In [1]:

```
import pandas as pd
import numpy as np
```

In [2]:

```
movies = pd.read_csv('tmdb_5000_movies.csv')
credits = pd.read_csv('tmdb_5000_credits.csv')
```

In [3]:

```
movies.head(1)
```

Out[3]:

	budget	genres	homepage	id	keywords	original_language
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en

In [4]:

```
credits.head(1)
```

Out[4]:

	movie_id	title	cast	crew
0	19995	Avatar	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...

In [5]:

```
# credits = credits.rename(columns={'movie_id': 'id'}) # Rename movie_id to id
```

In [6]:

```
credits.head(1)
```

Out[6]:

	movie_id	title	cast	crew
0	19995	Avatar	[{"cast_id": 242, "character": "Jake Sully", "...	[{"credit_id": "52fe48009251416c750aca23", "de...

In [7]:

```
movies = movies.merge(credits, on='title') # Convert both dataframe in single data
```

In [8]:

```
movies.head(1)
```

Out[8]:

	budget	genres	homepage	id	keywords	original_language
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en

1 rows × 23 columns

Data Preprocessing

In [9]:

```
movies.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4809 entries, 0 to 4808
Data columns (total 23 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   budget                4809 non-null   int64  
 1   genres                4809 non-null   object 
 2   homepage              1713 non-null   object 
 3   id                    4809 non-null   int64  
 4   keywords              4809 non-null   object 
 5   original_language     4809 non-null   object 
 6   original_title        4809 non-null   object 
 7   overview              4806 non-null   object 
 8   popularity            4809 non-null   float64
 9   production_companies  4809 non-null   object 
 10  production_countries  4809 non-null   object 
 11  release_date          4808 non-null   object 
 12  revenue               4809 non-null   int64  
 13  runtime               4807 non-null   float64
 14  spoken_languages      4809 non-null   object 
 15  status                4809 non-null   object 
 16  tagline               3965 non-null   object 
 17  title                 4809 non-null   object 
 18  vote_average          4809 non-null   float64
 19  vote_count            4809 non-null   int64  
 20  movie_id              4809 non-null   int64  
 21  cast                  4809 non-null   object 
 22  crew                  4809 non-null   object 
dtypes: float64(3), int64(5), object(15)
memory usage: 864.2+ KB
```

In [10]:

```
# genres
# id
# keywords
# title
# overview
# cast
# crew
```

In [11]:

```
movies = movies[['genres', 'id','title', 'keywords', 'overview', 'cast', 'crew']]
```

In [12]:

```
movies.isnull().sum()
```

```
Out[12]: genres      0
         id          0
         title       0
         keywords    0
         overview    3
         cast        0
         crew        0
         dtype: int64
```

```
In [13]: movies.head(2)
```

Out[13]:

	genres	id	title	keywords	overview	cast	
0	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	19995	Avatar	[{"id": 1463, "name": "culture clash"}, {"id":...	In the 22nd century, a paraplegic Marine is di...	[{"cast_id": 242, "character": "Jake Sully", "...	[{"cre "52fe48009251416c750a
1	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	285	Pirates of the Caribbean: At World's End	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	Captain Barbossa, long believed to be dead, ha...	[{"cast_id": 4, "character": "Captain Jack Spa...	[{"cre "52fe4232c3a36847f800

null value handling

```
In [14]: movies.isnull().sum()
```

```
Out[14]: genres      0
         id          0
         title       0
         keywords    0
         overview    3
         cast        0
         crew        0
         dtype: int64
```

```
In [15]: movies.dropna(inplace=True)
```

```
In [16]: movies.isnull().sum()
```

```
Out[16]: genres      0
         id          0
         title       0
         keywords    0
         overview    0
         cast        0
         crew        0
         dtype: int64
```

In [17]:

```python
movies.iloc[0].genres
```

Out[17]:

```
'[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}, {"id": 14, "nam
e": "Fantasy"}, {"id": 878, "name": "Science Fiction"}]'
```

Data Preprocessing for genres & keywords column

In [ ]:

```python
import ast
def convert(obj):
    l = []
    for i in ast.literal_eval(obj):
        l.append(i['name'])
    return l
movies['genres']=movies['genres'].apply(convert)
movies['keywords']=movies['keywords'].apply(convert)
```

Data Preprocessing for cast column

In [19]:

```python
def convertCast(obj):
    l = []
    counter = 0
    for i in ast.literal_eval(obj):
        if counter != 4:
            l.append(i['name'])
            counter += 1
        else:
            break
    return l
movies['cast']=movies['cast'].apply(convert)
```

Data Preprocessing for crew column

In [20]:

```python
def fetch_director(text):
    L = []
    for i in ast.literal_eval(text):
        if i['job'] == 'Director': # Only Director Name
            L.append(i['name'])
    return L
movies['crew']=movies['crew'].apply(fetch_director)
```

Data Preprocessing for overview column (Convert into a list)

In [21]:

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

Remove space between Names

In [22]:

```python
def collapse(L):
    L1 = []
    for i in L:
        L1.append(i.replace(" ",""))
    return L1
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 [23]:
```
movies.head()
```

Out[23]:

	genres	id	title	keywords	overview	cast	
0	[Action, Adventure, Fantasy, ScienceFiction]	19995	Avatar	[cultureclash, future, spacewar, spacecolony, ...	[In, the, 22nd, century,, a, paraplegic, Marin...	[SamWorthington, ZoeSaldana, SigourneyWeaver, ...	[James
1	[Adventure, Fantasy, Action]	285	Pirates of the Caribbean: At World's End	[ocean, drugabuse, exoticisland, eastindiatrad...	[Captain, Barbossa,, long, believed, to, be, d...	[JohnnyDepp, OrlandoBloom, KeiraKnightley, Ste...	[Gore
2	[Action, Adventure, Crime]	206647	Spectre	[spy, basedonnovel, secretagent, sequel, mi6, ...	[A, cryptic, message, from, Bond's, past, send...	[DanielCraig, ChristophWaltz, LéaSeydoux, Ralp...	[Sa
3	[Action, Crime, Drama, Thriller]	49026	The Dark Knight Rises	[dccomics, crimefighter, terrorist, secretiden...	[Following, the, death, of, District, Attorney...	[ChristianBale, MichaelCaine, GaryOldman, Anne...	[Christop
4	[Action, Adventure, ScienceFiction]	49529	John Carter	[basedonnovel, mars, medallion, spacetravel, p...	[John, Carter, is, a, war-weary,, former, mili...	[TaylorKitsch, LynnCollins, SamanthaMorton, Wi...	[Andre

concatenate all columns into tags

In [24]:
```
movies['tags'] = movies['overview'] + movies['genres'] + movies['keywords'] + movie
```

After concatenate all columns into tags Drop old columns

In [25]:
```
new = movies.drop(columns=['overview','genres','keywords','cast','crew'])
new.head(1)
```

Out[25]:

	id	title	tags
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...

tags list convert into string

In [26]:

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

Out[26]:

	id	title	tags
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...
1	285	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...
2	206647	Spectre	A cryptic message from Bond’s past sends him o...
3	49026	The Dark Knight Rises	Following the death of District Attorney Harve...
4	49529	John Carter	John Carter is a war-weary, former military ca...

Tags string convert into lowercase

In [27]:

```
new['tags'] = new['tags'].apply(lambda x: x.lower())
new.head()
```

Out[27]:

	id	title	tags
0	19995	Avatar	in the 22nd century, a paraplegic marine is di...
1	285	Pirates of the Caribbean: At World's End	captain barbossa, long believed to be dead, ha...
2	206647	Spectre	a cryptic message from bond’s past sends him o...
3	49026	The Dark Knight Rises	following the death of district attorney harve...
4	49529	John Carter	john carter is a war-weary, former military ca...

Vectorization

In [28]:

```
new['tags'][0]
```

Out[28]:

```
'in the 22nd century, a paraplegic marine is dispatched to the moon pandora on a u
nique mission, but becomes torn between following orders and protecting an alien c
ivilization. action adventure fantasy sciencefiction cultureclash future spacewar 
spacecolony society spacetravel futuristic romance space alien tribe alienplanet c
gi marine soldier battle loveaffair antiwar powerrelations mindandsoul 3d samworth
ington zoesaldana sigourneyweaver stephenlang michellerodriguez giovanniribisi joe
ldavidmoore cchpounder wesstudi lazalonso dileeprao mattgerald seananthonymoran ja
sonwhyte scottlawrence kellykilgour jamespatrickpitt seanpatrickmurphy peterdillon
kevindorman kelsonhenderson davidvanhorn jacobtomuri michaelblain-rozgay joncurry 
lukehawker woodyschultz petermensah soniayee jahnelcurfman ilramchoi kylawarren li
saroumain debrawilson chrismala taylorkibby jodielandau julielamm cullenb.madden j
osephbradymadden frankietorres austinwilson sarawilson tamicawashington-miller luc
ybriant nathanmeister gerryblair matthewchamberlain paulyates wraywilson jamesgayl
yn melvinlenoclarkiii carvonfutrell brandonjelkes micahmoch hanniyahmuhammad chris
tophernolen christaoliver aprilmariethomas bravitaa.threatt colinbleasdale mikebod
nar mattclayton nicoledionne jamieharrison allanhenry anthonyingruber ashleyjeffer
y deanknowsley josephmika-hunt terrynotary kaipantano loganpithyou stuartpollock r
aja garethruck rhiansheehan t.j.storm jodietaylor aliciavela-bailey richardwhitesi
de nikiezambo julenerenee jamescameron'
```

In [29]:

```python
from nltk.stem.porter import PorterStemmer
ps = PorterStemmer()
```

In [30]:

```python
def stem(text):
    y = []
    for i in text.split():
        y.append(ps.stem(i))
    return " ".join(y)
```

In [31]:

```python
new['tags'] = new['tags'].apply(stem)
```

In [32]:

```python
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000, stop_words='english')
vector = cv.fit_transform(new['tags']).toarray()
```

In [33]:

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

In [34]:

```python
similarity = cosine_similarity(vector)
```

In [35]:

```python
def recommend(movie):
    movie_index = new[new['title'] == movie].index[0]
    distance = similarity[movie_index]
    movie_list = sorted(list(enumerate(distance)), reverse=True, key = lambda x:x[1

    for i in movie_list:
        print(new.iloc[i[0]].title)
```

In [40]:

```python
recommend("Avatar")
```

```
Aliens vs Predator: Requiem
Predator
Battle: Los Angeles
Falcon Rising
Independence Day
Titan A.E.
```

In [37]:

```
import pickle
pickle.dump(new, open('movies.pkl', 'wb'))
```

In [ ]: