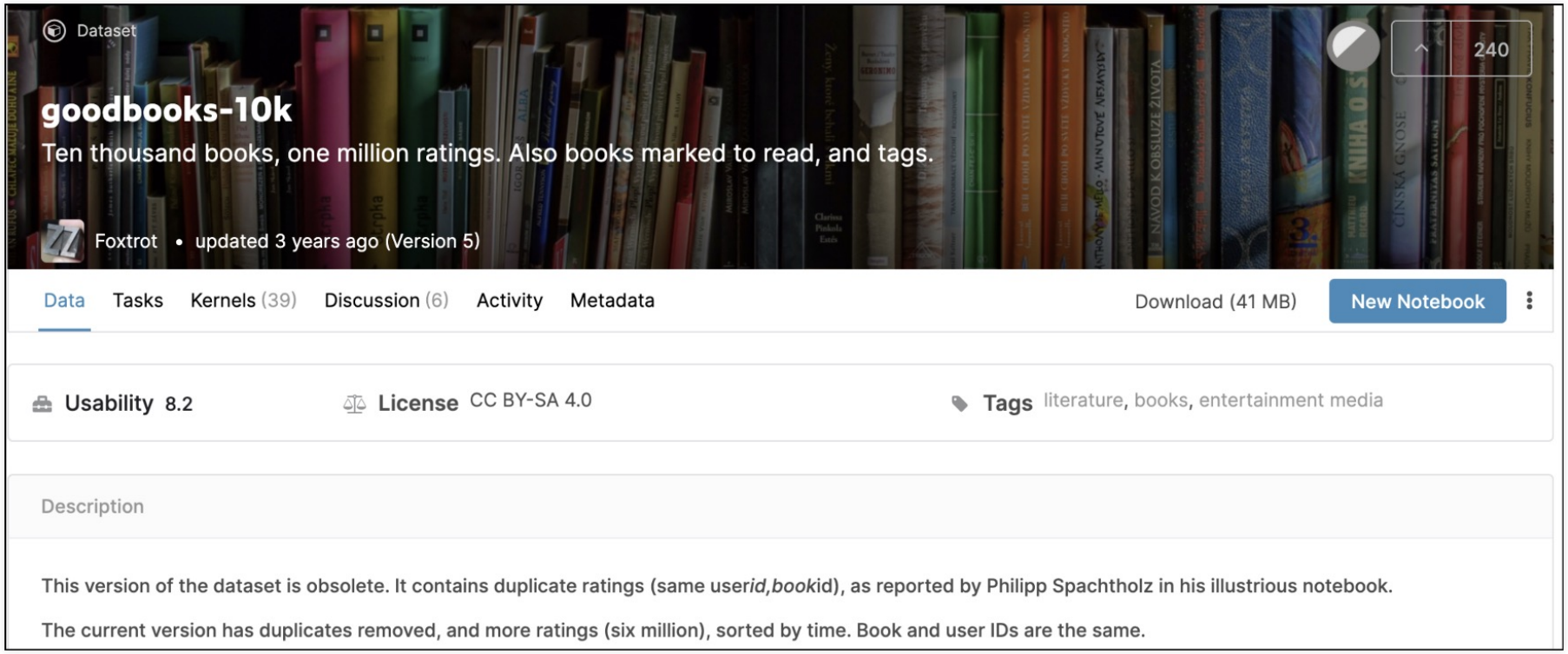
Chapter 33. Good Books recommendations

Good Books

zero-base /

데이터



- https://www.kaggle.com/zygmunt/goodbooks-10k

zero-base /

데이터 내용

books.csv (3.14 MB) — 20 of 23 columns — Views

	isbn13	authors	original_publication	original_title	title	language_code	avera
7% / 0% / 93%	195m – 9790b	4664 unique values	-1.75k – 2.02k	[null] 6% / [empty] 0% / Other (9273) 94%	9964 unique values	eng 63% / en-US 21% / Other (24) 16%	2.47
	9.78043902348e+12	Suzanne Collins	2008.0	The Hunger Games	The Hunger Games (The Hunger Games, #1)	eng	
	9.78043955493e+12	J.K. Rowling, Mary GrandPré	1997.0	Harry Potter and the Philosopher's Stone	Harry Potter and the Sorcerer's Stone (Harry Potter, #1)	eng	
	9.78031601584e+12	Stephenie Meyer	2005.0	Twilight	Twilight (Twilight, #1)	en-US	
	9.78006112008e+12	Harper Lee	1960.0	To Kill a Mockingbird	To Kill a Mockingbird	eng	
	9.78074327356e+12	F. Scott Fitzgerald	1925.0	The Great Gatsby	The Great Gatsby	eng	
	9.78052547881e+12	John Green	2012.0	The Fault in Our Stars	The Fault in Our Stars	eng	

zero-base /

데이터 경로 확인

```
import numpy as np
import pandas as pd

import os
print(os.listdir("./goodbooks-10k/"))
```

```
['book_tags.csv', 'sample_book.xml', 'tags.csv', 'ratings.csv', 'to_read.csv',
'books.csv']
```

zero-base /

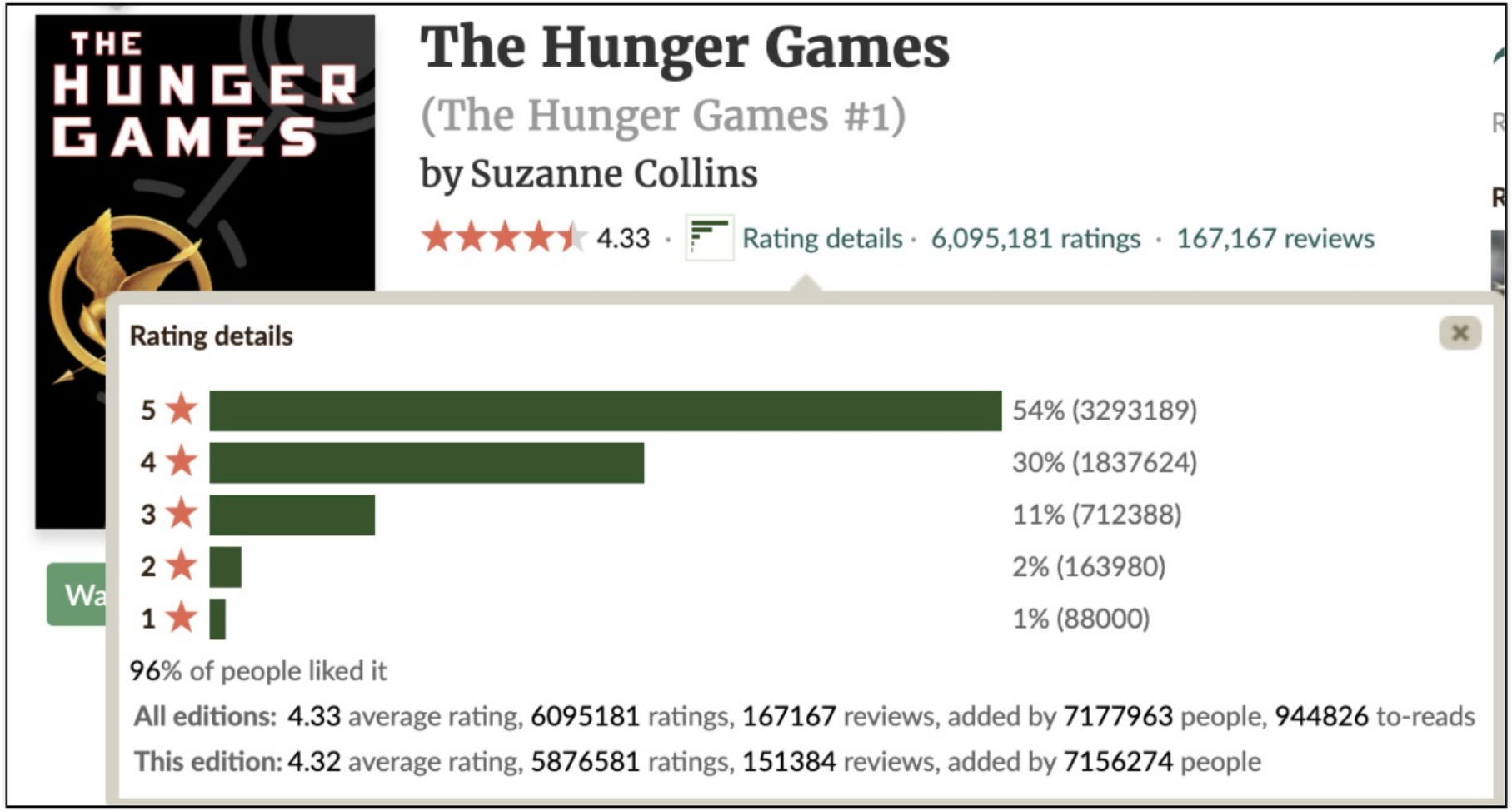
books.csv

```
books = pd.read_csv('./goodbooks-10k/books.csv', encoding = "ISO-8859-1")
books.head()
```

iews_count	ratings_1	ratings_2	ratings_3	ratings_4	ratings_5	image_url	small_image_url
	66715	127936	560092	1481305	2706317	https://images.gr-assets.com/books/1447303603m...	https://images.gr-assets.com/books/1447303603s...
	75504	101676	455024	1156318	3011543	https://images.gr-assets.com/books/1474154022m...	https://images.gr-assets.com/books/1474154022s...
	456191	436802	793319	875073	1355439	https://images.gr-assets.com/books/1361039443m...	https://images.gr-assets.com/books/1361039443s...
	60427	117415	446835	1001952	1714267	https://images.gr-assets.com/books/1361975680m...	https://images.gr-assets.com/books/1361975680s...
	86236	197621	606158	936012	947718	https://images.gr-assets.com/books/1490528560m...	https://images.gr-assets.com/books/1490528560s...

zero-base /

ratings 1,2,3,4,5의 의미



zero-base /

ratings.csv

```
ratings = pd.read_csv('./goodbooks-10k/ratings.csv', encoding = "ISO-8859-1")
ratings.head()
```

	book_id	user_id	rating
0	1	314	5
1	1	439	3
2	1	588	5
3	1	1169	4
4	1	1185	4

zero-base /

book_tags.csv

```
book_tags = pd.read_csv('./goodbooks-10k/book_tags.csv', encoding = "ISO-8859-1")
book_tags.head()
```

	goodreads_book_id	tag_id	count
0	1	30574	167697
1	1	11305	37174
2	1	11557	34173
3	1	8717	12986
4	1	33114	12716

zero-base /

tags.csv

```
tags = pd.read_csv('./goodbooks-10k/tags.csv')
tags.tail()
```

	tag_id	tag_name
34247	34247	Childrens
34248	34248	Favorites
34249	34249	Manga
34250	34250	SERIES
34251	34251	favourites

zero-base /

book_tags와 tags를 merge

```
tags_join_DF = pd.merge(book_tags, tags, left_on='tag_id',
                        right_on='tag_id', how='inner')
tags_join_DF.head()
```

	goodreads_book_id	tag_id	count	tag_name
0	1	30574	167697	to-read
1	2	30574	24549	to-read
2	3	30574	496107	to-read
3	5	30574	11909	to-read
4	6	30574	298	to-read

zero-base /

to_read.csv

```
to_read = pd.read_csv('./goodbooks-10k/to_read.csv')
to_read.head()
```

	user_id	book_id
0	1	112
1	1	235
2	1	533
3	1	1198
4	1	1874

zero-base /

books의 aurthors

```
books['authors'][:5]
```

```
0                 Suzanne Collins
1    J.K. Rowling, Mary GrandPrÃ©
2                 Stephenie Meyer
3                      Harper Lee
4             F. Scott Fitzgerald
Name: authors, dtype: object
```

zero-base /

authors로 Tfidf 수행

```
from sklearn.feature_extraction.text import TfidfVectorizer

tf = TfidfVectorizer(analyzer='word', ngram_range=(1, 2),
                     min_df=0, stop_words='english')
tfidf_matrix = tf.fit_transform(books['authors'])
tfidf_matrix
```

zero-base /

유사도 측정

```
from sklearn.metrics.pairwise import linear_kernel

cosine_sim = linear_kernel(tfidf_matrix, tfidf_matrix)
cosine_sim
```

```
array([[1., 0., 0., ..., 0., 0., 0.],
       [0., 1., 0., ..., 0., 0., 0.],
       [0., 0., 1., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 1., 0., 0.],
       [0., 0., 0., ..., 0., 1., 0.],
       [0., 0., 0., ..., 0., 0., 1.]])
```

zero-base /

The Hobbit의 index는 6

```
titles = books['title']
indices = pd.Series(books.index, index=books['title'])
indices['The Hobbit']
```

```
6
```

zero-base /

유사도 값 호출

```
cosine_sim[indices['The Hobbit']]
```

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

zero-base /

유사도 결과를 인덱스를 가진 list 형으로

```
cosine_sim[indices['The Hobbit']].shape
```

```
(10000,)
```

```
list(enumerate(cosine_sim[indices['The Hobbit']]))[:3]
```

```
[(0, 0.0), (1, 0.0), (2, 0.0)]
```

zero-base /

가장 유사한 책의 인덱스 찾기

```
sim_scores = list(enumerate(cosine_sim[indices['The Hobbit']]))
sim_scores = sorted(sim_scores, key=lambda x: x[1], reverse=True)
sim_scores[:3]
```

```
[(6, 1.0), (18, 1.0), (154, 1.0)]
```

zero-base /

작가로 본 유사 책 검색

```
sim_scores = sim_scores[1:11]
book_indices = [i[0] for i in sim_scores]
titles.iloc[book_indices]
```

```
154            The Two Towers (The Lord of the Rings, #2)
160     The Return of the King (The Lord of the Rings,...
188     The Lord of the Rings (The Lord of the Rings, ...
963     J.R.R. Tolkien 4-Book Boxed Set: The Hobbit an...
4975        Unfinished Tales of NÃºmenor and Middle-Earth
2308                              The Children of HÃºrin
610               The Silmarillion (Middle-Earth Universe)
8271                   The Complete Guide to Middle-Earth
1128     The History of the Hobbit, Part One: Mr. Baggins
465                              The Hobbit: Graphic Novel
Name: title, dtype: object
```

zero-base /

book에 tag 포함

```
books_with_tags = pd.merge(books, tags_join_DF, left_on='book_id',
                           right_on='goodreads_book_id', how='inner')
books_with_tags.head()
```

tings_4	ratings_5	image_url	small_image_url	goodreads_book_id	tag_id	count	tag_name
31305	2706317	https://images.gr-assets.com/books/1447303603m...	https://images.gr-assets.com/books/1447303603s...	2767052	30574	11314	to-read
31305	2706317	https://images.gr-assets.com/books/1447303603m...	https://images.gr-assets.com/books/1447303603s...	2767052	11305	10836	fantasy
31305	2706317	https://images.gr-assets.com/books/1447303603m...	https://images.gr-assets.com/books/1447303603s...	2767052	11557	50755	favorites
31305	2706317	https://images.gr-assets.com/books/1447303603m...	https://images.gr-assets.com/books/1447303603s...	2767052	8717	35418	currently-reading
31305	2706317	https://images.gr-assets.com/books/1447303603m...	https://images.gr-assets.com/books/1447303603s...	2767052	33114	25968	young-adult

zero-base /

이번에는 tag로 Tfidf

```
tf1 = TfidfVectorizer(analyzer='word', ngram_range=(1, 2),
                      min_df=0, stop_words='english')
tfidf_matrix1 = tf1.fit_transform(books_with_tags['tag_name'].head(10000))
cosine_sim1 = linear_kernel(tfidf_matrix1, tfidf_matrix1)
```

zero-base /

추천책을 반환하는 함수

```
titles1 = books['title']
indices1 = pd.Series(books.index, index=books['title'])

def tags_recommendations(title):
    idx = indices1[title]
    sim_scores = list(enumerate(cosine_sim1[idx]))
    sim_scores = sorted(sim_scores, key=lambda x: x[1], reverse=True)
    sim_scores = sim_scores[1:11]
    book_indices = [i[0] for i in sim_scores]
    return titles.iloc[book_indices]
```

zero-base /

태그로 찾아본 The Hobbits와 유사책

```
tags_recommendations('The Hobbit').head(20)
```

```
16              Catching Fire (The Hunger Games, #2)
31                                    Of Mice and Men
107    Confessions of a Shopaholic (Shopaholic, #1)
125                         Dune (Dune Chronicles #1)
149                                       The Red Tent
206          One for the Money (Stephanie Plum, #1)
214                                   Ready Player One
231              The Gunslinger (The Dark Tower, #1)
253          Shiver (The Wolves of Mercy Falls, #1)
313                           Inkheart (Inkworld, #1)
Name: title, dtype: object
```

zero-base /

임시로 book id 마다 tag를 붙이고

```
temp_df = books_with_tags.groupby('book_id')['tag_name'].apply(' '.join).reset_index()
temp_df.head()
```

	book_id	tag_name
0	1	to-read fantasy favorites currently-reading yo...
1	2	to-read fantasy favorites currently-reading yo...
2	3	to-read fantasy favorites currently-reading yo...
3	5	to-read fantasy favorites currently-reading yo...
4	6	to-read fantasy young-adult fiction harry-pott...

zero-base /

그걸 books에 합치고

```
books = pd.merge(books, temp_df, left_on='book_id', right_on='book_id', how='inner')
books.head()
```

ratings_1	ratings_2	ratings_3	ratings_4	ratings_5	image_url	small_image_url	tag_name
66715	127936	560092	1481305	2706317	https://images.gr-assets.com/books/1447303603m...	https://images.gr-assets.com/books/1447303603s...	to-read fantasy favorites currently-reading yo...
75504	101676	455024	1156318	3011543	https://images.gr-assets.com/books/1474154022m...	https://images.gr-assets.com/books/1474154022s...	to-read fantasy favorites currently-reading yo...
456191	436802	793319	875073	1355439	https://images.gr-assets.com/books/1361039443m...	https://images.gr-assets.com/books/1361039443s...	to-read fantasy favorites currently-reading yo...

zero-base /

저자이름과 태그를 합치고

```
books['corpus'] = (pd.Series(books[['authors', 'tag_name']]
                .fillna('')
                .values.tolist()
                ).str.join(' '))
books['corpus'][:3]
```

```
0    Suzanne Collins to-read fantasy favorites curr...
1    J.K. Rowling, Mary GrandPrÃ© to-read fantasy f...
2    Stephenie Meyer to-read fantasy favorites curr...
Name: corpus, dtype: object
```

zero-base /

Tfidf를 수행하고

```
tf_corpus = TfidfVectorizer(analyzer='word', ngram_range=(1, 2),
                            min_df=0, stop_words='english')
tfidf_matrix_corpus = tf_corpus.fit_transform(books['corpus'])
cosine_sim_corpus = linear_kernel(tfidf_matrix_corpus, tfidf_matrix_corpus)

titles = books['title']
indices = pd.Series(books.index, index=books['title'])
```

zero-base /

추천 함수를 만들고

```
def corpus_recommendations(title):
    idx = indices1[title]
    sim_scores = list(enumerate(cosine_sim_corpus[idx]))
    sim_scores = sorted(sim_scores, key=lambda x: x[1], reverse=True)
    sim_scores = sim_scores[1:11]
    book_indices = [i[0] for i in sim_scores]
    return titles.iloc[book_indices]
```

zero-base /

Hobbit과 비슷한 것은

```
corpus_recommendations("The Hobbit")
```

```
188     The Lord of the Rings (The Lord of the Rings, ...
154            The Two Towers (The Lord of the Rings, #2)
160     The Return of the King (The Lord of the Rings,...
18      The Fellowship of the Ring (The Lord of the Ri...
610              The Silmarillion (Middle-Earth Universe)
4975        Unfinished Tales of NÃºmenor and Middle-Earth
2308                               The Children of HÃºrin
963     J.R.R. Tolkien 4-Book Boxed Set: The Hobbit an...
465                              The Hobbit: Graphic Novel
8271                   The Complete Guide to Middle-Earth
Name: title, dtype: object
```

zero-base /

Twilight과 비슷한 것은

```
corpus_recommendations("Twilight (Twilight, #1)")
```

```
51                                  Eclipse (Twilight, #3)
48                                 New Moon (Twilight, #2)
991                      The Twilight Saga (Twilight, #1-4)
833                           Midnight Sun (Twilight, #1.5)
731     The Short Second Life of Bree Tanner: An Eclip...
1618    The Twilight Saga Complete Collection  (Twilig...
4087    The Twilight Saga: The Official Illustrated Gu...
2020              The Twilight Collection (Twilight, #1-3)
72                                  The Host (The Host, #1)
219     Twilight: The Complete Illustrated Movie Compa...
Name: title, dtype: object
```

zero-base /

Romeo와 Juliet과 유사한 것은

```
corpus_recommendations("Romeo and Juliet")
```

```
352                      Othello
769                Julius Caesar
124                       Hamlet
153                      Macbeth
247    A Midsummer Night's Dream
838       The Merchant of Venice
854                 Twelfth Night
529       Much Ado About Nothing
713                    King Lear
772      The Taming of the Shrew
Name: title, dtype: object
```