**데이터 수집 보고서**

**1) 조 이름(**단독 진행이면 본인 이름 작성**)**

|  |
| --- |
| 이하진 |

**2) 조 구성원**

|  |  |  |  |
| --- | --- | --- | --- |
| **학번** | **이름** | **이메일** | **역할** |
| 2017150432 | 이하진 | hlee2745@gmail.com |  |
|  |  |  |  |
|  |  |  |  |

**3) 프로젝트 주제**

|  |
| --- |
| 음원차트 실시간 사용자 추이를 통해 음원 분류하기 |

**4) 주제 선정 이유 및 수행 목적**

|  |
| --- |
| 2018년쯤부터, 몇몇 가수들의 음원이 비정상적인 추이를 보이며 급격하게 차트 순위가 상승하는 현상이 반복되면서 음원 사재기 논란이 화두에 올랐다. 또한, 예전부터 문제로 지적되어 왔던 대형 팬덤의 “스트리밍 공세”(팬들이 여러 기기를 사용해 스트리밍을 24시간 동안 켜 두면서 자기 가수의 음원성적을 올리는 것)도 여전했다. 이러한 상황 속에서, 더 이상 음원차트가 대중의 선호도를 반영하지 못한다는 비판이 나왔다. 그래서 실시간 사용자 추이를 분석하여 음원차트에 있는 여러 음원들을 “대중형 음원”, “팬덤형 음원”, “사재기 의심 음원” 등으로 분류하고, 이를 통해 현재 음원 차트가 얼마나 대중의 선호도를 반영하는지 알아보고자 한다. |

**5) 선택한 주제에 필요한 데이터**

|  |
| --- |
| 음원 사이트의 실시간 스트리밍 데이터  (국내 최대 규모 음원사이트인 Melon의 실시간 스트리밍 데이터를 사용하고 싶었지만, 새벽 1시부터 아침 7시까지의 데이터가 제공이 안되는 관계로 국내 두번째 규모의 음원사이트인 Genie의 실시간 스트리밍 정보 사용) |

**6) 사용 데이터 형식(**예: html, xml, json…**)**

|  |
| --- |
| 텍스트 데이터를 크롤링해서 json으로 저장 |

**7) 데이터 수집방법 및 출처(**예: 공공데이터, 네이버, 웹 페이지 크롤링 등**)**

|  |
| --- |
| 크롤링 시 활용한 기법(API, 라이브러리 사용)도 작성  방법: 웹 크롤링  출처: 가이섬.com의 Genie 실시간 차트 (탑100)  <https://xn--o39an51b2re.com/genie/chart/realtime/>  사용된 크롤링 라이브러리: requests 패키지와 BeautifulSoup 사용 |

**8) 프로젝트 수행을 통해 얻을 수 있는 기대효과**

|  |
| --- |
| 현재 음원 시장에 대한 현황을 파악하고, 비정상적인 음원 소비를 통한 음원 차트 순위 왜곡이 얼마나 심한지 수치화 함으로써 경각심을 가질 수 있다. |

**9) 데이터 수집 증빙(파이썬 코드는 별도의 파일로 제출)**

|  |
| --- |
| url (ex. <https://xn--o39an51b2re.com/genie/chart/realtime/20191231/23> ) 마지막 부분에 있는 날짜와 시간을 변경하면서 사용. datetime 라이브러리를 사용해서 2019년 12월 31일 23시부터 2023년 5월 13일 23시까지 1시간 단위로 시각을 변경하며 http request를 보냈다. 보낸 request에 대해 돌아온 response의 body를 BeautifulSoup에 인수로 넣어서 객체화 한 후, 태그를 찾는 방식으로 Genie 실시간 탑100 차트에 있는 (“노래 제목”, “가수 이름”, “누적 청취자수”, “누적 재생수”) 정보를 긁어낸다. 만약 페이지에 찾는 정보가 없거나 결측치로 ‘-‘가 들어가 있는 경우 건너뛰도록 코드를 작성했다. 이렇게 3년 6개월치에 육박하는 Genie 실시간 탑100 차트 정보를 모았고, 이 정보는 collections 모듈에 있는 defaultdict 구조에 저장한 후 json 파일로 내보냈다.  파이썬 코드는 <https://github.com/bossacode/Genie-Music-Chart-Analysis> 에 있는 data\_scraping 폴더 확인 |

**9) 6월 21일까지 전체 수행 일정**

|  |
| --- |
| ~5.28: data cleaning & EDA 완료  ~6.4: K-means 알고리즘 활용한 분류 결과 도출  ~6.11: 결과 시각화  ~6.18: 최종 결과 정리 & 보고서 작성 |