In [1]:
```python
import numpy as np
import pandas as pd
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

In [2]:
```python
movies = pd.read_csv('tmdb_5000_movies.csv')
credits = pd.read_csv('tmdb_5000_credits.csv')
```

In [3]:
```python
movies.shape
```

Out[3]:
```
(4803, 20)
```

In [4]:
```python
credits.head()
```

Out[4]:

	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...

In [5]:
```python
movies = movies.merge(credits,on='title')
```

In [6]:
```python
movies.head()
```

Out[6]:

	budget	genres	homepage	id	keywords	original_langua
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "nam...	http://www.thedarkknightrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853,...	
4	260000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://movies.disney.com/john-carter	49529	[{"id": 818, "name": "based on novel"}, {"id":...	

5 rows × 23 columns

In [7]:
```python
import ast
```

In [8]:
```python
def convert(text):
    L = []
    for i in ast.literal_eval(text):
        L.append(i['name'])
    return L
```

In [9]:
```python
movies.dropna(inplace=True)
```

In [10]:
```python
movies['genres'] = movies['genres'].apply(convert)
movies.head()
```

Out[10]:

	budget	genres	homepage	id	keywords	original_language
0	237000000	[Action, Adventure, Fantasy, Science Fiction]	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	er
1	300000000	[Adventure, Fantasy, Action]	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	er
2	245000000	[Action, Adventure, Crime]	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	er
3	250000000	[Action, Crime, Drama, Thriller]	http://www.thedarkknightrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853,...	er
4	260000000	[Action, Adventure, Science Fiction]	http://movies.disney.com/john-carter	49529	[{"id": 818, "name": "based on novel"}, {"id":...	er

5 rows × 23 columns

In [49]:
```python
movies['keywords'] = movies['keywords'].apply(convert)
movies.head()
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Input In [49], in <cell line: 1>()
----> 1 movies['keywords'] = movies['keywords'].apply(convert)
      2 movies.head()

File ~\anaconda3\lib\site-packages\pandas\core\series.py:4433, in Series.apply(self, func, convert_dtype, args, **kwargs)
   4323 def apply(
   4324     self,
   4325     func: AggFuncType,
   (...)
   4328     **kwargs,
   4329 ) -> DataFrame | Series:
   4330     """
   4331     Invoke function on values of Series.
   4332 
   (...)
   4431     dtype: float64
   4432     """
-> 4433     return SeriesApply(self, func, convert_dtype, args, kwargs).apply()

File ~\anaconda3\lib\site-packages\pandas\core\apply.py:1082, in SeriesApply.apply(self)
   1078 if isinstance(self.f, str):
   1079     # if we are a string, try to dispatch
   1080     return self.apply_str()
-> 1082 return self.apply_standard()

File ~\anaconda3\lib\site-packages\pandas\core\apply.py:1137, in SeriesApply.apply_standard(self)
   1131         values = obj.astype(object)._values
   1132         # error: Argument 2 to "map_infer" has incompatible type
   1133         # "Union[Callable[..., Any], str, List[Union[Callable[..., Any],
 str]],
   1134         # Dict[Hashable, Union[Union[Callable[..., Any], str],
   1135         # List[Union[Callable[..., Any], str]]]]]"; expected
   1136         # "Callable[[Any], Any]"
-> 1137         mapped = lib.map_infer(
   1138             values,
   1139             f,  # type: ignore[arg-type]
   1140             convert=self.convert_dtype,
   1141         )
   1143 if len(mapped) and isinstance(mapped[0], ABCSeries):
   1144     # GH#43986 Need to do list(mapped) in order to get treated as nested
   1145     #  See also GH#25959 regarding EA support
   1146     return obj._constructor_expanddim(list(mapped), index=obj.index)

File ~\anaconda3\lib\site-packages\pandas\_libs\lib.pyx:2870, in pandas._libs.lib.map_infer()

Input In [45], in convert(text)
      1 def convert(text):
      2     L = []
----> 3     for i in ast.literal_eval(text):
      4         L.append(i['name'])
      5     return L

File ~\anaconda3\lib\ast.py:105, in literal_eval(node_or_string)
    103                 return left - right
    104     return _convert_signed_num(node)
--> 105 return _convert(node_or_string)

File ~\anaconda3\lib\ast.py:104, in literal_eval.<locals>._convert(node)
    102         else:
    103             return left - right
--> 104 return _convert_signed_num(node)

File ~\anaconda3\lib\ast.py:78, in literal_eval.<locals>._convert_signed_num(node)
     76     else:
     77         return - operand
---> 78 return _convert_num(node)

File ~\anaconda3\lib\ast.py:69, in literal_eval.<locals>._convert_num(node)
     67 def _convert_num(node):
     68     if not isinstance(node, Constant) or type(node.value) not in (int, float, complex):
---> 69         _raise_malformed_node(node)
     70     return node.value

File ~\anaconda3\lib\ast.py:66, in literal_eval.<locals>._raise_malformed_node(node)
     65 def _raise_malformed_node(node):
---> 66     raise ValueError(f'malformed node or string: {node!r}')

ValueError: malformed node or string: ['culture clash', 'future', 'space war', 'space colony', 'society', 'space travel', 'futuristic', 'romance', 'space', 'alien', 'tribe', 'alien planet', 'cgi', 'marine', 'soldier', 'battle', 'love affair',
```

In [13]:
```python
import ast
ast.literal_eval('[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"},
```

Out[13]:
```
[{'id': 28, 'name': 'Action'},
 {'id': 12, 'name': 'Adventure'},
 {'id': 14, 'name': 'Fantasy'},
 {'id': 878, 'name': 'Science Fiction'}]
```

In [14]:
```python
def convert3(text):
    L = []
    counter = 0
    for i in ast.literal_eval(text):
        if counter < 3:
            L.append(i['name'])
        counter+=1
    return L
```

In [15]:
```python
movies['cast'] = movies['cast'].apply(convert)
movies.head()
```

Out[15]:

	budget	genres	homepage	id	keywords	original_language
0	237000000	[Action, Adventure, Fantasy, Science Fiction]	http://www.avatarmovie.com/	19995	[culture clash, future, space war, space colon...	er
1	300000000	[Adventure, Fantasy, Action]	http://disney.go.com/disneypictures/pirates/	285	[ocean, drug abuse, exotic island, east india ...	er
2	245000000	[Action, Adventure, Crime]	http://www.sonypictures.com/movies/spectre/	206647	[spy, based on novel, secret agent, sequel, mi...	er
3	250000000	[Action, Crime, Drama, Thriller]	http://www.thedarkknightrises.com/	49026	[dc comics, crime fighter, terrorist, secret i...	er
4	260000000	[Action, Adventure, Science Fiction]	http://movies.disney.com/john-carter	49529	[based on novel, mars, medallion, space travel...	er

5 rows × 23 columns

In [16]:
```python
movies['cast'] = movies['cast'].apply(lambda x:x[0:3])
```

In [17]:
```python
def fetch_director(text):
    L = []
    for i in ast.literal_eval(text):
        if i['job'] == 'Director':
            L.append(i['name'])
    return L
```

In [18]:
```python
movies['crew'] = movies['crew'].apply(fetch_director)
```

In [19]:
```python
def collapse(L):
    L1 = []
    for i in L:
        L1.append(i.replace(" ",""))
    return L1
```

In [20]:
```python
movies['cast'] = movies['cast'].apply(collapse)
movies['crew'] = movies['crew'].apply(collapse)
movies['genres'] = movies['genres'].apply(collapse)
movies['keywords'] = movies['keywords'].apply(collapse)
```

In [21]:
```python
movies.head()
```

Out[21]:

	budget	genres	homepage	id	keywords	original_l
0	237000000	[Action, Adventure, Fantasy, ScienceFiction]	http://www.avatarmovie.com/	19995	[cultureclash, future, spacewar, spacecolony, ...	
1	300000000	[Adventure, Fantasy, Action]	http://disney.go.com/disneypictures/pirates/	285	[ocean, drugabuse, exoticisland, eastindiatrad...	
2	245000000	[Action, Adventure, Crime]	http://www.sonypictures.com/movies/spectre/	206647	[spy, basedonnovel, secretagent, sequel, mi6, ...	
3	250000000	[Action, Crime, Drama, Thriller]	http://www.thedarkknightrises.com/	49026	[dccomics, crimefighter, terrorist, secretiden...	
4	260000000	[Action, Adventure, ScienceFiction]	http://movies.disney.com/john-carter	49529	[basedonnovel, mars, medallion, spacetravel, p...	

5 rows × 23 columns

In [22]:
```python
movies['overview'] = movies['overview'].apply(lambda x:x.split())
```

In [23]:
```python
movies['tags'] = movies['overview'] + movies['genres'] + movies['keywords'] + movi
```

In [24]:
```python
new = movies.drop(columns=['overview','genres','keywords','cast','crew'])
```

In [25]:
```python
new['tags'] = new['tags'].apply(lambda x: " ".join(x))
new.head()
```

Out[25]:

	budget	homepage	id	original_language	original_title	popula
0	237000000	http://www.avatarmovie.com/	19995	en	Avatar	150.437
1	300000000	http://disney.go.com/disneypictures/pirates/	285	en	Pirates of the Caribbean: At World's End	139.082
2	245000000	http://www.sonypictures.com/movies/spectre/	206647	en	Spectre	107.376
3	250000000	http://www.thedarkknightrises.com/	49026	en	The Dark Knight Rises	112.312
4	260000000	http://movies.disney.com/john-carter	49529	en	John Carter	43.926

In [26]:
```python
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000,stop_words='english')
```

In [27]:
```python
vector = cv.fit_transform(new['tags']).toarray()
```

In [28]:
```python
vector.shape
```

Out[28]:
```
(1494, 5000)
```

In [29]:
```python
from sklearn.metrics.pairwise import cosine_similarity
```

In [30]:
```python
similarity = cosine_similarity(vector)
```

In [31]:
```python
similarity
```

Out[31]:
```
array([[1.        , 0.08134892, 0.05423261, ..., 0.        , 0.05504819,
        0.02469324],
       [0.08134892, 1.        , 0.05882353, ..., 0.04428074, 0.        ,
        0.        ],
       [0.05423261, 0.05882353, 1.        , ..., 0.        , 0.        ,
        0.        ],
       ...,
       [0.        , 0.04428074, 0.        , ..., 1.        , 0.        ,
        0.04032389],
       [0.05504819, 0.        , 0.        , ..., 0.        , 1.        ,
        0.        ],
       [0.02469324, 0.        , 0.        , ..., 0.04032389, 0.        ,
        1.        ]])
```

In [32]:
```python
new[new['title'] == 'The Lego Movie'].index[0]
```

Out[32]:
```
744
```

In [33]:
```python
def recommend(movie):
    index = new[new['title'] == movie].index[0]
    distances = sorted(list(enumerate(similarity[index])),reverse=True,key = lambd
    for i in distances[1:6]:
        print(new.iloc[i[0]].title)
```

In [37]:
```python
recommend('Gandhi')
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
Input In [37], in <cell line: 1>()
----> 1 recommend('Gandhi')

Input In [33], in recommend(movie)
      1 def recommend(movie):
----> 2     index = new[new['title'] == movie].index[0]
      3     distances = sorted(list(enumerate(similarity[index])),reverse=True,key = lambda x: x[1])
      4     for i in distances[1:6]:

File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:5039, in Index.__getitem__(self, key)
   5036 if is_integer(key) or is_float(key):
   5037     # GH#44051 exclude bool, which would return a 2d ndarray
   5038     key = com.cast_scalar_indexer(key, warn_float=True)
-> 5039     return getitem(key)
   5041 if isinstance(key, slice):
   5042     # This case is separated from the conditional above to avoid
   5043     # pessimization com.is_bool_indexer and ndim checks.
   5044     result = getitem(key)

IndexError: index 0 is out of bounds for axis 0 with size 0
```

In [34]:
```python
import pickle
```

In [35]:
```python
pickle.dump(new,open('movie_list.pkl','wb'))
pickle.dump(similarity,open('similarity.pkl','wb'))
```

In [36]:
```python
recommend('Gandhi')
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
Input In [36], in <cell line: 1>()
----> 1 recommend('Gandhi')

Input In [33], in recommend(movie)
      1 def recommend(movie):
----> 2     index = new[new['title'] == movie].index[0]
      3     distances = sorted(list(enumerate(similarity[index])),reverse=True,key = lambda x: x[1])
      4     for i in distances[1:6]:

File ~\anaconda3\lib\site-packages\pandas\core\indexes\base.py:5039, in Index.__getitem__(self, key)
   5036 if is_integer(key) or is_float(key):
   5037     # GH#44051 exclude bool, which would return a 2d ndarray
   5038     key = com.cast_scalar_indexer(key, warn_float=True)
-> 5039     return getitem(key)
   5041 if isinstance(key, slice):
   5042     # This case is separated from the conditional above to avoid
   5043     # pessimization com.is_bool_indexer and ndim checks.
   5044     result = getitem(key)

IndexError: index 0 is out of bounds for axis 0 with size 0
```

In [ ]: