군집화 문제점**

3.6.1.4. Community detection

문서 벡터를 그래프로 변환하고 모듈성(Modularity)이 최대가되는 군집을 탐색해봤다. 먼저 문서 벡터를 그래프로 변환할 때 엣지 생성 기준을 정해줘야 한다는 단점이 있다. 테스트시에는 코사인 유사도가 0.8 이상인 문서 노드 끼리 엣지를 설정했으나 데이터에 따라 이를 조정해줘야 할 것으로 보인다. Community detection의 시간복잡도는 NP-Class로 실시간 서비스로 활용하기 부적합하며 분류성능도 좋지 않았다.

3.6.1.5. DBSCAN

DBSCAN은 밀도 기반 군집화 알고리즘으로 군집수를 지정해주지 않아도 되는 장점이 있다. 또한 군집 조건에 충족하지 않는 데이터를 외상치로 처리한다. 다만 군집 생성 기준인 엡실론을 설정해야하며, 최적 엡실론을 구하는 방법이 정의되지 않은 상태다. 최적 엡실론을 산출하는 논문을 참고했으나 결과는 좋지 않았다.

논문에서는 각 문서로부터 n번째(n은 군집이 형성되기 위한 문서 수)로 먼 문서와의 거리가 급증하는 지점을 최적 엡실론이라고 설명했다. 즉 외상치가 급증하는 지점의 거리를 최적 엡실론이라고 판단하는 것이다.

하지만 프로젝트 데이터는 n번째로 먼 문서와의 거리가 완만하게 증가하는 형태를 보였다. 문서 벡터가 골고루 넓게 퍼져있는 것으로 예상돼, 이 방법을 적용하기는 어렵다고 판단했다.

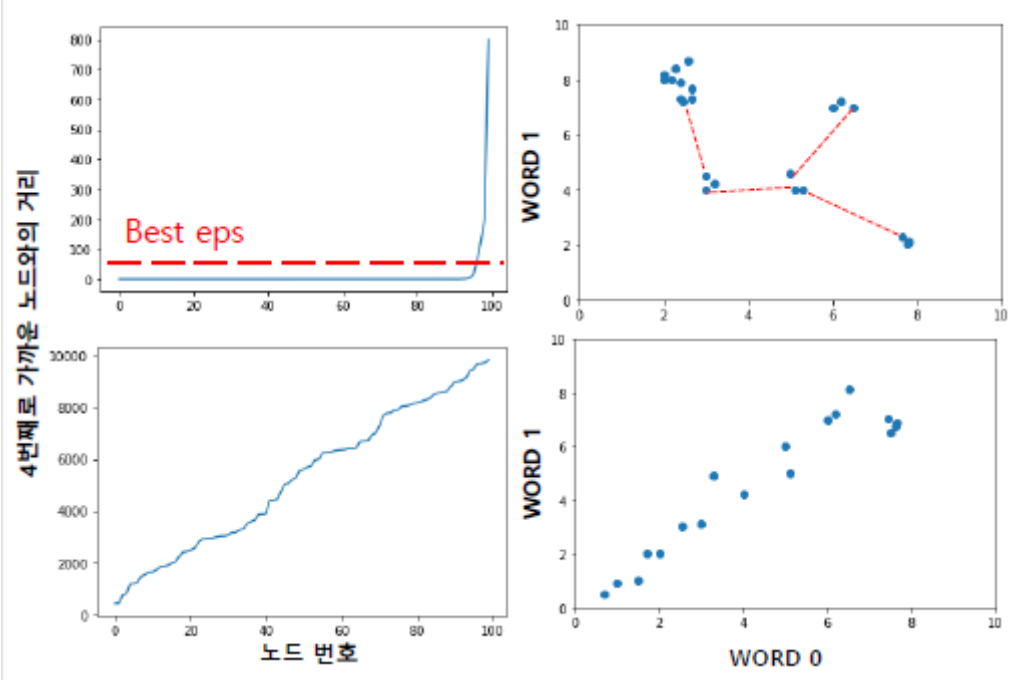



**그림 14 4번째로 가까운 노드와의 거리와 예상 벡터 분포(위 : 논문, 아래 : 테스트)**

3.6.2. 대표문서 선정

대표문서는 키워드를 추출하고 키워드를 기반으로 문서에 점수를 매기는 방식과 문서 대표 벡터를 추출하고 문서와의 유사도를 산출하는 방식을 테스트했다.

문서에 랭킹을 할당하는 알고리즘은 크게 3가지 종류가 있다. **선호도 기반 랭킹(Usage based Ranking)**은 사용자의 선호도를 기반으로 문서에 점수를 산출하는 방법으로, 사용자 정보가 없는 본 프로젝트에는 적용하기 어렵다. **링크 기반 랭킹(Link Based Ranking)**의 대표적인 알고리즘은 구글 검색 엔진의 페이지 랭크(Page Rank)다. 다른 문서에서 인용이 많이될수록(연결된 다른 문서가 많을수록) 높은 점수를 산출하는 방식으로 문서간 관계를 정의하기 어려운 기사 데이터에서 단기간에 구현할 수 없다. 이에 말뭉치 대표 키워드를 기반으로 문서 점수를 산출하는 **내용 기반 랭크(Content Based Ranking)** 방식을 적용했다.

3.6.2.1. 키워드 추출 + 대표 문서 선정

LDA의 키워드는 단어가 특정 주제에 들어갈 확률을 계산하고, 가장 확률이 높은 주제로 단어를 할당한다. 토픽 군집에는 하나의 주제가 있다고 가정하면, LDA로 추출한 키워드는 Top N으로 추출한 키워드와 같다. 이에 LDA로 키워드를 추출하는것은 무의미하다고 판단했다.

3.6.2.2. 대표 벡터 추출 + 대표 문서 선정

SVD로 토픽 내 대표 벡터를 설정하고 코사인 유사도가 가장 높은 문서를 선정하는 방식을 테스트했다. 앞서 기술한것과 같이 SVD역시 방향이 일정하지 않은(180도를 기준으로 변할 수 있는)벡터를 반환한다. 이에 산출된 문서의 대표성을 보장할 수 없다는 단점이 있다.

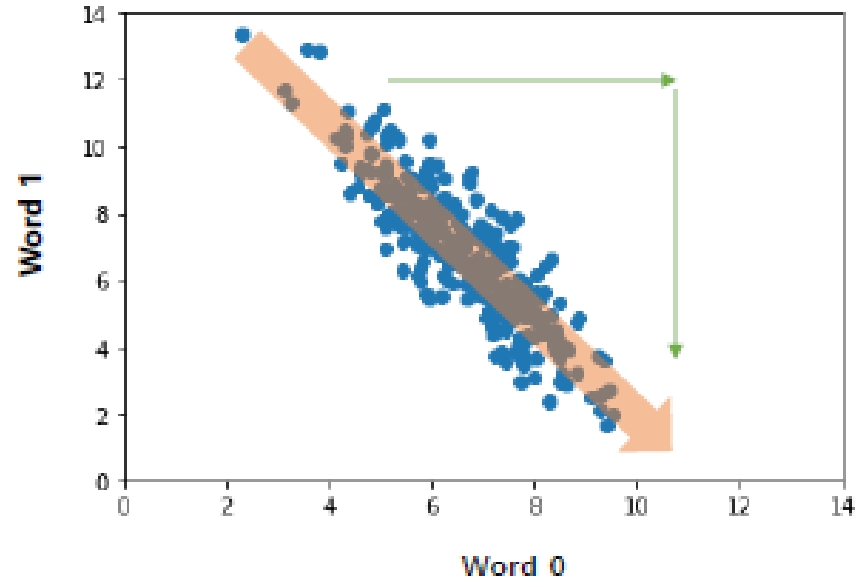



**그림 15 SVD 키워드 추출 문제**

# 4. 기대효과

## 4.1. 개선 방안

**유효문서 추출 과정**

유효문서를 추출하는 과정에서 90% 이상의 데이터를 잃었다. 빅카인즈 뉴스 API는 본문에 검색어를 1번 이상 포함하는 모든 문서를 추출하기에 검색어와 관련없는 문서가 많이 검색됐다. 데이터 크롤링하는 과정에서 보완할 수 있는 문제라고 판단한다.

**유의어 처리 과정**

유의어 처리를 진행하지 않았다. 말뭉치로 word2vec 모델을 만들고 유사도가 높은 단어를 유의어로 정의하려 했으나, 의미상 유의어가 아닌 단어도 유사도가 높게 나오는 경우가 빈번했다. 유의어 처리를 진행한다면 더 좋은 분류 성능을 얻을 수 있다고 생각한다.

**카테고리 할당 과정**

기업의 카테고리는 '경제'와 'IT' 가 대부분이다. 이에 카테고리간 데이터 불균형이 심하게 발생했다. 뉴스 정보에 따라 카테고리를 새로 정의하는 작업이 필요하다.

**성능검증**

문서 분류 문제 특성상 정형화된 성능 검증을 진행하지 못했다. 프로젝트 초기에 '토픽'을 정의하고 라벨링한 뒤 정확도를 측정하면 객관적으로 성능을 검증할 수 있을 것이다. 그러나 본 프로젝트에서는 눈으로 확인하는 작업만 진행해 아쉬움이 남는다.

**발전 방안**

감정분석을 통해 SWOT 분석등을 진행한다면 뉴스 기반 기업 분석 시스템으로 발전할 수 있을 것이라 예상한다.

## 4.2. 기대 효과

본 결과물의 기대효과는 1) 기사 소비자 편의성 2) 기사의 내재적 가치 3) 사업성의 관점으로 설명할 수 있다.

**기사 소비자 편의성 향상**

기사를 하나하나 읽어볼 필요 없이 관심사의 이슈 동향을 한눈에 파악할 수 있다. 또한, 비슷한 기사를 제거함으로써 효율적으로 이슈에 대한 세부정보를 제공할 수 있다.

**기사의 내재적 가치 제고**

인사이트(Insight)를 얻으려면 다양한 정보를 취합해야 한다. 기사는 정보량이 많다는 점에서 인사이트를 얻기에 좋은 소스(Source)지만, 기사 소비자의 역량에 따라 이를 재구성하는데 많은 시간이 소요된다는 단점이 있다. 자동으로 뉴스 이슈를 추출하고, 이를 시간순으로 재배열한다면 산재된 문서를 콘텐츠(Contents)로 재가공할 수 있다.

**사업성**

단기적으로는 취업포털의 서비스로 사용할 수 있다. 중기적으로는 홍보대행사 등 기업에 대한 이슈 조사가 필요한 기업에 서비스를 제공할 수 있다. 장기적으로는 기업의 대표 이슈에 대한 SWOT분석, 감정분석을 통해 더 정밀한 기업 분석 시스템으로 활용할 수 있다. 또한, 이슈 동향과 유사도가 낮은 기사를 가짜뉴스로 판별하는 가짜뉴스 판별기로도 활용할 수 있다.

# 5. 개발후기


| 성명 | 후기 |
|---|---|
| 김현호 | 프로젝트의 목표와 해결방안을 선정하고 결과를 만들면서 많은 것을 배울 수 있었습니다. 팀장님의 지시 아래 각자 팀원들이 최대 효율을 낼 수 있었고, 각자의 의견이 잘 반영되어 프로젝트를 진행하는 동안 내부적인 마찰 없이 프로젝트를 잘 완료할 수 있었습니다. 자신의 일이 끝나면 서로서로 도와주는 팀원들 덕분에 어려운 문제를 직면해도 항상 해결할 수 있어서 더 뜻깊은 프로젝트였습니다. |

| | |
|---|---|
| 노성문 | 각자의 역할과 능력을 발휘해준 팀원들에게 고맙다는 말을 하고싶습니다. 이번 프로젝트를 통해 프로젝트매니징 역할을 통해 팀원들간 소통하는 법과 산출물이 어떤 의미를 가지는 지 생각해보는 의미가 되었습니다. 과정에서도 결과물에서도 개인적으로 의미있는 프로젝트였습니다.<br><br>개인적으로 머신러닝 기법과 코딩을 많이 경험하지 못해 아쉽지만 개인 프로젝트를 통해 채워 나갈 예정입니다. |
| 백승현 | 학생 프로젝트가 대게 그렇듯 시작은 창대했으나 끝은 미약하다. 계획의 중요성을 통감했다. 시간이 좀 걸리더라도 프로젝트 초반부터 최대한 세밀하게 계획하는 것이 좋다고 느꼈다.<br><br>팀워크는 훌륭했다. 팀원 모두 독단적이지 않고 경청하는 성향이라, 최대한 많은 의견을 나누고 신중하게 방향을 택했다. 회의할 때 마다 팀운이 좋다고 자신했다.<br><br>결과물은 만족스럽지 못하지만 과정에서 많은 공부를 했다. 부족함이 눈에 밟히나 조금은 성장한 것 같다. |
| 안성윤 | 비전공자로 처음 겪는 프로그래밍 프로젝트여서 부담감도 컸으며 부족한 능력탓에 자신감도 떨어져있었습니다. 팀원들에게 많은 격려와 도움을 받으며 자신감도 회복하고 부담감이 많던 프로젝트에서 흥미를 느끼는 프로젝트로 바뀌었습니다. 이번 프로젝트를 통해서 기술관련 내용도 많이 배웠지만 팀 프로젝트의 진행 방향과 일정 관리 등 팀원과의 커뮤니케이션을 많이 배웠습니다. 앞으로 겪을 새로운 팀 프로젝트에도 큰 도움이 되겠다라고 생각합니다. |

# 6. 참고문헌


[1] Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, Jörg Sander , "OPTICS: Ordering Points To Identify the Clustering Structure", *ACM SIGMOD'99 Int. Conf. on Management of Data*, Philadelphia PA, 1999

[2] Nadia Rahmah and Imas Sukaesih Sitanggang, "Determination of Optimal Epsilon (Eps) Value on DBSCAN Algorithm to Clustering Data on Peatland Hotspots in Sumatra", *2016 IOP Conf. Ser.: Earth Environ. Sci. 31 012012*, 2016

[3] Robertson, Stephen; Zaragoza, Hugo, "The Probabilistic Relevance Framework: BM25 and Beyond NOW Publishers", *Inc. ISBN 978-1-60198-308-4*, 2009