

# Movie Recommender System

Dataset source: [https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata?select=tmdb\\_5000\\_movies.csv](https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata?select=tmdb_5000_movies.csv)

In [1]:

```
#Importing all required libraries
import numpy as np
import pandas as pd
import ast
from sklearn.feature_extraction.text import CountVectorizer
```

In [2]:

```
movies = pd.read_csv("tmdb_5000_movies.csv")
credits = pd.read_csv("tmdb_5000_credits.csv")
movies.head(1)
# credits.head(1)
```

Out[2]:

	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	production_companies	pi
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "name..."]	http://www.avatarmovie.com/	1995	[{"id": 1463, "name": "culture clash"}, {"id": ...]	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577	[{"name": "Ingenious Film Partners", "id": 289...]	

In [3]:

```
#We firstly need to merge the two datasets for a combined version to extract various fields from the dataset
movies = movies.merge(credits, on="title") #merging the contents on the bases of title
movies.head(1)
```

Out[3]:

	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	production_companies	...
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "name..."]	http://www.avatarmovie.com/	1995	[{"id": 1463, "name": "culture clash"}, {"id": ...]	en	Avatar	In the 22nd century, a paraplegic Marine is	150.437577	[{"name": "Ingenious Film Partners", "id": 289...]	

budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	production_companies	...
			{"id":...						di...	



```
In [4]: #Removing unwanted Fields from the dataset to simply data to processes Later  
# fields required are: genres , id , keywords , original_title , overview , crew , cast  
  
movies = movies[['genres','id', 'keywords','original_title','overview','cast','crew']]  
# movies.info
```

```
In [5]: # preprocessing of data  
def genres(obj):  
    gen = []  
    for i in ast.literal_eval(obj):  
        gen.append(i['name'])  
    return gen  
  
# movies['genres'] = movies['genres'].apply(genres)
```

```
In [6]: movies.dropna(inplace=True)
```

```
In [7]: # movies.head(1)  
# similarly we repeat the same process for keywords, cast and crew  
movies['keywords'].head(1).values  
  
def keywords(obj):  
    gen = []  
    for i in ast.literal_eval(obj):  
        gen.append(i['name'])  
    return gen  
  
movies['keywords']=movies['keywords'].apply(keywords)
```

```
In [8]: movies.head(1)
```

```
Out[8]:
```

	genres	id	keywords	original_title	overview	cast	crew
0	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Thriller"}]	19995	culture clash, future, space war, space	Avatar	In the 22nd century, a paraplegic Marine is di...	[{"cast_id": 242, "character": "Jake Sully"}, {"cast_id": 243, "character": "Dr. Evelyn Cabe...	[{"credit_id": "52fe48009251416c750aca23", "de...

	genres	id	keywords	original_title	overview	cast	crew
--	--------	----	----------	----------------	----------	------	------

```
In [9]:  
movies['cast'][0]  
def cast(obj):  
    cnt = 0  
    cast = []  
    for i in ast.literal_eval(obj):  
        if cnt != 3:  
            cast.append(i['name'])  
            cnt += 1  
        else:  
            break  
  
    return cast  
  
# movies['cast'].apply(cast)  
movies['cast'] = movies['cast'].apply(cast)  
# print('.')  
In [10]: movies.head(1)
```

	genres	id	keywords	original_title	overview	cast	crew
0	[{"id": 28, "name": "Action"}, {"id": 12, "name": "Sci-Fi"}, {"id": 14, "name": "Thriller"}]	19995	culture clash, future, space war, space colon...	Avatar	In the 22nd century, a paraplegic Marine is di...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[{"credit_id": "52fe48009251416c750aca23", "de..."}]

In [11]:

```

movies['crew'][0]
def director(obj):
    director = []
    for i in ast.literal_eval(obj):
        if i['job'] == "Director":
            director.append(i['name'])
            break
    return director

# movies['crew'].apply(director)
movies['crew'] = movies['crew'].apply(director)

```

In [12]:

```
movies.head(1)
```

Out[12]:

	genres	id	keywords	original_title	overview	cast	crew
0	[{"id": 28, "name": "Action"}, {"id": 12, "name": ...]	19995	culture clash, future, space war, space colon...	Avatar	In the 22nd century, a paraplegic Marine is di...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[James Cameron]

In [13]:

```

# movies['overview'][0]
# movies['overview'].apply(lambda j:j.split())
movies.head()

```

Out[13]:

	genres	id	keywords	original_title	overview	cast	crew
0	[{"id": 28, "name": "Action"}, {"id": 12, "name": ...]	19995	culture clash, future, space war, space colon...	Avatar	In the 22nd century, a paraplegic Marine is di...	[Sam Worthington, Zoe Saldana, Sigourney Weaver]	[James Cameron]
1	[{"id": 12, "name": "Adventure"}, {"id": 14, "name": ...]	285	[ocean, drug abuse, exotic island, east india ...]	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	[Johnny Depp, Orlando Bloom, Keira Knightley]	[Gore Verbinski]
2	[{"id": 28, "name": "Action"}, {"id": 12, "name": ...]	206647	[spy, based on novel, secret agent, sequel, mi...]	Spectre	A cryptic message from Bond's past sends him o...	[Daniel Craig, Christoph Waltz, Léa Seydoux]	[Sam Mendes]
3	[{"id": 28, "name": "Action"}, {"id": 80, "name": ...]	49026	[dc comics, crime fighter, terrorist, secret]	The Dark Knight Rises	Following the death of District Attorney Harve...	[Christian Bale, Michael Caine, Gary Oldman]	[Christopher Nolan]

	genres	id	keywords	original_title	overview	cast	crew
4	["name": "nam...", "id": 28, "name": "Action"], {"id": 12, "name": "sci-fi", "name": "adventure"}]	49529	[based on novel, mars, war, space travel]	John Carter	John Carter is a war-weary, former military	[Taylor Kitsch, Lynn Collins, Samantha Morton]	[Andrew Stanton]

```
In [14]: movies['overview'][0]
```

```
Out[14]: '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 [15]: movies.dropna(inplace=True)  
# movies['overview'].apply(lambda x:x.split())
```

```
In [16]: movies['overview']= movies['overview'].apply(lambda x:x.split())
```

```
In [17]: #Fixing any issues due to spacing between any two attributes  
#eg: sam harrington would turn into 2 different tags --> 'sam', 'harrington'  
movies['genres'] = movies['genres'].apply(lambda x:[i.replace(" ", "") for i in x])  
movies['keywords'] = movies['keywords'].apply(lambda x:[i.replace(" ", "") for i in x])  
movies['cast'] = movies['cast'].apply(lambda x:[i.replace(" ", "") for i in x])  
movies['crew'] = movies['crew'].apply(lambda x:[i.replace(" ", "") for i in x])  
  
movies.head(1)
```

```
In [18]: #creating a combined fields
movies['tagz'] = movies['genres'] + movies['keywords'] + movies['overview'] + movies['cast'] + movies['crew']
movies.head(1)
```

```
Out[18]: genres    id    keywords    original_title    overview    cast    crew    tagz
```

```
In [20]: # updated_movies['tagz'].apply(lambda x: ' '.join(x))  
# !pip install nltk  
import nltk
```

```
In [21]: from nltk.stem.porter import PorterStemmer  
ps = PorterStemmer()
```

```
def stem(txt):
    l = []
    for i in txt.split():
        l.append(ps.stem(i))
    strr = " ".join(l)
    return strr

# updted_movies['tagz'].head(1).values
# updted_movies['tagz'].apply(stem)
# updted_movies['tagz'] = updted_movies['tagz'].apply(stem)
```

In [22]:

```
# updted_movies['tagz'][0]
# df.columns= df.columns.str.lower()
# updted_movies['tagz'].apply(lambda x:x.lower())      #facing error due to x.lower instead we implement a short

def low(obj):
    l = []
    for i in obj:
        l.append(i.lower())
    return l

# print(low(updted_movies['tagz'][0]))

# updted_movies['tagz'] = updted_movies['tagz'].apply(low)
updted_movies['tagz'] = updted_movies['tagz'].apply(lambda x: " ".join(x))
# new.head()
updted_movies['tagz'][1]
```

C:\Users\lbs19\AppData\Local\Temp\ipykernel\_17504\713437769.py:14: 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](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
updted_movies['tagz'] = updted_movies['tagz'].apply(lambda x: " ".join(x))
```

Out[22]:  
[ { "id" : 12 , "name" : "Adventure" } , { "id" : 14 , "name" : "Fantasy" } , { "id" : 28 , "name" : "Action" } ] ocean drugabuse exoticisland eastindiatradingcompany loveofone'slife traitor s hipwreck strongwoman ship alliance calypso afterlife fighter pirate swashbuckler aftercreditsstinger Captain Barbossa, long believed to be dead, has come back to life and is headed to the edge of the Earth with Will Turner and Elizabeth Swann. But nothing is quite as it seems. JohnnyDepp OrlandoBloom KeiraKnightley GoreVerbinski'

In [23]:

```
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features=5000, stop_words='english')
# max features denotes the maximum number of words
```

In [24]:

```
vectors = cv.fit_transform(updted_movies['tagz']).toarray()
```

In [25]:

```
vectors
```

```
Out[25]: 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)
```

```
In [26]: cv.get_feature_names()  
  
#issues which the feature names ---> data is being repeated in different forms  
# eg: action, actions  
# where the base word is action  
  
# We will performing a operation to convert these terms into a single word
```

```
Out[26]: ['000',  
          '007',  
          '10',  
          '100',  
          '11',  
          '12',  
          '13',  
          '14',  
          '15',  
          '16',  
          '17',  
          '18',  
          '18th',  
          '19',  
          '1930s',  
          '1940s',  
          '1944',  
          '1950',  
          '1950s',  
          '1960s',  
          '1970s',  
          '1971',  
          '1974',  
          '1976',  
          '1980',  
          '1980s',
```

'1985',  
'1990s',  
'19th',  
'19thcentury',  
'20',  
'200',  
'2009',  
'20th',  
'21st',  
'23',  
'24',  
'25',  
'30',  
'300',  
'3d',  
'40',  
'50',  
'500',  
'60',  
'60s',  
'70',  
'70s',  
'aaron',  
'aaroneckhart',  
'abandoned',  
'abducted',  
'abigailbreslin',  
'abilities',  
'ability',  
'able',  
'aboard',  
'abuse',  
'abusive',  
'academic',  
'academy',  
'accept',  
'accepted',  
'accepts',  
'access',  
'accident',  
'accidental',  
'accidentally',

'accompanied',  
'accomplish',  
'account',  
'accountant',  
'accused',  
'ace',  
'achieve',  
'act',  
'acting',  
'action',  
'actionhero',  
'actions',  
'activist',  
'activities',  
'activity',  
'actor',  
'actors',  
'actress',  
'acts',  
'actual',  
'actually',  
'adam',  
'adams',  
'adamsandler',  
'adamshankman',  
'adaptation',  
'adapted',  
'addict',  
'addicted',  
'addiction',  
'adolescence',  
'adopt',  
'adopted',  
'adoption',  
'adopts',  
'adrienbrody',  
'adult',  
'adultery',  
'adulthood',  
'adults',  
'advantage',  
'adventure',

```
'adventures',
'advertising',
'advice',
'affair',
'affairs',
'affection',
'affections',
'affghanistan',
'africa',
'african',
'africanamerican',
'aftercreditsstinger',
'afterlife',
'aftermath',
'age',
'aged',
'agedifference',
'agency',
'agenda',
'agent',
'agents',
'aggressive',
'aging',
'ago',
'agree',
'agrees',
'ahead',
'aid',
'aided',
'aids',
'ailing',
'air',
'airplane',
'airplanecrash',
'airport',
'aka',
'al',
'alabama',
'alan',
'alaska',
'albert',
'alcohol',
```

'alcoholic',  
'alcoholism',  
'alecbaldwin',  
'alex',  
'alexkendrick',  
'alfredhitchcock',  
'ali',  
'alice',  
'alien',  
'alieninvasion',  
'alienlife',  
'aliens',  
'alike',  
'alive',  
'allen',  
'alliance',  
'allied',  
'allies',  
'allow',  
'allowing',  
'allows',  
'ally',  
'alongside',  
'alpacino',  
'alter',  
'alternate',  
'alternative',  
'alzheimer',  
'amanda',  
'amandapeet',  
'amandaseyfried',  
'amateur',  
'amazing',  
'ambassador',  
'ambition',  
'ambitious',  
'ambulance',  
'ambush',  
'america',  
'american',  
'americanabroad',  
'americanfootball',

'americans',  
'amid',  
'amidst',  
'amnesia',  
'amp',  
'amsterdam',  
'amusement',  
'amusementpark',  
'amy',  
'amyadams',  
'amysmart',  
'analyst',  
'anarchiccomedy',  
'ancient',  
'ancientrome',  
'ancientworld',  
'anderson',  
'andiemacdowell',  
'andrew',  
'android',  
'andy',  
'andygarcía',  
'angel',  
'angelabassett',  
'angeles',  
'angelinajolie',  
'angels',  
'anger',  
'anglee',  
'angry',  
'animal',  
'animalattack',  
'animalhorror',  
'animals',  
'animated',  
'animation',  
'anna',  
'annafaris',  
'anne',  
'annehathaway',  
'annemoss',  
'annettebening',

'annie',  
'anniversary',  
'annual',  
'answer',  
'answers',  
'ant',  
'anthology',  
'anthony',  
'anthonyanderson',  
'anthonyhopkins',  
'anthropomorphism',  
'anti',  
'antics',  
'antihero',  
'antoinefuqua',  
'antoniobanderas',  
'antonyelchin',  
'apart',  
'apartheid',  
'apartment',  
'ape',  
'apes',  
'apocalypse',  
'apocalyptic',  
'apparent',  
'apparently',  
'appear',  
'appears',  
'apple',  
'appointed',  
'apprentice',  
'approach',  
'approaches',  
'approaching',  
'april',  
'aquarium',  
'arab',  
'arch',  
'archaeologist',  
'archeology',  
'architect',  
'arctic',

```
'area',
'aren',
'arena',
'argument',
'arise',
'aristocrat',
'armed',
'arms',
'army',
'arnold',
'arnoldschwarzenegger',
'arrangedmarriage',
'arrangement',
'arrest',
'arrested',
'arrival',
'arrive',
'arrives',
'arriving',
'arrogant',
'art',
'arthur',
'artificialintelligence',
'artist',
'artistic',
'artists',
'arts',
'ashley',
'ashleyjudd',
'ashtonkutcher',
'asia',
'aside',
'ask',
'asked',
'asking',
'asks',
'aspirations',
'aspiring',
'assassin',
'assassinate',
'assassination',
'assassins',
```

```
'assault',
'assigned',
'assignment',
'assistant',
'assumes',
'asteroid',
'astronaut',
'astronauts',
'asylum',
'athlete',
'atomicbomb',
'attack',
'attacked',
'attacks',
'attempt',
'attempting',
'attempts',
'attempttoescape',
'attending',
'attends',
'attention',
'attic',
'attitude',
'attorney',
'attracted',
'attraction',
'attractive',
'audience',
'audiences',
'audition',
'august',
'aunt',
'austin',
'australia',
'australian',
'author',
'authorities',
'authority',
'autism',
'auto',
'avenge',
'average',
```

'avoid',  
'awaits',  
'awakens',  
'award',  
'away',  
'awry',  
'ax',  
'babe',  
'baby',  
'bachelor',  
'backdrop',  
'background',  
'backgrounds',  
'bad',  
'bag',  
'bahamas',  
'bail',  
'balance',  
'ball',  
'ballet',  
'baltimore',  
'band',  
'bandits',  
'bangkok',  
'banished',  
'bank',  
'banker',  
'bankrobber',  
'bankrobbery',  
'bar',  
'barely',  
'bargained',  
'barn',  
'barney',  
'barry',  
'barrylevinson',  
'bars',  
'base',  
'baseball',  
'based',  
'basedoncomicbook',  
'basedongraphicnovel',

```
'basedonnovel',
'basedonplay',
'basedonstagemusical',
'basedontrueevents',
'basedontruestory',
'basedontvseries',
'basedonvideogame',
'basedonyoungadultnovel',
'basement',
'basketball',
'batman',
'battle',
'battlefield',
'battles',
'battling',
'bay',
'beach',
'bear',
'bears',
'beast',
'beasts',
'beat',
'beating',
'beautiful',
'beautifulwoman',
'beauty',
'becky',
'becominganadult',
'bed',
'bedroom',
'bee',
'beer',
'befriends',
'began',
'begin',
'beginning',
'begins',
'behavior',
'beings',
'belief',
'beliefs',
'believe',
```

'believed',  
'believes',  
'believing',  
'bell',  
'belong',  
'belongs',  
'beloved',  
'ben',  
'benaffleck',  
'beneath',  
'benfoster',  
'beniciodeltoro',  
'benjamin',  
'benjaminbratt',  
'benkingsley',  
'bennett',  
'benstiller',  
'bent',  
'berlin',  
'best',  
'bestfriend',  
'bet',  
'beth',  
'betrayal',  
'betrayed',  
'bettemidler',  
'better',  
'betty',  
'beverly',  
'bible',  
'bid',  
'big',  
'bigger',  
'biggest',  
'biker',  
'bikini',  
'billhader',  
'billionaire',  
'billmurray',  
'billnighy',  
'billpaxton',  
'billpullman',

'billy',  
'billybobthornton',  
'billycrudup',  
'billycrystal',  
'biography',  
'bird',  
'birth',  
'birthday',  
'bisexual',  
'bishop',  
'bit',  
'bite',  
'bitter',  
'bizarre',  
'black',  
'blackmagic',  
'blackmail',  
'blackpeople',  
'blacksmith',  
'blade',  
'blame',  
'blind',  
'bliss',  
'block',  
'blonde',  
'blood',  
'bloodsplatter',  
'bloodthirsty',  
'bloody',  
'blow',  
'blue',  
'board',  
'boarding',  
'boardingschool',  
'boat',  
'bob',  
'bobby',  
'bobbyfarrelly',  
'bobhoskins',  
'bodies',  
'body',  
'bodyguard',

'bold',  
'bollywood',  
'bomb',  
'bombing',  
'bond',  
'bonds',  
'bone',  
'book',  
'books',  
'border',  
'bored',  
'boredom',  
'boring',  
'born',  
'boss',  
'boston',  
'botched',  
'bound',  
'boundaries',  
'bounty',  
'bountyhunter',  
'bourne',  
'box',  
'boxer',  
'boxing',  
'boy',  
'boyfriend',  
'boys',  
'bradleycooper',  
'bradpitt',  
'brain',  
'brand',  
'brave',  
'bravery',  
'brazil',  
'brazilian',  
'break',  
'breakdown',  
'breaking',  
'breaks',  
'brendanfraser',  
'brendangleeson',

```
'brent',
'brettratner',
'brian',
'briandepalma',
'bride',
'bridge',
'brief',
'brien',
'bright',
'brilliant',
'bring',
'bringing',
'brings',
'brink',
'britain',
'british',
'britishsecretservice',
'brittanymurphy',
'broadway',
'broke',
'broken',
'broker',
'brooklyn',
'brooks',
'brothel',
'brother',
'brotherbrotherrelationship',
'brothers',
'brothersisterrelationship',
'brought',
'brown',
'bruce',
'brucewillis',
'brutal',
'brutality',
'brutally',
'bryansinger',
'buck',
'buddies',
'buddy',
'buddycomedy',
'budget',
```

```
'build',
'building',
'built',
'bully',
'bullying',
'bumbling',
'bunny',
'burglar',
'buried',
'burned',
'bus',
'bush',
'business',
'businessman',
'bust',
'busy',
'butler',
'buy',
'cabin',
'caesar',
'cage',
'cairo',
'cal',
'california',
'called',
'calls',
'calvin',
'camcorder',
'came',
'camera',
'cameraman',
'cameras',
'camerondiaz',
'camp',
'campaign',
'campbell',
'camping',
'campus',
'canada',
'canadian',
'cancer',
'candidate',
```

'candy',  
'cannibal',  
'capable',  
'capital',  
'capitalism',  
'capt',  
'captain',  
'captive',  
'capture',  
'captured',  
'captures',  
'car',  
'caraccident',  
'carchase',  
'carcrash',  
'card',  
'care',  
'career',  
'carefree',  
'caretaker',  
'caribbean',  
'carjourney',  
'carl',  
'carlagugino',  
'carmen',  
'carol',  
'carolina',  
'carrace',  
'carrie',  
'carry',  
'carrying',  
'cars',  
'cartel',  
'carter',  
'cartoon',  
'caryelwes',  
'case',  
'caseyaffleck',  
'cash',  
'casino',  
'cast',  
'castle',

```
'cat',
'cataclysm',
'catastrophe',
'catch',
'catches',
'cateblanchett',
'catherinedeneuve',
'catherinekeener',
'catherinezeta',
'catholic',
'catholicism',
'cattle',
'caught',
'cause',
'caused',
'causes',
'causing',
'cavalry',
'cave',
'cavemen',
'celebrate',
'celebrated',
'celebration',
'celebrity',
'cell',
'cellphone',
'cemetary',
'center',
'centered',
'centers',
'central',
'centuries',
'century',
'ceo',
'ceremony',
'certain',
'chad',
'chain',
'chainsaw',
'challenge',
'challenged',
'challenges',
```

```
'champion',
'championship',
'chance',
'change',
'changed',
'changes',
'changing',
'channingtatum',
'chaos',
'chaotic',
'chapter',
'character',
'characters',
'charge',
'charged',
'charismatic',
'charles',
'charlie',
'charliesheen',
'charlizetheron',
'charlotte',
'charm',
'charming',
'chase',
'chased',
'chauffeur',
'cheating',
'cheerleader',
'chef',
'chemical',
'cher',
'chicago',
'chicken',
'chief',
'child',
'childabuse',
'childhero',
'childhood',
'childprodigy',
'children',
'chilling',
'china',
```

```
'chinese',
'chip',
'chiwetelejiofor',
'chloe',
'chloëgracemoretz',
'chloësevigny',
'chocolate',
'choice',
'choices',
'choose',
'chosen',
'chris',
'chriscolumbus',
'chriscooper',
'chrisevans',
'chrishemsworth',
'chrisklein',
'chrispine',
'chrisrock',
'christ',
'christian',
'christianbale',
'christianity',
'christianslater',
'christinaapplegate',
'christinaricci',
'christine',
'christmas',
'christmasparty',
'christmastree',
'christopher',
'christopherlloyd',
'christophernolan',
'christopherplummer',
'christopherwalken',
'christophwaltz',
'chrisweitz',
'chronicle',
'chronicles',
'chuck',
'church',
'cia',
```

```
'ciaránhinds',
'cigarettesmoking',
'cillianmurphy',
'cindy',
'cinema',
'circle',
'circuit',
'circumstances',
'circus',
'cities',
'citizens',
'city',
'civil',
'civilization',
'civilwar',
'claim',
'claims',
'claire',
'clairedanes',
'clan',
'clark',
'clash',
'class',
'classes',
'classic',
'classmate',
'classmates',
'classroom',
'claudevandamme',
'clay',
'clean',
'clear',
'clerk',
'client',
'clients',
'climate',
'climbing',
'clinteastwood',
'clique',
'cliveowen',
'clock',
'clone',
```

'cloning',  
'close',  
'closed',  
'closer',  
'club',  
'clubs',  
'clue',  
'clueless',  
'clues',  
'clutches',  
'coach',  
'coast',  
'cocaine',  
'cocky',  
'code',  
'cody',  
'coffin',  
'cohen',  
'col',  
'cold',  
'coldwar',  
'cole',  
'colin',  
'colinfarrell',  
'colinfirth',  
'collapse',  
'colleague',  
'colleagues',  
'collect',  
'collection',  
'collector',  
'college',  
'collide',  
'collision',  
'colonel',  
'colony',  
'color',  
'colorado',  
'colorful',  
'coma',  
'combat',  
'combined',

```
'come',
'comeback',
'comedian',
'comedic',
'comedy',
'comes',
'comet',
'comfort',
'comic',
'comics',
'coming',
'comingofage',
'comingout',
'command',
'commander',
'commercial',
'commit',
'commitment',
'committed',
'common',
'communication',
'community',
'companion',
'company',
'compete',
'competing',
'competition',
'complete',
'completely',
'complex',
'complicated',
'complications',
'composer',
'computer',
'computervirus',
'conan',
'concert',
'conclusion',
'condition',
'confession',
'confidence',
'confident',
```

```
'conflict',
'confront',
'confronted',
'confused',
'congress',
'conman',
'connected',
'connection',
'connell',
'connor',
'conquer',
'conscience',
'consequences',
'conservative',
'considered',
'conspiracy',
'constant',
'constantly',
'construction',
'contact',
'contain',
'contemporary',
'contend',
'contest',
'continue',
'continues',
'continuing',
'contract',
'control'.
```

In [27]:

```
# example
ps.stem('dancing')
```

Out[27]:

```
'danc'
```

In [28]:

```
ps.stem("Adventure Fantasy Action ocean drugabuse exoticisland eastindiatradingcompany loveofone slife traitor shipwreck strong
```

Out[28]:

```
'adventure fantasy action ocean drugabuse exoticisland eastindiatradingcompany loveofone slife traitor shipwreck strongwoman shi
p alliance calypso afterlife fighter pirate swashbuckler aftercreditsstinger captain barbossa, long believed to be dead, has com
e back to life and is headed to the edge of the earth with will turner and elizabeth swann. but nothing is quite as it seems. jo
hnnydepp orlandobloom keiraknightley goreverbinski'
```

In [ ]:

```
# Similarities between two movies can be calc by mmeasuring the distance between them in a co ordinate system (in our case 4806x5000) # for the given calc we  
have two methods # 1. Euclidean distance # 2. Cosine distance # For dataset of higher dimension euclidean distance fails (reason --> Curse of Dimensionality) a  
Simple relation can be given by smaller the distance greater the similarity i.e distance = 1/similarites
```

In [29]:

```
from sklearn.metrics.pairwise import cosine_similarity  
similarities = cosine_similarity(vectors)  
# cosine_similarity(vectors).shape
```

In [30]:

```
# sorted(list(enumerate(similarities[0])), reverse=True, key = lambda x:x[1])[1:6]  
# updted_movies.head(0)
```

In [31]:

```
def recomm(movie):  
    movie_index = updted_movies[updted_movies['original_title'] == movie].index[0]  
    # print(movie_index)  
    distances = sorted(list(enumerate(similarities[movie_index])), reverse=True, key = lambda x:x[1])  
    for i in distances[1:6]:  
        print(updted_movies.iloc[i[0]].original_title)
```

In [32]:

```
recomm('Tangled')
```

逃出生天  
Hross í oss  
The Black Stallion  
Malevolence  
Enchanted

In [33]:

```
recomm('Skyfall')
```

Spectre  
Never Say Never Again  
Quantum of Solace  
Octopussy  
Dr. No

```
In [34]: recomm('Iron Man 3')
```

```
Iron Man 2  
Iron Man  
Captain America: Civil War  
Avengers: Age of Ultron  
X-Men
```

```
In [35]: import pickle
```

```
In [36]: pickle.dump(updted_movies,open('movies_list.pkl','wb'))
```

```
In [37]: updted_movies.head(0)
```

```
Out[37]: id original_title tagz
```

```
In [ ]:
```