In [1]:

```python
import numpy as np
import pandas as pd
from ast import literal_eval

from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity
```

```
/Users/kunalshriwas/opt/anaconda3/lib/python3.8/site-packages/pandas/core/computation/expressions.py:20
: UserWarning: Pandas requires version '2.7.3' or newer of 'numexpr' (version '2.7.1' currently install
ed).
  from pandas.core.computation.check import NUMEXPR_INSTALLED
```

In [2]:

```python
credits_df = pd.read_csv("tmdb_5000_credits.csv")
movies_df = pd.read_csv("tmdb_5000_movies.csv")
```

In [3]:

```python
credits_df.head()
```

Out[3]:

| | movie_id | title | cast | crew |
|---|---|---|---|---|
| **0** | 19995 | Avatar | [{"cast_id": 242, "character": "Jake Sully", "... | [{"credit_id": "52fe48009251416c750aca23", "de... |
| **1** | 285 | Pirates of the Caribbean: At World's End | [{"cast_id": 4, "character": "Captain Jack Spa... | [{"credit_id": "52fe4232c3a36847f800b579", "de... |
| **2** | 206647 | Spectre | [{"cast_id": 1, "character": "James Bond", "cr... | [{"credit_id": "54805967c3a36829b5002c41", "de... |
| **3** | 49026 | The Dark Knight Rises | [{"cast_id": 2, "character": "Bruce Wayne / Ba... | [{"credit_id": "52fe4781c3a36847f81398c3", "de... |
| **4** | 49529 | John Carter | [{"cast_id": 5, "character": "John Carter", "c... | [{"credit_id": "52fe479ac3a36847f813eaa3", "de... |

In [4]:
```
movies_df.head()
```

Out[4]:

| genres | homepage | id | keywords | original_language | original_title | overview | popularity | production_comp |
|---|---|---|---|---|---|---|---|---|
| [{"id": 28, "name": "Action"}, {"id": 12, "nam... | http://www.avatarmovie.com/ | 19995 | [{"id": 1463, "name": "culture clash"}, {"id":... | en | Avatar | In the 22nd century, a paraplegic Marine is di... | 150.437577 | [{"name": "Ing Film Partners |
| [{"id": 12, "name": venture"}, d": 14, "... | http://disney.go.com/disneypictures/pirates/ | 285 | [{"id": 270, "name": "ocean"}, {"id": 726, "na... | en | Pirates of the Caribbean: At World's End | Captain Barbossa, long believed to be dead, ha... | 139.082615 | [{"name": "Walt I Pictures", "id": 2 |
| [{"id": 28, "name": "Action"}, {"id": 12, "nam... | http://www.sonypictures.com/movies/spectre/ | 206647 | [{"id": 470, "name": "spy"}, {"id": 818, "name... | en | Spectre | A cryptic message from Bond's past sends him o... | 107.376788 | [{"name": "Col Pictures", "i {" |
| [{"id": 28, "name": "Action"}, {"id": 80, "nam... | http://www.thedarkknightrises.com/ | 49026 | [{"id": 849, "name": "dc comics"}, {"id": 853,... | en | The Dark Knight Rises | Following the death of District Attorney Harve... | 112.312950 | [{"name": "Lege Pictures", "id" |
| [{"id": 28, "name": "Action"}, {"id": 12, "nam... | http://movies.disney.com/john-carter | 49529 | [{"id": 818, "name": "based on novel"}, {"id":... | en | John Carter | John Carter is a war-weary, former military ca... | 43.926995 | [{"name": "Walt I Pictures", "i |

In [5]:

```python
# extract only ID, TITLE, CAST, CREW ,and merge with ID

credits_df.columns = ['id','title','cast','crew']
movies_df= movies_df.merge(credits_df,on = 'id')
movies_df.head()
```

Out[5]:

| | budget | genres | homepage | id | keywords | original_language | original_title | overview | popular |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 237000000 | [{"id": 28, "name": "Action"}, {"id": 12, "nam... | http://www.avatarmovie.com/ | 19995 | [{"id": 1463, "name": "culture clash"}, {"id":... | en | Avatar | In the 22nd century, a paraplegic Marine is di... | 150.4375 |
| **1** | 300000000 | [{"id": 12, "name": "Adventure"}, {"id": 14, "... | http://disney.go.com/disneypictures/pirates/ | 285 | [{"id": 270, "name": "ocean"}, {"id": 726, "na... | en | Pirates of the Caribbean: At World's End | Captain Barbossa, long believed to be dead, ha... | 139.0826 |
| **2** | 245000000 | [{"id": 28, "name": "Action"}, {"id": 12, "nam... | http://www.sonypictures.com/movies/spectre/ | 206647 | [{"id": 470, "name": "spy"}, {"id": 818, "name... | en | Spectre | A cryptic message from Bond's past sends him o... | 107.3767 |
| **3** | 250000000 | [{"id": 28, "name": "Action"}, {"id": 80, "nam... | http://www.thedarkknightrises.com/ | 49026 | [{"id": 849, "name": "dc comics"}, {"id": 853,... | en | The Dark Knight Rises | Following the death of District Attorney Harve... | 112.3129 |
| **4** | 260000000 | [{"id": 28, "name": "Action"}, {"id": 12, "nam... | http://movies.disney.com/john-carter | 49529 | [{"id": 818, "name": "based on novel"}, {"id":... | en | John Carter | John Carter is a war-weary, former military | 43.9269 |

ca...

5 rows × 23 columns

In [6]:

```
movies_df.columns
```

Out[6]:

```
Index(['budget', 'genres', 'homepage', 'id', 'keywords', 'original_language',
       'original_title', 'overview', 'popularity', 'production_companies',
       'production_countries', 'release_date', 'revenue', 'runtime',
       'spoken_languages', 'status', 'tagline', 'title_x', 'vote_average',
       'vote_count', 'title_y', 'cast', 'crew'],
      dtype='object')
```

In [8]:

```
features = ['cast','crew','keywords','genres']

for feature in features:
    movies_df[feature]= movies_df[feature].apply(literal_eval)

movies_df[features].head()
```

Out[8]:

| | cast | crew | keywords | genres |
|---|---|---|---|---|
| **0** | [{'cast_id': 242, 'character': 'Jake Sully', '... | [{'credit_id': '52fe48009251416c750aca23', 'de... | [{'id': 1463, 'name': 'culture clash'}, {'id':... | [{'id': 28, 'name': 'Action'}, {'id': 12, 'nam... |
| **1** | [{'cast_id': 4, 'character': 'Captain Jack Spa... | [{'credit_id': '52fe4232c3a36847f800b579', 'de... | [{'id': 270, 'name': 'ocean'}, {'id': 726, 'na... | [{'id': 12, 'name': 'Adventure'}, {'id': 14, '... |
| **2** | [{'cast_id': 1, 'character': 'James Bond', 'cr... | [{'credit_id': '54805967c3a36829b5002c41', 'de... | [{'id': 470, 'name': 'spy'}, {'id': 818, 'name... | [{'id': 28, 'name': 'Action'}, {'id': 12, 'nam... |
| **3** | [{'cast_id': 2, 'character': 'Bruce Wayne / Ba... | [{'credit_id': '52fe4781c3a36847f81398c3', 'de... | [{'id': 849, 'name': 'dc comics'}, {'id': 853,... | [{'id': 28, 'name': 'Action'}, {'id': 80, 'nam... |
| **4** | [{'cast_id': 5, 'character': 'John Carter', 'c... | [{'credit_id': '52fe479ac3a36847f813eaa3', 'de... | [{'id': 818, 'name': 'based on novel'}, {'id':... | [{'id': 28, 'name': 'Action'}, {'id': 12, 'nam... |

In [22]:

```
#movies_df['crew'][0]
```

In [15]:

```
# create a function to extract director name

def get_director(x):
    for i in x:
        if i['job']=='Director':
            return i['name']
    return np.nan
```

In [16]:

```
def get_list(x):
    if isinstance(x,list):
        names = [i['name'] for i in x]

        if len(names)>3:
            names = names[:3]
        return names
    return []
```

In [18]:

```
# lets apply above both function on dataset

movies_df['director'] = movies_df['crew'].apply(get_director)
features = ['cast','keywords','genres']
for feature in features:
    movies_df[feature] = movies_df[feature].apply(get_list)
```

In [19]:

```
movies_df.columns
```

Out[19]:

```
Index(['budget', 'genres', 'homepage', 'id', 'keywords', 'original_language',
       'original_title', 'overview', 'popularity', 'production_companies',
       'production_countries', 'release_date', 'revenue', 'runtime',
       'spoken_languages', 'status', 'tagline', 'title_x', 'vote_average',
       'vote_count', 'title_y', 'cast', 'crew', 'director'],
      dtype='object')
```

In [21]:

```python
movies_df[['original_title','cast','director','keywords','genres']].head()
```

Out[21]:

| | original_title | cast | director | keywords | genres |
|---|---|---|---|---|---|
| **0** | Avatar | [Sam Worthington, Zoe Saldana, Sigourney Weaver] | James Cameron | [culture clash, future, space war] | [Action, Adventure, Fantasy] |
| **1** | Pirates of the Caribbean: At World's End | [Johnny Depp, Orlando Bloom, Keira Knightley] | Gore Verbinski | [ocean, drug abuse, exotic island] | [Adventure, Fantasy, Action] |
| **2** | Spectre | [Daniel Craig, Christoph Waltz, Léa Seydoux] | Sam Mendes | [spy, based on novel, secret agent] | [Action, Adventure, Crime] |
| **3** | The Dark Knight Rises | [Christian Bale, Michael Caine, Gary Oldman] | Christopher Nolan | [dc comics, crime fighter, terrorist] | [Action, Crime, Drama] |
| **4** | John Carter | [Taylor Kitsch, Lynn Collins, Samantha Morton] | Andrew Stanton | [based on novel, mars, medallion] | [Action, Adventure, Science Fiction] |

In [23]:

```python
def clean_data(row):
    if isinstance(row,list):
        return [str.lower(i.replace(" ","")) for i in row]
    else:
        if isinstance(row,str):
            return str.lower(row.replace(" ",""))
        else:
            return ""

features = ['cast','keywords','director','genres']

for feature in features:
    movies_df[feature]  = movies_df[feature].apply(clean_data)

```

In [24]:

```
movies_df[['cast','keywords','director','genres']].head()
```

Out[24]:

| | cast | keywords | director | genres |
|---|---|---|---|---|
| **0** | [samworthington, zoesaldana, sigourneyweaver] | [cultureclash, future, spacewar] | jamescameron | [action, adventure, fantasy] |
| **1** | [johnnydepp, orlandobloom, keiraknightley] | [ocean, drugabuse, exoticisland] | goreverbinski | [adventure, fantasy, action] |
| **2** | [danielcraig, christophwaltz, léaseydoux] | [spy, basedonnovel, secretagent] | sammendes | [action, adventure, crime] |
| **3** | [christianbale, michaelcaine, garyoldman] | [dccomics, crimefighter, terrorist] | christophernolan | [action, crime, drama] |
| **4** | [taylorkitsch, lynncollins, samanthamorton] | [basedonnovel, mars, medallion] | andrewstanton | [action, adventure, sciencefiction] |

In [25]:

```
def create_group(features):
    return ' '.join(features['keywords'])+ ' '+' '.join(features['cast'])+ ' ' +' '.join(features['c


movies_df['group'] = movies_df.apply(create_group,axis = 1)

```

```
0    cultureclash future spacewar samworthington zo...
1    ocean drugabuse exoticisland johnnydepp orland...
2    spy basedonnovel secretagent danielcraig chris...
3    dccomics crimefighter terrorist christianbale ...
4    basedonnovel mars medallion taylorkitsch lynnc...
Name: group, dtype: object
```

In [26]:

```
print(movies_df['group'].head(10))
```

```
0    cultureclash future spacewar samworthington zo...
1    ocean drugabuse exoticisland johnnydepp orland...
2    spy basedonnovel secretagent danielcraig chris...
3    dccomics crimefighter terrorist christianbale ...
4    basedonnovel mars medallion taylorkitsch lynnc...
5    dualidentity amnesia sandstorm tobeymaguire ki...
6    hostage magic horse zacharylevi mandymoore don...
7    marvelcomic sequel superhero robertdowneyjr. c...
8    witch magic broom danielradcliffe rupertgrint ...
9    dccomics vigilante superhero benaffleck henryc...
Name: group, dtype: object
```

In [27]:

```
count_vect = CountVectorizer(stop_words='english')
count_matrix = count_vect.fit_transform(movies_df['group'])
print(count_matrix.shape)
```

```
(4803, 9290)
```

In [36]:

```
cosine_sim = cosine_similarity(count_matrix, count_matrix)
cosine_sim.shape
```

Out[36]:

```
(4803, 4803)
```

In [38]:

```
movies_df = movies_df.reset_index()
indices = pd.Series(movies_df.index , index = movies_df['original_title'])
```

In [39]:

```
indices.head()
```

Out[39]:

```
original_title
Avatar                                      0
Pirates of the Caribbean: At World's End    1
Spectre                                     2
The Dark Knight Rises                       3
John Carter                                 4
dtype: int64
```

In [46]:

```
def get_recommendation(title,cosine_sim = cosine_sim):
    idx = indices[title]
    similarity_score = list(enumerate(cosine_sim[idx]))
    similarity_score = sorted(similarity_score,key = lambda x : x[1],reverse=True)
    similarity_score = similarity_score[1:11]
    movies_indices = [ind[0] for ind in similarity_score]
    movies = movies_df['original_title'].iloc[movies_indices]
    return movies
```

In [47]:

```python
print(get_recommendation('The Dark Knight Rises'),cosine_sim)
```

```
65                  The Dark Knight
119                   Batman Begins
4638    Amidst the Devil's Wings
3073             Romeo Is Bleeding
1986                        Faster
3326                Black November
1503                        Takers
303                       Catwoman
747                 Gangster Squad
1149               American Hustle
Name: original_title, dtype: object [[1.         0.33333333 0.22222222 ... 0.         0.         0.
]
 [0.33333333 1.         0.22222222 ... 0.         0.         0.        ]
 [0.22222222 0.22222222 1.         ... 0.         0.         0.        ]
 ...
 [0.         0.         0.         ... 1.         0.         0.        ]
 [0.         0.         0.         ... 0.         1.         0.        ]
 [0.         0.         0.         ... 0.         0.         1.        ]]
```

In [48]:

```
print(get_recommendation('The Avengers'),cosine_sim)
```

```
7                   Avengers: Age of Ultron
26               Captain America: Civil War
79                               Iron Man 2
169     Captain America: The First Avenger
174                     The Incredible Hulk
85      Captain America: The Winter Soldier
31                               Iron Man 3
33                    X-Men: The Last Stand
68                                 Iron Man
94                  Guardians of the Galaxy
Name: original_title, dtype: object [[1.         0.33333333 0.22222222 ... 0.         0.         0.
]
 [0.33333333 1.         0.22222222 ... 0.         0.         0.        ]
 [0.22222222 0.22222222 1.         ... 0.         0.         0.        ]
 ...
 [0.         0.         0.         ... 1.         0.         0.        ]
 [0.         0.         0.         ... 0.         1.         0.        ]
 [0.         0.         0.         ... 0.         0.         1.        ]]
```

In [ ]:

```

```