파이널

인스턴스 구성화면

안녕하세요 데이터베이스와 인프라부분을 설마하게된 문광식이라고합니다.

저희는 5개의 인스턴스 구성으로 운영을하였습니다.

원래대로라면 각각의 서비스 모듈마다 인스턴스를 생성하여 운영을 해야하지만, 저희의 서비스 규모가 작고, 비용절감을 목적으로 서비스모듈끼리 묶어서 인스턴스를 구성하였습니다.

왼쪽에 보이시는 인스턴스는 웹서버 인스턴스입니다.

도커 컴포즈를 통해서 엔진엑스 구니콘 장고를 띄웠고 빠른 로그처리와 검색을 하기위해서 같은인스턴스안에 logstash와 ES를 설치하였습니다.

다음은 데이터베이스 인스턴스입니다.

회원의 정보와 게시판의 정보들이 있는 mysql

영화정보들이 있는 mongoDB

사용자에게 빠르게 보여줄 데이터가 있는 redis가 있습니다.

저희는 도커의 컨테이너로 구축할수도 있지만 그렇게 된다면 도커로인한 추가적인 계층이 생기고 이러한 계층은 지연을 유발할수있고, 도커의 목적은 트래픽이 많아지면 오케스트레이션을 통해서 컨테이너를 늘리는것이 장점이지만 데이터베이스를 늘려도 의미가 없다고 판단하여 리얼서버로 설치를 하였습니다.

기술셋 화면

저희는 데이터베이스부분, 로그분석부분, 배치프로그래밍부분 이렇게 크게 3가지 부분으로 나눌수 있습니다.

제일먼저 데이터베이스 부분에대해 설명드리겠습니다.

저희는 3개의 데이터베이스를 사용합니다.

도커를 통해서 데이터베이스를 구성할수는 있지만 도커로인한 추가적인계층을 통해 지연이 발생할수도 있고,

도커의 트래픽발생시 대처하는방식은 컨테이너를 늘리는것이지만 데이터베이스의 컨테이너를 늘려도 의미가 없다고 판단하여 리얼서버로 설치하였습니다.

회원정보나 정해진 형태의 데이터인 정형데이터를 관리하기위한 MySQL

영화정보나 정해진 형태가 없는 데이터인 비정형 데이터를 관리하기위한 mongoDB

사용자들에게 빠르게 보여주기위한 redis를 사용하였습니다.

로그분석부분을 설명드리겠습니다.

저희는 장고에 logging이라는 기능을 통해서 사용자들의 로그를 파일로 기록하며 파일이 기록될때마다

logstash가 log를 가져와 알맞게 매핑을한뒤 카프카 클러스터에게 넘기게 됩니다.

그런 후 카프카의 토픽에 접근하여 데이터를 가져오는 consumer파이썬 파일을 통해서 데이터를 알맞은 데이터베이스에 보내게 됩니다.

이렇게 보내진 로그데이터들을 통해서 사용자의 영화의 선호도를 측정하고 분석하게 됩니다.

다음 배치프로그래밍 부분을 설명드리겠습니다.

저희는 극장과 OTT에서 개봉예정되는 영화의 정보들을 하루단위로 배치를 돌리게 됩니다.

이렇게 수집한 데이터를 mongoDB에 적재되고, 포스터를 스토리지에 저장되게됩니다.

이렇게 되면서 저희팀은 포스터의 사진과 로그파일등을 대량으로 관리하는 구글 스토리지를 데이터 레이크의 역할을, 분석에 필요한 데이터를 모아놓은 빅쿼리를 데이터 마트의 역할을 수행하고있습니다.

어려웠던점

카프카 : 원래는 프로듀서와 브로커 컨슈머가 유기적으로 동작하여 데이터베이스로 로그가 바로가야하지만 컨테이너로 클러스터를 구축하는과정에서 카프카커넥터,빅쿼리싱크 컨테이너에서 연결이 안되는 어려움을 겪었습니다.

하지만 보완점으로 파이썬 파일을 임의로 만들어서 카프카의 토픽에 접근하여 빅쿼리나 레디스로 로그데이터를 적재하였습니다. 향후에는 카프카내부에서 유기적으로 작동하는 것이 목표입니다.

ES : 저희는 배치를 통해 개봉예정작들의 검색을 구현하기 위해서 logstash가 mongoDB에 들어오는 새로운 데이터를 포착하고 그것을 ES에 보내면서 신작또한 검색하게 하는것이 목표였습니다.

하지만 mongoDB logstash ES가 연동은 되었지만 logstash가 mongoDB의 새로운데이터를 포착하지 못하고 log또한 없어 구현할수 없었습니다.

향후계획은 말씀드린 파이프라인을 구축하는것을 목표로 하고 있습니다.

데이터 크롤링 : 저희는 다양한 ott에서 크롤링을 하는것이 목표였지만 각 ott 서버에서 차단하는 현상이 발생한 후

그에대한 대책으로 tmdb와 imdb등의 외국 영화사이트의 api를 통해서 수집하려고 하였지만, 외국 사이트이다보니 배우들의 정보가 영어로 표기되는 등 원하는 형태의 데이터가 아니여서 , 최종적으로 키노라이츠에서 데이터수집을 하게되었습니다. 하지만 이곳에서도 배우들의 정보 부재 문제가 발생하여 결국 배우들의 정보를 제외하고 정보 제공을 하게 되었습니다.

이상으로 발표 마치겠습니다.