

NAME: Vishwajeet Dnyaneshwar Deshmukh

ROLL NO.: 2231030

Mini Project :- Develop a movie recommendation model using the scikit-learn library in python

```
In [2]: import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity
df = pd.read_csv("movie_dataset.csv")
df
```

Out[2]:

	index	budget	genres	homepage	id	
0	0	237000000	Action Adventure Fantasy Science Fiction	http://www.avatarmovie.com/	19995	cu
1	1	300000000	Adventure Fantasy Action	http://disney.go.com/disneypictures/pirates/	285	O €
2	2	245000000	Action Adventure Crime	http://www.sonypictures.com/movies/spectre/	206647	sp sec
3	3	250000000	Action Crime Drama Thriller	http://www.thedarkknightises.com/	49026	
4	4	260000000	Action Adventure Science Fiction	http://movies.disney.com/john-carter	49529	
...
4798	4798	220000	Action Crime Thriller		NaN	9367 stat
4799	4799	9000	Comedy Romance		NaN	72766
4800	4800	0	Comedy Drama Romance TV Movie	http://www.hallmarkchannel.com/signedsealeddel...	231617	
4801	4801	0	NaN	http://shanghaicalling.com/	126186	
4802	4802	0	Documentary		NaN	25975

4803 rows × 24 columns

```

In [9]: features = ['keywords', 'cast', 'genres', 'director']
def combine_features(row):
    return row['keywords']+" "+row['cast']+" "+row['genres']+" "+row['director']
for feature in features:
    df[feature] = df[feature].fillna('')

df["combined_features"] = df.apply(combine_features,axis=1)
cv = CountVectorizer()
count_matrix = cv.fit_transform(df["combined_features"])
cosine_sim = cosine_similarity(count_matrix)
def get_title_from_index(index):
    return df[df.index == index]["title"].values[0]
def get_index_from_title(title):
    return df[df.title == title]["index"].values[0]

movie_user_likes = "Shanghai Calling"
movie_index = get_index_from_title(movie_user_likes)
similar_movies = list(enumerate(cosine_sim[movie_index]))
sorted_similar_movies = sorted(similar_movies,key=lambda x:x[1],reverse=True)[1:]
i=0
print("Top 5 similar movies to "+movie_user_likes+" are:\n")
for element in sorted_similar_movies:
    print(get_title_from_index(element[0]))
    i=i+1
    if i>5:
        break

```

Top 5 similar movies to Shanghai Calling are:

Frailty
 The Fifth Estate
 Harry Potter and the Chamber of Secrets
 Big Hero 6
 The Hadza: Last of the First
 Doogal

In []: