

Movie Recommender System

Neha Agarwal(IT - 9), Avik Samanta (IT - 4),
Akshay Kumar Singh(IT - 3)
(Report)

Abstract

In this project, a movie recommender system is built based on the Amazon Review on musical Instrument dataset and Amazon Songs dataset. We used collaborative filtering method and K-Nearest Neighbour to predict user's movie rating and we can recommend rating of musical Instrument to customers, which they potentially give high ratings according to prediction. The root-mean-square error (RMSE) is calculated to carry out evaluation.

Introduction

A recommender system or a recommendation system (sometimes replacing "system" with a synonym such as platform or engine) is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item. They are primarily used in commercial applications, Services like Amazon, Netflix, and YouTube are typical examples of recommender system users. Recommender systems cannot only help the users find their favorite products, but also bring potential profit to online service providers.

DataSet

Songs data:-

In this dataset, we have a large no of user vs Song Id data where, users rated more than one song according to their preference and The SongsRating 2M dataset contains 2000000 ratings for 127771 movies from 200000 users. Each user may or may not have rated the songs. We will find a abstract function that can try to predict

rating for a song which are not rated by user and according to the rating we will recommend those songs to same user.that may like according to their activity of rating

Musical Instrument :- In this dataset, we have a 10261 data, which contains 9 column ['reviewerID' , 'asin' , 'reviewerName' , 'helpful' , 'reviewText' , 'Overall'] where, users rated more than one song according to their preference.

Method:-

Collaborative Filtering:-

There are three steps in the algorithm:

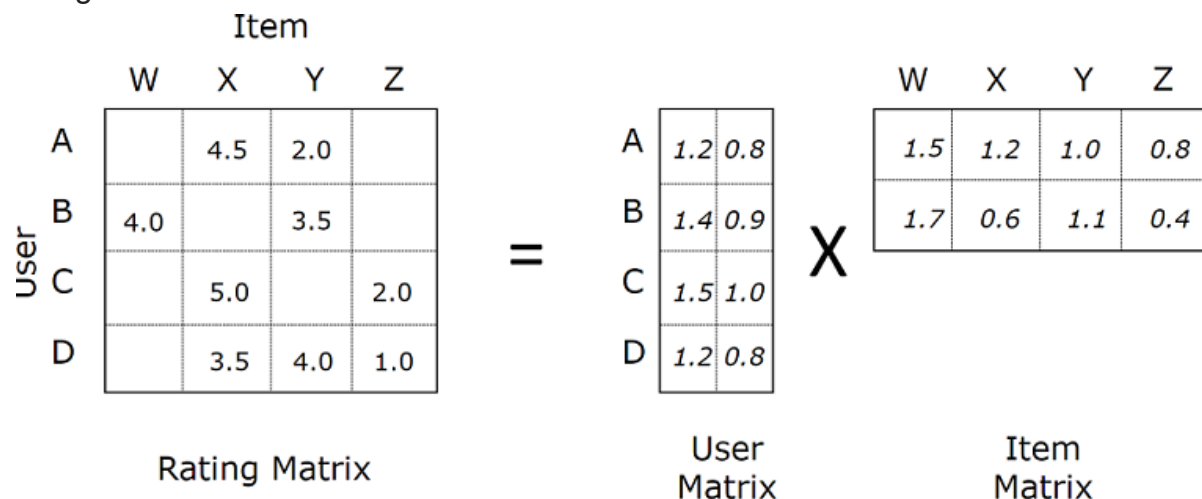
1. Initialize $x^{(i)}, \dots, x^{(nm)}, \theta^{(1)}, \dots, \theta^{(nu)}$ to small random values.
2. Minimize $J(x^{(1)}, \dots, x^{(nm)}, \theta^{(1)}, \dots, \theta^{(nu)})$ using gradient descent.

E.g. for every $j = 1, \dots, n_u, i = 1, \dots, n_m$:

$$x_k^{(i)} := x_k^{(i)} - \alpha \left\{ \sum_{j:r(i,j)=1} ((\theta_k^{(j)})^T x^{(i)} - y^{(i,j)}) \theta_k^{(j)} + \lambda x_k^{(i)} \right\}$$

$$\theta_k^{(j)} := \theta_k^{(j)} - \alpha \left\{ \sum_{i:r(i,j)=1} ((\theta_k^{(j)})^T x^{(i)} - y^{(i,j)}) x_k^{(i)} + \lambda \theta_k^{(j)} \right\}$$

3. For a user with parameters θ and a movie with (learned) feature x , predict a star rating of θx .



So that we assume some features of songs, on the basis of that features a user will predict its rating.

Only assigning features for a song will not help much but we have also some parameters that for a user too.

Both the feature of the songs and parameters of a user can solve our problem.

And one more things both the parameter and the features are same in numbers but may different in values.

We assigning 5 features for songs and 5 parameters for a user and set a small random value for both features and parameters

we create two matrix USER (200000 , 5) and Song (127771, 5) Assigns small random values to each of the index in matrix.

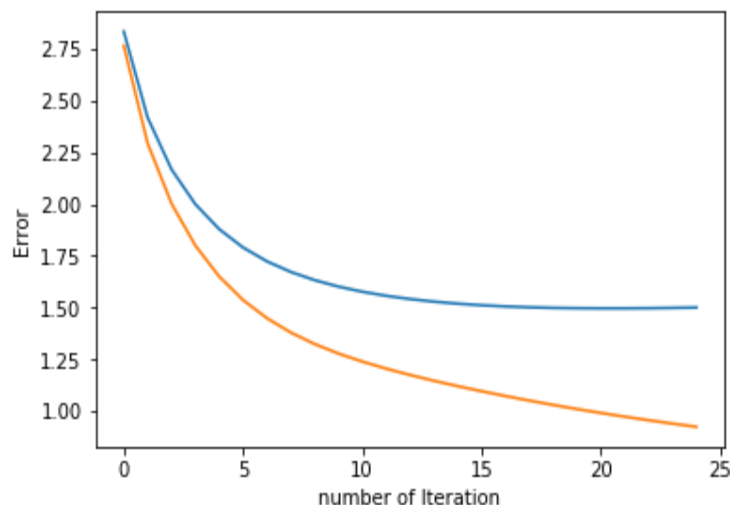
We have used the same equation given up to the matrix to learn the parameter And give the result

Result:-

This graph shows the rate of learning with respect to the no of iteration

Orange line shows RMSE with Train data

Blue line shows RMSE with Test Data



We will observe that now if we increase the no of iteration then Training error will decrease but Test error will be static

After iteration 10 the Test Error

have Straight line w.r.t iteration so in no need to increase iteration

	Songs	Prediction	rating
127725	136689	0.938648	1
26599	28414	2.695108	3
36031	38524	1.511098	2
99738	106721	1.510433	2
123148	131781	0.946126	1
108532	116129	2.439266	2
73297	78459	2.204791	2

Conclusion

In our project, collaborative filtering algorithm is used to predict user's song rating. The Songs dataset, which has 2 million ratings, is selected in our project and divided into training set and test set. The RMSE method is used for algorithm evaluation. According to evaluation result, our movie recommender system has pretty good prediction performance.

Reference

1. <https://towardsdatascience.com/intro-to-recommender-system-collaborative-filtering-64a238194a26>
2. https://en.wikipedia.org/wiki/Collaborative_filtering
3. <https://www.coursera.org/lecture/machine-learning/collaborative-filtering-algorithm-f26nH>