

# **유튜브 영상 타이틀 및 썸네일 분석을 통한 분야별 영상 전략 수립**

**마케팅 A조**  
**김석호 | 민세연 | 손정용 | 이소민**

# CONTENTS

1 프로젝트 목적

2 가상 시나리오 설정

3 데이터 소개

4 데이터 크롤링

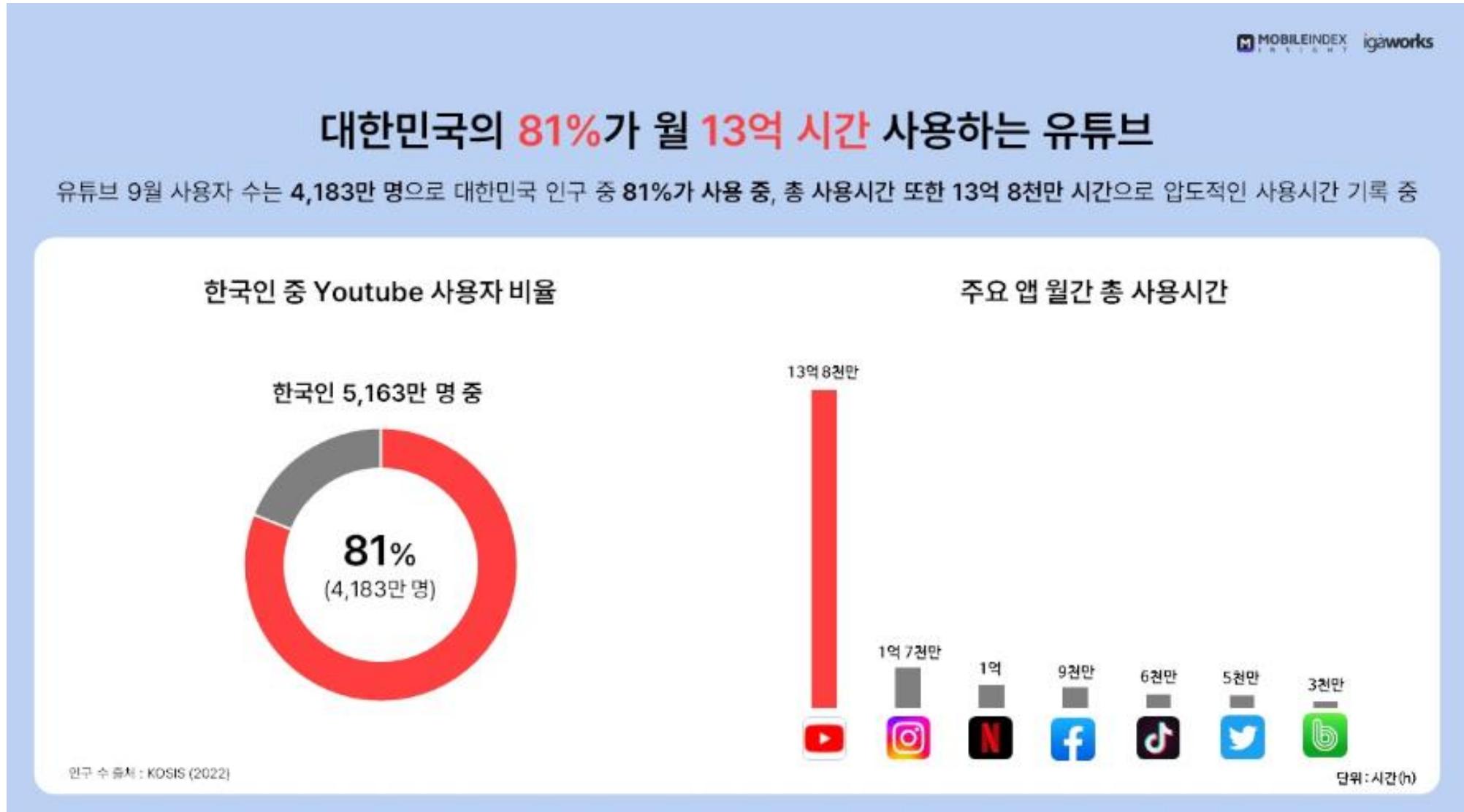
5 컨텐츠 소재 분석 & 결과

6 영상 타이틀 분석 & 결과

7 영상 썸네일 분석 & 결과

8 기대효과

# 01 프로젝트 목적



# 01 프로젝트 목적

MCN (Multi Channel Network)

: 다중 채널 네트워크는 인터넷 스트리밍 플랫폼에서 활동하는 사람들의 기획사



**SANDBOX**

샌드박스 네트워크  
소속 크리에이터 팀 수 &  
매출액

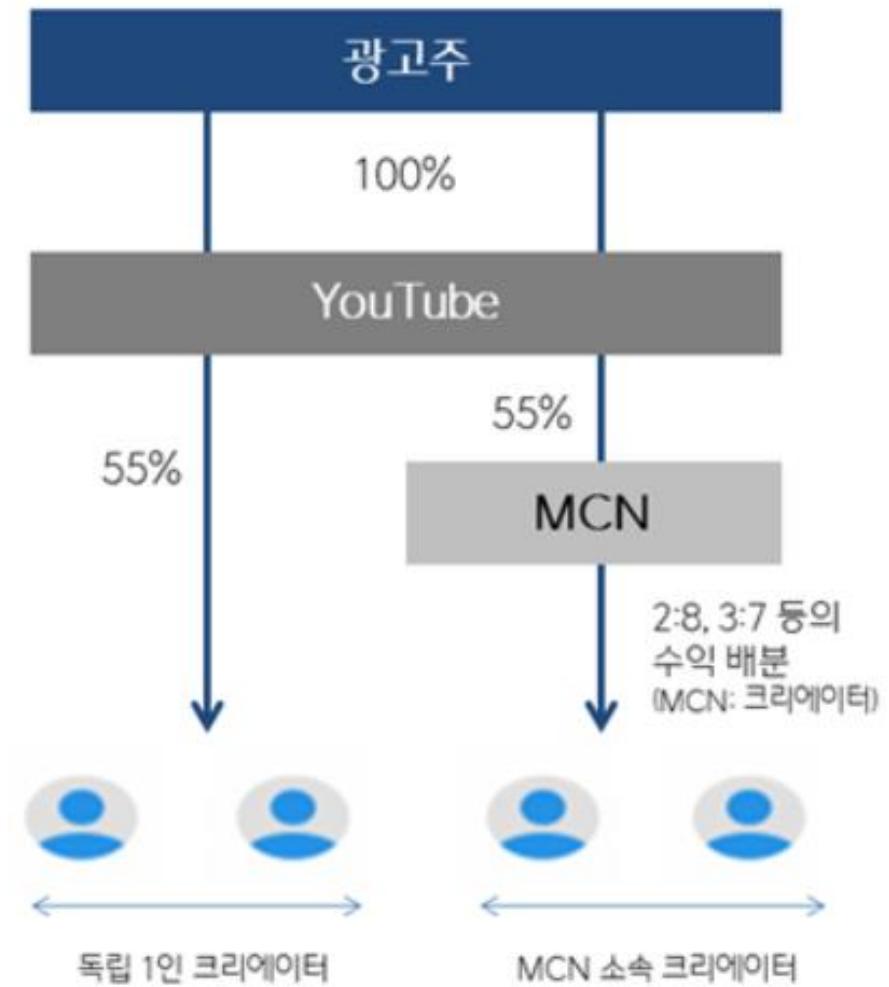
데이터 제공: 샌드박스 네트워크, 2020.12.25 기준



# 01 프로젝트 목적

✓ 수익구조의 변화 필요

광고 위주의 수익 → 신인 크리에이터 육성



# 01 프로젝트 목적

## ▣ 동영상



[초보 유튜버분들을 위한 유튜브 꿀팁 강의!](#)

YouTube · 대도서관TV (buzzbean11)  
2021. 1. 14.



[초보 유튜버 꿀팁 3가지 & 시작 7일 후기 \(유튜브 크리에이터\)](#)

YouTube · 주부아빠 Daddy J  
2018. 11. 24.



[초보 유튜버 꿀팁 - 조회수 구독자 늘리는법 1편 - 좋은 제목 ...](#)

YouTube · 주부아빠 Daddy J  
2018. 12. 13.



[초보 유튜버 꿀팁 - 썸네일 만드는 법 & 시작 14일 후기 \(유튜브 ...\)](#)

YouTube · 주부아빠 Daddy J  
2018. 12. 1.

모두 보기 →

<http://www.kocw.net> › home › search › kemView

[초보 유튜버를 위한 콘텐츠 제작 및 동영상 편집 \(인덕대학교\)](#)

초보 유튜버를 위한 콘텐츠 제작 및 동영상 편집 (인덕대학교). 신한대학교; 신종우; 0. 1863.  
(/5.0); (0); 강의담기; 공유; 오류접수. 오류접수(차시를 선택하세요).

<https://taling.me> › Talent › Detail ▾

[초보 유튜버를 위한 1:1 비대면 강의 - 탈잉](#)

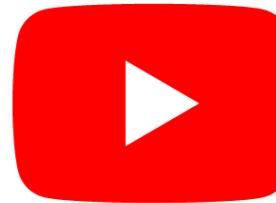
초보 유튜버를 위한 1:1 비대면 강의. 16,500원(시간) ... 책이나 다른 유튜버의 안내 영상만으로는 어떻게 해야 할지 감이 안잡히시는 분들 ...

<https://brunch.co.kr> › ... ▾

[초보 유튜버가 주제 고르는 방법 유튜브 시작하기 - 브런치](#)

2022. 12. 9. — 그래서 오늘 12만 유튜버 포리얼님이 알려주는 돈이 되는 주제 찾는 방법 4가지를 작성해 ... 초보 엄마 벗어나기 같은 강의를 할 수도 있고.,

# 01 프로젝트 목적



# YouTube

## 타이틀

“10명중 8명은 콘텐츠의 타이틀만 읽고 넘어간다”  
-미국의 콘텐츠 마케팅 회사 Quick Sprout-

- 콘텐츠를 클릭하게 하는 제목의 힘은 매우 중요

## 썸네일

- 사용자가 영상을 직관적으로 이해하고 흥미를 느끼게 하여 시청을 유도하는 효과가 있음
- 영상의 썸네일을 디자인할 때 어떠한 요소를 고려해야 할지 아는 것이 중요

# 01 프로젝트 목적

- 키워드 빈도분석을 통한 컨텐츠 소재 파악
- 텍스트 분석을 통한 타이틀 구성
- 이미지 분석을 통한 썸네일 추천



**트렌드 컨텐츠 제작지원  
서비스 구축**

## 02 가상 시나리오 설정



# SANDBOX

월드컵 스타 조규성 선수가 유튜브를 시작하고자  
할 때, 샌드박스 크리에이터 지원팀은 조규성 선수<sup>에게 어떤 영상 컨텐츠를 추천해 줄 수 있을까?</sup>



최근 카테고리별 조회수가 높은 영상 데이터를 분석  
하여 컨텐츠 소재 및 영상 타이틀&썸네일 추천!

# 03 데이터 소개



## 필터

업로드 날짜	구분	길이	기능별	정렬기준
지난 1시간	동영상	4분 미만	실시간	관련성
오늘	채널	4~20분	4K	업로드 날짜
이번 주	재생목록	20분 초과	HD	조회수
이번 달	영화		자막	평점
올해			크리에이티브 커먼즈 360° VR180 3D HDR 위치 구입한 항목	

# 03 데이터 소개

- 기간 : 올해(2022)
- 조건 1) 조회수 높은 순 (조회수 10000회 이상)  
2) 영상 길이 : 4분~20분
- 분야

생활·여행 (브이로그)	 <p>vlog   비오는 날엔 베이컨치즈김치전, 오징어 순대 데리야끼 치킨 덮밥 만들어 먹고 카... 조회수 57만회 • 13일 전 지현장</p> <p>영상에 나온 것을 베이컨치즈김치전   오징어순대   김치칼국수   데리야끼치킨덮밥   불닭 꿩이들 만명하세요 :) 다들 이번주도 잘 보냈... 자막</p>	먹방	 <p>줄어서 먹는 기사식당 햄박스테이크? 수유 동 돈까스와 햄박 먹방 조회수 53만회 • 1시간 전 tzuyang쯔양</p> <p>뷰티관련 협찬(화장품, 성형외과, 피부과 등) 및 다이어트 제품군 협찬(약,보조제)은 일절 진행하지 않고 있습니다. 이 부분 양해 부탁 ... 새 동영상 자막</p>
리뷰	 <p>이병헌X하정우 역대급 케미 폭발한 영화 ㅋㅋ [영화리뷰/결말포함] 조회수 48만회 • 4주 전 시미드네</p> <p>영화리뷰 #결말포함 #sf영화 #하정우 #이병헌 인네하세요 영화가 스며드는 "시 며드네"입니다. 즐겁게 보시고 오늘도 즐거운 하루 ... 16:00</p>	운동·헬스	 <p>뱃살 아랫뱃살 최고의 운동 조회수 774만회 • 1년 전 Thankyou BUBU</p> <p>뱃살, 영상 하나로 끝내고 싶다! 오늘 영상을 만들기 위해 저희 부부가 자막</p>
패션	 <p>나이대별 옷 잘 입는 법 (쉽게 알리드름~) 10대부터 50대까지 13:59</p> <p>나이대별 쉽게 옷 잘 입는 법✓(10대~50대까지) 조회수 43만회 • 4개월 전 AliceFunk 알리스펑크</p> <p>세상에서 제일 쉽게 옷 잘 입는 방법 옷 잘 입는 사람이란 건 놀까요? 본인에게 4K</p>		

# 04 데이터 크롤링

## 【셀레니움을 활용한 유튜브 영상 크롤링】

```
#라이브러리 불러오기
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
import time
import re
import pandas as pd
import urllib.request
```

```
#크롬 브라우저 생성  
dr = webdriver.Chrome('/Users/jungy/Downloads/chromedriver')  
dr.get('https://www.youtube.com/')
```

```
#검색어 입력  
keyword=input('검색어를 입력하시오 : ')  
elem = dr.find_element(By.XPATH,'/html/body/ytd-app/div[1]/div/ytd-masthead/div[3]/div[2]/ytd-searchbox/form/div[1]/div[1]/input')  
elem.send_keys(keyword)  
elem.send_keys(Keys.ENTER)
```

```
#검색 조건 입력  
dr.find_element(By.XPATH, '/html/body/ytd-app/div[1]/ytd-page-manager/ytd-search/div[1]/ytd-two-column-search-results-renderer/div[2]/div/ytd-search-filter-menu-renderer/div[1]/div/ytd-search-filter-item-details-renderer').click()  
dr.find_element(By.XPATH, '/html/body/ytd-app/div[1]/ytd-page-manager/ytd-search/div[1]/ytd-two-column-search-results-renderer/div[2]/div/ytd-search-filter-item-details-renderer/div[1]/div/ytd-search-filter-item-details').send_keys('인기')  
dr.find_element(By.XPATH, '/html/body/ytd-app/div[1]/ytd-page-manager/ytd-search/div[1]/ytd-two-column-search-results-renderer/div[2]/div/ytd-search-filter-item-details-renderer/div[1]/div/ytd-search-filter-item-details').send_keys(Keys.RETURN)  
dr.find_element(By.XPATH, '/html/body/ytd-app/div[1]/ytd-page-manager/ytd-search/div[1]/ytd-two-column-search-results-renderer/div[2]/div/ytd-search-filter-item-details-renderer/div[1]/div/ytd-search-filter-item-details').send_keys(Keys.RETURN)  
dr.find_element(By.XPATH, '/html/body/ytd-app/div[1]/ytd-page-manager/ytd-search/div[1]/ytd-two-column-search-results-renderer/div[2]/div/ytd-search-filter-item-details-renderer/div[1]/div/ytd-search-filter-item-details').send_keys(Keys.RETURN)  
dr.find_element(By.XPATH, '/html/body/ytd-app/div[1]/ytd-page-manager/ytd-search/div[1]/ytd-two-column-search-results-renderer/div[2]/div/ytd-search-filter-item-details-renderer/div[1]/div/ytd-search-filter-item-details').send_keys(Keys.RETURN)  
dr.find_element(By.XPATH, '/html/body/ytd-app/div[1]/ytd-page-manager/ytd-search/div[1]/ytd-two-column-search-results-renderer/div[2]/div/ytd-search-filter-item-details-renderer/div[1]/div/ytd-search-filter-item-details').send_keys(Keys.RETURN)
```

# 04 데이터 크롤링

## [셀레니움을 활용한 유튜브 영상 크롤링]

#검색된 동영상 수 확인

```
contents=dr.find_elements(By.CSS_SELECTOR, '#contents > ytd-video-renderer')
len(contents)
```

#썸네일, 제목, 조회수, 기간, 유튜버, 장르 저장해둘 리스트 생성

```
thumbnails=[]
titles=[]
views=[]
timelines=[]
youtubers=[]
genre=[]
```

#썸네일, 제목, 조회수, 기간, 유튜버, 장르 저장

```
for content in contents:
    thumbnail=content.find_element(By.ID, 'thumbnail').find_element(By.TAG_NAME, 'img').get_attribute('src')
    thumbnails.append(thumbnail)
    title=content.find_element(By.ID, 'video-title').text
    titles.append(title)
    view=content.find_element(By.ID, 'metadata-line').text.split('\n')[0].split(' ')[1]
    views.append(view)
    timeline=content.find_element(By.ID, 'metadata-line').text.split('\n')[1]
    timelines.append(timeline)
    youtuber=content.find_element(By.ID, 'channel-info').find_element(By.CSS_SELECTOR, '#text > a').text
    youtubers.append(youtuber)
    genre.append(keyword)
```

#Pandas프레임으로 변환

```
df=pd.DataFrame(zip(titles,views,int(timelines),youtubers,genre,thumbnails),columns=['title','view','timeline','youtuber','genre','thumbnails'])
```

# 04 데이터 크롤링

## [데이터 크롤링 결과]

```
#조회수 10000 이상 영상만 추출  
df1=df[df['view']>=10000]
```

	title	view	timeline	youtuber	genre	thumbnail
0	Mukbang Big Bottle Candy drink 보틀 캔디 젤리 먹방 DON...	530000000	10개월 전	DONA 도나	먹방	<a href="https://i.ytimg.com/vi/MWshfHzTE/hq720.jpg?sqp...">https://i.ytimg.com/vi/MWshfHzTE/hq720.jpg?sqp...</a>
1	Red VS Blue Food Dessert Mukbang 빨간색 파란색 디저트 먹...	210000000	6개월 전	DONA 도나	먹방	<a href="https://i.ytimg.com/vi/wpJYpknITk4/hq720.jpg?sqp...">https://i.ytimg.com/vi/wpJYpknITk4/hq720.jpg?sqp...</a>
2	Big Food VS Small Food Challenge 대왕 음식 챌린지 대왕 ...	110000000	8개월 전	DONA 도나	먹방	<a href="https://i.ytimg.com/vi/HBRecULSEbk/hq720.jpg?sqp...">https://i.ytimg.com/vi/HBRecULSEbk/hq720.jpg?sqp...</a>
3	ASMR MUKBANG  직접 만든 치토스 양념치킨 치즈스틱 핫도그 짜파게티 먹방 ...	98220000	11개월 전	설기양SULGI	먹방	<a href="https://i.ytimg.com/vi/HQsswovXtPU/hq720.jpg?sqp...">https://i.ytimg.com/vi/HQsswovXtPU/hq720.jpg?sqp...</a>
4	Mukbang Jello Shooter race 틱톡 챌린지 먹방 TikTok Ch...	36820000	5개월 전	HUBABU	먹방	<a href="https://i.ytimg.com/vi/_l29Ec9nkZg/hq720.jpg?sqp...">https://i.ytimg.com/vi/_l29Ec9nkZg/hq720.jpg?sqp...</a>
...	...	...	...	...	...	...
437	[먹방브이로그] 밑빠진 승이에 소주붓기 술이 승승승~~   맛나 반점, 오징어볶음, ...	15000	1일 전	승이어트 soong'iet	먹방	<a href="https://i.ytimg.com/vi/U5dxacAdCpc/hq720.jpg?sqp...">https://i.ytimg.com/vi/U5dxacAdCpc/hq720.jpg?sqp...</a>
438	꽃돼지 머릿고기 수육 김장배추김치 굴무침 카레밥 짜장밥 한식먹방[korean foo...	14000	18시간 전	[Flower pig]꽃돼지	먹방	<a href="https://i.ytimg.com/vi/iVT6mVpfG9g/hqdefault.j...">https://i.ytimg.com/vi/iVT6mVpfG9g/hqdefault.j...</a>
439	[리얼먹방]매콤한 양념과 달큰한 무 넣고[[코다리무조림]] 파래 무침과 달래장같이 겉...	13000	2일 전	혁이네집밥	먹방	<a href="https://i.ytimg.com/vi/HHQlb7T5bak/hq720.jpg?sqp...">https://i.ytimg.com/vi/HHQlb7T5bak/hq720.jpg?sqp...</a>
440	이삭토스트 신메뉴 미트팡?! 먹방!! [ Isaac Toast Mukbang ]	11000	19시간 전	마빡	먹방	<a href="https://i.ytimg.com/vi/HJhXknWLZCU/hq720.jpg?sqp...">https://i.ytimg.com/vi/HJhXknWLZCU/hq720.jpg?sqp...</a>
441	두툼바삭 무공돈까스에 치즈돈까스 뜨끈한 계란우동 바삭한 새우튀김까지 리얼먹방 Por...	10000	4일 전	배고픈 뺑떡이 [Hungry B.D.]	먹방	<a href="https://i.ytimg.com/vi/PESC6p1GE60/hqdefault.j...">https://i.ytimg.com/vi/PESC6p1GE60/hqdefault.j...</a>

# 05 컨텐츠 소재 분석 & 결과

## [키워드 빈도 분석]

- KoNLPy라이브러리의 Okt와 Kkma 클래스 사용
- 브이로그, 리뷰, 운동의 경우 Okt를 통해 명사, 동사, 형용사, 부사를 분석
- 패션, 먹방의 경우 Kkma를 통해 명사를 분석하고 Okt로 나머지 품사를 분석

```
1 mukbang=pd.read_csv('/content/drive/MyDrive/DScover_A조/메인 프로젝트/주제 크롤링/먹방.csv')

1 okt = Okt()
2 sentences_tag = []
3 for sentence in mukbang['title']:
4     morph = okt.pos(sentence)
5     sentences_tag.append(morph)
6     print(morph)

#pos를 이용해 title 열에 대해 형태소 분석 및 품사 태깅

1 adj_list = []
2 verb_list = []
3 adverb_list = []
4 alpha_list = []
5
6 for my_sentence in sentences_tag:
7     for word, tag in my_sentence:
8         if tag in ['Adjective']:
9             adj_list.append(word)
10        if tag in ['Verb']:
11            verb_list.append(word)
12        if tag in ['Adverb']:
13            adverb_list.append(word)
14        if tag in ['Alpha']:
15            alpha_list.append(word)

#품사별 단어를 리스트에 추가
```

# 05 컨텐츠 소재 분석 & 결과

## [키워드 빈도 분석-형용사, 동사, 부사, 알파벳 분석]

- collections 모듈의 Counter 클래스를 이용해 품사별 등장빈도가 높은 단어 확인

## [형용사]

```
1 from collections import Counter  
2 adj_counts = Counter(adj_list)  
3 adj_counts.most_common(20)  
  
[('매운', 20),  
 ('마라', 12),  
 ('입니다', 8),  
 ('맛있는', 6),  
 ('없는', 6),  
 ('신기한', 5),  
 ('다양한', 4),  
 ('있을까요', 3),  
 ('미친', 3),  
 ('있을까', 3),  
 ('빼고파서', 3),  
 ('미', 3),  
 ('많은', 3),  
 ('아닌가요', 3),  
 ('좋아', 3),  
 ('힘든', 2),  
 ('시원한', 2),  
 ('많은데', 2),  
 ('유명한', 2),  
 ('맛있음', 2)]
```

[동사]

```
1 verb_counts = Counter(verb_list)
2 verb_counts.most_common(20)

[('만든', 37),
 ('먹는', 14),
 ('챌', 9),
 ('하는', 8),
 ('먹으면', 8),
 ('무침', 5),
 ('먹을', 5),
 ('먹어', 5),
 ('탄', 5),
 ('먹었더니', 4),
 ('꿔', 3),
 ('먹고', 3),
 ('하면', 3),
 ('가서', 3),
 ('일까', 3),
 ('하기', 3),
 ('뜨', 3),
 ('골라', 3),
 ('크', 3),
 ('와', 2)]
```

[부사]

```
1 adverb_counts = Counter(adverb_list)
2 adverb_counts.most_common(10)

[(('다', 15),
 ('하', 9),
 ('너무', 7),
 ('함께', 5),
 ('혹', 4),
 ('팡팡', 3),
 ('매주', 3),
 ('이렇게', 3),
 ('없이', 2),
 ('드디어', 2)])
```

## [알파벳]

```
1 alpha_counts = Counter(alpha_list)
2 alpha_counts.most_common(30)

[('MUKBANG', 178),
 ('ASMR', 160),
 ('Mukbang', 66),
 ('EATING', 63),
 ('SOUND', 53),
 ('REAL', 39),
 ('ft', 35),
 ('mukbang', 33),
 ('SUB', 28),
 ('FRIED', 25),
 ('Spicy', 25),
 ('CHICKEN', 24),
 ('NOODLES', 23),
 ('AND', 22),
 ('Korean', 21),
 ('Eating', 18),
 ('DESSERT', 17),
 ('EP', 16),
 ('Beef', 15),
 ('show', 15)]
```

# 05 컨텐츠 소재 분석 & 결과

## [키워드 빈도 분석-명사분석]

```
1 from konlpy.tag import Kkma  
2 kkma = Kkma()  
3 noun_list=[]  
4 for i in mukbang['title']:  
5     noun_list.append(kkma.nouns(i))  
6 print(noun_list)  
  
[['보', '보틀', '틀', '캔디', '젤리', '먹', '먹방', '방', '나'], ['빨간색', '파란색', '디저트', '먹', '먹방', '방', '나'],
```

```
1 dic={}  
2 for i in noun_list:  
3     for word in i:  
4         if word not in dic:  
5             dic[word]=1  
6         else:  
7             dic[word]+=1  
8 print(dic)  
  
{'보': 2, '보틀': 1, '틀': 1, '캔디': 1, '젤리': 17, '먹': 416, '먹방': 375, '방': 415, '나': 7, '빨간색': 1, '파란색': 1,
```

```
1 kkma_d1 = sorted(dic.items(), key=lambda x: x[1], reverse=True)  
2 kkma_d2=[item for item in kkma_d1 if item[1]>=10]  
3 print(kkma_d2)  
  
('먹', 416), ('방', 415), ('먹방', 375), ('리얼', 102), ('ㅣ', 78), ('리얼먹방', 65), ('레', 54), ('치킨', 47),
```

# 05 컨텐츠 소재 분석 & 결과

## [키워드 빈도 분석-명사분석]

## [먹방]

```
1 stopwords=['사운드','리얼먹방','먹','방','먹방','ㅣ','레','집','불','티','시','유','3','개','1','2','아이스','신','전','피','5','안','수','10','우','원','만','분','하이',
2 df=[item for item in kkma_d2 if item[0] not in stopwords] #의미없는 단어는 불용어로 처리해 제거
3 df
```

('리얼', 102),  
('치킨', 47),  
('밥', 47),  
('디저트', 33),  
('리얼사운드', 33),  
('치즈', 32),  
('김치', 28),  
('면', 28),  
(('통', 28),  
('밋', 28),  
('닭', 25),  
('양념', 24),  
('대왕', 21),  
('크림', 21),  
('떡볶이', 21),  
('대창', 21),  
('음식', 20),  
('직접', 20),  
('라면', 20),  
('닭볶음', 20),  
('젤리', 17),  
('불닭볶음면', 17),  
('구이', 16),  
('편의점', 16),  
('꿀', 16),  
('조합', 16),  
('짬뽕', 16),  
('파김치', 16),  
('아이스크림', 15),  
('소세지', 14),  
('계란', 14),

# 05 컨텐츠 소재 분석 & 결과

## [키워드 빈도 분석-명사분석]

### [패션]

[('웃', 76),  
('코디', 74),  
('여자', 53),  
('가을', 49),  
('남자', 47),  
('2022', 46),  
('룩', 44),  
('중년', 40),  
('법', 34),  
('트렌드', 33),  
('스타일', 32),  
('가지', 31),  
('중년패션코디', 28),  
('겨울', 27),  
('스타일링', 26),  
('추천', 26),  
('룩', 25),  
('여름', 23),  
('요즘', 21),  
('템', 21),  
('룩북', 20),  
('워크', 20),  
('데일리', 19),  
('하울', 19),  
('2023', 18),  
('유행', 18),  
('브랜드', 17),  
('한국', 16),  
('아이템', 15),  
('데일리룩', 15),  
('법여자', 15),  
('명품', 14),  
('봄', 14),  
('정리', 13),  
('율', 12),  
('런던', 11),]

### [브이로그]

[('일상', 75),  
('카페', 73),  
('여행', 46),  
('알바', 37),  
('먹방', 36),  
('피시방', 28),  
('공부', 26),  
('복학생', 23),  
('맛집', 22),  
('영상', 21),  
('일본', 21),  
('살', 21),  
('첫', 20),  
('준비', 19),  
('커플', 18),  
('대학생', 18),  
('음료', 18),  
('데이트', 18),  
('쇼핑', 18),  
('환자', 17),  
('캠핑', 16),  
('직장인', 16),  
('겨울', 15),  
('날', 15),  
('미국', 15),  
('나', 15),  
('하루', 14),  
('다이어트', 14),  
('일주일', 14),  
('시험', 13),  
('생일', 13),  
('학교', 12),  
('디저트', 12),  
('제조', 12),]

### [리뷰]

[('결말', 186),  
('포함', 184),  
('애니', 73),  
('영화', 41),  
('남자', 19),  
('실화', 17),  
('세계', 17),  
('충격', 17),  
('사람', 13),  
('여자', 13),  
('방법', 13),  
('역대', 13),  
('일본', 12),  
('추천', 12),  
('진짜', 12),  
('버전', 12),  
('공개', 11),  
('풀', 11),  
('분석', 11),  
('킹', 11),  
('최후', 10),  
('지구', 10),  
('쿠션', 10),  
('이유', 10)]

### [운동]

[('뱃살', 73),  
('루틴', 65),  
('하체', 65),  
('방법', 60),  
('근육', 51),  
('어깨', 50),  
('최고', 38),  
('가슴', 38),  
('다이어트', 35),  
('걷기', 35),  
('살', 34),  
('법', 33),  
('엉덩이', 33),  
('지방', 32),  
('중년', 28),  
('가장', 28),  
('복근', 26),  
('내장', 25),  
('등', 24),  
('스쿼트', 22),  
('효과', 22),  
('방송', 21),  
('대신', 21),  
('배', 20),  
('매일', 20),  
('이유', 19),  
('허리', 19),  
('산소', 19),  
('전신', 18),  
('근력', 18),  
('식단', 17),]

# 05 컨텐츠 소재 분석 & 결과

## [워드 클라우드를 통한 시각화]

- 각 카테고리별 등장 횟수가 10번 이상인 명사만을 고려

```
from wordcloud import WordCloud
import matplotlib.pyplot as plt
font_path = '/usr/share/fonts/truetype/nanum/NanumGothic.ttf'

wordcloud = WordCloud(
    font_path = font_path,
    max_font_size = 100,
    width = 700,
    height = 400,
    background_color = 'white',
    max_words = 55
)

#먹방
wordcloud.generate_from_frequencies(mukbang_words)
#wordcloud.to_file('mukbang_wordcloud.png')
print('먹방')
plt.axis("off")
plt.imshow(wordcloud, interpolation='bilinear')
plt.show()
```



## [먹방]

# 05 컨텐츠 소재 분석 & 결과

## [워드 클라우드를 통한 시각화]



[브이로그]



## [리뷰]



## [운동]



[패션]

# 06 영상 타이틀 분석 & 결과

## [텍스트 유사도 분석]

- 텍스트를 TF-IDF로 벡터화 한 이후에, 벡터화된 각 문장 간의 유사도를 측정하여 카테고리 별 키워드에 따른 타이틀 예시를 추출

```
1 mukbang=pd.read_csv('/content/drive/MyDrive/DScover_A조/메인 프로젝트/주제 크롤링/먹방.csv')

1 from sklearn.feature_extraction.text import TfidfVectorizer

1 vectorizer=TfidfVectorizer(min_df =2 , ngram_range=(1,2))
2 # min_df-몇개의 문서에서 등장했는지,
3 # ngram 1=단어 1개씩 묶음(unigram)
4 #         2=단어 2개를 한 묶음으로 취급(bigram)

1 features=vectorizer.fit_transform(mukbang['title'])
2 features.todense()                                     # 제목 벡터화

matrix([[0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]])
```

```
1 feature_names=vectorizer.get_feature_names_out()

1 import numpy as np
2 dtm_np = np.array(features.todense()) # dtm document term matrix
```

# 06 영상 타이틀 분석 & 결과

## [텍스트 유사도 분석]

```
1 from sklearn.metrics.pairwise import cosine_similarity

1 cosine_sim=cosine_similarity(dtm_np,dtm_np)          # 코사인유사도연산

1 cosine_sim.shape

(442, 442)

1 text='꿀조합'
2 title_in=[]
3 for i,title in enumerate(mukbang['title']):
4     if text in title:
5         title_in.append((i,title))
6 title_in

[(27,
  'SUB)밥도둑 간장게장 간장새우 먹방! 🍜🍤 흰쌀밥에 김부각 달걀후라이까지 꿀조합 리얼사운드 Soy Sauce Marinated Crab & Shrimp Mukbang Asmr'),
 (35,
  'SUB)새로나온 불닭짬뽕에 왕새우 넣어서 먹방! 🍜 스팸달걀김밥 통김밥까지 라면 분식 꿀조합 리얼사운드 Buldak Ramyeon & Kimbab Mukbang Asmr'),
 (38,
  'SUB)꾸덕족족 진진짜라에 불닭팽이버섯 불닭가리비 먹방! 🔥 파김치까지 꿀조합 리얼사운드 JjajangRamyeon & Buldak Enoki Mushroom Mukbang Asmr'),
 (40,
  '순두부 열라면과 대왕 유부초밥 꿀조합♥ 먹방'),
 (62,
  'SUB)육즙팡팡 바삭촉촉한 통대창 먹방! 대창 막창 파김치까지(ft. 맥주🍺) 꿀조합 리얼사운드 Daechang Makchang Mukbang Asmr'),
 (135,
  'SUB)육즙팡팡 바삭빠삭 통대창에 곱창 염통 먹방! 파김치까지 대창 꿀조합 리얼사운드 Crispy Beef Intestine Mukbang ASMR'),
 (190,
  'SUB)살살 녹는 통연어 2kg 먹방! 직접 만든 연어국수까지 연어파티 꿀조합 리얼사운드 Salmon Mukbang Asmr'),
 (199,
  'SUB)버거킹 신메뉴 골든치즈렐라 와퍼 먹방! 🍔 통새우와퍼 타バス코버거에 너겟킹 치즈스틱 바삭킹까지 치킨 햄버거 꿀조합 리얼사운드 Burger King Mukbang Asmr'),
```

# 06 영상 타이틀 분석 & 결과

## [텍스트 유사도 분석]

```
1 #idx = 조회 수가 높은 타이틀 순서
2 idx=title_in[0][0]
3 sim_scores=list(enumerate(cosine_sim[idx]))          # 해당 키워드와 모든 제목의 유사도를 가져옴

1 sim_scores=sorted(sim_scores,key=lambda x :x[1], reverse=True)    # 유사도에 따른 제목 정렬

1 sim_scores=sim_scores[0:11]                                         # 가장 유사한 11개 제목의 인덱스를 반환
2 sim_scores

[(27, 1.0),
 (332, 0.3828825200552326),
 (261, 0.3816886579013365),
 (318, 0.3322824666946257),
 (190, 0.31432227368644217),
 (108, 0.30493423217749055),
 (35, 0.30439366044728766),
 (199, 0.28299610157238353),
 (196, 0.2701268358642922),
 (212, 0.2664343147642917),
 (38, 0.26540043090582466)]
```

## [먹방] 키워드: 꿀조합

```
1 mukbang.iloc[title_indices]['title']

27 SUB)밥도둑 간장게장 간장새우 먹방! 🍜🔥 흰쌀밥에 김부각 달걀후라이까지 꿀조합 리...
332 업기떡볶이 신메뉴 리얼사운드 먹방 | 중국당면 짜장업떡 오돌뼈 주먹밥 | Soy-s...
261 SUB)뜨끈킬킬 돼지곱창전골 먹방! 🔥 당면 우동사리에 볶음밥 주먹밥까지 꿀조합 리얼사...
318 SUB)뿌링클 맛초킹 치킨 먹방! ♦뿌링핫도그 치즈볼까지 단짠단짠 꿀조합 리얼사운드 ...
190 SUB)살살 녹는 통연어 2kg 먹방! 직접 만든 연어국수까지 연어파티 꿀조합 리얼...
108 대왕 연어초밥 1kg + 새우장 & 개장 2kg 먹방 🔥 밥도둑 | GIANT SALM...
35 SUB)새로나온 불닭짬뽕에 왕새우 넣어서 먹방! 🔮 스팸달걀김밥 통김밥까지 라면 분식...
199 SUB)버거킹 신메뉴 골든치즈렐라 와퍼 먹방! 🍔 통새우와퍼 타バス코버거에 너겟킹 치...
196 SUB)얼얼한 마라샹궈 먹방 🔥 (ft. 크림새우) Mara Xiangguo Crea...
212 SUB)꼬독쫀득 매운 국물닭발에 닭발튀김 맥주 먹방! 🍺 주먹밥 계란찜까지 닭발 꿀조...
38 SUB)꾸덕족족 진진짜라에 불닭팽이버섯 불닭가리비 먹방! 🔥 파김치까지 꿀조합 리얼사...
Name: title, dtype: object
```

# 06 영상 타이틀 분석 & 결과

## [텍스트 유사도 분석]

### [패션] 키워드: 코디

```
1 fashion.iloc[title_indices]['title']  
  
22      한물간 패션은 노노~ 한끗차이로 달라지는 부츠코디 🍃 님아 이렇게 입지마오..  
244      미미짱의 패션은 누구에게나 멋있을까?  
147      겨울에 꼭! 무조건 이렇게 입으세요✓ 평범한 옷도 달라지는 요즘 데일리룩 코디 👩  
383      이렇게 입으면 싼티나요..! 한끗차이로 저렴해보이는 패션 특징  
128      부티나지 않는 패션은?/중년패션코디/ 싼티나는 여자/ 옷 코디 잘하는법의 패션실수  
164      초등학생 5학년?🤔 갤민아 허언증이 왜 이렇게 심해?!!  
262      이렇게 귀여워도 되나 싶을 정도로 귀여운 강아지 패션  
410      패션, 옷에 관심을 갖게 되면 사람이 이렇게 변합니다 ㅋㅋㅋㅋ  
67      '육상 여신' 김민지 사복 패션은 처음이지? 트와이스 다현 실물 영접+육상 아이돌 ...  
110      패션은 거들뿐! 멋보다 웃음을 택한 비투비가 코디한 가을룩은?? | 배달의 프로들2...  
39      "이렇게 입으면 허수다" 구독자 독서실 패션까지 훈수질 해버리기  
Name: title, dtype: object
```

### [브이로그] 키워드: 캠핑

```
1 vlog.iloc[title_indices]['title']  
  
0      쓸아지는 폭우 우중 캠핑 . 빗소리 . 캠핑 브이로그 . 솔로캠핑 .  
102     쓸아지는 빗속에서 텐트 안 힐링 우중캠핑 .캠핑 브이로그 . 솔로캠핑 . 두부 김치전 .  
15      폭우 우중캠핑 | 가족과 함께 멈추지 않는 빗소리 | 캠핑 브이로그  
27      캠핑 브이로그 . 극한의 눈보라 속 생존하기 . 폭설 솔로캠핑  
465      건장한 여자친구와 함께 떠난 첫 캠핑 브이로그  
184     대형견이 4년만에 드디어 수영을한다고?! 캠핑 브이로그 🐕 MY DOG IS SWI...  
90      캠핑 브이로그 . 폭설 내린 날 강아지 그리고 설중캠핑 . 화목난로 스테이크  
428     먹방 브이로그) 캠핑은 모르겠고 먹성은 알아요 ( 생채비빔밥 시래기국 송어회 짜파게...  
204     ENG) 끼니 챙기다가 하루 다 가는 브이로그 / 먹으려 간 반나절 캠핑(feat....  
30      앨범깡 브이로그 💕 팬싸컷 채우려고 200장 앨범깡 • 오프깡 • 준등기 • 핫트랙스...  
33      도둑질 브이로그  
Name: title, dtype: object
```

# 06 영상 타이틀 분석 & 결과

## [텍스트 유사도 분석]

### [리뷰] 키워드: 세계

```
1 review.iloc[title_indices]['title']

7 전 세계 132개국 선판매 확정!! 《범죄도시2》 영화리뷰
1 말이 필요없는 [범죄도시2] 영화리뷰 (마동석, 손석구, 최귀화, 박지환 aka 장...
90 [영화리뷰/결말포함] 전 세계 사람들이 모두 사라진 세상에서 살아남은 생존자들
392 [후리뷰] 아스날 1위 확정!!! 근데 제주스 다음에는 골 넣어줄꺼죠??
234 수백년간 차별받아온 '수인'들의 세계:1부 [영화리뷰/결말포함]
237 세계 정복을 위해 마왕이 한 선택 【애니리뷰/결말포함】
415 앞에선 약한 척하며 뒷 세계 최강자의 삶 [애니리뷰/결말포함][어둠의 실력자가 되고...
297 《공조2》영화리뷰 (현빈, 유해진, 윤아, 다니엘 헤니 주연/출연)
465 《데시벨》배우 인터뷰 & 영화리뷰 (김래원, 이종석, 정상훈, 박병은, 차은우)
25 10대 도둑에게 정x을 넣어 참교육하는 노인 💔(영화리뷰)(결말포함)(맨인더다크)
34 무시받던 약골 찌질이가 본모습을 보이면 벌이지는 일 [영화리뷰/결말포함]
Name: title, dtype: object
```

### [운동] 키워드: 하체

```
1 workout.iloc[title_indices]['title']

24 스쿼트 없이 하체를 키우는 최고의 하체운동
452 스쿼트 '없이 '하체근육' 을 좋아지게 만드는 최고의 운동 방법! (대퇴사두, 엉덩이, ...
125 황철순과 하체운동
216 빠르게 팔을 키우는 최고의 팔 운동 프로그램 [믿고 보는 Jeff형]
321 홈트 필라테스 하체운동
50 근손실 없이 무조건 살 빼지는 최고의 '유산소운동' 방법!!
456 무릎관절에 무리 없는 다리근육 키우는 최고의 엉덩이 근육운동방법! [무릎 통증 없는...
260 걷기 말고 이 하체 운동 3분씩 하세요! 40대 50대 이후 다리근육 키우는 최고의...
341 엉덩이 짹짝 채워주는 최고의 힙업운동! 리버스 하이퍼 [힙딥운동]
178 매일 한번만 이걸 하세요! 운동 초보자도 나잇살 군살 다 빼드립니다. 빠르게 근력을...
121 [SUB] 금요일 올데이 다이어트식단 / 간식 / 치팅 / 하체운동 Vlog
Name: title, dtype: object
```

# 06 영상 타이틀 분석 & 결과

## [LSTM을 이용한 제목 생성]

- 최빈 형용사, 동사, 부사 분석을 활용한 예시 타이틀 생성

```
1 df = pd.read_csv('/content/drive/MyDrive/DScover_A조/메인 프로젝트/주제 크롤링/먹방.csv')
```

```
1 title = []
2 # 헤드라인의 값들을 리스트로 저장
3 title.extend(list(df.title.values))
4 title[:5]

['Mukbang Big Bottle Candy drink 보틀 캔디 젤리 먹방 DONA 도나',
 'Red VS Blue Food Dessert Mukbang 빨간색 파란색 디저트 먹방 DONA 도나',
 'Big Food VS Small Food Challenge 대왕 음식 챌린지 대왕 눈알젤리 먹방 DONA 도나',
 'ASMR MUKBANG! 직접 만든 치토스 양념치킨 치즈스틱 핫도그 짜파게티 먹방 & 레시피 FRIED CHICKEN AND BLACK BEAN NOODLES EATING',
 'Mukbang Jello Shooter race 턱톡 챌린지 먹방 TikTok Challenge compilation by Hubabu']
```

```
1 def reprocessing(raw_sentence):
2     preproceseed_sentence = raw_sentence.encode("utf8").decode("ascii",'ignore')
3     # 구두점 제거와 동시에 소문자화
4     return ''.join(word for word in preproceseed_sentence if word not in punctuation).lower()
5
6 prepocessed_headline = [reprocessing(x) for x in title]      # 구두점제거및단어소문자화
7 prepocessed_headline[:5]
```

```
'mukbang big bottle candy drink dona',
'red vs blue food dessert mukbang dona',
'big food vs small food challenge dona',
'asmr mukbang fried chicken and black bean noodles eating',
'mukbang jello shooter race tiktok challenge compilation by hubabu']
```

# 06 영상 타이틀 분석 & 결과

## [LSTM을 이용한 제목 생성]

```
1 tokenizer = Tokenizer()
2 tokenizer.fit_on_texts(title)
3 vocab_size = len(tokenizer.word_index) + 1
4 print('단어 집합의 크기 : %d' % vocab_size)

단어 집합의 크기 : 2602

1 sequences = list()
2
3 for sentence in title:                                # 정수인코딩 및 훈련데이터구성
4
5     # 각 샘플에 대한 정수 인코딩
6     encoded = tokenizer.texts_to_sequences([sentence])[0]
7     for i in range(1, len(encoded)):
8         sequence = encoded[:i+1]
9         sequences.append(sequence)
10
11 sequences[:11]

[[2, 192],
 [2, 192, 310],
 [2, 192, 310, 49],
 [2, 192, 310, 49, 311],
 [2, 192, 310, 49, 311, 620],
 [2, 192, 310, 49, 311, 620, 621],
 [2, 192, 310, 49, 311, 620, 621, 39],
 [2, 192, 310, 49, 311, 620, 621, 39, 1],
 [2, 192, 310, 49, 311, 620, 621, 39, 1, 193],
 [2, 192, 310, 49, 311, 620, 621, 39, 1, 193, 194],
 [312, 62]]
```

# 06 영상 타이틀 분석 & 결과

## [LSTM을 이용한 제목 생성]

```
1 index_to_word = {}
2 for key, value in tokenizer.word_index.items(): # 인덱스를 단어로 바꾸기 위해 index_to_word를 생성
3     index_to_word[value] = key
4
5 print('빈도수 상위 582번 단어 : {}'.format(index_to_word[582]))
```

빈도수 상위 582번 단어 : 만드는

```
1 max_len = max([len(l) for l in sequences])
2 print('샘플의 최대 길이 : {}'.format(max_len))
```

# 가장 긴 샘플 길이 확인  
샘플의 최대 길이 : 24

```
1 sequences = pad_sequences(sequences, maxlen=max_len, padding='pre')
2 print(sequences[:3])
```

# 전체 샘플의 길이를 동일하게 만들어줌

```
1 sequences = np.array(sequences)
2 X = sequences[:, :-1]
3 y = sequences[:, -1]
```

# 맨 우측 단어만 레이블로 분리

```
1 y = to_categorical(y, num_classes=vocab_size)
```

# 레이블 y에 대해 원 핫 인코딩 수행

```
1 embedding_dim = 10
2 hidden_units = 128
3
4 model = Sequential()
5 model.add(Embedding(vocab_size, embedding_dim))
6 model.add(LSTM(hidden_units))
7 model.add(Dense(vocab_size, activation='softmax'))
8 model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
9 model.fit(X, y, epochs=200, verbose=2)
```

# 모델 설계 및 학습

# 06 영상 타이틀 분석 & 결과

## [LSTM을 이용한 제목 생성]

```
1 def sentence_generation(model, tokenizer, current_word, n): # 모델, 토크나이저, 현재 단어, 반복할 횟수
2     init_word = current_word
3     sentence = '' # 문장생성함수정의
4
5     # n번 반복
6     for _ in range(n):
7         encoded = tokenizer.texts_to_sequences([current_word])[0]
8         encoded = pad_sequences([encoded], maxlen=max_len-1, padding='pre')
9
10    # 입력한 X(현재 단어)에 대해서 y를 예측하고 y(예측한 단어)를 result에 저장.
11    result = model.predict(encoded, verbose=0)
12    result = np.argmax(result, axis=1)
13
14    for word, index in tokenizer.word_index.items():
15        # 만약 예측한 단어와 인덱스와 동일한 단어가 있다면
16        if index == result:
17            break
18
19    # 현재 단어 + ' ' + 예측 단어를 현재 단어로 변경
20    current_word = current_word + ' ' + word
21
22    # 예측 단어를 문장에 저장
23    sentence = sentence + ' ' + word
24
25    sentence = init_word + sentence
26    return sentence
```

## [먹방]

```
1 print(sentence_generation(model, tokenizer, '이렇게', 7))
이렇게 통으로 먹으니까 맛있네요 ☆ 대하구이 전어회에 소주한잔 |

1 print(sentence_generation(model, tokenizer, '오늘', 5))
오늘 솔뚜껑삽겹살에 김치 가득 레전드 조합

1 print(sentence_generation(model, tokenizer, '떡볶이', 5))
떡볶이 서열 먹방 서열 1등 떡볶이는

1 print(sentence_generation(model, tokenizer, '리얼', 5))
리얼 걸바속쫀 누텔라 딸기 크로풀에 몰티저스

1 print(sentence_generation(model, tokenizer, '리얼 사운드', 5))
리얼 사운드 대왕 먹방 🍒 대왕이 미취게 땡겨버린다…후식 볶음밥까지

1 print(sentence_generation(model, tokenizer, '미친', 9))
미친 서열 먹방 대결로 이겨보라고 마술사가 식신을 마술로 이기는 방법
```

# 06 영상 타이틀 분석 & 결과

## [LSTM을 이용한 제목 생성]

### [패션]

```
1 print(sentence_generation(model, tokenizer, '좋아하는', 5))
```

좋아하는 패션 유행은 돌고 돈다 진짜

```
1 print(sentence_generation(model, tokenizer, '이렇게', 4))
```

이렇게 입으면 여자들이 무조건 설렘

```
1 print(sentence_generation(model, tokenizer, '고급스런', 3))
```

고급스런 중년가을패션 귀티나는 가을분위기연출

```
1 print(sentence_generation(model, tokenizer, '세련된', 5))
```

세련된 패션 트렌드 10분안에 떼 먹여드림

```
1 print(sentence_generation(model, tokenizer, '요즘', 7))
```

요즘 20대 남자 패션 코디 어떻게 입어 😊

```
1 print(sentence_generation(model, tokenizer, '2022년', 5))
```

2022년 최고의 wearable 겨울 패션 트렌드

### [브이로그]

```
1 print(sentence_generation(model, tokenizer, '소소한', 3))
```

소소한 일상 브이로그 ❤️ 동물생일파티

```
1 print(sentence_generation(model, tokenizer, '좋은', 5))
```

좋은 브이로그 일본여행 고인물의 시 부야 여행법

```
1 print(sentence_generation(model, tokenizer, '재밌는', 4))
```

재밌는 vlog 피시방 알바 브이로그

```
1 print(sentence_generation(model, tokenizer, '일상', 5))
```

일상 브이로그 19개월 아기와 경주여행 황리단길

```
1 print(sentence_generation(model, tokenizer, '쇼핑', 4))
```

쇼핑 이건 그지인지 절약인지 모르겠음

```
1 print(sentence_generation(model, tokenizer, '캠핑', 7))
```

캠핑 브이로그 극한의 눈보라 속 생존하기 폭설 솔로캠핑

# 06 영상 타이틀 분석 & 결과

## [LSTM을 이용한 제목 생성]

### [리뷰]

```
1 print(sentence_generation(model, tokenizer, '드디어', 5))
```

드디어 밝혀지는 🔥 ss급 능력자 소녀와 구자윤의 관계

```
1 print(sentence_generation(model, tokenizer, '이렇게', 5))
```

이렇게 잽나는 줄 몰랐네요 영화리뷰 결말포함

```
1 print(sentence_generation(model, tokenizer, '진짜', 5))
```

진짜 복수 를 보여주는 애니【애니리뷰 결말포함】

```
1 print(sentence_generation(model, tokenizer, '비싼', 6))
```

비싼 값을 할까 neo qled vs oled

```
1 print(sentence_generation(model, tokenizer, '충격', 6))
```

충격 이 소주 왜 맛있지 진짜 강추합니다

```
1 print(sentence_generation(model, tokenizer, '오로지', 5))
```

오로지 '복수' 만을 그려온 소년 상 편

### [운동]

```
1 print(sentence_generation(model, tokenizer, '무조건', 4))
```

무조건 살빠지는 운동 🔥 10분만에 칼로리 불태우기

```
1 print(sentence_generation(model, tokenizer, '하루', 5))
```

하루 15분 뱃살빼는 최고의 운동 매운맛

```
1 print(sentence_generation(model, tokenizer, '빠르게', 5))
```

빠르게 칼로리 버닝 서서하는 유산소 다이어트

```
1 print(sentence_generation(model, tokenizer, '열심히', 5))
```

열심히 운동하는데 왜 몸이 안 좋아질까

```
1 print(sentence_generation(model, tokenizer, '어떻게', 7))
```

어떻게 '이 동작' 을 9분만 했더니 볼록한 '아랫뱃살' 이 사라졌다 🔥

```
1 print(sentence_generation(model, tokenizer, '엄청난', 7))
```

엄청난 통증 없이 누워서 하는 최고의 하체근력강화 운동

# 07 영상 썸네일 분석 & 결과

## [썸네일 이미지 저장 후 불러오기]

- 이전에 저장한 썸네일 링크를 사진 파일로 저장 후 배열의 형태로 변환

```
#썸네일을 이미지로 저장
```

```
cnt=0
for img in df1['thumbnail']:
    urlib.request.urlretrieve(img, keyword + str(cnt) + ".jpg")
    cnt+=1
```

```
#저장된 이미지 썸네일 리스트로 불러오기
```

```
path = r"/Users/jungy/Downloads/먹방"
os.chdir(path)
mukbang = []
with os.scandir(path) as files:
    for file in files:
        if file.name.endswith('.jpg'):
            mukbang.append(file.name)
print(mukbang)
```

```
#이미지 배열화
```

```
img_list=[]
for img in fashion:
    image=load_img(img, target_size=(224,224))
    image_np = np.array(image)
    img_list.append(image_np)
img_np=np.array(img_list)
```

# 07 영상 썸네일 분석 & 결과

## [VGG-16 모델을 활용하여 이미지 특징 추출]

```
#VGG16 모델 불러오기(OUTPUT)
model = VGG16()
model = Model(inputs = model.inputs, outputs = model.layers[-2].output)

#이미지 특징 추출 할수 경의
def extract_features(file, model):
    img = load_img(file, target_size=(224,224))
    img = np.array(img)
    reshaped_img = img.reshape(1,224,224,3)
    imgx = preprocess_input(reshaped_img)
    features = model.predict(imgx, use_multiprocessing=True)
    return features

data = {}
p = r"/Users/jungy/Downloads/먹방"

#이미지 특징 추출
for img in mukbang:
    try:
        feat = extract_features(img,model)
        data[img] = feat
    except:
        with open(p,'wb') as file:
            pickle.dump(data,file)

filenames= np.array(list(data.keys()))
feat = np.array(list(data.values()))
print(feat.shape)

feat = feat.reshape(-1,4096)
print(feat.shape)
```

# 07 영상 썸네일 분석 & 결과

## [PCA 차원 축소 & KMEANS CLUSTERING]

- PCA를 활용하여 여러 변수를 2차원으로 축소
- KMEANS를 활용하여 3개의 군집으로 CLUSTERING

```
from sklearn.cluster import KMeans  
from sklearn.decomposition import PCA
```

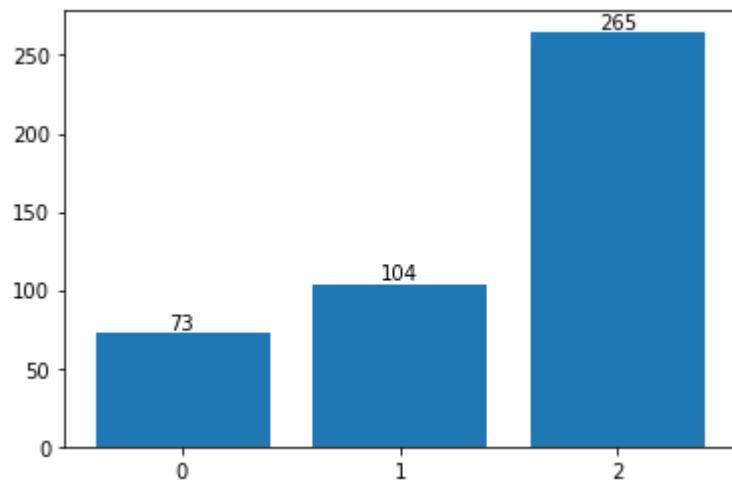
```
#PCA를 통한 차원 축소  
pca = PCA(n_components=2, random_state=22)  
pca.fit(feat)  
x = pca.transform(feat)
```

```
#Kmeans를 활용한 clustering  
kmeans = KMeans(n_clusters=3, random_state=22)  
kmeans.fit(x)
```

```
#Kmeans 군집별 데이터 수 시각화  
label=kmeans.labels_.tolist()  
total=Counter(label)  
keys=total.keys()  
values=total.values()  
bar=plt.bar(keys,values)  
plt.xticks([0,1,2])  
  
for rect in bar:  
    height = rect.get_height()  
    plt.text(rect.get_x() + rect.get_width()/2.0, height, '%.f' % height, ha='center', va='bottom', size = 10)  
plt.show()
```

# 07 영상 썸네일 분석 & 결과

[CLUSTERING 결과]



[0번 군집]



[1번 군집]



[2번 군집]



# 07 영상 썸네일 분석 & 결과

[먹방 카테고리 대표 썸네일 분석 결과]

먹방



# 07 영상 썸네일 분석 & 결과

[카테고리별 대표 썸네일 분석 결과]

생활·여행 (브이로그)



리뷰



운동



패션



## 08 기대 효과

- 컨텐츠 분야에 대한 경험이 적은 1인 크리에이터들에게 썸네일 디자인 및 영상 타이틀 선정을 위한 가이드라인을 제시할 것으로 기대함
- MCN(Multi Channel Network) 기업의 소속 크리에이터들에게 인기 있는 컨텐츠 소재를 통한 마케팅 전략, 신인 육성에 도움을 줄 것으로 기대함