유튜브, 구글, 니코동 크롤링에 관한 고찰

1. 서론

크롤러 제작 과정에서 제작 페이지에서의 크롤러의 역할에 대한 실용성과 효용성에 대하여 계속하여 의문점을 갖게 됨.
이에 본격적인 제작에 앞서 제작 웹 사이트와 관련하여 크롤링 할 소스(source)에 관한 고찰을 하게 됨.

2. 본론

(1) 서버에 올려서 크롤러를 사용하는 것은 효용성이 있는가?

- 서버에 올려두고 스케줄러에 맞춰 자동으로 동작하는 크롤러를 만든다고 했을 때, 서버 상에서 부담도 적고 인기곡 혹은 추천곡을 작성할 때 지표가 될 데이터를 모으는 역할을 할 수가 있다.
  다만, 실제로 서버 상에서 시간에 맞추어 동작하는 배치 프로그램을 만들기 위해서는 공부가 더 필요할 것이다.

(2) 가수별로 노래를 크롤링하는 것은 효용성이 있는가?

- 오랜 고민을 하게 된 원인 첫 번째는 상위노출 관련이다.
  유튜브를 켜고 검색을 하며 고민해 보았다.
  유튜브는 한 가수의 이름을 검색해보면, 사용자가 입력한 검색어와 연관도가 높은 순으로 위에서부터 정렬을 하는 모습을 볼 수 있다.
  즉 해당 곡의 원작자가 직접 유튜브에 곡을 업로드 한 경우라면 원작자의 채널이 상위 노출되는 것이다.
  그렇게 크롤러의 데이터를 약간의 가공을 거쳐, 사이트 제작에서 음악의 링크를 달 때 원작자의 영상으로 손쉽게 링크를 할 수 있다.

  그런데 여기서 한 가지 예외가 발생한다.

'Nea - Some Say', 'Sam Ryder - Tiny riot' 같은 곡을 검색해보면 원작자가 본인의 곡을 직접 업로드 했음에도 불구하고 최상단에 원작자의 채널이 노출되지 않는다.
데이터의 가공이 복잡해지는 것이다.
더불어 유튜브의 검색 결과 반환은 AJAX를 이용한 무한 스크롤 방식을 사용하는데, 요청에 의한 가장 첫 HTML 데이터에 원작자의 채널이 없다면...

첨언으로 구글 검색의 경우 상위 노출되는 로직 분석을 못 했다... 너무 마구잡이로 상위 노출되기도 하고 전혀 연관 없는 결과가 나오기도 했다.
니코동의 경우는 니코동에서 활발히 활동한 우타이테의 이름으로 검색했을 때 우타이테가 불렀던 노래들이 원작자의 계정으로 과거부터 현재까지 모두 검색 결과로 나타났다.
아마 니코동은 우타이테 활동과 관련해 특화되어 있기 때문인 듯.
그러나 대표적인 유튜브 활동 가수로 '츠유'를 검색해봤으나 관련 영상이 하나도 없다.
'요루시카'의 검색결과는 유튜브 상위노출 예외와 같이 원작자가 검색 결과로 나오기는 하나, 하단에 노출되는 모습을 보였다.

사실 위는 더 많은 가공으로 보완할 수도 있는 부분이기는 하다.
문제는 두 번째, **효용성**에 관한 것이다.
크롤링을 통하여 어떤 데이터를 가져와야하는 것인지 생각해보았다.
해당 가수가 부른 노래의 제목, 영상을 업로드한 날짜, 그 영상의 링크, 미리보기를 위한 썸네일의 URL, 해당 곡의 가사, 작사가와 작곡가, 제작자들의 코멘트, 조회수 등등.
굳이 짜내어 보자면 위와 같은 것들이 될 것이다.
이런 데이터들을 JSON이든 DB든 저장한다고 해보자.
크롤링을 진행할 소스에는 위와 같은 데이터들을 가지고 있을 수도 있고, 그렇지 않을 수도 있다.
가수별 크롤링을 할 때 소스가 될 각 영상마다 담고 있는 데이터가 각양각색이라 데이터 저장 시 데이터가 비는 경우가 발생할 것이다.
프론트엔드에서 이를 가져가 사용하면 당연히 오류가 날 것이다.
자, 그렇다면 이를 보완하기 위해서 영상마다 가사 데이터가 비어있는 경우, 자동으로 해당 노래를 검색하여 가사를 가져오는 코드를 작성한다고 해보자.
물론 가사뿐만이 아니라 작사가와 작곡가 데이터가 비어있는 경우, 조회수가 비어있는 경우, 업로드 날짜가 불일치하는 경우 등 예외 처리를 모든 경우에 대하여 코드를 작성해야 할 것이다.

과연 이것이 효율적인 방법일까? 라는 의문점이 들게 된다.
그렇다면 그냥 애초부터 각 데이터마다 수집하는 크롤러를 다르게 하여 여러 개를 만들어서 돌린다고 해보자.
한 가수의 노래 목록을 수집하는 크롤러, 목록을 가져와서 각 노래마다의 가사를 수집하는 크롤러, 조회수를 수집하는 크롤러, 작곡가 작사가를 수집하는 크롤러 등 각각 만들고 하나의 DB 테이블에 저장을 하여 바로 사용할 수 있게끔 설계했다고 하자는 것이다.
아, 그러면 그렇게 코드를 작성하면 되겠구나! 하는 생각이 든다.
그러나 과연 이것이 효용성이 있다고 할 수 있을까? 라는 질문이 생기게 된다.
왜냐하면 이러한 크롤러들은 서버 상에 적재해서 스케줄에 의하여 주기적으로 실행하는 배치 프로그램이 아니다.
사이트 제작에 들어갈 내용을 수집하는 크롤러이다.
즉, 추가적으로 내용을 추가할 때마다 한번 실행하고 말 프로그램이라는 것이다.
그렇다면 그냥 구글의 강력한 검색기능과 ~~나무위키~~를 통해 빠르게 정보를 얻을 수가 있는데 이러한 크롤러들을 통하여 정보를 얻는 것이 과연 효용성이 있다고 할 정도로 가치가 있을까?

3. 결론

새벽 감성으로 글을 쓰다보니까 뭔 말이 ~~쥰내게~~ 길어졌는데 사실 요약하자면 그냥 이런거다.

안에 들어갈 내용은 그냥 직접 구하는 게 낫지 않을까?
크롤러를 통한 데이터 구성은 오히려 비효율적일지도 모른다.

대신에, 추천곡이나 인기곡에 대한 지표가 될 만한 요소들만 크롤러로 만들어서 서버에 올려놓고 돌리면서 DB에 저장하는 게 크롤러 파트는 더 깔끔하게 할 수 있지 않을까? 하는 내 생각이다.

24Hits 차트라는 게 있다.
네이버의 실시간 검색어도 그랬고, 멜론의 실시간 차트 혹은 top 100도 그랬고, 여러 가지 큰 문제점을 가지고 있었다.
단합하면 쉽게 순위 조작이 가능하고, 한번 위로 올라간 검색어

혹은 노래는 다른 사람에게 상위 노출되기 때문에 계속 열람을 하게 되면서 순위권 밖으로 잘 떨어지기가 힘들다는 것이다.
즉, 이런 문제점들로 인해서 '순위' 라는 것이 의미가 없어진 것.
24Hits 차트는 이런 문제점들을 개선하여 만든 리더보드이다.

어차피 일반적인 음반과는 달라서 신곡에 관해서는 끊임없이 체크하기도 힘들테니 수 천곡의 음악들은 천천히 데이터베이스에 저장해나가기로 하는 것이 어떨까.
우선적으로 top 100을 구성하기 위해 24Hits 차트를 적용하든지 자료 수집을 진행하고, 느낌별 플레이리스트를 만들고 난 이후면 내 생각엔 충분히 이미 DB 상에 많은 노래들이 저장되어 있을 거라고 생각한다.