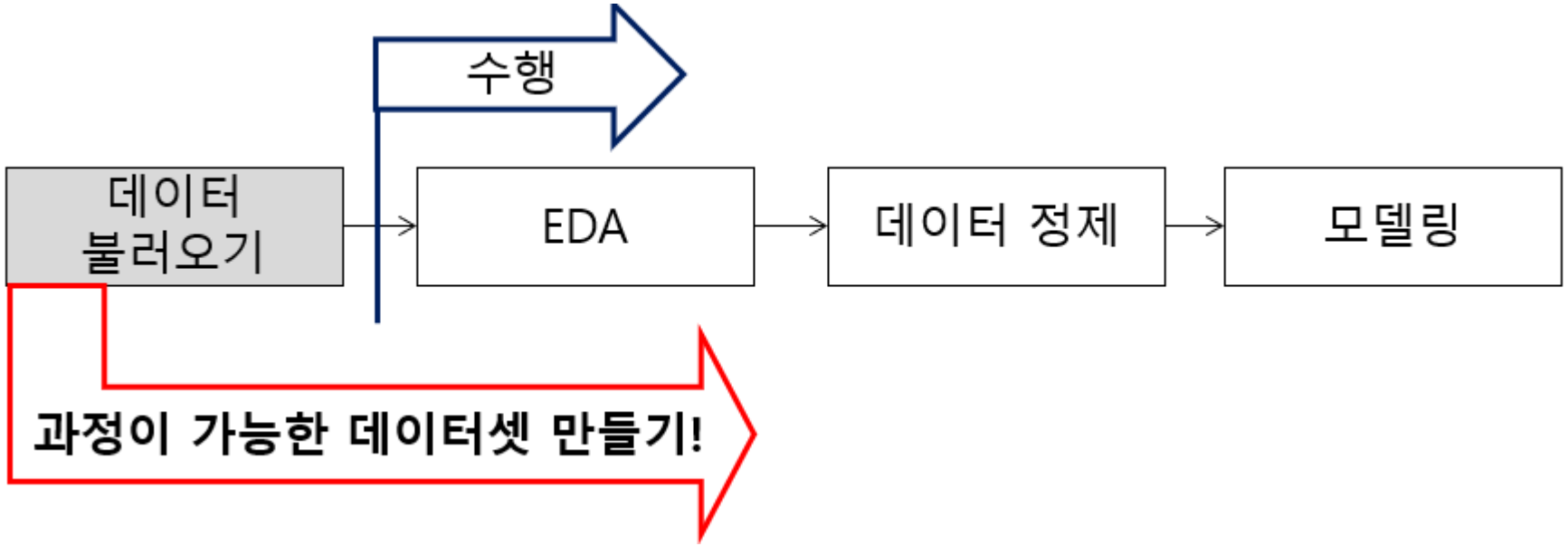
AI를 위한 기초데이터통계학

Final Project
Build Your Own Training Dataset

2019-12-18

Contents

01. Overview



rottentomatoes 사이트에서 영화 평점과 리뷰 데이터 수집

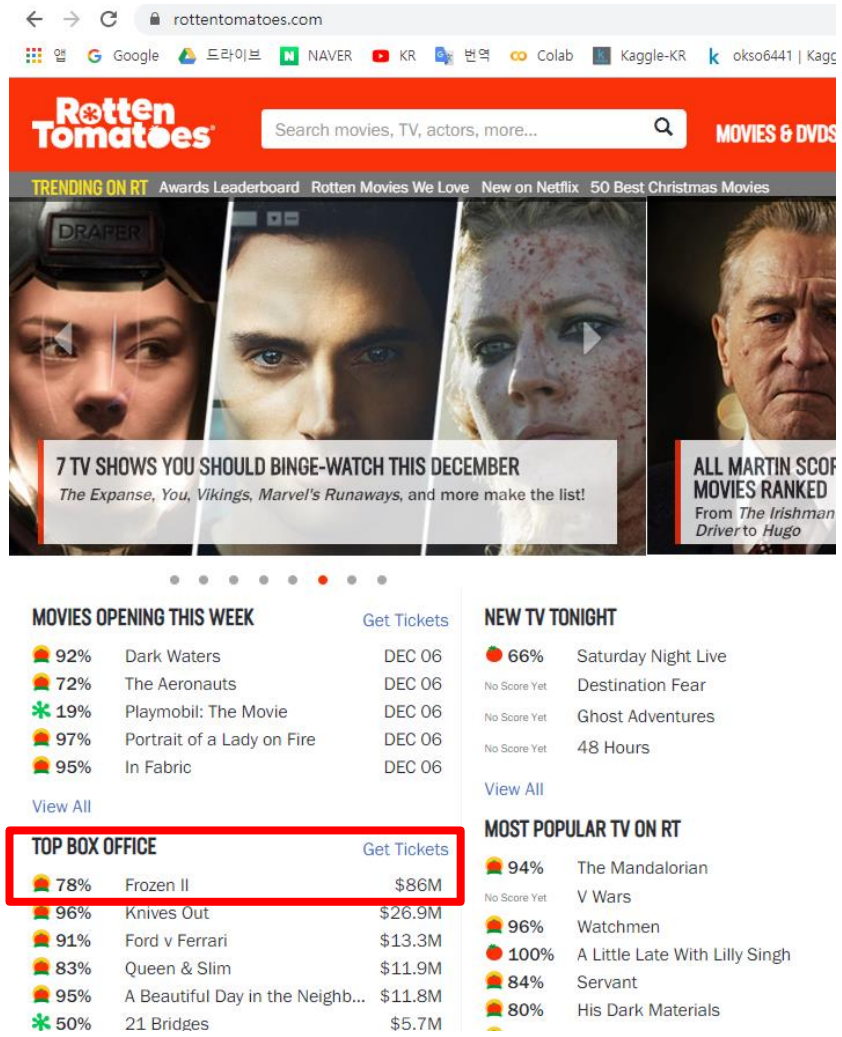
크롤링(Crawling):

- 웹 페이지의 하이퍼링크를 순회하면서 웹 페이지를 다운로드 하는 방법

02. 크롤링 - 영화 선정

2019.12.08일 기준 TOP BOX OFFICE

겨울왕국2 선정



02. 크롤링 > 대상 URL 확인

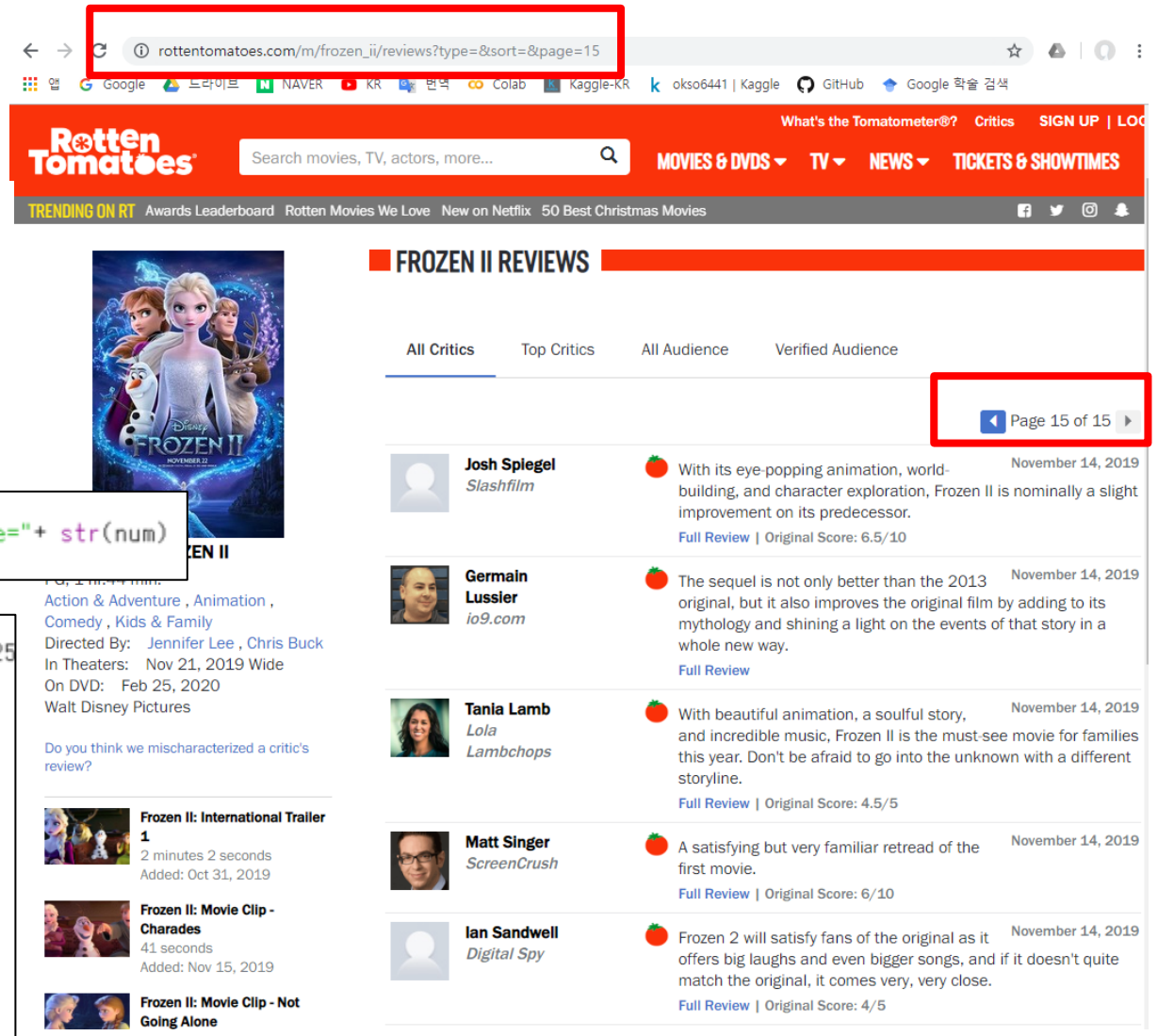
겨울왕국 리뷰 페이지

- 15페이지 분량

https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=15
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=14
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=13
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=12
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=11
…
https://www.rottentomatoes.com/m/해당영화/리뷰페이지?type=&sort=&page=페이지번호

```
for num in range(1, 16):
    url = "https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page="+ str(num)
    print(url)
```

```
>>>
 RESTART: C:₩Users₩ksh₩AppData₩Local₩Programs₩Python₩Python37₩Scripts₩DCIT67125
FP_1.py
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=1
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=2
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=3
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=4
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=5
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=6
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=7
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=8
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=9
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=10
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=11
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=12
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=13
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=14
https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page=15
```



0.2 크롤링> 라이브러리 설치

크롤링을 위한 requests, beatifulsoup4 설치

```
C:₩Users₩ksh₩AppData₩Local₩Programs₩Python₩Python37>pip install beautifulsoup4
Collecting beautifulsoup4
  Downloading https://files.pythonhosted.org/packages/3b/c8/a55eb6ea11cd7e5ac4bacdf92bac4693b90d3ba79268be16527555e186f0/beautifulsoup4-4.8.1-py3-none-any.whl (101kB)
     |████████████████████████████████| 102kB 243kB/s
Collecting soupsieve>=1.2 (from beautifulsoup4)
  Downloading https://files.pythonhosted.org/packages/81/94/03c0f04471fc245d08d0a99f7946ac228ca98da4fa75796c507f61e688c2/soupsieve-1.9.5-py2.py3-none-any.whl
Installing collected packages: soupsieve, beautifulsoup4
Successfully installed beautifulsoup4-4.8.1 soupsieve-1.9.5
WARNING: You are using pip version 19.2.3, however version 19.3.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```

```
C:₩Users₩ksh₩AppData₩Local₩Programs₩Python₩Python37>pip install requests
Collecting requests
  Downloading https://files.pythonhosted.org/packages/51/bd/23c926cd341ea6b7dd0b2a00aba99ae0f828be89d72b2190f27c11d4b7fb/requests-2.22.0-py2.py3-none-any.whl (57kB)
     |████████████████████████████████| 61kB 206kB/s
Collecting urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 (from requests)
  Downloading https://files.pythonhosted.org/packages/b4/40/a9837291310ee1ccc242ceb6ebfd9eb21539649f193a7c8c86ba15b98539/urllib3-1.25.7-py2.py3-none-any.whl (125kB)
     |████████████████████████████████| 133kB 726kB/s
Collecting chardet<3.1.0,>=3.0.2 (from requests)
  Downloading https://files.pythonhosted.org/packages/bc/a9/01ffebfb562e4274b6487b4bb1ddec7ca55ec7510b22e4c51f14098443b8/chardet-3.0.4-py2.py3-none-any.whl (133kB)
     |████████████████████████████████| 143kB ...
Collecting idna<2.9,>=2.5 (from requests)
  Downloading https://files.pythonhosted.org/packages/14/2c/cd551d81dbe15200be1cf41cd03869a46fe7226e7450af7a6545bfc474c9/idna-2.8-py2.py3-none-any.whl (58kB)
     |████████████████████████████████| 61kB 3.8MB/s
Collecting certifi>=2017.4.17 (from requests)
  Downloading https://files.pythonhosted.org/packages/b9/63/df50cac98ea0d5b006c55a399c3bf1db9da7b5a24de7890bc9cfd5dd9e99/certifi-2019.11.28-py2.py3-none-any.whl (156kB)
     |████████████████████████████████| 163kB ...
Installing collected packages: urllib3, chardet, idna, certifi, requests
Successfully installed certifi-2019.11.28 chardet-3.0.4 idna-2.8 requests-2.22.0 urllib3-1.25.7
WARNING: You are using pip version 19.2.3, however version 19.3.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```

02. 크롤링

BeautifulSoup 으로 크롤링

```
import requests
from bs4 import BeautifulSoup

for num in range(1, 16):
    url = requests.get("https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page="+ str(num))
    soup = BeautifulSoup(url.text, "html.parser")
    print(soup)
```

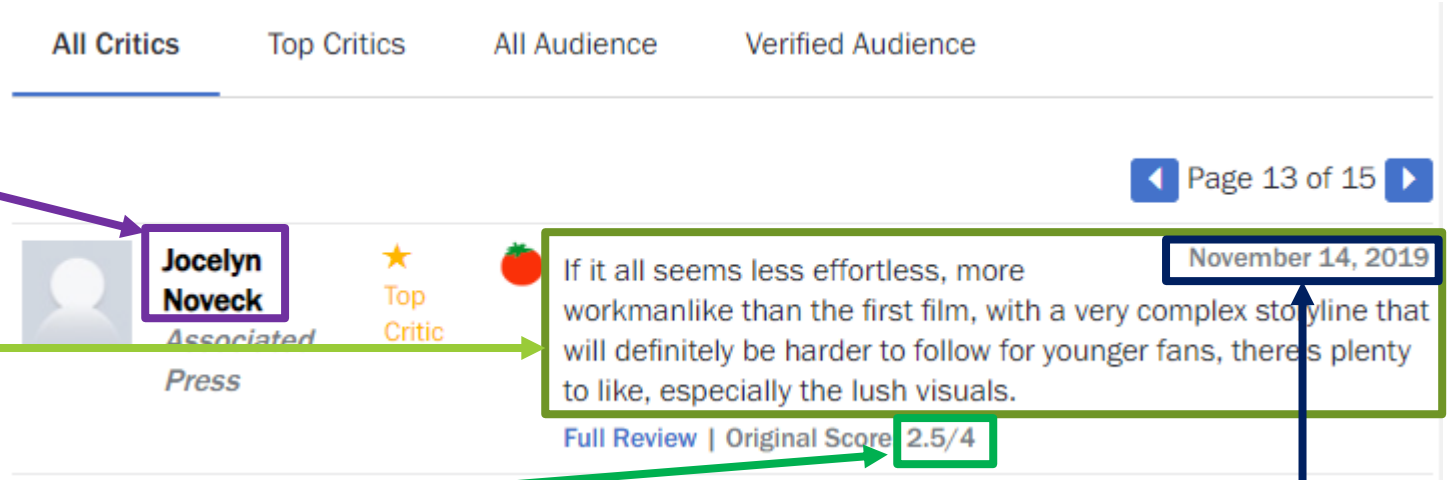
```
>>>
 RESTART: C:₩Users₩ksh₩AppData₩Local₩Programs₩Python₩Python37₩Scripts₩DCIT67125_FP_1.py
Squeezed text (3277 lines).
<!DOCTYPE html>

<html dir="ltr" lang="en" xmlns:fb="http://www.facebook.com/2008/fbml" xmlns:og="http://opengraphprotocol.org/schema/">
<head prefix="og: http://ogp.me/ns# flixstertomatoes: http://ogp.me/ns/apps/flixstertomatoes#">
<!-- salt=lay-def-02-juRm -->
<meta content="text/html; charset=utf-8" http-equiv="Content-Type"/>
<meta content="ie=edge" http-equiv="x-ua-compatible"/>
<meta content="width=device-width, initial-scale=1" name="viewport"/>
<title>Frozen II - Movie Reviews</title>
```

크롤링(Crawling): 웹 페이지의 하이퍼링크를 순회하면서 웹 페이지를 다운로드 하는 방법

02. 크롤링> 페이지에서 추출 대상 확인

원하는 추출 대상

- 리뷰어
- text 리뷰
- 숫자 평점
- 리뷰 날짜



02. 크롤링> 리뷰 라인(row) 추출

리뷰가 포함된 라인 요소 확인

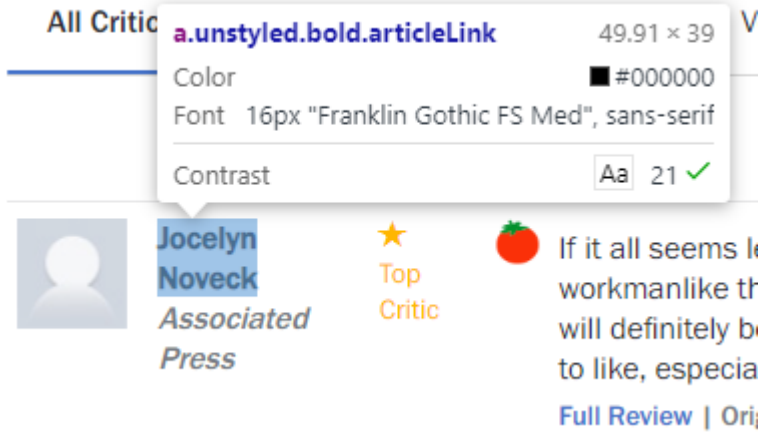

```python
import requests
from bs4 import BeautifulSoup

for num in range(1, 16):
    url = requests.get("https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page="+ str(num))
    soup = BeautifulSoup(url.text, "html.parser")

    all_review = soup.findAll("div", attrs={"class": "row review_table_row"})

    print(all_review)
```

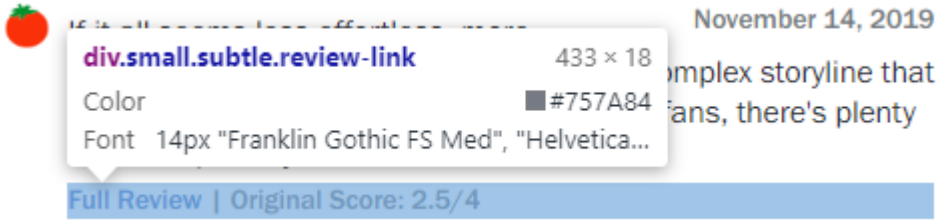
```
[<div class="row review_table_row">
<div class="col-xs-8">
<div class="col-sm-7 col-xs-16 critic_img">
<img class="critic_thumb fullWidth" src="https://d2a5cgar23scu2.cloudfront
</div>
<div class="col-sm-4 col-xs-8 top_critic col-sm-push-13">
</div>
<div class="col-sm-13 col-xs-24 col-sm-pull-4 critic_name">
<a class="unstyled bold articleLink" href="/critic/josh-spiegel-16191">Jos
<br>
<a href="/source-1996">
<em class="subtle critic-publication">Slashfilm</em>
</a>
</br></div>
</div>
<div class="col-xs-16 review_container">
```

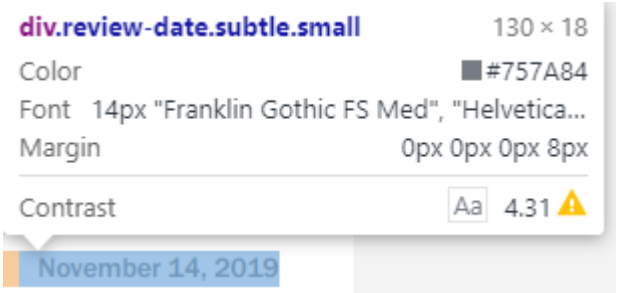
02. 크롤링> 라인에서 추출 요소 확인

요소: 리뷰어, text 리뷰, 숫자 평점, 리뷰 날짜

구분	요소	class
리뷰어	a	unstyled bold articleLink
text 리뷰	div	the_review
숫자평점	div	small subtle review-link
리뷰날짜	div	review-date subtle small





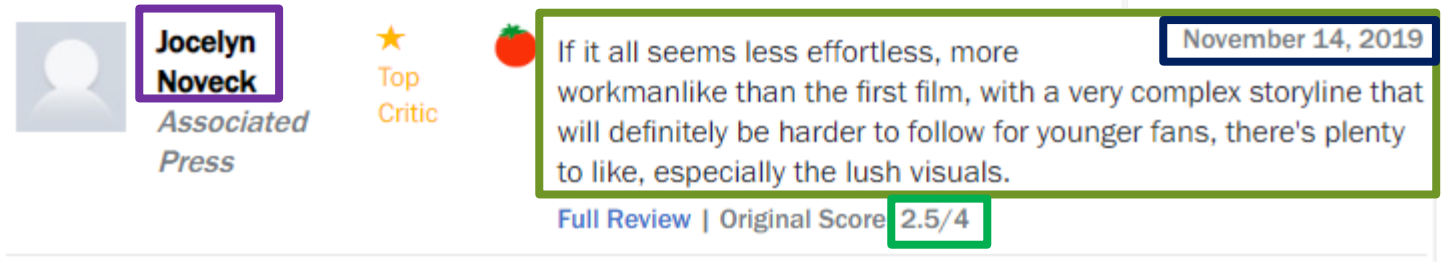


02. 크롤링> 라인별로 데이터 추출

```python
import requests
from bs4 import BeautifulSoup

for num in range(1, 16):
    url = requests.get("https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page="+ str(num))
    soup = BeautifulSoup(url.text, "html.parser")

    all_review = soup.findAll("div", attrs={"class": "row review_table_row"})

    for line in all_review:
        reviewer = line.find("a", attrs={"class": "unstyled bold articleLink"}).get_text()
        review = line.find("div", attrs={"class": "the_review"}).get_text()
        review_point_original = line.find("div", attrs={"class": "small subtle review-link"}).get_text()
        review_dt = line.find("div", attrs={"class": "review-date subtle small"}
```

구분	요소	class
리뷰어	a	unstyled bold articleLink
text 리뷰	div	the_review
숫자평점	div	small subtle review-link
리뷰날짜	div	review-date subtle small

```
Jocelyn Noveck
                                    If it all seems less effortless, more workmanlike than the first film, with a very complex storyline that will d
efinitely be harder to follow for younger fans, there's plenty to like, especially the lush visuals.
Full Review
                                                                    | Original Score: 2.5/4
November 14, 2019
```



02. 크롤링> 추출된 데이터 csv 파일로 저장

2차원 리스트로 생성, padnas Dataframe 으로 변환해서 저장

```
C:₩Users₩ksh₩AppData₩Local₩Programs₩Python₩Python37>pip install pandas
Collecting pandas
  Downloading https://files.pythonhosted.org/packages/02/d0/1e8e60e61e748338e3a40e42f5dfeee63ccdecfc4f0894122b890bfb009a/pandas-0.25.3-cp37-cp37m-win_amd64.whl (9.2MB)
     |                                                | 9.2MB 6.4MB/s
Requirement already satisfied: python-dateutil>=2.6.1 in c:₩users₩ksh₩appdata₩local₩programs₩python₩python37₩lib₩site-packages (from pandas) (2.8.0)
Collecting pytz>=2017.2 (from pandas)
  Downloading https://files.pythonhosted.org/packages/e7/f9/f0b53f88060247251bf481fa6ea62cd0d25bf1b11a87888e53ce5b7c8ad2/pytz-2019.3-py2.py3-none-any.whl (509kB)
     |                                                | 512kB ...
Requirement already satisfied: numpy>=1.13.3 in c:₩users₩ksh₩appdata₩local₩programs₩python₩python37₩lib₩site-packages (from pandas) (1.17.2)
Requirement already satisfied:                                                                                  eutil>=2.6.1->pandas) (1.12.0)
Installing collected packages:
Successfully installed pandas-
WARNING: You are using pip vers
You should consider upgrading
```

```python
import requests
from bs4 import BeautifulSoup
import pandas as pd

a = []

for num in range(1, 16):
    url = requests.get("https://www.rottentomatoes.com/m/frozen_ii/reviews?type=&sort=&page="+ str(num))
    soup = BeautifulSoup(url.text, "html.parser")

    all_review = soup.findAll("div", attrs={"class": "row review_table_row"})

    for line in all_review:
        reviewer = line.find("a", attrs={"class": "unstyled bold articleLink"}).get_text()
        review = line.find("div", attrs={"class": "the_review"}).get_text()
        review_point_original = line.find("div", attrs={"class": "small subtle review-link"}).get_text()
        review_dt = line.find("div", attrs={"class": "review-date subtle small"}).get_text()
        print(review_dt)
        b = []
        b.insert(0, reviewer)
        b.insert(1, review)
        b.insert(2, review_point_original)
        b.insert(3, review_dt)

        a.append(b)

data = pd.DataFrame(a)
data.to_csv('frozen_ii.csv')
```

02. 크롤링> 저장된 csv 파일

	A	B	C	D
1	REVIEWER	REVIEW	SCORE	REVIEW_DT
2	Edwin Arnaudin	A waste of beautiful animation.	Full Review | Original Score: 2/5	December 10, 2019
3	Sameen Amer	The storyline overall feels forced and clunky.	Full Review | Original Score: 2.5/5	December 9, 2019
4	Stephen Romei	The songs, perhaps not as compelling as in the original, still make the heart beat faster now and then.	Full Review | Original Score: 3/5	December 9, 2019

frozen_ii.csv

03. 데이터 가공/정제> 불러오기

생성한 csv 읽어오기

In[1]:

```
import pandas as pd
data = pd.read_csv("../input/frozen_ii.csv", encoding='ISO-8859-1')
```

+ Code + Markdown

In[2]:

```
data.head()
```

Out[2]:

	REVIEWER	REVIEW	SCORE	REVIEW_DT
0	Edwin Arnaudin	\n A waste ...	\nFull Review\n ...	\n December 10,...
1	Sameen Amer	\n The stor...	\nFull Review\n ...	\n December 9, ...
2	Stephen Romei	\n The song...	\nFull Review\n ...	\n December 9, ...
3	Sarah Gopaul	\n It's all...	\nFull Review\n	\n December 9, ...
4	Josh Larsen	\n ...a tor...	\nFull Review\n ...	\n December 9, ...

03. 데이터 가공/정제> 데이터 전처리 – text 리뷰

text 리뷰 데이터 전처리 – 영어 텍스트 전처리

In[3]:

```
import re
from bs4 import BeautifulSoup
from nltk.corpus import stopwords

stop_words = set(stopwords.words('english')) # 영어 불용어들의 set을 만든다.
```

In[4]:

```
def preprocessing_review(review):
    # 불용어 제거는 옵션으로 선택 가능하다.

    # 1. HTML 태그 제거
    review_text = BeautifulSoup(review, "html5lib").get_text()

    # 2. 영어가 아닌 특수문자들을 공백(" ")으로 바꾸기
    review_text = re.sub("[^a-zA-Z]", " ", review_text)

    # 3. 대문자들을 소문자로 바꾸고 공백단위로 텍스트들 나눠서 리스트로 만든다.
    words = review_text.lower().split()

    # 4. 불용어들을 제거

    #영어에 관련된 불용어 불러오기
    stops = set(stopwords.words("english"))
    # 불용어가 아닌 단어들로 이루어진 새로운 리스트 생성
    words = [w for w in words if not w in stops]
    # 5. 단어 리스트를 공백을 넣어서 하나의 글로 합친다.
    clean_review = ' '.join(words)

    return clean_review
```

03. 데이터 가공/정제> 데이터 전처리 – text 리뷰

text 리뷰 데이터 전처리 – 결과

In[5]:

```
clean_reviews = []

for review in data['REVIEW']:
    clean_reviews.append(preprocessing_review(review))
```

	REVIEWER	REVIEW	SCORE	REVIEW_DT
0	Edwin Arnaudin	\n A waste ...	\nFull Review\n ...	\n December 10,...
1	Sameen Amer	\n The stor...	\nFull Review\n ...	\n December 9, ...
2	Stephen Romei	\n The song...	\nFull Review\n ...	\n December 9, ...
3	Sarah Gopaul	\n It's all...	\nFull Review\n	\n December 9, ...
4	Josh Larsen	\n ...a tor...	\nFull Review\n ...	\n December 9, ...

	REVIEWER	REVIEW	SCORE	REVIEW_DT
0	Edwin Arnaudin	waste beautiful animation	4.0	2019-12-10
1	Sameen Amer	storyline overall feels forced clunky	NaN	2019-12-09
2	Stephen Romei	songs perhaps compelling original still make h...	6.0	2019-12-09
3	Sarah Gopaul	still fantastical likely falls slightly short ...	NaN	2019-12-09
4	Josh Larsen	torturously convoluted extension already compl...	5.0	2019-12-09

03. 데이터 가공/정제> 데이터 전처리 – 숫자 평점

숫자 평점의 문제점(3)

- 1) 숫자 평점이 존재하지 않는 리뷰가 존재함

Jocelyn Noveck
Associated Press
Top Critic

November 14, 2019

If it all seems less effortless, more workmanlike than the first film, with a very complex storyline that will definitely be harder to follow for younger fans, there's plenty to like, especially the lush visuals.

Full Review | Original Score: 2.5/4

Kristen Lopez
FanSided

November 14, 2019

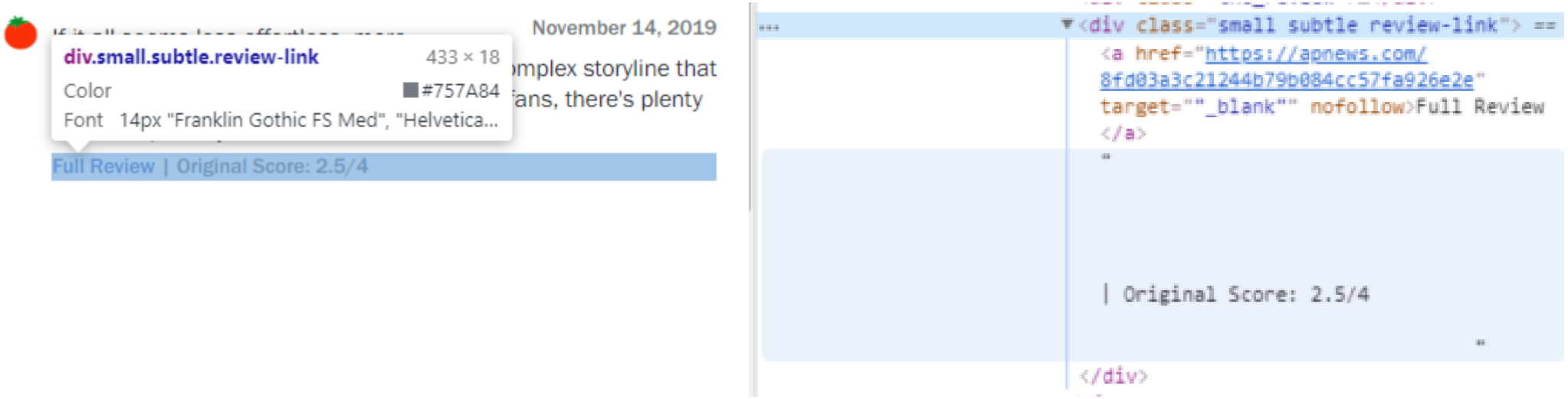
Frozen 2 never rises above mediocre, answering questions that never needed to be asked and creating a burdensome narrative to tie everything together.

Full Review

03. 데이터 가공/정제> 데이터 전처리 – 숫자 평점

숫자 평점의 문제점(3)

- 2) 숫자 평점만을 특정하는 요소가 존재하지 않아
  부가적인 정보가 포함됨

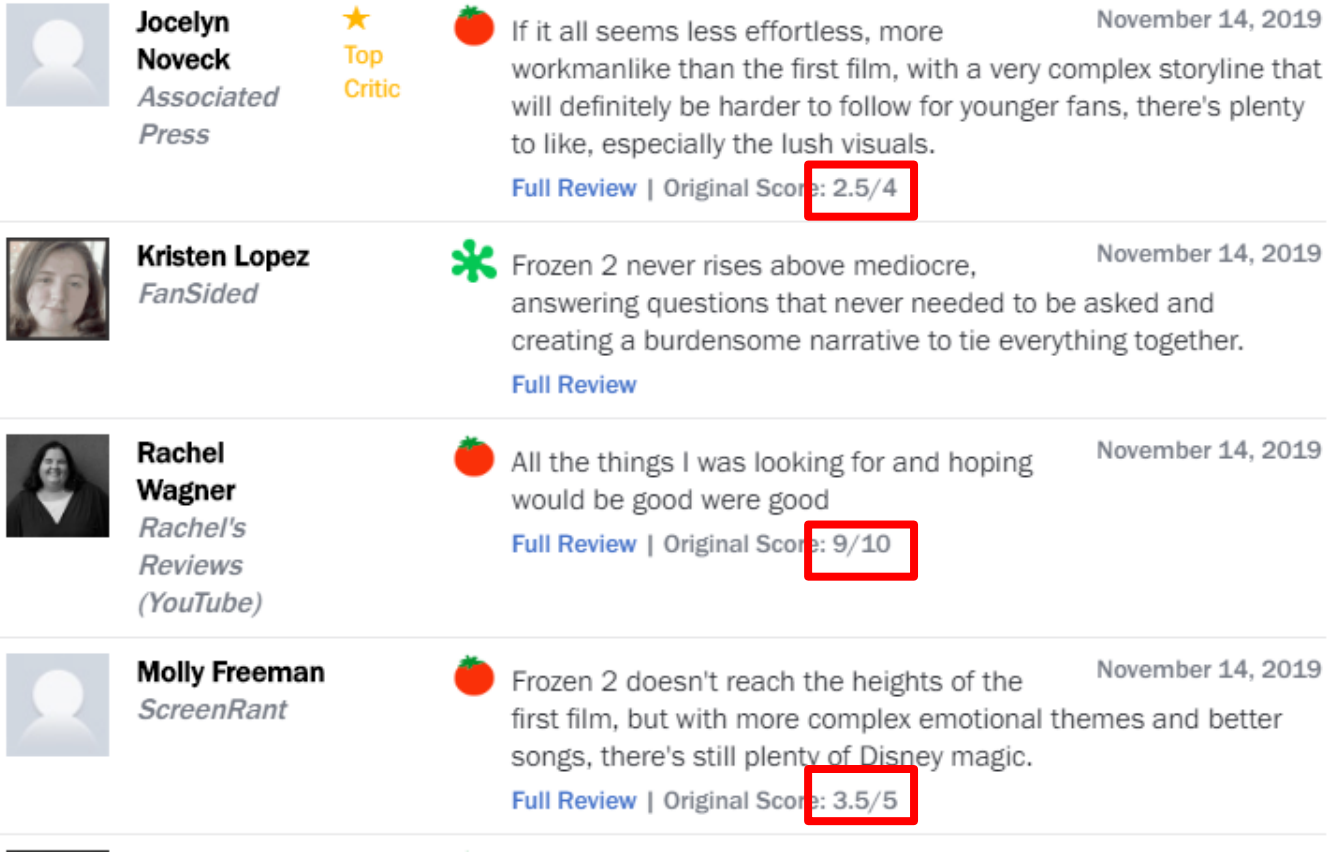


03. 데이터 가공/정제> 데이터 전처리 – 숫자 평점

숫자 평점의 문제점(3)

- 3) 숫자 평점의 총점의 유형이 다양함

숫자유형					문자유형 (특수문자 포함)	
분수유형				정수유형		
2/4,	1/5,	4/10,	84/100,	5,	A,	-,
2.5/4,	1.5/5,	5/10,		10	A-,	
3/4,	2/5,	6/10,			B+,	
3.5/4,	2.5/5,	6.5/10,			B	
	3/5,	6.85/10,			B,	
	3.5/5,	7/10,			B-,	
	4/5,	8/10,			B-minus,	
	4.5/5,	8.5/10,			C+,	
	5/5,	9/10,			C,	
					C-	



03. 데이터 가공/정제> 데이터 전처리 – 숫자 평점

In[6]:

```
import numpy as np

def preprocessing_score(score):

    if score.find(":") == -1:
        review_text = np.nan
    else:
        review_text = score[score.find(":")+1:].replace("\n","").str

        if review_text.find("/") == 1:

            """
            1. 숫자 분수 유형 - 10진수로 변환
              1-1. x/4   -> X 2.5
              1-2. x/5   -> X 2
              1-3. x/10  -> x
              1-4. x/100 -> / 10
            """

            den = float(review_text[review_text.find("/")+1:])
            num = float(review_text[:review_text.find("/")])
            if den == 4.0:
                review_text = num * 2.5
            elif den == 5.0:
                review_text = num * 2
            elif den == 10.0:
                review_text = num
            elif den == 100.0:
                review_text = num / 2
        else:

            """
            2. 문자유형
            """
            if review_text[:len("A")] == "A": # 'A': 9, 'A-': 9
                review_text = 9
            elif review_text[:len("B")] == "B": # 'B+': '8', 'B': '8', 'B-': '8', 'B-minus': '8'
                review_text = 8
            elif review_text[:len("C")] == "C": # 'C+': '7', 'C': '7', 'C-': '7'
                review_text = 7
            else: # '-'
                review_text = np.nan

return review_text
```

03. 데이터 가공/정제> 데이터 전처리

숫자 평점 & 리뷰 날짜 데이터 전처리 – 결과

In[7]:

```
clean_scores = []

for score in data['SCORE']:
    clean_scores.append(preprocessing_score(score))
```

```
import datetime
clean_review_dt = pd.to_datetime(data['REVIEW_DT'].str.strip())
type(clean_review_dt)
```

```
pandas.core.series.Series
```

	REVIEWER	REVIEW	SCORE	REVIEW_DT
0	Edwin Arnaudin	\n A waste ...	\nFull Review\n ...	\n December 10,...
1	Sameen Amer	\n The stor...	\nFull Review\n ...	\n December 9, ...
2	Stephen Romei	\n The song...	\nFull Review\n ...	\n December 9, ...
3	Sarah Gopaul	\n It's all...	\nFull Review\n	\n December 9, ...
4	Josh Larsen	\n ...a tor...	\nFull Review\n ...	\n December 9, ...

	REVIEWER	REVIEW	SCORE	REVIEW_DT
0	Edwin Arnaudin	waste beautiful animation	4.0	2019-12-10
1	Sameen Amer	storyline overall feels forced clunky	NaN	2019-12-09
2	Stephen Romei	songs perhaps compelling original still make h...	6.0	2019-12-09
3	Sarah Gopaul	still fantastical likely falls slightly short ...	NaN	2019-12-09
4	Josh Larsen	torturously convoluted extension already compl...	5.0	2019-12-09

04. 데이터 활용> 데이터 분석을 위한 전처리

결측치 처리

```
clean_data['SCORE'].dropna().mean()
```

Out[23]:

```
6.996688741721854
```

In[24]:

```
clean_data.loc[clean_data.SCORE.isnull(), 'SCORE'] = clean_data['SCORE'].dropna().mean()
```

+ Code + Markdown

In[25]:

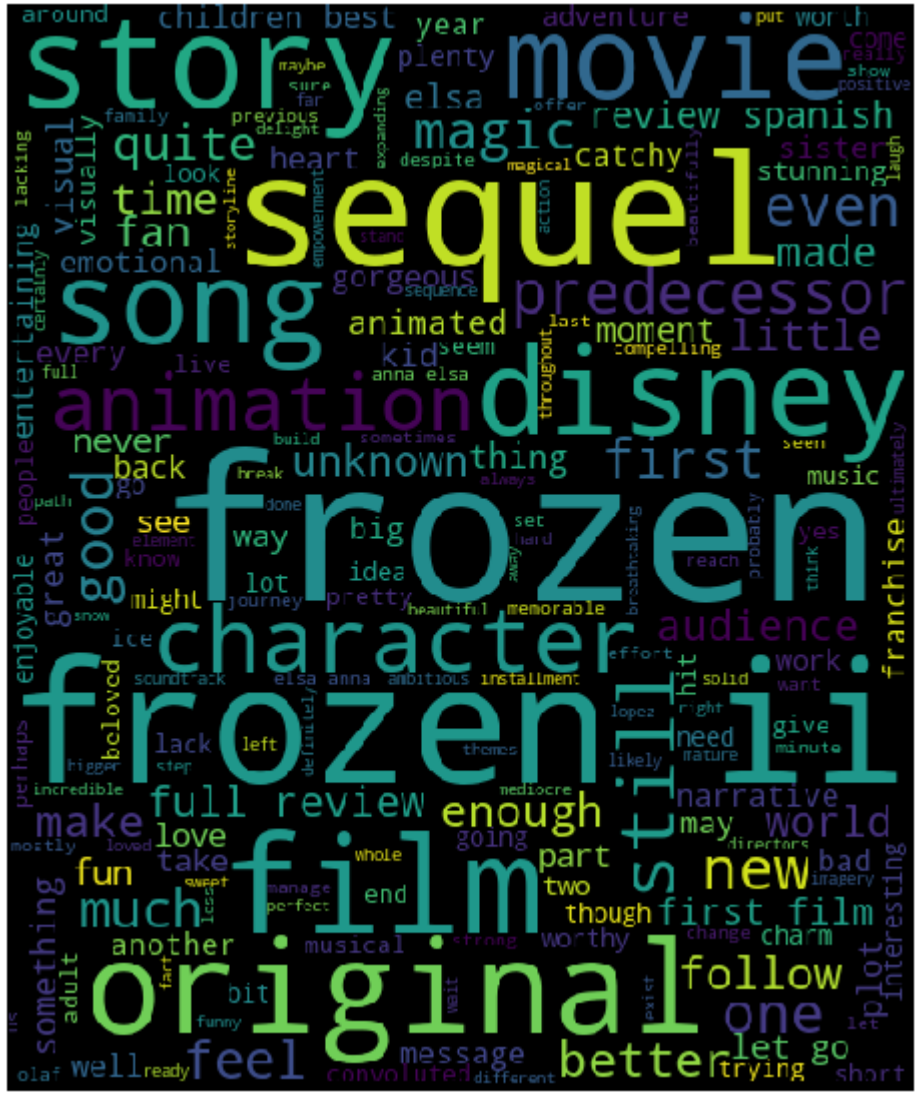
```
clean_data.head()
```

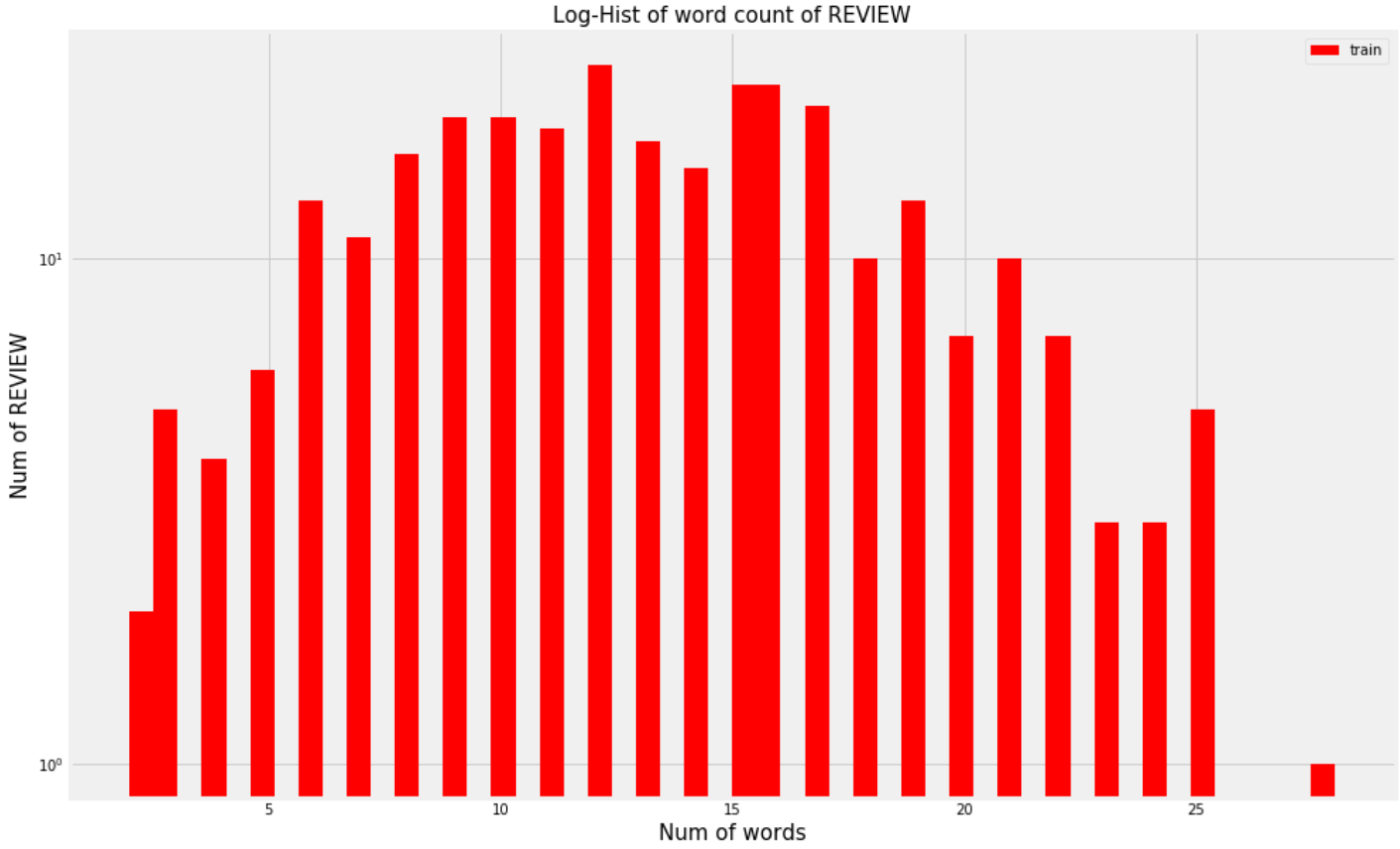
Out[25]:

	REVIEWER	REVIEW	SCORE	REVIEW_DT
0	Edwin Arnaudin	waste beautiful animation	4.000000	2019-12-10
1	Sameen Amer	storyline overall feels forced clunky	6.996689	2019-12-09
2	Stephen Romei	songs perhaps compelling original still make h...	6.000000	2019-12-09
3	Sarah Gopaul	still fantastical likely falls slightly short ...	6.996689	2019-12-09
4	Josh Larsen	torturously convoluted extension already compl...	5.000000	2019-12-09

	REVIEWER	REVIEW	SCORE	REVIEW_DT
0	Edwin Arnaudin	waste beautiful animation	4.0	2019-12-10
1	Sameen Amer	storyline overall feels forced clunky	NaN	2019-12-09
2	Stephen Romei	songs perhaps compelling original still make h...	6.0	2019-12-09
3	Sarah Gopaul	still fantastical likely falls slightly short ...	NaN	2019-12-09
4	Josh Larsen	torturously convoluted extension already compl...	5.0	2019-12-09

04. 데이터 활용> 가시화


story
movie
sequel
song
frozen
frozen ii
animation
disney
character
film
original
predecessor
magic
review
spanish
quite
time
fan
even
made
little
moment
animated
unknown
thing
first
music
franchise
audience
work
still
new
full review
enough
narrative
world
make
love
take
fun
much
another
musical
follow
one
better
feel
message
let go
trying
good
great
enjoyable
beloved
never
back
see
way
big
idea
lot
pretty
memorable
lack
left
need
give
part
two
though
first film
charm
bad
year
plenty
elsa
children
best
adventure
heart
catchy
stunning
sister
emotional
gorgeous
every
kid
anna elsa
visual
visually
entertaining
people
enjoyable
incredible
well
ready
convoluted
different
short
plot
interesting
something
adult
bit
funny
musical
worthy
change
manage
perfect
end
whole
swept
lack
installment
ambitious
effort
solid
wart
lopez
right
likely
nature
minute
directors
imagery
mediocre
going
themes
trying
let go
exist
wait
fart
strong
olaf
mostly
loved
perhaps
trigger
step
definitely
scundtrack
ice
snow
journey
beautifully
breathtaking
probably
think
reach
yes
set
hard
away
always
sometimes
build
break
done
pull
element
sequence
compelling
throughout
last
seem
live
full
look
previous
delight
far
sure
maybe
family
offer
expanding
despite
magical
action
empowerment
storyline
lacking
certainly
laugh
show
really
positive
come
put
worth
around


Log-Hist of word count of REVIEW
train
Num of REVIEW
Num of words
10^1
10^0
5
10
15
20
25

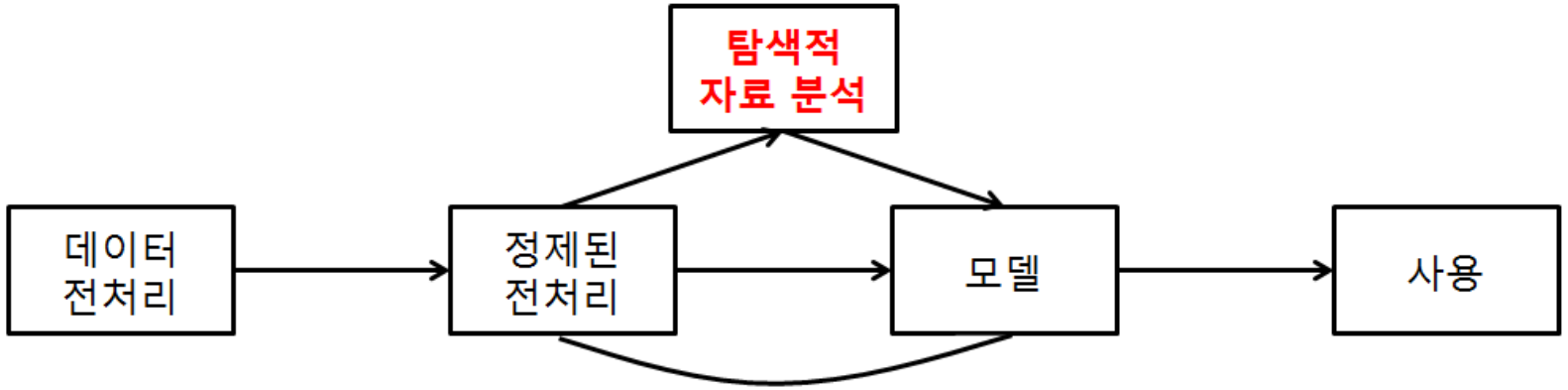
04. 데이터 활용> EDA

EDA탐색적 데이터 분석(EDA: Exploratory Data Analysis)

- 정해진 틀 없이 데이터에 대하여 최대한 많은 정보를 뽑아냄
- 평균값, 중앙값, 최솟값, 최댓값, 범위, 분포, 이상치 등

```
clean_data.describe()
```

	SCORE
count	292.000000
mean	6.996689
std	1.074938
min	2.000000
25%	6.996689
50%	6.996689
75%	7.500000
max	10.000000



04. 데이터 활용> EDA

```
for col in clean_data.columns:
    msg = 'column: {:>10}\t Percent of NaN value: {:.2f}%'.format(col, 100 * (clean_data[col].isnull().sum() / clean_data[col].shape[0]))
    print(msg)
```

```
column:   REVIEWER       Percent of NaN value: 0.00%
column:     REVIEW       Percent of NaN value: 0.00%
column:      SCORE       Percent of NaN value: 48.29%
column:  REVIEW_DT       Percent of NaN value: 0.00%
```

```
print('리뷰 단어 개수 최대 값: {}'.format(np.max(word_counts)))
print('리뷰 단어 개수 최소 값: {}'.format(np.min(word_counts)))
print('리뷰 단어 개수 평균 값: {:.2f}'.format(np.mean(word_counts)))
print('리뷰 단어 개수 표준편차: {:.2f}'.format(np.std(word_counts)))
print('리뷰 단어 개수 중간 값: {}'.format(np.median(word_counts)))
# 사분위의 대한 경우는 0~100 스케일로 되어있음
print('리뷰 단어 개수 제 1 사분위: {}'.format(np.percentile(word_counts, 25)))
print('리뷰 단어 개수 제 3 사분위: {}'.format(np.percentile(word_counts, 75)))
```

```
리뷰 단어 개수 최대 값: 28
리뷰 단어 개수 최소 값: 2
리뷰 단어 개수 평균 값: 13.26
리뷰 단어 개수 표준편차: 5.21
리뷰 단어 개수 중간 값: 13.0
리뷰 단어 개수 제 1 사분위: 9.0
리뷰 단어 개수 제 3 사분위: 17.0
```

```
print('평점 최대 값: {}'.format(np.max(clean_data['SCORE'])))
print('평점 최소 값: {}'.format(np.min(clean_data['SCORE'])))
print('평점 평균 값: {:.2f}'.format(np.mean(clean_data['SCORE'])))
print('평점 표준편차: {:.2f}'.format(np.std(clean_data['SCORE'])))
print('평점 중간 값: {}'.format(np.median(clean_data['SCORE'])))
# 사분위의 대한 경우는 0~100 스케일로 되어있음
print('평점 제 1 사분위: {}'.format(np.percentile(clean_data['SCORE'], 25)))
print('평점 제 3 사분위: {}'.format(np.percentile(clean_data['SCORE'], 75)))
```

```
평점 최대 값: 10.0
평점 최소 값: 2.0
평점 평균 값: 7.00
평점 표준편차: 1.07
평점 중간 값: 6.996688741721854
평점 제 1 사분위: 6.996688741721854
평점 제 3 사분위: 7.5
```

Q & A