

컨텐츠 주제와 타겟 국가 선정에 따른 성장가능성 예측 2

컴퓨터공학과 2022108115

이우석

지난 주차 진행 상황

01 문제점

학습 데이터의 양이 부족


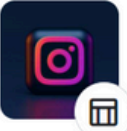




지난 발표에서 Category, Audience Country을 통해 Followers을 예측하는 모델과 Category, Audience Country, Followers을 통해 Engagement avg을 예측하는 모델을 학습시켰으나, 두 모델 모두 정확도가 매우 낮았습니다.

학습데이터가 1000개에 불과했기 때문에, 이것이 문제라고 생각함

01 문제점

다른 데이터셋 조사

해당 문제를 해결하기 위해 캐글의 여러 데이터셋을 조사했지만 Audience Country가 포함된 데이터셋은 찾을 수 없었으며, 데이터셋의 크기 또한 충분히 큰 데이터셋은 존재하지 않았음.

	Instagram Data Dataset · 2y ago · by Amir Motefaker Instagram Reach Analysis Data	36 3,731 downloads
	Top Instagram Influencers Data (Cleaned) Dataset · 2y ago · by SJ Analyzing data of top Instagram influencers	168 9,050 downloads
	Instagram Play Store Reviews Dataset · 1y ago · by Saloni Jhalani Instagram App Reviews dataset is a comprehensive collection of user reviews ...	40 1,760 downloads
	Instagram fake spammer genuine accounts Dataset · 6y ago · by Bardiya Bakhshandeh Predict whether an Instagram user is fake/spammer or not	105 8,482 downloads
	Threads, an Instagram app Reviews Dataset · 1y ago · by Saloni Jhalani The Threads, an Instagram App Reviews dataset is a comprehensive collectio...	68 2,862 downloads
	instagram-analysis Dataset · 1y ago · by Shubham Sadawarti	35

01 문제점

Trending YouTube Video Statistics

Daily statistics for trending YouTube videos



다른 데이터셋 확보

데이터셋을 조사하던중 기존에 학습하려던 인스타그램은 아니지만,
또다른 큰 시장인 유튜브에 관한 데이터셋을 발견함

해당 데이터셋은 10개국의 유튜브 정보를, 국가별로 적게는 3,000개에서 34,000개까지 조사하여
모델 학습을 위한 충분한 데이터라고 판단함

02 데이터셋

Trending YouTube Video Statistics

해당 데이터는 미국, 영국, 독일, 캐나다, 프랑스, 러시아, 멕시코, 한국, 일본, 인도 총 10개국의 유튜브 동영상에 대한 데이터를 포함합니다.
데이터는 하루의 최대 200개의 인기 동영상을 기록합니다.

Data Explorer

Version 115 (539.22 MB)

- {i} CA_category_id.json
- CAvideos.csv
- {i} DE_category_id.json
- DEvideos.csv
- {i} FR_category_id.json
- FRvideos.csv
- {i} GB_category_id.json
- GBvideos.csv
- {i} IN_category_id.json
- INvideos.csv
- {i} JP_category_id.json
- JPvideos.csv
- {i} KR_category_id.json
- KRvideos.csv
- {i} MX_category_id.json
- MXvideos.csv
- {i} RU_category_id.json
- RUvideos.csv
- {i} US_category_id.json
- USvideos.csv

02 데이터셋

▲ title ≡	▲ channel_ti... ≡	🔗 category_id ≡	📅 publish_ti... ≡	▲ tags ≡
좋아 by 민서_윤 종신_종니 답가	라꾸마코리아	22	2017-11- 13T07:07:36.000 Z	라꾸마 "윤종 신" "종니" "종 아" "살레" "민 서"
JSA 귀순 북한군 총격 부상	Edward	25	2017-11- 13T10:59:16.000 Z	JSA "귀순" "북한 군" "총격" "부 상" "JSA 귀순 북 한군 총격 부상"
나몰라패밀리 운동	나몰라패밀리 핫쇼	22	2017-11-	아디다스 "배배

Trending YouTube Video Statistics

해당 데이터의 column은 총 10개로 구성되어 있습니다.
비디오 id, 인기동영상으로 게시된 날짜, 제목, 채널명,
카테고리, 업로드 날짜, 태그, 시청수, 좋아요, 싫어요로 구성되어 있습니다.

이 중 모델 학습에 사용할 항목은 카테고리, 좋아요, 시청수와
국가별로 구분된 파일로 존재하는 csv파일을 하나로 합치고 부여한 국가명을 사용할 예정입니다.

02 데이터셋

데이터셋의 카테고리 id

해당 데이터셋의 카테고리는 숫자로된 id로 구분됩니다.
숫자별 카테고리 분류는 다음과 같습니다.

1, Film & Animation
2, Autos & Vehicles
10, Music
15, Pets & Animals
17, Sports
18, Short Movies
19, Travel & Events
20, Gaming
21, Videoblogging
22, People & Blogs
23, Comedy
24, Entertainment
25, News & Politics
26, Howto & Style
27, Education

28, Science & Technology
29, Nonprofits & Activism
30, Movies
31, Anime/Animation
32, Action/Adventure
33, Classics
34, Comedy
35, Documentary
36, Drama
37, Family
38, Foreign
39, Horror
40, Sci-Fi/Fantasy
41, Thriller
42, Shorts

03 데이터 전처리

데이터에서 필요한 항목만 남기고
글자를 숫자로 변환하고, 결측값처리 등의
전처리 과정을 진행하였습니다.
전처리 과정을 진행하며 국가 Column을 추가하였습니다

모든 국가의 csv파일을 하나로 합쳐
하나의 파일로 만들었습니다.

```
import pandas as pd
import os

# 특정 국가의 CSV 파일 경로
file_path = '/content/RUvideos.csv' # 파일 경로를 실제 경로로 변경

# 전처리된 데이터를 저장할 리스트
processed_data = []

# 파일을 읽어오기 (인코딩 오류가 있을 수 있으므로 encoding을 시도해봄)
try:
    # CSV 파일 읽기
    chunk = pd.read_csv(file_path, encoding='utf-8', low_memory=False)

    # 국가 이름을 추출
    country_name = 'Russia'

    # 필요한 열만 선택하기
    filtered_chunk = chunk[['category_id', 'views', 'likes']].copy()

    # 결측값 처리
    filtered_chunk['views'] = filtered_chunk['views'].fillna(0)
    filtered_chunk['likes'] = filtered_chunk['likes'].fillna(0)

    # category_id를 숫자형으로 변환 (문제가 있을 경우 0으로 처리)
    filtered_chunk['category_id'] = pd.to_numeric(filtered_chunk['cate

    # 'country' 컬럼에 해당 국가명을 추가
    filtered_chunk['country'] = country_name

    # 처리된 데이터를 리스트에 추가
```

03 데이터 전처리

합쳐진 데이터셋이 국가별로 몰려있어서 파일의 행을 무작위로 섞은 후 저장하였습니다.

```
category_id  views  likes  country
0           22   62408    334   Russia
1           22  330043   43841   Russia
2           24  424596   49854   Russia
3           22  112851    3566   Russia
4           24  243469   36216   Russia
```

```
[32] # 전체 데이터를 무작위로 섞기
      shuffled_data = final_data.sample(frac=1).reset_index(drop=True)

      # 무작위로 섞은 데이터를 CSV 파일로 저장
      shuffled_data.to_csv('/content/shuffled_data.csv', index=False)

      # 결과 확인
      print("파일이 무작위로 섞여 저장되었습니다.")
```

🔄 파일이 무작위로 섞여 저장되었습니다.

```
rows_10003_10009 = shuffled_data.iloc[10002:10009]

print(rows_10003_10009)
```

```
category_id  views  likes  country
10002         22    9275     373   Mexico
10003         22   46082    1102   Mexico
10004         10  7518990  1573046  Russia
10005         10   300850   32066   Japan
10006         26   46347    2606   Russia
10007         10   12498    1018   Mexico
10008         24  1327815   32895    USA
```

04 모델 학습

이렇게 확보한 데이터를 사용할 모델을 Pycaret을 이용하여 결정 및 학습을 할 예정입니다.

```
from pycaret.regression import *
import pandas as pd

# 전처리된 데이터 로드
data = pd.read_csv('/content/preprocessed_data.csv')

# PyCaret 환경 설정 (타겟 변수 'Followers'로 설정)
exp = setup(data, target='Followers', categorical_features=['Category', 'Audience Country'], session_id=123)

# 모델 비교 후 선택
best_model = compare_models()

# 모델 학습
final_model = create_model(best_model)

# 모델 튜닝
tuned_model = tune_model(final_model)

# 예측
predictions = predict_model(tuned_model, data)

# 예측된 값 확인
print(predictions.head())
```

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	Light Gradient Boosting Machine	4612437.9725	310267911844320.5625	17614423.4037	0.7929	0.2749	0.1222

Thank you

감사합니다.