

Movie Recommendation DTtttttttttt

October 2, 2023

```
[56]: import numpy as np
import pandas as pd
```

```
[57]: dfcred=pd.read_csv('tmdb_5000_credits.csv')
```

```
[58]: dfmov=pd.read_csv('tmdb_5000_movies.csv')
```

```
[59]: dfmov.head(2)
```

```
[59]:      budget                                     genres \
0  237000000 [{"id": 28, "name": "Action"}, {"id": 12, "nam...
1  300000000 [{"id": 12, "name": "Adventure"}, {"id": 14, "...

                                     homepage      id \
0                http://www.avatarmovie.com/  19995
1  http://disney.go.com/disneypictures/pirates/   285

                                     keywords original_language \
0 [{"id": 1463, "name": "culture clash"}, {"id": "...      en
1 [{"id": 270, "name": "ocean"}, {"id": 726, "na...      en

                                     original_title \
0                                     Avatar
1  Pirates of the Caribbean: At World's End

                                     overview popularity \
0  In the 22nd century, a paraplegic Marine is di...  150.437577
1  Captain Barbossa, long believed to be dead, ha...  139.082615

                                     production_companies \
0 [{"name": "Ingenious Film Partners", "id": 289...
1 [{"name": "Walt Disney Pictures", "id": 2}, {""...

                                     production_countries release_date      revenue \
0 [{"iso_3166_1": "US", "name": "United States o...  2009-12-10  2787965087
1 [{"iso_3166_1": "US", "name": "United States o...  2007-05-19   961000000

      runtime                                     spoken_languages      status \
```

```

0    162.0 [{"iso_639_1": "en", "name": "English"}, {"iso... Released
1    169.0 [{"iso_639_1": "en", "name": "English"}] Released

```

```

tagline \
0          Enter the World of Pandora.
1 At the end of the world, the adventure begins.

```

```

title vote_average vote_count
0      Avatar          7.2      11800
1 Pirates of the Caribbean: At World's End  6.9      4500

```

```
[60]: dfcred.head(2)
```

```

[60]: movie_id          title \
0      19995          Avatar
1       285 Pirates of the Caribbean: At World's End

```

```

cast \
0 [{"cast_id": 242, "character": "Jake Sully", "...
1 [{"cast_id": 4, "character": "Captain Jack Spa...

```

```

crew
0 [{"credit_id": "52fe48009251416c750aca23", "de...
1 [{"credit_id": "52fe4232c3a36847f800b579", "de...

```

```
[61]: dfcred.nunique()
```

```

[61]: movie_id    4803
title          4800
cast           4761
crew           4776
dtype: int64

```

```
[62]: dfcred.isna().sum()
```

```

[62]: movie_id    0
title          0
cast           0
crew           0
dtype: int64

```

```
[63]: dfmov.nunique()
```

```

[63]: budget          436
genres            1175
homepage          1691
id                4803
keywords          4222

```

```

original_language      37
original_title         4801
overview               4800
popularity             4802
production_companies   3697
production_countries   469
release_date           3280
revenue                3297
runtime                156
spoken_languages        544
status                  3
tagline                 3944
title                  4800
vote_average            71
vote_count              1609
dtype: int64

```

```
[64]: dfmov.isna().sum()
```

```

[64]: budget          0
      genres          0
      homepage       3091
      id              0
      keywords        0
      original_language  0
      original_title   0
      overview         3
      popularity       0
      production_companies  0
      production_countries  0
      release_date      1
      revenue           0
      runtime           2
      spoken_languages   0
      status            0
      tagline           844
      title             0
      vote_average       0
      vote_count         0
      dtype: int64

```

```
[65]: dfmov = pd.merge(dfmov, dfcred, left_on='id', right_on='movie_id', how='inner')
```

```
[66]: dfmov.head(1)
```

```

[66]:      budget          genres \
0  237000000 [{"id": 28, "name": "Action"}, {"id": 12, "nam...

```

```

        homepage      id \
0  http://www.avatarmovie.com/  19995

        keywords original_language \
0  [{"id": 1463, "name": "culture clash"}, {"id":...      en

        original_title      overview \
0  Avatar  In the 22nd century, a paraplegic Marine is di...

        popularity      production_companies ... \
0  150.437577  [{"name": "Ingenious Film Partners", "id": 289...  ...

        spoken_languages      status \
0  [{"iso_639_1": "en", "name": "English"}, {"iso...  Released

        tagline  title_x  vote_average  vote_count  movie_id \
0  Enter the World of Pandora.  Avatar      7.2      11800      19995

        title_y      cast \
0  Avatar  [{"cast_id": 242, "character": "Jake Sully", "...

        crew
0  [{"credit_id": "52fe48009251416c750aca23", "de...

[1 rows x 24 columns]

```

```
[67]: dfmov.columns
```

```
[67]: Index(['budget', 'genres', 'homepage', 'id', 'keywords', 'original_language',
        'original_title', 'overview', 'popularity', 'production_companies',
        'production_countries', 'release_date', 'revenue', 'runtime',
        'spoken_languages', 'status', 'tagline', 'title_x', 'vote_average',
        'vote_count', 'movie_id', 'title_y', 'cast', 'crew'],
        dtype='object')
```

1 pandas profiling

```
[68]: # import pandas_profiling

# # Generate the pandas profiling report
# profile = dfmov.profile_report(title="Pandas Profiling Report")

# # Save the report as an HTML file
# profile.to_file("dfmov_profile_report.html")
```

2 budget vs popularity revenue

3 spoken language

3.0.1 status vs popularity vs legend me and revenue like b4 release and after releas affect on populaity or voting avg

3.0.2 voting avd vas

```
[69]: dfmov.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4803 entries, 0 to 4802
Data columns (total 24 columns):
#   Column                Non-Null Count  Dtype
---  -
0   budget                4803 non-null   int64
1   genres                4803 non-null   object
2   homepage              1712 non-null   object
3   id                    4803 non-null   int64
4   keywords              4803 non-null   object
5   original_language     4803 non-null   object
6   original_title        4803 non-null   object
7   overview              4800 non-null   object
8   popularity            4803 non-null   float64
9   production_companies  4803 non-null   object
10  production_countries  4803 non-null   object
11  release_date          4802 non-null   object
12  revenue               4803 non-null   int64
13  runtime              4801 non-null   float64
14  spoken_languages      4803 non-null   object
15  status                4803 non-null   object
16  tagline               3959 non-null   object
17  title_x               4803 non-null   object
18  vote_average          4803 non-null   float64
19  vote_count            4803 non-null   int64
20  movie_id              4803 non-null   int64
21  title_y               4803 non-null   object
22  cast                  4803 non-null   object
23  crew                  4803 non-null   object
dtypes: float64(3), int64(5), object(16)
memory usage: 938.1+ KB
```

```
[70]: dfmov.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4803 entries, 0 to 4802
Data columns (total 24 columns):
#   Column                Non-Null Count  Dtype
---
```

```

---  -----
0  budget          4803 non-null  int64
1  genres          4803 non-null  object
2  homepage        1712 non-null  object
3  id              4803 non-null  int64
4  keywords        4803 non-null  object
5  original_language  4803 non-null  object
6  original_title   4803 non-null  object
7  overview        4800 non-null  object
8  popularity      4803 non-null  float64
9  production_companies  4803 non-null  object
10 production_countries  4803 non-null  object
11 release_date     4802 non-null  object
12 revenue          4803 non-null  int64
13 runtime          4801 non-null  float64
14 spoken_languages  4803 non-null  object
15 status           4803 non-null  object
16 tagline          3959 non-null  object
17 title_x          4803 non-null  object
18 vote_average     4803 non-null  float64
19 vote_count       4803 non-null  int64
20 movie_id         4803 non-null  int64
21 title_y          4803 non-null  object
22 cast             4803 non-null  object
23 crew             4803 non-null  object
dtypes: float64(3), int64(5), object(16)
memory usage: 938.1+ KB

```

```

[71]: import pandas as pd

dfmov['release_date'] = pd.to_datetime(dfmov['release_date']).dt.year

```

4 feature selection

```

[72]: l=['genres','id','keywords','title_x','overview','cast','crew']

```

```

[73]: dfmov1=dfmov[l]

```

```

[74]: dfmov1.head(3)

```

```

[74]:
          genres      id \
0  [{"id": 28, "name": "Action"}, {"id": 12, "nam...  19995
1  [{"id": 12, "name": "Adventure"}, {"id": 14, "...   285
2  [{"id": 28, "name": "Action"}, {"id": 12, "nam...  206647

          keywords \
0  [{"id": 1463, "name": "culture clash"}, {"id":...

```

```

1 [{"id": 270, "name": "ocean"}, {"id": 726, "na...
2 [{"id": 470, "name": "spy"}, {"id": 818, "name...

                                title_x \
0                                Avatar
1 Pirates of the Caribbean: At World's End
2                                Spectre

                                overview \
0 In the 22nd century, a paraplegic Marine is di...
1 Captain Barbossa, long believed to be dead, ha...
2 A cryptic message from Bond's past sends him o...

                                cast \
0 [{"cast_id": 242, "character": "Jake Sully", "...
1 [{"cast_id": 4, "character": "Captain Jack Spa...
2 [{"cast_id": 1, "character": "James Bond", "cr...

                                crew
0 [{"credit_id": "52fe48009251416c750aca23", "de...
1 [{"credit_id": "52fe4232c3a36847f800b579", "de...
2 [{"credit_id": "54805967c3a36829b5002c41", "de...

```

5 Data preprocessing

```
[75]: dfmov1.isna().sum()
```

```

[75]: genres      0
      id          0
      keywords    0
      title_x     0
      overview    3
      cast        0
      crew        0
      dtype: int64

```

```
[76]: dfmov1=dfmov1.dropna()
```

```
[77]: dfmov1.isna().sum()
```

```

[77]: genres      0
      id          0
      keywords    0
      title_x     0
      overview    0
      cast        0

```

```
crew          0
dtype: int64
```

```
[78]: dfmov1.duplicated().sum()
```

```
[78]: 0
```

```
[79]: import ast
def helpme(listdict):
    kk=[]
    for i in ast.literal_eval(listdict):
        kk.append(i['name'])
    return kk
```

```
[80]: dfmov1['genres']=dfmov1['genres'].apply(helpme)
```

```
[81]: dfmov1['keywords']=dfmov1['keywords'].apply(helpme)
```

6 getting top 3 cast of each movie

```
[82]: def helpme3(listdict):
    kk=[]
    c=0
    for i in ast.literal_eval(listdict):
        if c<3:
            kk.append(i['name'])
            c+=1
        else: break
    return kk
```

```
[83]: dfmov1['cast']=dfmov1['cast'].apply(helpme3)
```

```
[84]: dfmov1.head(3)
```

```
[84]:
```

	genres	id \
0	[Action, Adventure, Fantasy, Science Fiction]	19995
1	[Adventure, Fantasy, Action]	285
2	[Action, Adventure, Crime]	206647

	keywords \
0	[culture clash, future, space war, space colon...
1	[ocean, drug abuse, exotic island, east india ...
2	[spy, based on novel, secret agent, sequel, mi...

	title_x \
0	Avatar


```

1 Pirates of the Caribbean: At World's End
2 Spectre

                                overview \
0 In the 22nd century, a paraplegic Marine is di...
1 Captain Barbossa, long believed to be dead, ha...
2 A cryptic message from Bond's past sends him o...

                                cast \
0 [Sam Worthington, Zoe Saldana, Sigourney Weaver]
1 [Johnny Depp, Orlando Bloom, Keira Knightley]
2 [Daniel Craig, Christoph Waltz, Léa Seydoux]

                                crew
0 [{"credit_id": "52fe48009251416c750aca23", "de...
1 [{"credit_id": "52fe4232c3a36847f800b579", "de...
2 [{"credit_id": "54805967c3a36829b5002c41", "de...

```

```

[85]: def helpme4(listdict):
      kk=[]
      for i in ast.literal_eval(listdict):
          if i['job']=='Writer' or i['job']=='Director':
              kk.append(i['name'])
      return kk

```

```

[86]: dfmov1['crew']=dfmov1['crew'].apply(helpme4)

```

```

[87]: df=pd.DataFrame()

```

```

[88]: dfmov1.head()

```

```

[88]:
                                genres      id \
0 [Action, Adventure, Fantasy, Science Fiction]  19995
1 [Adventure, Fantasy, Action]  285
2 [Action, Adventure, Crime]  206647
3 [Action, Crime, Drama, Thriller]  49026
4 [Action, Adventure, Science Fiction]  49529

```

```

                                keywords \
0 [culture clash, future, space war, space colon...
1 [ocean, drug abuse, exotic island, east india ...
2 [spy, based on novel, secret agent, sequel, mi...
3 [dc comics, crime fighter, terrorist, secret i...
4 [based on novel, mars, medallion, space travel...

```

```

                                title_x \
0 Avatar

```

```

1 Pirates of the Caribbean: At World's End
2                               Spectre
3                               The Dark Knight Rises
4                               John Carter

```

overview \

```

0 In the 22nd century, a paraplegic Marine is di...
1 Captain Barbossa, long believed to be dead, ha...
2 A cryptic message from Bond's past sends him o...
3 Following the death of District Attorney Harve...
4 John Carter is a war-weary, former military ca...

```

cast \

```

0 [Sam Worthington, Zoe Saldana, Sigourney Weaver]
1 [Johnny Depp, Orlando Bloom, Keira Knightley]
2 [Daniel Craig, Christoph Waltz, Léa Seydoux]
3 [Christian Bale, Michael Caine, Gary Oldman]
4 [Taylor Kitsch, Lynn Collins, Samantha Morton]

```

crew

```

0 [James Cameron, James Cameron]
1 [Gore Verbinski]
2 [Sam Mendes]
3 [Christopher Nolan]
4 [Andrew Stanton]

```

```
[89]: dfmov1['overview'] = dfmov1['overview'].apply(lambda x: x.split())
```

```
[90]: dfmov1.head()
```

```
[90]:
          genres      id \
0 [Action, Adventure, Fantasy, Science Fiction]  19995
1 [Adventure, Fantasy, Action]                285
2 [Action, Adventure, Crime]                 206647
3 [Action, Crime, Drama, Thriller]           49026
4 [Action, Adventure, Science Fiction]        49529

```

keywords \

```

0 [culture clash, future, space war, space colon...
1 [ocean, drug abuse, exotic island, east india ...
2 [spy, based on novel, secret agent, sequel, mi...
3 [dc comics, crime fighter, terrorist, secret i...
4 [based on novel, mars, medallion, space travel...

```

title_x \

```

0 Avatar
1 Pirates of the Caribbean: At World's End

```

```

2                               Spectre
3       The Dark Knight Rises
4                               John Carter

```

```

                                overview \
0 [In, the, 22nd, century,, a, paraplegic, Marin...
1 [Captain, Barbossa,, long, believed, to, be, d...
2 [A, cryptic, message, from, Bond's, past, send...
3 [Following, the, death, of, District, Attorney...
4 [John, Carter, is, a, war-weary,, former, mili...

```

```

                                cast \
0 [Sam Worthington, Zoe Saldana, Sigourney Weaver]
1 [Johnny Depp, Orlando Bloom, Keira Knightley]
2 [Daniel Craig, Christoph Waltz, Léa Seydoux]
3 [Christian Bale, Michael Caine, Gary Oldman]
4 [Taylor Kitsch, Lynn Collins, Samantha Morton]

```

```

                                crew
0 [James Cameron, James Cameron]
1 [Gore Verbinski]
2 [Sam Mendes]
3 [Christopher Nolan]
4 [Andrew Stanton]

```

```

[91]: dfmov1['genres']=dfmov1['genres'].apply(lambda x:[i.replace(" ","") for i in x])
dfmov1['keywords']=dfmov1['keywords'].apply(lambda x:[i.replace(" ","") for i_
↳in x])
dfmov1['cast']=dfmov1['cast'].apply(lambda x:[i.replace(" ","") for i in x])
dfmov1['crew']=dfmov1['crew'].apply(lambda x:[i.replace(" ","") for i in x])
dfmov1['overview']=dfmov1['overview'].apply(lambda x:[i.replace(" ","") for i_
↳in x])

```

```

[92]: dfmov1.head(1)

```

```

[92]:                                     genres      id \
0 [Action, Adventure, Fantasy, ScienceFiction]  19995

                                keywords title_x \
0 [cultureclash, future, spacewar, spacecolony, ...  Avatar

                                overview \
0 [In, the, 22nd, century,, a, paraplegic, Marin...

                                cast                                crew
0 [SamWorthington, ZoeSaldana, SigourneyWeaver] [JamesCameron, JamesCameron]

```

```
[93]: dfmov1['decision']=dfmov1['genres']+dfmov1['keywords']+dfmov1['overview']+dfmov1['cast']+dfmov1['cast2']
dfmov11=dfmov1[['title_x','id','decision']]
```

```
[94]: dfmov11
```

```
[94]:
```

	title_x	id	\
0	Avatar	19995	
1	Pirates of the Caribbean: At World's End	285	
2	Spectre	206647	
3	The Dark Knight Rises	49026	
4	John Carter	49529	
...	
4798	El Mariachi	9367	
4799	Newlyweds	72766	
4800	Signed, Sealed, Delivered	231617	
4801	Shanghai Calling	126186	
4802	My Date with Drew	25975	

	decision
0	[Action, Adventure, Fantasy, ScienceFiction, c...
1	[Adventure, Fantasy, Action, ocean, drugabuse,...
2	[Action, Adventure, Crime, spy, basedonnovel, ...
3	[Action, Crime, Drama, Thriller, dccomics, cri...
4	[Action, Adventure, ScienceFiction, basedonnov...
...	...
4798	[Action, Crime, Thriller, unitedstates-mexicob...
4799	[Comedy, Romance, A, newlywed, couple's, honey...
4800	[Comedy, Drama, Romance, TVMovie, date, loveat...
4801	[When, ambitious, New, York, attorney, Sam, is...
4802	[Documentary, obsession, camcorder, crush, dre...

[4800 rows x 3 columns]

```
[95]: dfmov11['decision']=dfmov11['decision'].apply(lambda x:" ".join(x))
```

C:\Users\Kundan Mourya\AppData\Local\Temp\ipykernel_11184\58274346.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
dfmov11['decision']=dfmov11['decision'].apply(lambda x:" ".join(x))

```
[96]: dfmov11
```

```
[96]:
```

	title_x	id	\
0	Avatar	19995	

```

1    Pirates of the Caribbean: At World's End      285
2                                           Spectre  206647
3                      The Dark Knight Rises      49026
4                      John Carter                49529
...
4798                      El Mariachi            9367
4799                      Newlyweds              72766
4800                      Signed, Sealed, Delivered 231617
4801                      Shanghai Calling         126186
4802                      My Date with Drew        25975

```

```

                                decision
0    Action Adventure Fantasy ScienceFiction cultur...
1    Adventure Fantasy Action ocean drugabuse exoti...
2    Action Adventure Crime spy basedonnovel secret...
3    Action Crime Drama Thriller dccomics crimefigh...
4    Action Adventure ScienceFiction basedonnovel m...
...
4798 Action Crime Thriller unitedstates-mexicobarri...
4799 Comedy Romance A newlywed couple's honeymoon i...
4800 Comedy Drama Romance TVMovie date loveatfirsts...
4801 When ambitious New York attorney Sam is sent t...
4802 Documentary obsession camcorder crush dreamgir...

```

[4800 rows x 3 columns]

```
[97]: dfmov11.to_csv('radkeee.csv', index=False)
```

```
[98]: dfmov11['decision']=dfmov11['decision'].apply(lambda x:x.lower())
```

C:\Users\Kundan Mourya\AppData\Local\Temp\ipykernel_11184\692145296.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
dfmov11['decision']=dfmov11['decision'].apply(lambda x:x.lower())

```
[99]: dfmov11
```

```

[99]:          title_x      id \
0          Avatar    19995
1  Pirates of the Caribbean: At World's End      285
2          Spectre   206647
3  The Dark Knight Rises   49026
4          John Carter   49529
...

```

4798	El Mariachi	9367
4799	Newlyweds	72766
4800	Signed, Sealed, Delivered	231617
4801	Shanghai Calling	126186
4802	My Date with Drew	25975

```

                                decision
0    action adventure fantasy sciencefiction cultur...
1    adventure fantasy action ocean drugabuse exoti...
2    action adventure crime spy basedonnovel secret...
3    action crime drama thriller dcomics crimefigh...
4    action adventure sciencefiction basedonnovel m...
...
4798 action crime thriller unitedstates-mexicobarri...
4799 comedy romance a newlywed couple's honeymoon i...
4800 comedy drama romance tvmovie date loveatfirsts...
4801 when ambitious new york attorney sam is sent t...
4802 documentary obsession camcorder crush dreamgir...

```

[4800 rows x 3 columns]

```
[100]: from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer(max_features=4000,stop_words='english')
vec=cv.fit_transform(dfmov1['decision']).toarray()
```

```
[101]: vec
```

```
[101]: 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]], dtype=int64)
```

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

```
[103]: def stem(gg):
        l=[]
        for i in gg.split():
            print
```

```
[104]: def stem(gg):
        l=[]
        for i in gg.split():
```

```

        l.append(ps.stem(i))
    return " ".join(l)
    #return l
dfmov11['decision']=dfmov11['decision'].apply(stem)

```

C:\Users\Kundan Mourya\AppData\Local\Temp\ipykernel_11184\3786805350.py:7:

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

```
dfmov11['decision']=dfmov11['decision'].apply(stem)
```

[105]: dfmov11

```

[105]:
          title_x      id \
0          Avatar   19995
1  Pirates of the Caribbean: At World's End    285
2          Spectre  206647
3  The Dark Knight Rises   49026
4      John Carter   49529
...
4798      El Mariachi    9367
4799      Newlyweds   72766
4800  Signed, Sealed, Delivered  231617
4801  Shanghai Calling  126186
4802  My Date with Drew   25975

```

```

          decision
0  action adventur fantasi sciencefict culturecla...
1  adventur fantasi action ocean drugabus exotici...
2  action adventur crime spi basedonnovel secreta...
3  action crime drama thriller dccomic crimefight...
4  action adventur sciencefict basedonnovel mar m...
...
4798  action crime thriller unitedstates-mexicobarri...
4799  comedi romanc a newlyw couple' honeymoon is up...
4800  comedi drama romanc tvmovi date loveatfirstsig...
4801  when ambiti new york attorney sam is sent to s...
4802  documentari obsess camcord crush dreamgirl eve...

```

[4800 rows x 3 columns]

[106]: dfmov11.to_csv('radkeee11.csv', index=False)

```
[107]: from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer(max_features=4000,stop_words='english')
vec=cv.fit_transform(dfmov11['decision']).toarray()
```

```
[108]: vec
```

```
[108]: 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]], dtype=int64)
```

```
[109]: dfmov11.shape
```

```
[109]: (4800, 3)
```

```
[110]: from sklearn.metrics.pairwise import cosine_similarity
```

```
[111]: ss=cosine_similarity(vec)
```

```
[112]: ss
```

```
[112]: array([[1.          , 0.08492078, 0.0877058 , ..., 0.0457509 , 0.
          0.          ],
          [0.08492078, 1.          , 0.06454972, ..., 0.02525381, 0.
          0.02727724],
          [0.0877058 , 0.06454972, 1.          , ..., 0.02608203, 0.
          0.          ],
          ...,
          [0.0457509 , 0.02525381, 0.02608203, ..., 1.          , 0.04307305,
          0.04408667],
          [0.          , 0.          , 0.          , ..., 0.04307305, 1.
          0.09304842],
          [0.          , 0.02727724, 0.          , ..., 0.04408667, 0.09304842,
          1.          ]])
```

```
[113]: def recommend(movie):
        indexx=dfmov11[dfmov11['title_x']==movie].index[0]
        #arraymovie=ss[indexx]
        movielist=sorted(list(enumerate(ss[indexx])),reverse=True,key=lambda x:
        ↪x[1])[1:6]
        for i in movielist:
            print(dfmov11.iloc[i[0]].title_x)
```

```
[114]: recommend('Batman Begins')
```


The Dark Knight
Batman
10th & Wolf
The Dark Knight Rises
Rockaway

```
[115]: recommend('Avatar')
```

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

```
[ ]:
```