

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

```
In [3]: movies = pd.read_csv('tmdb_5000_movies.csv')
```

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

```
Out[4]:
```

	budget	genres	homepage	id	keywords
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}]	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id": 1464, "name": "culture clash"}]
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "name": "Fantasy"}]	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "name": "pirates"}]
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}]	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name": "spy"}]
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "name": "Fantasy"}]	http://www.thedarkknighttrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853, "name": "dc comics"}]
4	260000000	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}]	http://movies.disney.com/john-carter	49529	[{"id": 818, "name": "based on novel"}, {"id": 819, "name": "based on novel"}]

```
In [6]: credits = pd.read_csv("tmdb_5000_credits.csv")
```

```
In [7]: credits.head()
```

Out [7]:

	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 [11]:

```
movies1 = movies.merge(credits,on='title')
```

In [12]:

```
movies1.head()
```

Out [12]:

	budget	genres	homepage	id	keywords
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}]	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id": 1464, "name": "culture clash"}]
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "name": "Action"}]	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "name": "na..."}]
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}]	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name": "name..."}]
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "name": "Action"}]	http://www.thedarkknighttrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853, "name": "dc comics"}]
4	260000000	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Adventure"}]	http://movies.disney.com/john-carter	49529	[{"id": 818, "name": "based on novel"}, {"id": 819, "name": "based on novel"}]

5 rows × 23 columns

Now we will ask ourselves which columns to pick (will they be useful for generating tags or not)

1. Genres
2. id
3. keywords
4. original language (here we have a little imbalanced data) so will discard this
5. original title or title
6. Overview (as content based LSTM we are using , we will pick this)
7. popularity (same as above reason, we will discard this), companies,countries, release date, revenue, runtime,spoken lang, status, tagline, vote average,vote count,
8. Cast n Crew (we will pick)

```
In [13]: movies2 = movies1[['movie_id',
                        'title', 'overview', 'genres', 'keywords', 'cast', 'crew' ]]
```

```
In [15]: movies2.head()
```

Out[15]:

	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...

Now we will do data preprocessing which includes (filling missing values,removal of duplicates)

```
In [16]: movies2.isnull().sum()
```

```
Out[16]: movie_id    0
title      0
overview   3
genres     0
keywords   0
cast       0
crew       0
dtype: int64
```

```
In [25]: movies3 = movies2.dropna()
```

```
In [26]: movies3.isnull().sum()
```

```
Out[26]: movie_id    0
         title      0
         overview   0
         genres     0
         keywords   0
         cast       0
         crew       0
         dtype: int64
```

```
In [27]: movies3.duplicated().sum()
```

```
Out[27]: 0
```

```
In [28]: movies3.iloc[0].genres
```

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

```
In [29]: import ast
```

```
In [32]: def convert(obj):
         l=[]
         for i in ast.literal_eval(obj):
             l.append(i['name'])
         return l
```

```
In [34]: movies3['genres'] = movies3['genres'].apply(convert)
```

```
In [36]: movies3['keywords'] = movies3['keywords'].apply(convert)
```

```
In [37]: movies3.head()
```

Out [37]:

	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 [38]:

```
def convert2(obj):
    l=[]
    counter=0
    for i in ast.literal_eval(obj):
        if counter!=3:
            l.append(i['name'])
            counter+=1
        else:
            break
    return l
```

In [40]:

```
movies3['cast']=movies3['cast'].apply(convert2)
```

In [41]:

```
movies3.head()
```

Out [41]:

	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 Weaver]	["52fe48009251416c
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]	["52fe4232c3a36847
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]	["54805967c3a36829
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]	["52fe4781c3a36847
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]	["52fe479ac3a3684

In [42]:

```
def fetch(obj):
    l=[]
    for i in ast.literal_eval(obj):
        if i['job']=='Director':
            l.append(i['name'])
            break
    return l
```

In [44]:

```
movies3['crew']=movies3['crew'].apply(fetch)
```

In [45]:

```
movies3.head()
```

Out [45]:

	movie_id	title	overview	genres	keywords	cast	crew
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 Weaver]	[James Cameron]
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]	[Gore Verbinski]
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]	[Sam Mendes]
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]	[Christopher Nolan]
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]	[Andrew Stanton]

In [46]:

```
movies3['overview']=movies3['overview'].apply(lambda x : x.split())
```

In [47]:

```
movies3.head()
```


Out [47]:

	movie_id	title	overview	genres	keywords	cast	crew
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 Weaver]	[James Cameron]
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]	[Gore Verbinski]
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]	[Sam Mendes]
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]	[Christopher Nolan]
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]	[Andrew Stanton]

In [51]:

```

movies3['genres']=movies3['genres'].apply(lambda x : [i.replace(" ", "") for i in x])
movies3['keywords']=movies3['keywords'].apply(lambda x : [i.replace(" ", "") for i in x])
movies3['cast']=movies3['cast'].apply(lambda x : [i.replace(" ", "") for i in x])
movies3['crew']=movies3['crew'].apply(lambda x : [i.replace(" ", "") for i in x])

```

In [52]:

```

movies3.head()

```

06/12/2022, 18:26

MRE

Out [52]:

	movie_id	title	overview	genres	keywords	cast	
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[Jam
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]	[Go
2	206647	Spectre	[A, cryptic, message, from, Bond's, past, send...	[Action, Adventure, Crime]	[spy, basedonnovel, secretagent, sequel, mi6, ...	[DanielCraig, ChristophWaltz, LéaSeydoux]	[S
3	49026	The Dark Knight Rises	[Following, the, death, of, District, Attorney...	[Action, Crime, Drama, Thriller]	[dccomics, crimefighter, terrorist, secretiden...	[ChristianBale, MichaelCaine, GaryOldman]	[Christ
4	49529	John Carter	[John, Carter, is, a, war-weary,, former, mili...	[Action, Adventure, ScienceFiction]	[basedonnovel, mars, medallion, spacetravel, p...	[TaylorKitsch, LynnCollins, SamanthaMorton]	[And

In [53]:

```
movies3['tags'] = movies3['overview'] + movies3['genres'] +  
movies3['keywords'] + movies3['cast'] + movies3['crew']
```

In [54]:

```
movies3.head()
```

Out [54]:

	movie_id	title	overview	genres	keywords	cast	
0	19995	Avatar	[In, the, 22nd, century,, a, paraplegic, Marin...	[Action, Adventure, Fantasy, ScienceFiction]	[cultureclash, future, spacewar, spacecolony, ...	[SamWorthington, ZoeSaldana, SigourneyWeaver]	[Jam
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]	[Go
2	206647	Spectre	[A, cryptic, message, from, Bond's, past, send...	[Action, Adventure, Crime]	[spy, basedonnovel, secretagent, sequel, mi6, ...	[DanielCraig, ChristophWaltz, LéaSeydoux]	[S
3	49026	The Dark Knight Rises	[Following, the, death, of, District, Attorney...	[Action, Crime, Drama, Thriller]	[dccomics, crimefighter, terrorist, secretiden...	[ChristianBale, MichaelCaine, GaryOldman]	[Christ
4	49529	John Carter	[John, Carter, is, a, war-weary,, former, mili...	[Action, Adventure, ScienceFiction]	[basedonnovel, mars, medallion, spacetravel, p...	[TaylorKitsch, LynnCollins, SamanthaMorton]	[And

In [57]:

```
movies_final = movies3[['movie_id', 'title', 'tags']]
movies_final['tags'] = movies_final['tags'].apply(lambda x : "
".join(x))
```

```
/var/folders/72/94yf9pqn1jldywq95n5rkh680000gn/T/ipykernel_56913/1833041197.
py:2: 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
movies_final['tags'] = movies_final['tags'].apply(lambda x : " ".join(x))
```

In [59]:

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

```
/var/folders/72/94yf9pqn1jldywq95n5rkh680000gn/T/ipykernel_56913/1555663356.
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
movies_final['tags'] = movies_final['tags'].apply(lambda x : x.lower())
```

In [61]:

```
movies_final.head()
```

Out [61]:

	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...

Now for the model, we gotta calculate similarity between texts, such that our engine will be able to recommend easily, but how to check similarity, i guess we can by similarity score, but here similarity can't be calculate as it is textual data (string form) [tags] So for this we will do vectorisation of text (bag of words) We won't consider stop words here

In [75]:

```
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000, stop_words='english')
```

In [76]:

```
import nltk
from nltk.stem.porter import PorterStemmer
ps=PorterStemmer()
def stem(text):
    y=[]
    for i in text.split():
        y.append(ps.stem(i))
    return " ".join(y)
```

In [77]:

```
movies_final['tags'] = movies_final['tags'].apply(stem)
```

```
/var/folders/72/94yf9pqn1jldywq95n5rkh680000gn/T/ipykernel_56913/2560892306.
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
movies_final['tags'] = movies_final['tags'].apply(stem)
```

In [78]:

```
cv.fit_transform(movies_final['tags']).toarray().shape
```

Out[78]:

```
(4806, 5000)
```

In [79]:

```
vectors = cv.fit_transform(movies_final['tags']).toarray()
```

```
In [80]: vectors[0]
```

```
Out[80]: array([0, 0, 0, ..., 0, 0, 0])
```

```
In [81]: cv.get_feature_names()
```

```
Out[81]: ['000',
'007',
'10',
'100',
'11',
'12',
'13',
'14',
'15',
'16',
'17',
'17th',
'18',
'18th',
'18thcenturi',
'19',
'1910',
'1920',
'1930',
'1940',
'1944',
'1950',
'1950s',
'1960',
'1960s',
'1970',
'1970s',
'1971',
'1974',
'1976',
'1980',
'1985',
'1990',
'1999',
'19th',
'19thcenturi',
'20',
'200',
'2003',
'2009',
'20th',
'21st',
'23',
'24',
'25',
'30',
'300',
'3d',
'40',
'50',
'500',
'60',
'70',
'80',
'aaron',
'aaroneckhart',
```

```
'concern',  
'concert',  
'concoct',  
'condit',  
'condition',  
'conduct',  
'confeder',  
'confess',  
'confid',  
'confin',  
'conflict',  
'confront',  
'confus',  
'congress',  
'conman',  
'connect',  
'connecticut',  
'connel',  
'connor',  
'conquer',  
'consequ',  
'consequences',  
'conserv',  
'consid',  
'conspir',  
'conspiraci',  
'conspiracy',  
'constant',  
'constantli',  
'construct',  
'consum',  
'contact',  
'contain',  
'contemporari',  
'contend',  
'content',  
'contest',  
'continu',  
'contract',  
'contractor',  
'control',  
'controversi',  
'convent',  
'converg',  
'convers',  
'convict',  
'convinc',  
'cook',  
...]
```

Higher dimensionality leads to failure of euclidian distance or manhola distances, so here we will use cosine similarity

```
In [82]: from sklearn.metrics.pairwise import cosine_similarity
```

```
In [84]: cosine_similarity(vectors)
```

```
Out[84]: array([[1.          , 0.08346223, 0.0860309 , ..., 0.04499213, 0.
        0.          ],
       [0.08346223, 1.          , 0.06063391, ..., 0.02378257, 0.
        0.02615329],
       [0.0860309 , 0.06063391, 1.          , ..., 0.02451452, 0.
        0.          ],
       ...,
       [0.04499213, 0.02378257, 0.02451452, ..., 1.          , 0.03962144,
        0.04229549],
       [0.          , 0.          , 0.          , ..., 0.03962144, 1.
        0.08714204],
       [0.          , 0.02615329, 0.          , ..., 0.04229549, 0.08714204,
        1.          ]])
```

```
In [87]: simi=cosine_similarity(vectors)
```

```
In [90]: simi[1]
```

```
Out[90]: array([0.08346223, 1.          , 0.06063391, ..., 0.02378257, 0.
        0.02615329])
```

```
In [92]: movies_final[movies_final['title']=='Avatar']
```

```
Out[92]:
```

	movie_id	title	tags
0	19995	Avatar	in the 22nd century, a parapleg marin is dispa...

```
In [102... def recommend(movie):
    movie_index =
movies_final[movies_final['title']==movie].index[0]
    distances = simi[movie_index]
    movies_list = sorted(list(enumerate(distances)),reverse=True,
key = lambda x:x[1])[1:6]
    for i in movies_list:
        print(movies_final.iloc[i[0]].title)
```

```
In [107... recommend('Avatar')
```

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

```
In [108... import pickle
pickle.dump(movies_final,open('movies.pkl','wb'))
```

```
In [109... pickle.dump(simi,open('simi.pkl','wb'))
```


Now code the below lines on Pycharm for deploying your streamlit app locally and then via streamlit cloud, heroku, deta,vercel, you can upload it publically

```
In [ ]: # import streamlit as st
# import pickle
# import pandas as pd
# import requests

# def fetch_poster(movie_id):
#     response =
requests.get('https://api.themoviedb.org/3/movie/{}?api_key=[your
api key]&language=en-US'.format(movie_id))
#     data = response.json()
#     return "https://image.tmdb.org/t/p/w300/" +
data['poster_path']
# def recommend(movie):
#     movie_index = movies[movies['title']==movie].index[0]
#     distances = simi[movie_index]
#     movies_list =
sorted(list(enumerate(distances)),reverse=True, key = lambda
x:x[1])[1:6]
#     recommend_movies=[]
#     posters=[]
#     for i in movies_list:
#         movie_id=movies.iloc[i[0]].movie_id
#         recommend_movies.append(movies.iloc[i[0]].title)
#         posters.append(fetch_poster(movie_id))
#     return recommend_movies,posters

# movies_dict = pickle.load(open('movies.pkl','rb'))
# movies = pd.DataFrame(movies_dict)
# simi = pickle.load(open('simi.pkl','rb'))

# st.title("Movie Recommendation Engine")
# selected_movie_name = st.selectbox('How would you like to be
connected?',movies['title'].values)

# if st.button('Recommend'):
#     names,posters=recommend(selected_movie_name)
#     col1, col2, col3 ,col4, col5= st.columns(5)
```

```
#     with col1:
#         st.text(names[0])
#         st.image(posters[0])

#     with col2:
#         st.text(names[1])
#         st.image(posters[1])

#     with col3:
#         st.text(names[2])
#         st.image(posters[2])

#     with col4:
#         st.text(names[3])
#         st.image(posters[3])

#     with col5:
#         st.text(names[4])
#         st.image(posters[4])
```