

```
In [1]: import numpy as np
import pandas as pd
```

```
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_lan
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id": ...	

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

```
In [6]: movies.head(1)
```

```
Out[6]:
```

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

1 rows × 23 columns

```
In [7]: # genres
# id
# keywords
# title
# overview
# cast
# crew
movies = movies[['movie_id', 'title', 'overview', 'genres', 'keywords', 'cast'
```

```
In [8]: movies.head()
```

```
Out[8]:
```

	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", "...	"52fe480
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...	"52fe423
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...	"5480596
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...	"52fe478
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...	"52fe479

```
In [9]: movies.isnull().sum()
```

```
Out[9]: movie_id    0
title            0
overview         3
genres           0
keywords         0
cast             0
crew            0
dtype: int64
```

```
In [10]: movies.iloc[0].genres
```

```
Out[10]: '[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fantasy"}, {"id": 878, "name": "Science Fiction"}]'
```

```
In [11]: # '[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fantasy"}, {"id": 878, "name": "Science Fiction"}]'
#['Action', 'Adventure', 'Fantasy', 'SciFi']
```

```
In [12]: 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()
```

Out [12]:

	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", ... }	"52fe4800
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...	"52fe4232c
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...	"54805967c
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...	"52fe4781c
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...	"52fe479a

```
In [13]: movies['keywords'] = movies['keywords'].apply(convert)
movies.head()
```

Out [13]:

	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...

```
In [14]: 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]: movies['cast'] = movies['cast'].apply(convert)
movies.head()
```

Out[15]:

	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...	[Sam Worthington, Zoe Saldana, Sigourney Weave...	"52fe4800
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 ...	[Johnny Depp, Orlando Bloom, Keira Knightley, ...	"52fe4232
2	206647	Spectre	A cryptic message from Bond's past sends him o...	[Action, Adventure, Crime]	[spy, based on novel, secret agent, sequel, mi...	[Daniel Craig, Christoph Waltz, Léa Seydoux, R...	"54805967
3	49026	The Dark Knight Rises	Following the death of District Attorney Harve...	[Action, Crime, Drama, Thriller]	[dc comics, crime fighter, terrorist, secret i...	[Christian Bale, Michael Caine, Gary Oldman, A...	"52fe4781
4	49529	John Carter	John Carter is a war-weary, former military ca...	[Action, Adventure, Science Fiction]	[based on novel, mars, medallion, space travel...	[Taylor Kitsch, Lynn Collins, Samantha Morton,...	"52fe4792

In [16]: `movies['overview'][0]`Out[16]: `'In the 22nd century, a paraplegic Marine is dispatched to the moon Pandora on a unique mission, but becomes torn between following orders and protecting an alien civilization.'`In [18]: `movies['overview'] = movies['overview'].apply(lambda x:x.split())`In [19]: `movies.head()`

Out [19]:

	movie_id	title	overview	genres	keywords	cast	
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, Science Fiction]	[culture clash, future, space war, space colon...	[Sam Worthington, Zoe Saldana, Sigourney Weave...	"52fe480
1	285	Pirates of the Caribbean: At World's End	[Captain, Barbossa,, long, believed, to, be, d...	[Adventure, Fantasy, Action]	[ocean, drug abuse, exotic island, east india ...	[Johnny Depp, Orlando Bloom, Keira Knightley, ...	"52fe4232
2	206647	Spectre	[A, cryptic, message, from, Bond's, past, send...	[Action, Adventure, Crime]	[spy, based on novel, secret agent, sequel, mi...	[Daniel Craig, Christoph Waltz, Léa Seydoux, R...	"54805967
3	49026	The Dark Knight Rises	[Following, the, death, of, District, Attorney...	[Action, Crime, Drama, Thriller]	[dc comics, crime fighter, terrorist, secret i...	[Christian Bale, Michael Caine, Gary Oldman, A...	"52fe478
4	49529	John Carter	[John, Carter, is, a, war-weary,, former, mili...	[Action, Adventure, Science Fiction]	[based on novel, mars, medallion, space travel...	[Taylor Kitsch, Lynn Collins, Samantha Morton,...	"52fe479

```
In [23]: movies['genres']= movies['genres'].apply(lambda x:[i.replace(" ",",") for
movies['keywords']= movies['keywords'].apply(lambda x:[i.replace(" ",",")
movies['cast']= movies['cast'].apply(lambda x:[i.replace(" ",",") for i in
movies['crew']= movies['crew'].apply(lambda x:[i.replace(" ",",") for i in
```

```
In [24]: movies.head()
```

Out [24]:	movie_id	title	overview	genres	keywords	cas
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	[SamWorthington, ZoeSaldana, SigourneyWeaver, ..
1	285	Pirates of the Caribbean: At World's End	[Captain, Barbossa,, long, believed, to, be, d...	[Adventure, Fantasy, Action]	[ocean, drugabuse, exoticisland, eastindiatrad...	[JohnnyDepp, OrlandoBloom, KeiraKnightley, Ste..
2	206647	Spectre	[A, cryptic, message, from, Bond's, past, send...	[Action, Adventure, Crime]	[spy, basedonnovel, secretagent, sequel, mi6, ...	[DanielCraig, ChristophWaltz, LéaSeydoux, Ralp..
3	49026	The Dark Knight Rises	[Following, the, death, of, District, Attorney...	[Action, Crime, Drama, Thriller]	[dccomics, crimefighter, terrorist, secretiden...	[ChristianBale, MichaelCaine, GaryOldman, Anne..
4	49529	John Carter	[John, Carter, is, a, war-weary,, former, mili...	[Action, Adventure, ScienceFiction]	[basedonnovel, mars, medallion, spacetravel, p...	[TaylorKitsch, LynnCollins, SamanthaMorton, Wi..

```
In [27]: movies['tags'] = movies['overview'] + movies['genres'] + movies['keywords']
```

```
In [28]: movies.head()
```



Out [28]:	movie_id	title	overview	genres	keywords	cas
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	[SamWorthington, ZoeSaldana, SigourneyWeaver, ..
1	285	Pirates of the Caribbean: At World's End	[Captain, Barbossa,, long, believed, to, be, d...	[Adventure, Fantasy, Action]	[ocean, drugabuse, exoticisland, eastindiatrad...	[JohnnyDepp, OrlandoBloom, KeiraKnightley, Ste..
2	206647	Spectre	[A, cryptic, message, from, Bond's, past, send...	[Action, Adventure, Crime]	[spy, basedonnovel, secretagent, sequel, mi6, ...	[DanielCraig, ChristophWaltz, LéaSeydoux, Ralp..
3	49026	The Dark Knight Rises	[Following, the, death, of, District, Attorney...	[Action, Crime, Drama, Thriller]	[dccomics, crimefighter, terrorist, secretiden...	[ChristianBale, MichaelCaine, GaryOldman, Anne..
4	49529	John Carter	[John, Carter, is, a, war-weary,, former, mili...	[Action, Adventure, ScienceFiction]	[basedonnovel, mars, medallion, spacetravel, p...	[TaylorKitsch, LynnCollins, SamanthaMorton, Wi..

```
In [31]: new_df = movies[['movie_id','title','tags']]
```

```
In [34]: new_df['tags']= new_df['tags'].apply(lambda x:" ".join(x))
```

```

/var/folders/bn/4hk3f4kn3bqb3hxf7bxjd64h0000gn/T/ipykernel_43623/323963808
0.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
new_df['tags']= new_df['tags'].apply(lambda x:" ".join(x))

```

In [35]: `new_df.head()`

Out[35]:

	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...

In [36]: `new_df['tags']= new_df['tags'].apply(lambda x:x.lower())`

```

/var/folders/bn/4hk3f4kn3bqb3hxf7bxjd64h0000gn/T/ipykernel_43623/130686010
9.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
new_df['tags']= new_df['tags'].apply(lambda x:x.lower())

```

In [37]: `new_df.head()`

Out[37]:

	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...

In [50]: `import nltk`

In [52]: `from nltk.stem.porter import PorterStemmer  
ps = PorterStemmer()`

In [59]: `def stem(text):  
 y = []  
  
 for i in text.split():  
 y.append(ps.stem(i))  
 return " ".join(y)`

In [60]: `new_df['tags'] = new_df['tags'].apply(stem)`

Out[60]:

```

0          in
1        captain
2           a
3        follow
4         john
...
4804        el
4805         a
4806    "signed,
4807         when
4808         ever
Name: tags, Length: 4806, dtype: object

```

In [39]: `from sklearn.feature_extraction.text import CountVectorizer  
cv = CountVectorizer(max_features = 5000, stop_words='english')`

In [42]: `vectors = cv.fit_transform(new_df['tags']).toarray()`

In [43]: `vectors`

```
Out[43]: array([[0, 0, 0, ..., 0, 0, 0],
               [0, 0, 0, ..., 0, 0, 0],
               [0, 0, 0, ..., 0, 0, 0],
               ...,
               [0, 0, 0, ..., 0, 0, 0],
               [0, 0, 0, ..., 0, 0, 0],
               [0, 0, 0, ..., 0, 0, 0]])
```

```
In [44]: vectors[0]
```

```
Out[44]: array([0, 0, 0, ..., 0, 0, 0])
```

```
In [63]: from sklearn.metrics.pairwise import cosine_similarity
```

```
In [72]: similarity = cosine_similarity(vectors)
```

```
In [81]: sorted(list(enumerate(similarity[0])), reverse=True, key=lambda x:x[1])[1:6]
```

```
Out[81]: [(1214, 0.23160277792655043),
          (1916, 0.23145502494313785),
          (582, 0.2309782890611944),
          (539, 0.2252817784447915),
          (507, 0.21912524504463887)]
```

```
In [73]: similarity[1]
```

```
Out[73]: array([0.07142857, 1.          , 0.07715167, ..., 0.02326211, 0.          ,
                0.          ])
```

```
In [110... def recommend(movie):
            movie_index = new_df[new_df['title'] == movie].index[0]
            distances = similarity[movie_index]
            movies_list = sorted(list(enumerate(distances)), reverse=True, key = lambda x: x[1])

            for i in movies_list:
                print(new_df.iloc[i[0]].title)
```

```
In [111... recommend('Batman Begins')
```

```
The Dark Knight
The Dark Knight Rises
Amidst the Devil's Wings
Batman
Batman & Robin
```

```
In [125... # Sample validation dataset (manually defined)
validation_data = {
    "Inception": {"The Dark Knight", "Interstellar", "Memento"},
    "Titanic": {"The Notebook", "Romeo + Juliet", "Pride and Prejudice"},
    "The Avengers": {"Iron Man", "Thor", "Captain America"},
}

# Test movies: the keys of the dictionary
```

```

test_movies = list(validation_data.keys()) # Ensure this part is include

# Modify the recommend function for better matching
def recommend(movie):
    try:
        # Normalize movie titles (to lowercase and strip whitespace)
        movie = movie.strip().lower()

        # Find the movie index by matching case-insensitively
        movie_index = movies[movies['title'].str.lower() == movie].index[0]
        distances = similarity[movie_index]
        movies_list = sorted(list(enumerate(distances)), reverse=True, key=lambda x: x[1])

        recommended_movies = [movies.iloc[i[0]].title for i in movies_list[:5]]
        return recommended_movies
    except (IndexError, Exception):
        return [] # Return an empty list instead of None if movie not found

# Function to evaluate precision and recall for a single movie
def evaluate_recommendations(movie, ground_truth, model_recommendations):
    # Convert recommendations to a set for comparison
    recommendations = set(model_recommendations)
    relevant_recommended = ground_truth.intersection(recommendations)

    # Calculate precision and recall
    precision = len(relevant_recommended) / len(recommendations) if recommendations else 0
    recall = len(relevant_recommended) / len(ground_truth) if ground_truth else 0

    return precision, recall

# Initialize lists to store precision and recall scores
precision_scores = []
recall_scores = []

# Evaluate for each movie in the validation dataset
for movie in test_movies:
    ground_truth = validation_data[movie] # Relevant movies for the test
    model_recommendations = recommend(movie) # Using the `recommend` function

    # Debugging: Print the recommendations for each movie
    print(f"Recommendations for '{movie}': {model_recommendations}")

    # If recommendations are empty, skip this movie
    if not model_recommendations:
        print(f"Warning: No recommendations found for {movie}. Skipping..")
        continue

    precision, recall = evaluate_recommendations(movie, ground_truth, model_recommendations)
    precision_scores.append(precision)
    recall_scores.append(recall)

# Calculate average precision and recall
if precision_scores and recall_scores:
    avg_precision = sum(precision_scores) / len(precision_scores)
    avg_recall = sum(recall_scores) / len(recall_scores)

```

```
average_precision = np.mean(precision_scores)
average_recall = np.mean(recall_scores)
print(f"Average Precision: {average_precision:.2f}")
print(f"Average Recall: {average_recall:.2f}")
else:
    print("No valid recommendations found, unable to calculate average pr
```

```
Recommendations for 'Inception': ['Timecop', 'The Walk', 'The Helix... Loa
ded', 'Star Trek II: The Wrath of Khan', 'The Score']
Recommendations for 'Titanic': ['Captain Phillips', 'In the Heart of the S
ea', 'Ghost Ship', 'The Notebook', 'The Bounty']
Recommendations for 'The Avengers': ['Avengers: Age of Ultron', 'Iron Man
3', 'Iron Man', 'Iron Man 2', 'Captain America: Civil War']
Average Precision: 0.13
Average Recall: 0.22
```

```
In [112... import pickle
```

```
In [113... pickle.dump(new_df,open('movies.pkl','wb'))
```

```
In [115... new_df['title'].values
```

```
Out[115... array(['Avatar', "Pirates of the Caribbean: At World's End", 'Spectre',
..., 'Signed, Sealed, Delivered', 'Shanghai Calling',
'My Date with Drew'], dtype=object)
```

```
In [117... pickle.dump(new_df.to_dict(),open('movies_dict.pkl','wb'))
```

```
In [119... pickle.dump(similarity,open('similarity.pkl' , 'wb'))
```

```
In [ ]:
```