In [6]:

```
import pandas as pd
from scipy.sparse import csr_matrix
from sklearn.neighbors import NearestNeighbors
from fuzzywuzzy import process
```

In [2]:

```
!pip install python-Levenshtein
```

```
Collecting python-Levenshtein
  Downloading python_Levenshtein-0.20.5-py3-none-any.whl (9.4 kB)
Collecting Levenshtein==0.20.5
  Downloading Levenshtein-0.20.5-cp39-cp39-win_amd64.whl (98 kB)
     -------------------------------------- 98.5/98.5 kB 156.8 kB/s eta 0:00:00
Collecting rapidfuzz<3.0.0,>=2.3.0
  Downloading rapidfuzz-2.11.1-cp39-cp39-win_amd64.whl (993 kB)
     ------------------------------------ 993.6/993.6 kB 243.0 kB/s eta 0:00:00
Installing collected packages: rapidfuzz, Levenshtein, python-Levenshtein
Successfully installed Levenshtein-0.20.5 python-Levenshtein-0.20.5 rapidfuzz-2.11.1

[notice] A new release of pip available: 22.1.2 -> 22.2.2
[notice] To update, run: python.exe -m pip install --upgrade pip
```

In [24]:

```
!pip install --upgrade pip
```

```
Requirement already satisfied: pip in c:\users\kmiyienda\appdata\local\programs\python\python39\lib\site-packages (22.2.2)
```

In [7]:

```
#importing files in .csv

movies='movies.csv'
ratings='ratings.csv'

df_movies=pd.read_csv(movies, usecols=['movieId','title'], dtype={'movieId':'int32','title':'str'})
df_ratings=pd.read_csv(ratings, usecols=['userId','movieId','rating'],dtype={'userId':'int32','movieId':'int32','rating':'float32'})
```

In [8]:

```
#checking to see if the files imported correctly

df_movies.head()
```

Out[8]:

	movieId	title
0	1	Toy Story (1995)
1	2	Jumanji (1995)
2	3	Grumpier Old Men (1995)
3	4	Waiting to Exhale (1995)
4	5	Father of the Bride Part II (1995)

In [9]:

```
#checking to see if the files imported correctly

df_ratings.head()
```

Out[9]:

	userId	movieId	rating
0	1	1	4.0
1	1	3	4.0
2	1	6	4.0
3	1	47	5.0
4	1	50	5.0

In [17]:

```
#sorting the data into a sparse matrix

movies_users=df_ratings.pivot(index='movieId', columns='userId',values='rating').fillna(0)

mat_movies_users=csr_matrix(movies_users.values)
```

In [11]:

```
#invoking KNN and passing it various parameteres

model_knn= NearestNeighbors(metric='cosine', algorithm='brute', n_neighbors=20)
```

In [12]:

```
model_knn.fit(mat_movies_users)
```

Out[12]:

```
▾ NearestNeighbors
NearestNeighbors(algorithm='brute', metric='cosine', n_neighbors=20)
```

In [23]:

```
#recommender function

def recommender(movie_name, data,model, n_recommendations ):
    model.fit(data)
    idx=process.extractOne(movie_name, df_movies['title'])[2]

    print('Movie Selected: ',df_movies['title'][idx], 'Index: ',idx)
    print('Here are your recommendations...')
    distances, indices=model.kneighbors(data[idx], n_neighbors=n_recommendations)

    for i in indices:
        print(df_movies['title'][i].where(i!=idx))



recommender('Independence Day', mat_movies_users, model_knn,20)
```

```
Movie Selected:  Independence Day (a.k.a. ID4) (1996) Index:  615
Here are your recommendations...
615                                                    NaN
546                             Mission: Impossible (1996)
418                                   Jurassic Park (1993)
594                                         Twister (1996)
910     Once Upon a Time in the West (C'era una volta ...
507                      Terminator 2: Judgment Day (1991)
224              Star Wars: Episode IV - A New Hope (1977)
0                                         Toy Story (1995)
592                                       Rock, The (1996)
31               Twelve Monkeys (a.k.a. 12 Monkeys) (1995)
938                                      Local Hero (1983)
314                                    Forrest Gump (1994)
897                  Cheech and Chong's Up in Smoke (1978)
97                                       Braveheart (1995)
334                                           Speed (1994)
1182                                           Fall (1997)
1575                                 Outsiders, The (1983)
968                            Arsenic and Old Lace (1944)
509                                          Batman (1989)
337                                       True Lies (1994)
Name: title, dtype: object
```