안녕하세요, 빅데이터와 열정6기 핫식스에서 영감을 얻은 빅식스 팀의 서비스, TNT 발표를 맡은 팀장 김승수 입니다. 발표시작하겠습니다.

목차는 다음과 같이 팀소개, 기획 배경 및 목표, 기술스택, 산출물, 서비스 소개, 시연 순으로 진행하겠습니다./

저희 팀은 5명 모두 풀스택으로 진행하였고 영현님이 배포담당 소영님이 지라를 추가적으로 맡으셔서 수고해주셨습니다./

기획 배경 및 목표입니다.

기획배경을 말씀드리기 전에 저희 서비스명을 왜 TNT로 지었는지 먼저 설명드리겠습니다. Today News Talk의 줄임말이자 강력한 폭탄의 이름으로, 하루에 두번 대규모의 데이터 처리를 통해 폭발적인 정보를 제공하는 저희의 서비스를 단번에 설명할 수 있는 이름이라 생각해 서비스명을 TNT로 결정하게되었습니다./

우리 주변에서는 종이 신문부터, 인터넷 기사, 그리고 뉴스까지 정말 많은 소식들을 접하게 됩니다. 4대 매체 및 인터넷 뉴스 이용율 추이를 보시면 10년전에 비해 모바일 및 인터넷을 통한 뉴스이용율이 크게 증가한 것을 볼 수 있습니다. 그만큼 이동하면서 빠른 시간내에 원하는 뉴스만 골라서 보는 사람이 많아 졌다는 뜻입니다./

저희는 수많은 뉴스 기사들을 한눈에 볼 수 있도록 분석하고 시각적으로 보여줌으로써 사용자들에게 의미있는 서비스를 제공하고자 했습니다.

대량의 뉴스 기사 데이터를 하둡과 하둡 에코시스템을 활용하여 분산처리하고 추가적인 라이브러리를 통해 크롤링 및 자연어 처리를 하여 데이터 속에서 가치를 창출하고자 하였습니다./

저희의 기술스택에 대해 설명 드리겠습니다.

데이터수집, 데이터 처리, 데이터 분석, 데이터 시각화에 따른 툴은 다음과 같습니다. 자세한 설명은 이어서 진행하겠습니다./

먼저 하둡에 대해 설명 드리겠습니다. 하둡은 하나의 성능 좋은 컴퓨터를 이용해 데이터를 처리하는 대신 적당한 성능의 컴퓨터 여러대를 클러스터화하고 대용량의 데이터를 클러스터에서 병렬로 동시에 처리해 처리속도를 높이는 것을 목적으로 하는 오픈소스 프레임워크입니다.

하둡의 주요 모듈에는 HDFS, Mapreduce 등이 있는데 HDFS는 분산저장을 위한 모듈이며 mapreduce는 분산되어 저장된 데이터를 병렬 처리할 수 있게 해주는 분산 처리 모듈입니다. 뉴스기사들의 키워드데이터를 저장하고 개수를 카운팅해주는 역할을 합니다./

아파치 너치는 웹사이트 크롤링을 위한 자바기반 오픈소스 웹 크롤러 소프트웨어이며 스펙상으로 100만페이지정도는 검색할 수 있습니다. 하지만 어떤 사건때문에 아파치 너치를 사용하지 못하고 JSOUP을 사용하여 크롤링을 진행하였습니다./

저희 서비스는 오전과 오후 5시, 하루에 두번 대략 한시간정도 소요되는크롤링 작업을 수행한뒤 데이터 가공을 진행합니다. 아파치 우지는 하둡의 잡을 관리하기 위한 워크플로우 스케줄링 시스템으로 해당 작업을 자동화하기 위해 우지를 적용하였습니다./

아파치 스쿱은 관계형 데이터베이스와 하둡의 분산파일저장시스템 사이에서 데이터 이동을 지원하는 툴입니다. 스쿱을 이용하면 관계형 DB의 데이터를 분산파일저장시스템에 임포트(import)하거나, 반대로 관계형 DB로 익스포트(export)할 수 있습니다./

코모란은 크롤링한 뉴스기사들에서 형태소를 추출하여 다양한 품사들로 분류가능한 자바 기반의 한국어 형태소 분석 라이브러리 입니다. 저희는 코모란을 사용하여 형태소를 분석한 뒤 일반명사, 고유명사, 의존명사등의 키워드들을 뽑아 분석하였습니다./

다만 명사들 중에서 일반적인 기사문에 자주 사용되는 단어나 뉴스기사의 주제를 나타내줄 수 있는 의미있는 단어가 아니라면 통계에서 제외해줄 필요를 느꼈습니다. 해당 작업은 TF-IDF 알고리즘으로 해결하였는데, 해당 알고리즘은 단어의 빈도와 역 문서 빈도를 사용하여 뉴스 기사내에서 각 단어들마다 중요한 정도를 가중치로 나타내주는 방식입니다. 값이 0에 가까울수록 모든 뉴스기사에 나오는 흔한 말이고 1에 가까울 수록 특정 뉴스기사에만 등장하는 주제키워드를 의미합니다. 저희는 시뮬레이션을 통해 tf-idf 값이 0.3 이상인 단어만을 통계분석을 위한 DB에 넣었고 낮은 확률로 나오는 의미없는 단어들은 블랙리스트에 등록하고 필터링 처리하여 통계 결과의 신뢰도를 높였습니다./

저희의 웹 인프라 구조는 다음과 같습니다. 젠킨스를 통해 배포 자동화를 했고 도커에 하둡 가상노드 3개를 띄우고 클러스터화 하였습니다. 뉴스 기사 및 회원관리등은 마리아디비를 사용했고 메인서비스에서 자주 호출될 가능성이 큰 상위 키워드들은 인메모리 데이터베이스인 레디스로 관리하였습니다./

이틀전 개발과정에서 ec2서버가 멀웨어에 감염되어 초기화한 사건이 있었습니다. 초기화후 하둡 인프라를 재 구축 하는 과정에서 웹 크롤러인 아파치 너치를 설치하지 못했고 해당 기능을 JSOUP으로 대체하였습니다. 이로인해 성능에 병목현상이 발생하였고 아쉬움이 남아있습니다./

다음으로 산출물 입니다.

저희가 초기에 구상했던 TNT의 주요기능입니다. 데이터의 분산처리와 성능 고도화를 위해, 다양하고 많은 서비스보다는 주제에 부합하는 핵심기능만 넣었습니다./

기능에 부합하는 저희의 요구사항 명세서입니다./

API명세서 입니다. /

ERD입니다. RDB에서 관리하는 회원, 공지사항, 뉴스 테이블만 작성하였습니다./

다음은 서비스소개를 해드리겠습니다./

저희 서비스의 크롤링 및 데이터 분석은 대략 한시간 내외로 걸립니다. 소요시간과 사용자들의 출퇴근 시간에 맞춰 오전 5시와 오후 5시에 크롤링작업을 시작합니다. 데이터분석까지 완료되어 오전6시와 오후 6시에 사용자에게 최신화된 정보를 한번에 제공합니다./

키워드를 통한 뉴스 검색이 가능하고 키워드들을 분석하여 가장 많이 나온 키워드 10개를 순위와 함께 보여주는 오늘의 키워드 기능입니다. /

상위 키워드들은 워드 클라우드 형태로도 보여지며 나온 횟수에 비례한 크기를 가지게 됩니다. 또한 언론사별 오늘 낸 기사의 수를 알수있고 상위키워드들이 실제로 몇번 사용됐는지를 함께 보여줍니다./

또한 수집된 뉴스 기사를 카테고리 별로 분류하여 오늘 뉴스들의 전반적인 경향이 어떠한지 시각화하여 제공합니다./

필터링검색을 통해 원하는 뉴스검색이 가능합니다./

뉴스는 모달 형태로 보여지며 원한다면 원문링크로 이동하여 뉴스를 볼 수 있습니다.

백에서 데이터가 처리되는 비중이 높아 메인페이지 및 간단한 뉴스검색 시연후 마치겠습니다.

감사합니다.