Youtube 댓글을 통한 음원의 비지도학습 분류

이름 : 안현

프로젝트 Github :

<https://github.com/Crane-lupert/museclassification>

기반 데이터 링크:

<https://www.kaggle.com/datasnaek/youtube>

목차

1. 서론
   1. 주제 선정 이유
   2. 데이터 선정 이유
   3. 분석 방법 선정
2. 본론
   1. 전처리(1/3)
   2. 전처리(2/3)
   3. 전처리(3/3)
   4. 차원 축소
   5. 머신러닝 도입 및 결과
3. 결론
   1. 결과 해석
   2. 실제 활용 여지
4. 서론
   1. 주제선정이유

음원은 다른 재화와 비교했을 때 창작에 기반한 상품이라는 명확한 특징을 가진다. 여타 상품들이 공정을 거쳐 기성품의 조립으로 만들어지는 반면, 음원은 이러한 공정과 궤를 달리하는 경향이 있다. 음원은 창작이라는 특이한 성향에 기인해서 만들어지고, 이는 상당한 리스크를 동반한다, 여타 상품의 경우 투자와 시장조사가 해당 상품의 기획이자 방향성이 되고, 결과의 실패가 있을지언정 방향성이 엇나가는 경우는 드물다. 하지만 음원은 수십억의 자본이 투자되어도 그 명확한 방향을 결정지을 수 없는 경우가 많고, 방향성이 엇나가면서 예상외의 성공을 거두는 경우도 많다.

2010년에 한국계 미국인이 핵심 멤버로 있는 Far East Movement의 ‘Like a G6’가 빌보드 1위를 차지한 경우가 대표적인 예이다. 당시 작곡가로 참여한 나일스 홀로웨다는 녹음중인 특정악기가 고장났고, 이 고장난 소리를 그대로 음원에 사용했다[[1]](#footnote-1). 해당 악기소리는 당시 통념상으로 사용이 안 될만한 악기소리였고, 해당 음원은 예상하던 방향성에서 벗어났다. 하지만 이 곡은 성공여부로만 본다면 해당 아티스트의 커리어 하이에 해당하는 곡일만큼 이견의 여지가 없는 곡이다. 다른 상품의 기획이 단순한 농사라고 한다면, 음원 상품의 기획은 무슨 종인지 모를 종자를 농사짓는 것과 같다.

이처럼 음원은 다른 상품에 비해 상품 기획의 양상이 다름에도 불구하고 이러한 부분에 대한 고찰은 여전히 부족한 편이다. 음원의 상품기획을 위한 적합한 마케팅믹스의 골자를 만들거나, 기존의 것들을 발전/개량시키기 위해서는 새로운 소비자행동 분석이 필요하다고 판단하였다. 이것이 해당 보고서의 주제를 음원으로 정한 이유이다.

* 1. 데이터선정이유

Youtube Music, Melon을 포함한 음원 서비스사에서 쓰고 있는 음원추천 알고리즘 중 Collaboration Filtering이란 것이 있다. 이는 사람들이 해당 음원에 대해 어떻게 생각하는지, 어떻게 관련된 글을 쓰고 반응하는지를 기반으로 해서 해당 음원에 대한 평가를 내린 뒤, 이를 기반으로 유사한 경향을 보이는 소비자들에게 음원을 추천해주는 것을 의미한다. 그리고 이러한 평가에 주로 쓰이는 것은 단어, 그중에서도 형용사이다. 하지만 이를 위해 다른 음원사이트의 Open API나 음원에 대한 감상평을 크롤링하여 쓰는것은 경제적 가치를 창출해 내기 힘들다. Open API의 경우 이미 해당 음원사에서 훨씬 더 정교하고 정확하게 다루고 있기 때문에 일개 대학생이 처리하는 것이 의미가 없다. 감상평 크롤링의 경우 분명히 해당 음원사보다 더욱 정교하게 다룰 수는 있으나, 해당 데이터를 추출하는 과정에서 수많은 노이즈가 데이터에 낄 확률도 높으며, 크롤링을 할 부분을 찾는 과정에서 추가적인 웹 개발 관련 지식이 필요하다.

위의 이유로 인해 가장 적합한 데이터는 kaggle에 있는 유튜브 댓글 데이터로 판단하였다. 해당 데이터는 2017년 하반기에 인기있었던 유튜브 영상들의 메타데이터(영상 제목, 조회수, 태그, 좋아요, 싫어요 수, 카테고리 ID 등등)으로 이루어진 GB\_Videos.csv, US\_Videos.csv와 해당 영상들에 달린 댓글들의 메타데이터(댓글 내용, 좋아요 수 등등)으로 이루어진 GB\_Comments.csv, US\_Comments.csv로 이루어진다. 물론 추가로 있는 GB\_category\_id.json, US\_category\_id.json으로도 위의 네 개의 csv와 동일한 기능을 할 수는 있으나, 이는 웹에 직접 게시할 때에 적합한 것이므로 일단 csv만을 이용하기로 한다.

해당 데이터 안에서 실제로 사용한 데이터는 US\_Videos.csv와 US\_Comments.csv이다. 이 두 데이터는 미국에서 수집된 것이고, 나머지 두 데이터는 영국에서 수집된 것이다. 두 데이터를 통합해서 사용할 수도 있지만, 이 경우 영국식 영어와 미국식 영어의 차이가 데이터의 경향을 흐리게 만들 수 있을 것 같아서 미국에서 수집된 데이터만 사용하겠다.

* 1. 분석 방법 선정

당연히 해당 영상들 중에서는 음악 비디오만 있는 것도, 형용사로만 이루어진 댓글이 있는 것도 아니니 전처리과정이 머신러닝앞에 선재한다.

가장 편한 방법은 장르에 따른 분류일 것이다. 하지만 이 방법은 장르간 혼합이 상당히 이뤄지고 있는 현재 음원시장의 추세에 맞지도 않을 뿐더러, 굳이 위의 데이터를 쓰지 않아도 할 수 있는 일이다. 해당 데이터를 통해 알고 싶은 것은 특정 노래를 좋아하는 사람들이 다른 노래들 중 어떤 노래를 좋아할지 이고, 이는 기존에 정해진 기준을 가지고 하는 것이 아니다. 즉, 지도학습보다는 비지도학습이 분석 방법으로선 더 정확하다.

비지도학습의 군집분석(clustering)중에선 계층적 군집분석(Hierachical Clustering)이 더욱 적합하다 할 수 있는데, k-평균 군집분석(k-means Clustering)처럼 전체를 단순하게 몇 개의 부분집합으로 분할하는 것이 아니라 집합의 추가적인 분할/통합을 할 수 있는 구조이고, 이러한 현황을 Dendrogram으로 시각화시킬 수 있기 때문이다.

1. 본론
   1. 전처리(1/3)

우선, 제목 페이지에 있는 기반 데이터 링크로 들어가서 하단부의 US\_Video.csv와 US\_Comments.csv를 다운받았다. 전술했듯이 해당 데이터에는 Youtube 음악 동영상만 들어있는것이 아니라, 다양한 비디오들에 대한 정보도 포함되어있다. 이를 최우선적으로 정제하는 이유는 프로젝트에 필요없는 데이터들이 이후 처리과정에 포함되게 될 경우 전체 처리속도를 가장 많이 늦출 것이라 판단했기 때문이다. 혹자의 경우 Docker기반 Oracle SQL(ver 11g)상에서 쿼리문을 통해 US\_Video.csv와 US\_Comments.csv의 데이터를 용도에 맞게 정제하였다. US\_Comments.csv 기준 약 63만개였던 댓글 데이터 수가 정제 이후 약 9만개로 축소되었다. 해당 처리 이후 SQL에서 다시 csv 로 꺼냈다. 파일명은 이전과 동일하게 하여, 기반 데이터(정제가 안된)와 혼동이 없도록 하였다.

그 다음은 파이썬으로 가서, 201501868\_NLP\_1.py 파일을 컴파일하면 된다. 우선, US\_Comments.csv에 전부 모여있던 댓글 데이터들을 Video\_ID의 이름을 따서 csv파일을 만든후 해당 파일들에 분류하였다. 이는 하나의 비디오가 하나의 음원과 같다는 가정 하에 분석을 진행하기 때문에 행한 것이다. 즉, 하나의 비디오에 해당하는 모든 댓글들의 집합체가 해당 비디오의 특성을 결정짓는 것이다.

Textblob의 품사 판별기능을 함수로 만든 뒤 바로 앞에서 분류한 댓글 데이터들 중에서 형용사계열만 남긴다. 다만 해당시점에서 US\_Video.csv에 있는 Video\_ID ‘Zy6vBxqlapw’의 댓글 데이터가 유실된 것이 발견되었다. SQL에 삽입된 데이터로 색인해본 결과 US\_Video.csv에는 있지만 US\_Comments.csv에는 하나도 존재하지 않는 것으로 확인한다. 즉, 전체 분석과정에는 영향을 미치지 않으므로 if 조건을 하나 더 삽입하였다.

그 다음에는 전체 댓글에 있던 형용사를 전부 추출하여 wordtotallist.csv파일로 만들었다. 단, 이 과정에서 한 비디오 안에 동일한 형용사가 3개 이하인 경우 wordtotallist.csv에 추가하지 못하도록 하였다. 또한 바로 전 처리에서 하나의 댓글 안에서 발견된 모든 형용사는 똑같은 칸에 전부 포함되도록 처리를 하였는데, 그 갯수가 30개를 넘긴 경우(즉, 하나의 댓글에 30개이상의 형용사가 들어간 경우) 또한 wordtotallist.csv에 추가하는 과정에서 배제하였다. 보통 해당 데이터는 도배성 댓글일 확률이 높아서 outliner로 작용하기 때문이다.

위와 같은 필터링을 거치더라도 wordtotallist.csv를 확인해보면 이모지, 스페인어 등의 데이터노이즈값들이 남아있다. 이러한 노이즈값들을 수작업으로 지워서 400~500개대로 만든 뒤 wordcountmanual.csv로 저장한다.

* 1. 전처리(2/3)

이전 전처리 과정에서 만들어낸 wordcountmanual.csv를 기반으로 인접행렬을 만든다. 상술하자면, 동일한 비디오 안에 형용사들 간의 네트워크 망의 연결 횟수를 모두 누적시켜서 인접행렬을 뽑아낸다. 예를 들어, 특정 비디오 댓글중에 아래 두 개가 포함되어있다 가정해보자. 하나의 댓글 안에 있지 않아도 상관없다.

‘good’ ‘deep’

이 경우, wordcountmanual을 index이자 column으로 하는 인접행렬은 [good][deep]좌표의 수와 [deep][good]좌표의 수를 1씩 추가한다. 단, 만약 위의 두 단어가 각각 다른 영상에 있는 댓글에서 나온 것이라면, 인접행렬에 추가할 수 없다. 댓글 하나에서 형용사가 30개 이상 추출된 경우도 당연히 제외한다. 사유는 이전단계와 동일하다.

위의 인접행렬을 다시 csv로 변환한 형태가 wordrelfin.csv이다. wordrelfin.csv를 Google Spread Sheet에 넣은 뒤, 각 index별 pagerank(=빈도수 총합), 평균, 재귀값, 재귀값제외평균, 재귀값제외 STDEV를 산출하면 아래와 같은 꼴이 된다. 여기서 재귀값이란, 동일한 단어를 index/column으로 가지고 있는 인접행렬의 값을 의미한다. 예를 들어, 어느 영상의 댓글목록이 다음과 같다 하자.

‘good’ ‘deep’ ‘good’

위의 목록에서, 앞에 있는 ‘good’을 good1, 뒤에 있는 ‘good’을 good2라고 칭하겠다. 실제 인접행렬에 추가되는 경우의 수는 다음과 같다.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Good1 | Deep | Good2 |
| Good1 | [good][good] | [good][deep] | [good][good] |
| Deep | [deep][good] | [deep][deep] | [deep][good] |
| Good2 | [good][good] | [good][deep] | [good][good] |

이처럼 인접행렬의 index/column이 동일할 경우, 해당 단어가 여러번 나오는 값을 index/column이 다른 경우와 달리 한 곳에 집적된다. 형용사의 모임이니 만큼 경향성을 가진 단어일수록 한 영상안에 여러 번 중복되어 들어가있을 것이고, 이에 따른 재귀값의 과대측정은 심각할 것으로 추정된다. 때문에 각 index(단어)별 빈도 수에 기반한 통계값 산출에 있어서 재귀값을 빼고 하는 것이 맞다고 판단하였고, 해당 산출데이터는 다음과 같은 꼴을 가진다.([전체 데이터는 여기에서](https://docs.google.com/spreadsheets/d/1AvR2s90oRYl6cV2gbT-EyvCMCyMeiMxIvlPsHnWvMvk/edit?usp=sharing))

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 총합 | 평균 | 재귀값 | 재귀제외평균 | STDEV(평균보정) |
| good | 5320 | 28.44919786 | 675 | 24.97311828 | 1742.126763 |
| beautiful | 3215 | 17.19251337 | 446 | 14.88709677 | 817.2925798 |
| best | 2864 | 15.31550802 | 226 | 14.1827957 | 619.8525921 |
| amazing | 2535 | 13.55614973 | 230 | 12.39247312 | 588.701902 |
| new | 2258 | 12.07486631 | 196 | 11.08602151 | 381.6557822 |
| great | 2151 | 11.5026738 | 156 | 10.72580645 | 356.1949002 |

여기서 STDEV값은 재귀값을 제외한 표준편차를 의미하는데, 기존 표준편차값에서 수작업으로 변형을 가해야 하며, 공식은 다음과 같다.

실제 원 데이터를 보게 되면, 재귀값을 제외한 평균/표준편차에서 기존 데이터상의 서열이 다소 바뀌는 현상을 확인할 수 있다. 여기서 평균보정STDEV(재귀값을 제외한 상태의 표준편차)를 기준으로 상위 36개 단어를 최종적으로 정해서, 해당 단어를 worddfterm.csv로 저장하였다.

* 1. 전처리(3/3)

이전 전처리과정에서 얻은 worddfterm.csv의 단어가 각 비디오마다 몇개가 있는지를 도출한 뒤, 도출된 수를 해당 비디오 전체 빈도수로 나눠서 비율로 전환해준다. 해당 값은 하위폴더인 csvres3 밑에 [해당 video\_id].csv로 저장된다. 예를 들어, csvres3/1NyMSWqIJDQ.csv에는 아래와 같은 표가 csv형태로 저장된다.

|  |  |
| --- | --- |
| good | 0.14510609 |
| beautiful | 0.08555784 |
| best | 0.07939767 |
| amazing | 0.04996578 |
| new | 0.04825462 |

(후략)

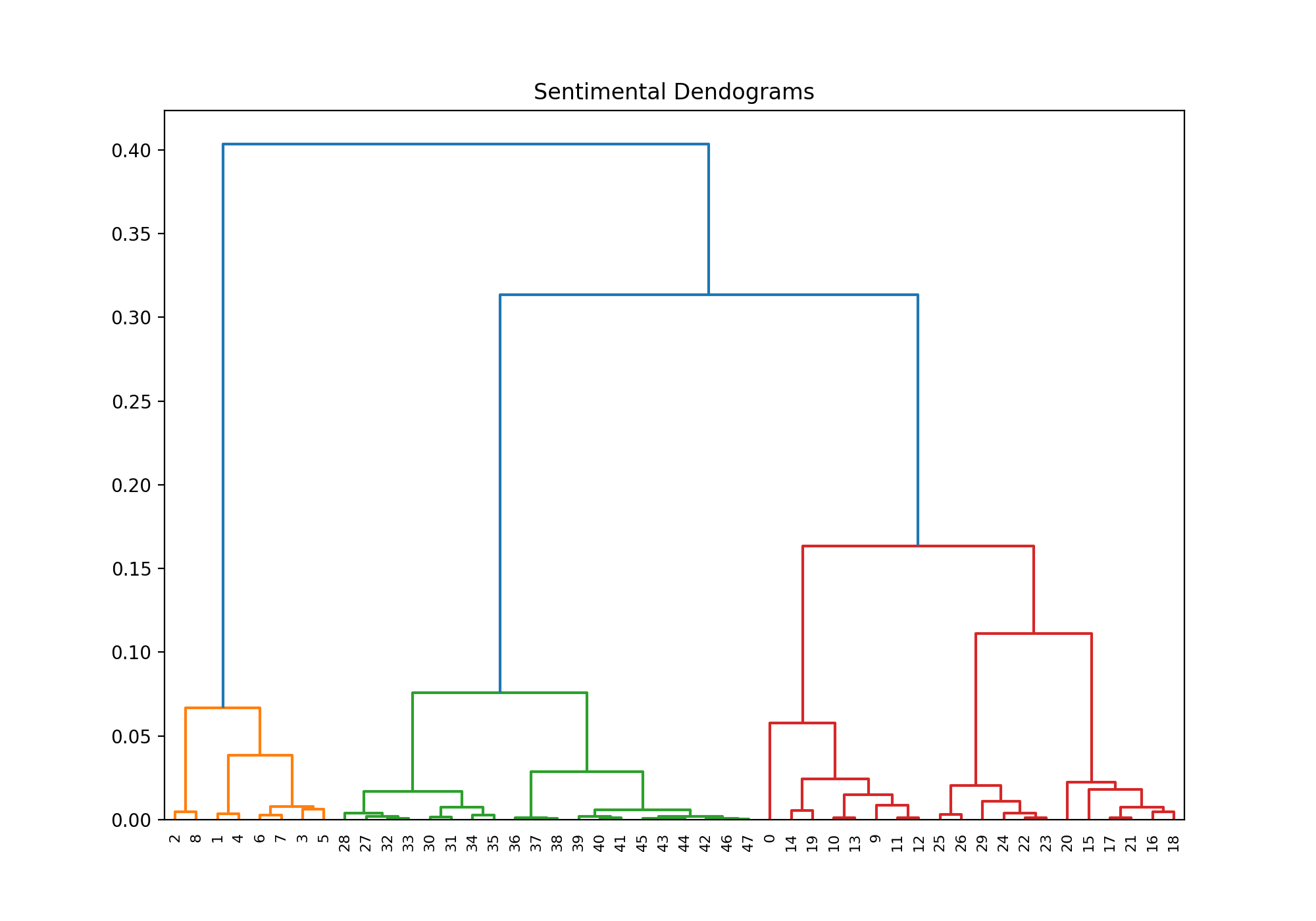
해당 데이터를 기반으로 파일제목으로 된 음원 유튜브영상의 특성을 부여할 수 있으며, 36차원의 좌표값으로 사용할 수 있다.

* 1. 차원 축소

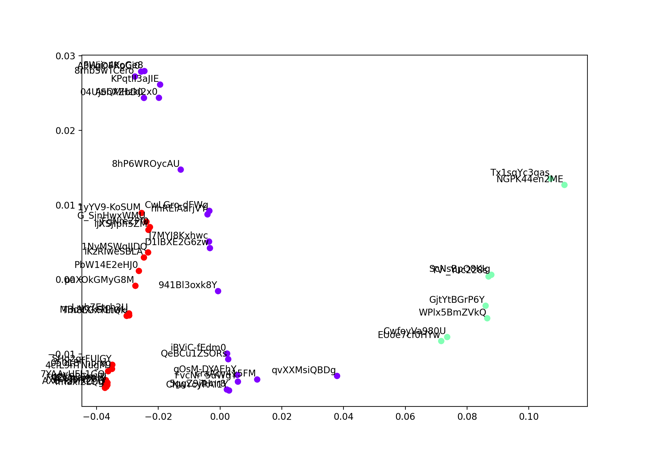
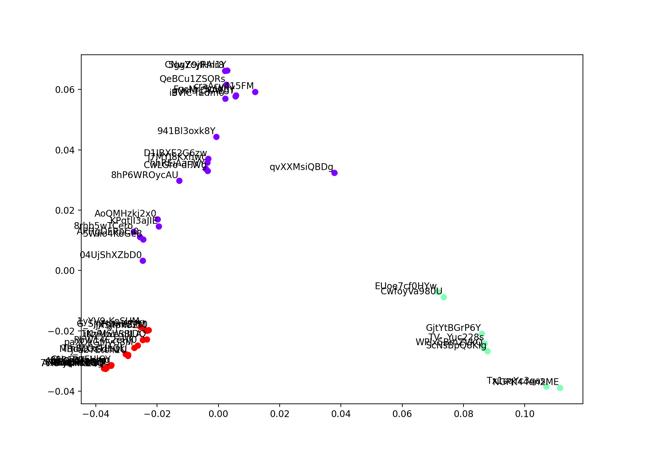
다만 이러한 36차원의 좌표값은 시각화가 매우 힘들어질 뿐만 아니라, 전체 데이터의 알고리즘 도출/실행시간도 너무 많이 잡아먹는다. 이러한 부분을 해결하기 위해 차원축소를 실행한다. 차원축소가 가능한 이유는 실제 데이터의 구도가 36차원이긴 해도 실제로는 낮은 차원에 수렴하는 경향이 있을것이라 생각하기 때문이다. 201501868\_ML\_4.py의 전반부가 여기에 해당하는 내용인데, 우선 이전 NLP\_3파트에서 res3에 보관한 데이터들을 하나의 데이터프레임으로 취합한 뒤, PCA를 이용해 설명률 95%를 갖춘 차원축소를 실행하였고, 그 결과 column이 4개로 줄어들었다(36차원에서 4차원으로 축소).

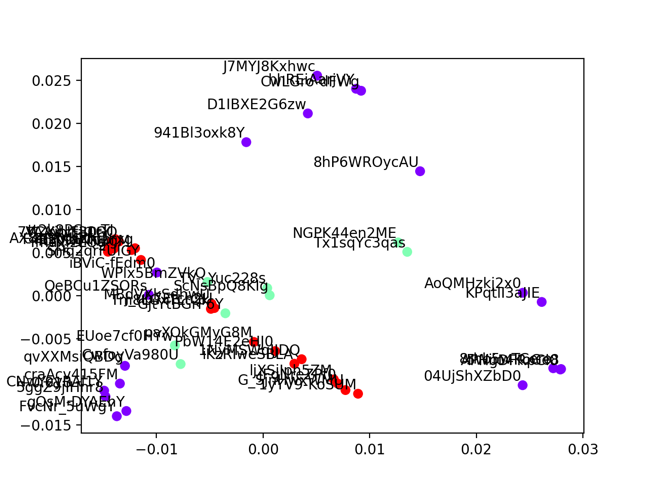
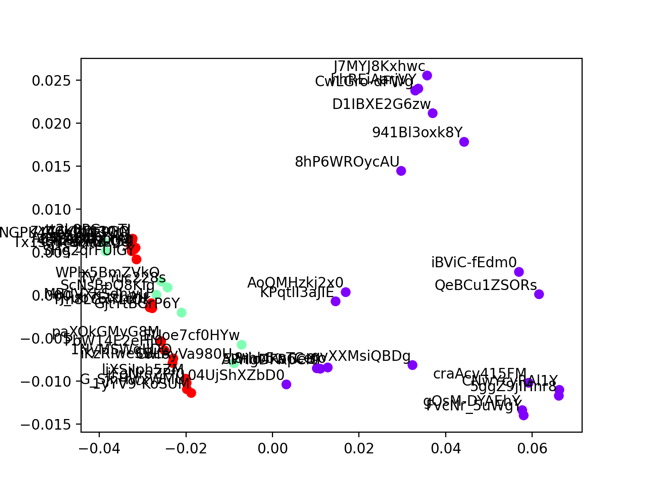
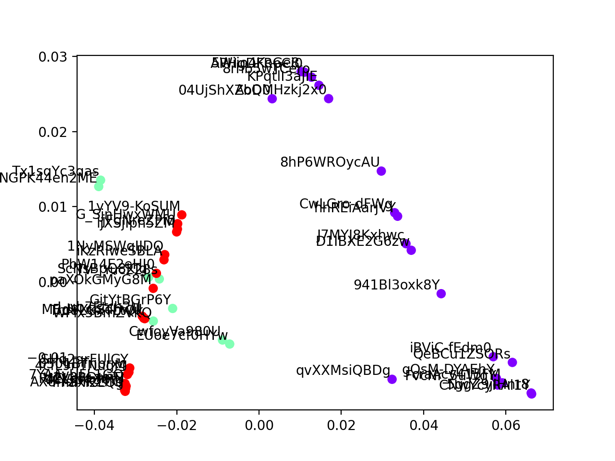
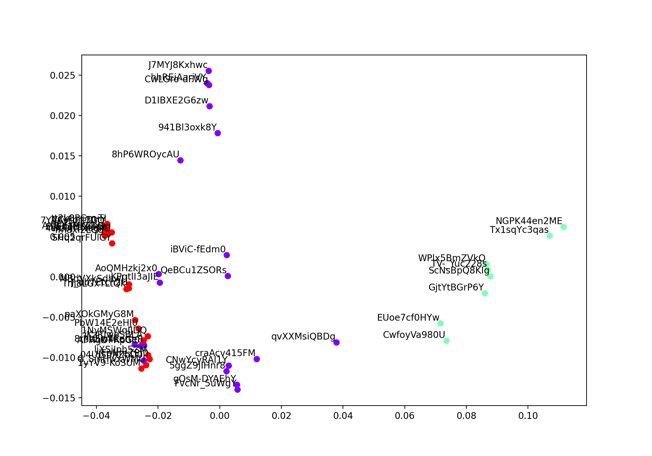
* 1. 머신러닝 도입 및 결과

서론에서 계획한 계층적 군집분석중 하나인 병합 군집분석(Agglomerative clustering)을 이용한다. 병합 군집분석은 계층적 군집을 만든 뒤 군집을 계속해서 만들어나가는 것을 의미한다. 군집에 한 번 소속된 개체는 다른 군집으로 이동할 수 없다는 특징을 가진다. 차원 축소를 하긴 했어도 여전이 column이 4개인(=4차원) 상태이므로 전체 데이터를 가독성있게 정리한 Dendogram을 출력하면 다음과 같다.



이전 과정에서 차원 축소까지 완료한 데이터프레임을 scipy(계층 군집분석 라이브러리)에 넣게 되면 위의 사진과 같은 결과를 얻을 수 있다. 수형도 상에서 가장 의미있게 나뉘는 군집갯수는 3개로 볼 수 있는데, y축 기준 0.17이후로는 윗단계의 수형도보다 더 짧은 간격으로 군집이 분화되기 때문이다. 이러한 부분을 6개의 평면그래프로 나타내면 아래와 같다. 6개의 평면도인 이유는 차원축소를 거친 데이터가 4차원이기 때문에 평면으로 이를 완벽하게 구현해내기 위해선 4C2=6 개의 평면이 필요해서이다.





위의 6개의 산포도는 전체 column 4개중 순서대로 1-2, 1-3, 1-4, 2-3, 2-4, 3-4 번째의 column을 각각 X/Y축으로 한 산포도이다. 원론적으로는 해당 6개의 평면을 조합해서 투사할 때 분류의 적합성을 확인할 수 있지만, 우리는 3차원에 살고 있으므로 이는 현실적으로 불가능하다.

1. 결론
   1. 결과 해석

위의 내용을 다시 csv파일에 나눠서 담으면 다음 표를 얻을 수 있다.(final\_classification\_result.csv)

|  |  |  |
| --- | --- | --- |
| VIDEO\_ID | TITLE | category\_number |
| GjtYtBGrP6Y | [OFFICIAL VIDEO] Away in a Manger – Pentatonix | 1 |
| 5ggZ9jIHnr8 | Alesso & Anitta - Is That For Me (Official Music Video) | 0 |
| -jFgNreZPf0 | ASKING ALEXANDRIA - Into The Fire (Official Music Video) | 2 |
| craAcv415FM | Bruno Mars - Just The Way You Are [ONE VOICE: SOMOS LIVE! Performance] | 0 |
| MBdVXkSdhwU | BTS (방탄소년단) 'DNA' Official MV | 2 |
| CwfoyVa980U | Charlie Puth - How Long [Official Video] | 1 |
| CwLGro-dFWg | Charlie Puth – How Long [Official Audio] | 0 |
| NGPK44en2ME | CNCO - Mamita (Audio) | 1 |
| 1NyMSWqIJDQ | Danielle Bregoli is BHAD BHABIE “Hi Bich / Whachu Know” (Official Music Video) | 2 |
| 8rhb5wTCero | dodie - 6/10 | 0 |
| iKzRIweSBLA | Ed Sheeran - Perfect [Official Lyric Video] | 2 |
| 7YAAyUFL1GQ | Fall Out Boy - The Last Of The Real Ones | 2 |
| -Ifnaxi2LQg | Fergie - You Already Know ft. Nicki Minaj | 2 |
| VCVjbrt30tY | Fifth Harmony - Deliver | 2 |
| TV-\_Yuc228s | Hayley Kiyoko - Feelings [Official Video] | 1 |
| gOsM-DYAEhY | Imagine Dragons - Whatever It Takes | 0 |
| 941Bl3oxk8Y | Invitation - Why Don't We [Official Music Video] | 0 |
| APHgDFRpCi0 | J Balvin, Willy William - Mi Gente featuring Beyoncé ft. Beyoncé | 0 |
| FvcNr\_5uWgY | Jessie J - Not My Ex | 0 |
| QeBCu1ZSORs | Kris Wu - Deserve ft. Travis Scott (Official Audio) | 0 |
| F4ELqraXx-U | Lana Del Rey - White Mustang (Official Video) | 2 |
| WPlx5BmZVkQ | Liam Payne - Bedroom Floor (Lyric Video) | 1 |
| D1IBXE2G6zw | Lin-Manuel Miranda - Almost Like Praying feat Artists for Puerto Rico [Music Video] | 0 |
| qvXXMsiQBDg | Louis Tomlinson - Just Like You (Lyric Video) | 0 |
| 1yYV9-KoSUM | MACKLEMORE FEAT KESHA - GOOD OLD DAYS (OFFICIAL MUSIC VIDEO) | 2 |
| 5Wiio4KoGe8 | Maroon 5 - What Lovers Do ft. SZA | 0 |
| Tx1sqYc3qas | Marshmello - Silence Ft. Khalid (Official Music Video) | 1 |
| ScNsBpQ8KIg | Martin Garrix & Matisse & Sadko - Forever (Official Music Video) | 1 |
| 04UjShXZbD0 | Melanie Martinez - Mad Hatter [Official Video] | 0 |
| CNwYcyRAl1Y | Michael Jackson - Blood On The Dance Floor X Dangerous (The White Panda Mash-Up) | 0 |
| 0501BTnbrxg | NF - Let You Down (Audio) | 2 |
| J7MYJ8Kxhwc | NF - Outcast | 0 |
| 4cfL9nTNugM | Niall Horan - Too Much to Ask (Audio) | 2 |
| ljXSjIph5ZM | Niall Horan - Too Much To Ask (Official) | 2 |
| G\_SjnHwxWMU | Noah Cyrus - Again ft. XXXTENTACION | 2 |
| 8hP6WROycAU | One More Light (Official Lyric Video) - Linkin Park | 0 |
| Tm8LGxTLtQk | One More Light (Official Video) - Linkin Park | 2 |
| hhREiAarjVY | Sam Smith - Pray (Official Audio) | 0 |
| KPqtlI3aJIE | Sam Smith - Too Good At Goodbyes (Live From Hackney Round Chapel) | 0 |
| AX8-YzMKZhQ | Sam Smith - Too Good At Goodbyes (Official Audio) | 2 |
| J\_ub7Etch2U | Sam Smith - Too Good At Goodbyes (Official Video) | 2 |
| SHq2qrFUlGY | Shakira - Perro Fiel (Official Video) ft. Nicky Jam | 2 |
| paXOkGMyG8M | Sia - Rainbow (From The 'My Little Pony: The Movie' Official Soundtrack) (Official Video) | 2 |
| EUoe7cf0HYw | Taylor Swift - Gorgeous (Lyric Video) | 1 |
| iBViC-fEdm0 | Wu-Tang Clan - People Say ft. Redman | 0 |
| tt2k8PGm-TI | ZAYN - Dusk Till Dawn ft. Sia | 2 |
| AoQMHzkj2x0 | ZAYN - Dusk Till Dawn ft. Sia (SING OFF vs. Madison Beer) | 0 |
| PbW14E2eHJ0 | Zedd, Liam Payne - Get Low (Street Video) | 2 |

대상이 된 것이 음원이 아닌 비디오 단위다 보니, 동일한 음원의 다른 버전이 동시에 기입된 것들도 있다. 주로 원 음원을 올린 비디오(lyrics video나 music video)와 무대 영상을 동시에 올린 비디오인 경우가 많다. Charlie Puth의 How Long의 경우 Video 영상과 Audio 영상에 동시에 올라와있고, 양쪽의 카테고리가 다르게 분류 되어 있는 것을 볼 수 있다. Sam smith 의 Too Good At Goodbyes의 경우 Audio, Video, 라이브 영상 총 3개가 리스트에 있고, 이중 라이브영상만 카테고리가 다르게 설정되어있다. Linkin Park의 One More Night의 경우에도 Video와 Audio의 카테고리가 다르게 되어있다. 이처럼 동일 음원의 비디오들을 다른 카테고리로 분류한 걸로 볼 때, 아무리 계층적 군집화가 정답이 없다 한들 이 연구는 실패라고 볼 수 있을 것이다. 그렇다면, 과연 어떤 부분 때문에 실패했을까? 부족한 전처리 과정, 언어적 문제, 정보습득의 시각적 편중이 그 원인으로 유추된다.

맨 처음 이 연구를 시작하게 된 이유 중 하나는 기존에 있던 감정 분석이 너무 단선적으로 되어있어서도 있다. 많은 감정 분석이 일차원적 이거나, 특정 단어에 기반해서 이루어진다. 인접 행렬을 만든 이유도 근본적으로는 단어 간 미묘한 의미의 차이를 정확하게 짚어 내기 위해서였다. 하지만 기존의 감정분석보다 더욱 정교한 알고리즘을 얻어내기에는 데이터가 너무 노이즈를 많이 포함하고 있다. 중간에 수작업으로 단어들을 편집하긴 했지만 이보다 더욱 정제된 알고리즘이 요구되는 것으로 보인다. Textblob의 품사탐색 알고리즘이 성능이 떨어지는 것도 이유 중 하나라고 할 수 있다.

뿐만 아니라, 어떤 단어가 어떻게 쓰일 지도 정확히 모른 상태로 단어만 보고 판단한 것도 문제라 할 수 있다. ‘Good’이라는 단어를 가지고 노래를 좋게 표현할 수도 있는 것이고, 나쁘게 표현할 수도 있다. Deep은 관형어로 쓰일 수도 있지만, 감정을 표현하는 형용사로도 쓰일 수 있다. Old의 경우 만약 해당 노래가 사자와 관련되어 있다면 형용사로서의 Old가 아닌 사자를 의미하는 The Old로 인해 많이 빈도수가 나올 것이다. 물론 알고리즘 도출에서 이렇게 의미 없는 변수가 많을 것을 예상 했었고, 이를 염두에 두어서 차원 축소를 진행했으나, 이를 전처리 이후에 줄이는게 아닌 전처리과정 에서 줄였어야 했다.

사람이 정보를 받아 들일 때 대부분의 정보는 시각에 편중된다. 라디오가 나오기 전까지 오랜 시간 동안 많은 사람들은 시각을 통해 대부분의 정보를 얻어왔다. 이러한 경향이 현대에도 남아있다. 공부한다 그러면 귀로 듣는 사람은 드물다. 대부분 신문이나 책, 인터넷을 통해 글자로 공부한다. 이로 인해 동일한 비디오여도 뮤직비디오가 나오게 되면 사람들이 음악에 집중하기 보다는 뮤직비디오안의 시각적 내용을 댓글에 더 표현할 수도 있다. 이는 Audio 영상들만 추려서 댓글 크롤링을 한 뒤 시행하면 개선될 여지가 있다.

머신러닝의 군집분석 모델상의 문제일 수도 있어서 동일한 형태의 분석을 K-평균 군집분석의 개량형태인 DBSCAN으로 군집분석을 추가로 실행한 결과(201501868\_ML\_5.py), Outliner로 추정되는 3개의 데이터를 제외하고는 병합 군집분석과 전부 동일하게 분류가 된 것으로 나타났다. 교차분석을 통해서 문제가 나오지 않았기에, 해당 프로젝트의 실패는 전처리 및 데이터 자체에만 있다고 볼 수 있다.

* 1. 결과 해석

결과는 실패했지만, 범위를 음원이 아닌 유튜브 영상으로 돌리게 되면 어느정도 활용의 여지는 있다고 본다. 현재 유튜브의 영상 추천 알고리즘과 비슷한 기능을 할 수 있기 때문이다. 댓글에 달린 감정명사를 기반으로 동일 카테고리의 비슷한 특징을 가진 비디오를 추천해줄 수 있다. 다만 이를 위해서는 전술했던 전처리의 개선이 반드시 이뤄져야 할 것이다.

결과가 실패로 끝난 이상 활용의 여지는 없지만, 만약 위의 원인들을 고친 결과 의미있는 결과값이 나온다면 다음과 같이 활용할 수 있다 : 음원추천 알고리즘 제안, 새로운 감정분석 알고리즘 개발.

현재 대부분의 음원 서비스사에서 사용하는 협업 기반 필터링 알고리즘은 해당 데이터를 해당 용도로 사용함을 알린 뒤 일일히 조사해서 모은다. 예를 들어, 유튜브의 경우 비디오 추천 알고리즘을 특정 비디오를 지목한 뒤 시청자들에게 ‘해당 비디오와 연관 있는 감정명사를 고르세요’같은 식으로 진행한다. 이는 데이터를 입력하는 사람이 의식한 상태에서 하는 것이기에 그 정확성이 떨어질 수 있고, 소비자입장에서 불편할 수 밖에 없다. 즉, 협업 기반 필터링의 전형적인 문제인 Cold Start를 피하기 위해 소비자에게 불편함을 줄 수 밖에 없다. 하지만 위의 알고리즘을 이용한다면 Youtube에서 사용자가 작성한 댓글 목록만 가지고 와도 어떤 노래를 좋아할지를 바로 추천해줄 수 있다. 엄연한 의미에서는 여전히 협업 기반 필터링에 해당하지만 해당 목적을 위해 소비자에게 해당 데이터를 직접적으로 요구하지 않아도 된다는 점에서 기존 협업 기반 필터링보다 한 단계 진보했다고 볼 수 있다.

뿐만 아니라, 위의 전처리단계에서 만들어낸 형용사(감정명사 포함)들의 인접행렬을 통해 기존의 감정분석 알고리즘을 한 단계 더 발전시킬 수도 있다. 수많은 단어들의 네트워크를 통해 단어 하나하나의 감정적 의미를 다른 단어로 환원할 뿐만 아니라, 다른 품사에 대한 알고리즘만 충분히 갖춰진다면 동일한 의미를 가진 다른 문장을 창조해낼 수도 있을 것이다. 뿐만 아니라, 해당 감정분석 시스템을 시간이 지나면서 지속적으로 발전시킬 수도 있는데, 새로운 형용사(또는 감정명사)를 넣기 위해서 해당 형용사가 들어간 문장을 끌고 오면 되기 때문이다.

1. “KSHMR explains how an accident helped him create 2010 hit ‘Like A G6’”, *We Rave You*, 2016.11.12 수정, 2020.11.02 접속, <https://weraveyou.com/2016/11/kshmr-like-g6-production/> [↑](#footnote-ref-1)