



(https://praxis.ac.in/data-science-program/?utm_source=Analytics%20India%20Magazine&utm_medium=Banner&utm_campaign=DS-Jan2021).

DEVELOPERS CORNER (HTTPS://ANALYTICSINDIAMAG.COM/CATEGORY/DEVELOPERS_CORNER/)

Singular Value Decomposition (SVD) & Its Application In Recommender System



BY DR. VAIBHAV KUMAR (HTTPS://ANALYTICSINDIAMAG.COM/AUTHOR/VAIBHAV-KUMARANALYTICSINDIAMAG-COM/)
25/03/2020



Download our Mobile App



(<https://play.google.com/store/apps/details?id=com.analyticsindiamag>).



(<https://apps.apple.com/us/app/id1502685162>).

Singular Value Decomposition (SVD), a classical method from linear algebra is getting popular in the field of data science and machine learning. This popularity is because of its application in developing recommender systems. There are a lot of online user-centric applications such as video players, music players, e-commerce applications, etc., where users are recommended with further items to engage with.

Finding and recommending many suitable items that would be liked and selected by users is always a challenge. There are many techniques used for this task and SVD is one of those techniques. This article presents a brief introduction to recommender systems, an introduction to singular value decomposition and its implementation in movie recommendation.



(<https://www.analytixlabs.co.in/>).

What is a Recommender System?

A recommender system is an intelligent system that predicts the rating and preferences of users on products. The primary application of recommender systems is finding a relationship between user and products in order to maximise the user-product engagement. The major application of recommender systems is in suggesting related video or music for generating a playlist for the user when they are engaged with a related item.

A similar application is in the field of e-commerce where customers are recommended with the related products, but this application involves some other techniques such as association rule learning. It is also used to recommend contents based on user behaviours on social media platforms and news websites.

There are two popular approaches used in recommender systems to suggest items to the users:-

- 1. Collaborative Filtering:** The assumption of this approach is that people who have liked an item in the past will also like the same in future. This approach builds a model based on the past behaviour of users. The user behaviour may include previously watched videos, purchased items, given ratings on items. In this way, the model finds an association between the users and the items. The model is then used to predict the item or a rating for the item in which the user may be interested. Singular value decomposition is used as a collaborative filtering approach in recommender systems.
- 2. Content-Based Filtering:** This approach is based on a description of the item and a record of the user's preferences. It employs a sequence of discrete, pre-tagged characteristics of an item in order to recommend additional items with similar properties. This approach is best suited when there is sufficient information available on the items but not on the users. Content-based recommender systems also include the opinion-based recommender system.

Apart from the above two approaches, there are few more approaches to build recommender systems such as multi-criteria recommender systems, risk-aware recommender systems, mobile recommender systems, and hybrid recommender systems (combining collaborative filtering and content-based filtering).

Singular Value Decomposition

The Singular Value Decomposition (SVD), a method from linear algebra that has been generally used as a dimensionality reduction technique in machine learning. SVD is a matrix factorisation technique, which reduces the number of features of a dataset by reducing the space dimension from N-dimension to K-dimension (where $K < N$). In the context of the recommender system, the SVD is used as a collaborative filtering technique. It uses a matrix structure where each row represents a user, and each column represents an item. The elements of this matrix are the ratings that are given to items by users.

Stay Connected

Get the latest updates and relevant offers by sharing your email.

ENTER YOUR EMAIL

SUBSCRIBE NOW

The factorisation of this matrix is done by the singular value decomposition. It finds factors of matrices from the factorisation of a high-level (user-item-rating) matrix. The singular value decomposition is a method of decomposing a matrix into three other matrices as given below:

$$A = USV^T$$

Where A is a $m \times n$ utility matrix, U is a $m \times r$ orthogonal left singular matrix, which represents the relationship between users and latent factors, S is a $r \times r$ diagonal matrix, which describes the strength of each latent factor and V is a $r \times n$ diagonal right singular matrix, which indicates the similarity between items and latent factors. The latent factors here are the characteristics of the items, for example, the genre of the music. The SVD decreases the dimension of the utility matrix A by extracting its latent factors. It maps each user and each item into a r -dimensional latent space. This mapping facilitates a clear representation of relationships between users and items.

Let each item be represented by a vector x_i and each user is represented by a vector y_u . The expected rating by a user on an item \hat{r}_{ui} can be given as:

$$\hat{r}_{ui} = x_i^T y_u$$

Here, \hat{r}_{ui} is a form of factorisation in singular value decomposition. The x_i and y_u can be obtained in a manner that the square error difference between their dot product and the expected rating in the user-item matrix is minimum. It can be expressed as:

$$Min(x, y) = \sum_{(u, i) \in K} (r_{ui} - x_i^T y_u)^2$$

In order to let the model generalise well and not overfit the training data, a regularisation term is added as a penalty to the above formula.

$$Min(x, y) = \sum_{(u, i) \in K} (r_{ui} - x_i^T y_u)^2 + \lambda(\|x_i\|^2 + \|y_u\|^2)$$

In order to reduce the error between the value predicted by the model and the actual value, the algorithm uses a bias term. Let for a user-item pair (u, i) , μ is the average rating of all items, b_i is the average rating of item i minus μ and b_u is the average rating given by user u minus μ , the final equation after adding the regularisation term and bias can be given as:

$$Min(x, y, b_i, b_u) = \sum_{(u, i) \in K} (r_{ui} - x_i^T y_u - \mu - b_i - b_u)^2 + \lambda(\|x_i\|^2 + \|y_u\|^2 + b_i^2 + b_u^2)$$

The above equation is the main component of the algorithm which works for singular value decomposition based recommendation system.

Singular Value Decomposition (SVD) based Movie Recommendation

Below is an implementation of singular value decomposition (SVD) based on collaborative filtering in the task of movie recommendation. This task is implemented in Python. For simplicity, the [MovieLens 1M Dataset](https://grouplens.org/datasets/movielens/1m/) (<https://grouplens.org/datasets/movielens/1m/>) has been used. This dataset has been chosen because it does not require any preprocessing as the main focus of this article is on SVD and recommender systems.

Import the required python libraries:

```
import numpy as np
import pandas as pd
```

Read the dataset from where it is downloaded in the system. It consists of two files 'ratings.dat' and 'movies.dat' which need to be read.

```
data = pd.io.parsers.read_csv('data/ratings.dat',
    names=['user_id', 'movie_id', 'rating', 'time'],
    engine='python', delimiter='::')

movie_data = pd.io.parsers.read_csv('data/movies.dat',
    names=['movie_id', 'title', 'genre'],
    engine='python', delimiter='::')
```

Create the rating matrix with rows as movies and columns as users.

```
ratings_mat = np.ndarray(
    shape=(np.max(data.movie_id.values), np.max(data.user_id.values)),
    dtype=np.uint8)
ratings_mat[data.movie_id.values-1, data.user_id.values-1] = data.rating.values
```

Normalise the matrix.

SEE ALSO

([https://a
nalyticsin
diamag.c
om/how-
to-build-
a-
content-
based-
movie-
recommen
dation-
system-
in-
python/](https://analyticsindiamag.com/how-to-build-a-content-based-movie-recommendation-system-in-python/))

DEVELOPERS CORNER ([HTTPS://ANALYTICSINDIAMAG.COM/CATEGORY/DEVELOPERS_CORNER/](https://analyticsindiamag.com/category/developers_corner/))

[How To Build A Content-Based Movie Recommendation System In Python](https://analyticsindiamag.com/how-to-build-a-content-based-movie-recommendation-system-in-python/) (<https://analyticsindiamag.com/how-to-build-a-content-based-movie-recommendation-system-in-python/>)

```
normalised_mat = ratings_mat - np.asarray([(np.mean(ratings_mat, 1))]).T
```

Compute the Singular Value Decomposition (SVD).

```
A = normalised_mat.T / np.sqrt(ratings_mat.shape[0] - 1)
U, S, V = np.linalg.svd(A)
```

Define a function to calculate the cosine similarity. Sort by most similar and return the top N results.

```
def top_cosine_similarity(data, movie_id, top_n=10):
    index = movie_id - 1 # Movie id starts from 1 in the dataset
    movie_row = data[index, :]
    magnitude = np.sqrt(np.einsum('ij, ij -> i', data, data))
    similarity = np.dot(movie_row, data.T) / (magnitude[index] * magnitude)
    sort_indexes = np.argsort(-similarity)
    return sort_indexes[:top_n]
```

Define a function to print top N similar movies.

```
def print_similar_movies(movie_data, movie_id, top_indexes):
    print('Recommendations for {0}: \n'.format(
        movie_data[movie_data.movie_id == movie_id].title.values[0]))
    for id in top_indexes + 1:
        print(movie_data[movie_data.movie_id == id].title.values[0])
```

Initialise the value of k principal components, id of the movie as given in the dataset, and number of top elements to be printed.

```
k = 50
movie_id = 10 # (getting an id from movies.dat)
top_n = 10
sliced = V.T[:, :k] # representative data
indexes = top_cosine_similarity(sliced, movie_id, top_n)
```

Print the top N similar movies.

```
print_similar_movies(movie_data, movie_id, indexes)
```

Binding all together:

```

#Importing Libraries
import numpy as np
import pandas as pd

#Reading dataset (MovieLens 1M movie ratings dataset: downloaded from https://grouplens
data = pd.io.parsers.read_csv('data/ratings.dat',
    names=['user_id', 'movie_id', 'rating', 'time'],
    engine='python', delimiter=':::')
movie_data = pd.io.parsers.read_csv('data/movies.dat',
    names=['movie_id', 'title', 'genre'],
    engine='python', delimiter=':::')

#Creating the rating matrix (rows as movies, columns as users)
ratings_mat = np.ndarray(
    shape=(np.max(data.movie_id.values), np.max(data.user_id.values)),
    dtype=np.uint8)
ratings_mat[data.movie_id.values-1, data.user_id.values-1] = data.rating.values

#Normalizing the matrix(subtract mean off)
normalised_mat = ratings_mat - np.asarray([(np.mean(ratings_mat, 1))]).T

#Computing the Singular Value Decomposition (SVD)
A = normalised_mat.T / np.sqrt(ratings_mat.shape[0] - 1)
U, S, V = np.linalg.svd(A)

#Function to calculate the cosine similarity (sorting by most similar and returning th
def top_cosine_similarity(data, movie_id, top_n=10):
    index = movie_id - 1 # Movie id starts from 1 in the dataset
    movie_row = data[index, :]
    magnitude = np.sqrt(np.einsum('ij, ij -> i', data, data))
    similarity = np.dot(movie_row, data.T) / (magnitude[index] * magnitude)
    sort_indexes = np.argsort(-similarity)
    return sort_indexes[:top_n]

# Function to print top N similar movies
def print_similar_movies(movie_data, movie_id, top_indexes):
    print('Recommendations for {0}: \n'.format(
        movie_data[movie_data.movie_id == movie_id].title.values[0]))
    for id in top_indexes + 1:
        print(movie_data[movie_data.movie_id == id].title.values[0])

#k-principal components to represent movies, movie_id to find recommendations, top_n p
k = 50
movie_id = 10 # (getting an id from movies.dat)
top_n = 10
sliced = V.T[:, :k] # representative data
indexes = top_cosine_similarity(sliced, movie_id, top_n)

#Printing the top N similar movies
print_similar_movies(movie_data, movie_id, indexes)

```

Some sample outputs are given below:

1 Comment



Can we use SVD result directly as predicted ratings or need to do any modifications?

Facebook Comments Dluain

WHAT'S YOUR REACTION?



EXCITED

O



HAPPY

0



IN LOVE

O



NOT SURE

0



SILLY

0



Vaibhav Kumar has experience in the field of Data Science and Machine Learning, including research and development. He holds a PhD degree in which he has worked in the area of Deep Learning for Stock Market Prediction. He has published/presented more than 15 research papers in international journals and conferences. He has an interest in writing articles related to data science, machine learning and artificial intelligence.

f SHARE

 TWEET

(<https://twitter.com/share?text=Singular%20Value%20Decomposition%20%28SVD%29%20%26%230>).

in(<https://www.linkedin.com/cws/share?url=https://analyticsindiamag.com/singular-value-decomposition-svd-application-recommender->

<https://wa.me/?text=Singular%20Value%20Decomposition%20%28SVD%29%20%26%23038%3B%20Its%20Application%20In%20Reco>
(mailto:?)

✖ subject=Singular%20Value%20Decomposition%20%28SVD%29%20%26%23038%3B%20Its%20Application%20In%20Recommender%20value-decomposition-svd-application-recommender-system/

🔗(<https://t.me/share/url?&text=Singular%20Value%20Decomposition%20%28SVD%29%20%26%23038%3B%20Its%20Application%20In>)

■ (<https://share.flipboard.com/bookmarklet/popout?v=2&title=Singular%20Value%20Decomposition%20%28SVD%29%20%26%23038%3F>)



(<https://www.adasci.org/cds/>).



(https://business.louisville.edu/learnmore/UofLMSBA/?utm_campaign=MSBA&utm_source=analyticsindia&utm_medium=display&utm_keyword=analyticsindia&utm_content=GetPaid)

OUR UPCOMING EVENTS

Full-Day Workshop: Build your own GANs from Scratch | 28th November | [Register here>>](https://www.adasci.org/continuous-learning/workshop-build-your-own-generative-adversarial-networks-gans-from-scratch/) (<https://www.adasci.org/continuous-learning/workshop-build-your-own-generative-adversarial-networks-gans-from-scratch/>).

Webinar On Data Science In The Post-COVID World | 28th November | [Register here>>](https://register.gotowebinar.com/register/4537424314496790287?source=article) (<https://register.gotowebinar.com/register/4537424314496790287?source=article>).

Webinar – Why & How to Automate Your Risk Identification | 9th Dec | [Register here>>](https://register.gotowebinar.com/register/3335244689234991886) (<https://register.gotowebinar.com/register/3335244689234991886>).

RELATED POSTS

DEVELOPERS CORNER
([HTTPS://ANALYTICSINDIAMAG.COM/CATEGORY/DEVELOPERS_CORNER/](https://analyticsindiamag.com/category/developers_corner/))

How To Build A Content-Based Movie Recommendation System In Python (<https://analyticsindiamag.com/how-to-build-a-content-based-movie-recommendation-system-in-python/>)

17/08/2020 · 6 MINS READ

(<https://analyticsindiamag.com/how-to-build-a-content-based-movie-recommendation-system-in-python/>)

Reinforcement Learning For Better Recommender Systems (<https://analyticsindiamag.com/reinforcement-learning-recommender-system-recsim-google-ai/>)

21/11/2019 · 3 MINS READ

(<https://analyticsindiamag.com/reinforcement-learning-recommender-system-recsim-google-ai/>)

DEVELOPERS CORNER
([HTTPS://ANALYTICSINDIAMAG.COM/CATEGORY/DEVELOPERS_CORNER/](https://analyticsindiamag.com/category/developers_corner/))

How To Build Your First Recommender System Using Python & MovieLens Dataset (<https://analyticsindiamag.com/how-to-build-your-first-recommender-system-using-python-movielens-dataset/>)

25/09/2019 · 4 MINS READ

(<https://analyticsindiamag.com/how-to-build-your-first-recommender-system-using-python-movielens-dataset/>)

OPINIONS ([HTTPS://ANALYTICSINDIAMAG.COM/CATEGORY/ARTICLES/](https://analyticsindiamag.com/category/articles/))

With Recommender Systems,

Humans Are Playing A Key Role In Curating & Personalising Content **(<https://analyticsindiamag.com/with-recommender-systems-humans-are-playing-a-key-role-in-curating-personalising-content/>)**

01/09/2019 · 3 MINS READ

[\(https://analyticsindiamag.com/with-recommender-systems-humans-are-playing-a-key-role-in-curating-personalising-content/\)](https://analyticsindiamag.com/with-recommender-systems-humans-are-playing-a-key-role-in-curating-personalising-content/)

DEVELOPERS CORNER
[\(HTTPS://ANALYTICSINDIAMAG.COM/CATEGORY/DEVELOPERS_CORNER/\)](https://analyticsindiamag.com/category/developers_corner/)

Can Singapore Startup Musiio Rival Spotify With Its Push For Personalisation? **(<https://analyticsindiamag.com/can-singapore-startup-musiio-rival-spotify-with-its-push-for-personalisation/>)**

25/06/2019 · 4 MINS READ

[\(https://analyticsindiamag.com/can-singapore-startup-musiio-rival-spotify-with-its-push-for-personalisation/\)](https://analyticsindiamag.com/can-singapore-startup-musiio-rival-spotify-with-its-push-for-personalisation/)

OPINIONS [\(HTTPS://ANALYTICSINDIAMAG.COM/CATEGORY/ARTICLES/\)](https://analyticsindiamag.com/category/articles/)

5 Open-Source Recommender Systems You Should Try For Your Next Project **(<https://analyticsindiamag.com/5-open-source-recommender-systems-you-should-try-for-your-next-project/>)**

05/06/2019 · 3 MINS READ

[\(https://analyticsindiamag.com/5-open-source-recommender-systems-you-should-try-for-your-next-project/\)](https://analyticsindiamag.com/5-open-source-recommender-systems-you-should-try-for-your-next-project/)

CONNECT WITH
US

[About Us](https://analyticsindiamag.com/about/)
[\(https://analyticsindiamag.com/about/\)](https://analyticsindiamag.com/about/)

OUR BRANDS

[MachineHack – ML Hackathons](#)

OUR
CONFERENCES

[Cypher](https://www.analyticsindiasummit.com/)
[\(https://www.analyticsindiasummit.com/\)](https://www.analyticsindiasummit.com/)

OUR VIDEOS

[Documentary – The Transition Cost](#)

BRAND PAGES

[Intel AI Hub](#)
[\(https://analyticsindiamag.com/intel-ai-hub/\)](https://analyticsindiamag.com/intel-ai-hub/)

LISTS

[Academic Rankings](#)

Advertise (https://www.machinehackathon.com/machinecon)	The MachineCon (https://www.youtube.com/on-intel/)	rankings/ (https://www.analyticsindiamag.com/advertise-with-us/)	AIM Research (https://themachinecon.conv=7pvGfJbzTTWk&list=PL9Kc1zSa4,6OzMfx01SJZOGpN_p371vbd)	Top Leaders (https://aimresearch.ai/)	Machine Learning (https://analyticsindiamag.com/write-for-us/)	Web Series – The Dating Scientists (https://www.youtube.com/watch?v=WQbKbIRKQsk&list=PL9Kc1zSa4,6OxZJqOEJa-qI55CtLZxFl2v)	ASSOCIATION OF DATA SCIENTISTS (https://www.analyticsindiamag.com/write-for-us/)	Data Scientists (https://analyticsindiamag.com/write-for-us/)
Write for us (https://analyticsindiamag.com/write-for-us/)	AWARDS (https://aimresearch.ai/)	Developers Summit (http://mlds.analyticsindiasummit.com/)	The Rising (https://rising.analyticsindiasummit.com/)	Chartered Data Scientist (TM) (https://www.adasci.org/cds/)	Plugin (https://plugin.analyticsindiasummit.com/awards/)	Podcasts – Simulated Reality (https://www.youtube.com/watch?v=gVzfVbGE&list=PL9Kc1zSa4,6OxZJqOEJa-qI55CtLZxFl2v)	Lattice – Machine Learning Journal (https://www.adasci.org/cds/)	Emerging Startups (https://analyticsindiamag.com/write-for-us/)
You Say, We Write (https://www.linkedin.in/company/analytics-india-magazine/jobs/)	Analytics100 (https://themachinecon.conv=7pvGfJbzTTWk&list=PL9Kc1zSa4,6OxZJqOEJa-qI55CtLZxFl2v)	4.0 under 4.0 Data Scientists (https://mlds.analyticsindiasummit.com/awards/)	EVENTS (https://rising.analyticsindiasummit.com/awards/)	Best Firms To Work For (https://www.adasci.org/cds/)	Data Science Excellence (https://mlds.analyticsindiasummit.com/awards/)	Analytics India Guru (https://www.youtube.com/watch?v=gVzfVbGE&list=PL9Kc1zSa4,6OxZJqOEJa-qI55CtLZxFl2v)	Continuous Learning (https://www.adasci.org/cds/)	Best Firms To Work For (https://analyticsindiamag.com/write-for-us/)
Contact Us (https://analyticsindiamag.com/contact-us/)	Women in AI Leadership (https://www.analyticsindiasummit.com/awards/)	AIM Fits.com Abouts/awards/ (https://mlds.analyticsindiasummit.com/awards/)	The Pretentious Geek (https://www.youtube.com/watch?v=oQDZMdevzgw&list=PL9Kc1zSa4,6OxZJqOEJa-qI55CtLZxFl2v)	Career Center (https://www.adasci.org/cds/)	Data Science Excellence (https://www.analyticsindiasummit.com/awards/)	Curiosum – AI Storytelling (https://www.youtube.com/watch?v=Z3Z7RiW1G9o&list=PL9Kc1zSa4,6OxZJqOEJa-qI55CtLZxFl2v)	Membership (https://www.adasci.org/cds/)	Trends (https://analyticsindiamag.com/write-for-us/)
MENTORSHIP								
AIM Mentorship Circle (https://analyticsindiamag.com/mentorship-circle/)								
Assisted Mentoring (https://analyticsindiamag.com/mentorship-circle/assisted-mentoring/)								

ABOUT US(HTTPS://ANALYTICSINDIAMAG.COM/ABOUT/)									
ADVERTISE(HTTPS://ANALYTICSINDIAMAG.COM/ADVERTISE-WITH-US/)									
WRITE FOR US(HTTPS://ANALYTICSINDIAMAG.COM/WRITE-FOR-US/)									
COPYRIGHT(HTTPS://ANALYTICSINDIAMAG.COM/COPYRIGHT-TRADEMARKS/)									
PRIVACY(HTTPS://ANALYTICSINDIAMAG.COM/PRIVACY-POLICY/)									
TERMS OF USE(HTTPS://ANALYTICSINDIAMAG.COM/TERMS-USE/)									
CONTACT US(HTTPS://ANALYTICSINDIAMAG.COM/CONTACT-US/)									