

```
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))
```

```
from google.colab import drive
drive.mount('/content/drive')
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force\_re

```
movies = pd.read_csv('/content/drive/MyDrive/tmdb_5000_movies.csv')
credits = pd.read_csv('/content/drive/MyDrive/tmdb_5000_credits.csv')
```

```
movies.head(2)
```

```

    budget    genres    homepage    id    keywords    original_language    original_title    overview

movies.shape

(4803, 20)

"nam...

credits.head()

```

	movie_id		title	cast	cr
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

```
movies = movies.merge(credits,on='title')
```

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

```
movies.head()
```

	movie_id	title	overview	genres	keywords	cast
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...	{{"id": 28, "name": "Action"}, {"id": 12, "nam...	{{"id": 1463, "name": "culture clash"}, {"id":...	{{"cast_id": 242, "character": "Jake Sully", "...
1	285	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	{{"id": 12, "name": "Adventure"}, {"id": 14, "...	{{"id": 270, "name": "ocean"}, {"id": 726, "na...	{{"cast_id": 4, "character": "Captain Jack Spa...
2	206647	Spectre	A cryptic message from Bond's past sends him o...	{{"id": 28, "name": "Action"}, {"id": 12, "nam...	{{"id": 470, "name": "spy"}, {"id": 818, "name...	{{"cast_id": 1, "character": "James Bond", "cr...
3	49026	The Dark Knight Rises	Following the death of District Attorney Harve...	{{"id": 28, "name": "Action"}, {"id": 80, "nam...	{{"id": 849, "name": "dc comics"}, {"id": 853,...	{{"cast_id": 2, "character": "Bruce Wayne / Ba...
4	49529	John Carter	John Carter is a war-weary, former military ca...	{{"id": 28, "name": "Action"}, {"id": 12, "nam...	{{"id": 818, "name": "based on novel"}, {"id":...	{{"cast_id": 5, "character": "John Carter", "c...

```

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

movies.dropna(inplace=True)
movies['genres'] = movies['genres'].apply(convert)
movies.head()

```

	movie_id	title	overview	genres	keywords	cast
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...	[Action, Adventure, Fantasy, Science Fiction]	{ "id": 1463, "name": "culture clash"}, { "id": ...	{ "cast_id": 242, "character": "Jake Sully", "...
1	285	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	[Adventure, Fantasy, Action]	{ "id": 270, "name": "ocean"}, { "id": 726, "na...	{ "cast_id": 4, "character": "Captain Jack Spa...
2	206647	Spectre	A cryptic message from Bond's past sends him o...	[Action, Adventure, Crime]	{ "id": 470, "name": "spy"}, { "id": 818, "name...	{ "cast_id": 1, "character": "James Bond", "cr...
3	49026	The Dark Knight Rises	Following the death of District Attorney Harve...	[Action, Crime, Drama, Thriller]	{ "id": 849, "name": "dc comics"}, { "id": 853, ...	{ "cast_id": 2, "character": "Bruce Wayne / Ba...
4	49529	John Carter	John Carter is a war-weary, former military ca...	[Action, Adventure, Science Fiction]	{ "id": 818, "name": "based on novel"}, { "id": ...	{ "cast_id": 5, "character": "John Carter", "c...

```
movies['keywords'] = movies['keywords'].apply(convert)
movies.head()
```

	movie_id	title	overview	genres	keywords	cast
0	19995	Avatar	In the 22nd century, a paraplegic Marine is di...	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	[{"cast_id": 242, "character": "Jake Sully", "...
1	285	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	[Adventure, Fantasy, Action]	[ocean, drug abuse, exotic island, east india ...	[{"cast_id": 4, "character": "Captain Jack Spa...
2	206647	Spectre	A cryptic message from Bond's past sends him o...	[Action, Adventure, Crime]	[spy, based on novel, secret agent, sequel, mi...	[{"cast_id": 1, "character": "James Bond", "cr...
3	49026	The Dark Knight Rises	Following the death of District Attorney Harve...	[Action, Crime, Drama, Thriller]	[dc comics, crime fighter, terrorist, secret i...	[{"cast_id": 2, "character": "Bruce Wayne / Ba...
4	49529	John Carter	John Carter is a war-weary, former military ca...	[Action, Adventure, Science Fiction]	[based on novel, mars, medallion, space travel...	[{"cast_id": 5, "character": "John Carter", "c...

```
import ast
```

```
data = ast.literal_eval("""
[
    {"id": 28, "name": "Action"},
    {"id": 12, "name": "Adventure"},
    {"id": 14, "name": "Fantasy"},
    {"id": 878, "name": "Science Fiction"}
]
""")
```

```
def convert3(text):
    L = []
    counter = 0
```

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

```
movies['cast'] = movies['cast'].apply(convert)  
movies.head()
```

```

movies['cast'] = movies['cast'].apply(lambda x:x[0:3])

n      10005      Avatar      in the 22nd century, a      Fantasy Science      [culture clash, future,      Saldana, Sigourney

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

=      2000.7      Species      Linda's past comes back.      Crime1      secret agent, sequel,      Waltz, Léa Sevdoux. R...

movies['crew'] = movies['crew'].apply(fetch_director)

3      49026      Piece      District Attorney Henry      Thriller1      terrorist, secret i      Caine, Gary Oldman, A

movies.sample(5)

```

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



```

    movie_id      title      overview      genres      keywords
0      19995      Avatar      In the 22nd century, a paraplegic Marine is di...      [Action, Adventure, Fantasy]      [culture, death, future, ...]
1      285      Pirates of the Caribbean: At World's End      Captain Barbossa, long believed to be dead, ha...      [Adventure, Fantasy, Action]      [ocean, drug abuse, ...]
2      206647      Spectre      A cryptic message from Bond's past sends him o...      [Action, Adventure, Thriller]      [James Bond, ...]
3      49026      The Dark Knight Rises      Following the death of District Attorney Harve...      [Action, Crime, Drama]      [Batman, ...]
4      49529      John Carter      John Carter is a war-weary, former military ca...      [Adventure, Fantasy, Action]      [John Carter, ...]

movies['overview'] = movies['overview'].apply(lambda x:x.split())
movies['tags'] = movies['overview'] + movies['genres'] + movies['keywords'] + movies['cast'] + movies['crew']

new = movies.drop(columns=['overview', 'genres', 'keywords', 'cast', 'crew'])
new['tags'] = new['tags'].apply(lambda x: " ".join(x))
new.head()

```

	movie_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...

```

from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000, stop_words='english')
vector = cv.fit_transform(new['tags']).toarray()
vector.shape
from sklearn.metrics.pairwise import cosine_similarity
similarity = cosine_similarity(vector)
similarity

```

```

array([[1.          , 0.08964215, 0.06071767, ..., 0.02519763, 0.0277885 ,
        0.          ],
       [0.08964215, 1.          , 0.06350006, ..., 0.02635231, 0.          ,
        0.          ],
       [0.06071767, 0.06350006, 1.          , ..., 0.02677398, 0.          ,
        0.          ],
       ...,

```

```
[0.02519763, 0.02635231, 0.02677398, ..., 1.          , 0.07352146,
 0.04774099],
[0.0277885 , 0.          , 0.          , ..., 0.07352146, 1.          ,
 0.05264981],
[0.          , 0.          , 0.          , ..., 0.04774099, 0.05264981,
 1.          ]]
```

```
new[new['title'] == 'The Lego Movie'].index[0]
def recommend(movie):
    index = new[new['title'] == movie].index[0]
    distances = sorted(list(enumerate(similarity[index])),reverse=True,key = lambda x: x[1])
    for i in distances[1:6]:
        print(new.iloc[i[0]].title)
    ....
```

```
recommend('Avatar')
```

```
Titan A.E.
Small Soldiers
Ender's Game
Aliens vs Predator: Requiem
Independence Day
```

```
user_input = input("Enter a movie name: ")
```

```
try:
    recommended_movies = recommend(user_input)

    if recommended_movies is not None:
        if recommended_movies.empty:
            print("Sorry, no recommendations found for the given movie.")
        else:
            print(recommended_movies)
    else:
        print("Sorry, no recommendations found for the given movie.")
except Exception as e:
    print(f"An error occurred: {e}")
```

```
Enter a movie name: Avatar
Titan A.E.
Small Soldiers
Ender's Game
Aliens vs Predator: Requiem
Independence Day
Sorry, no recommendations found for the given movie.
```