The Experiment Report of Machine Learning

SCHOOL: SCHOOL OF SOFTWARE ENGINEERING

SUBJECT: SOFTWARE ENGINEERING

Author:
Juncong Huang, Yunzhe Guo and Shumin Yan

Supervisor:
Mingkui Tan

Student ID：
201530611746, 201530611524 and 201530613337

Grade:
Undergraduate

December 26, 2017

Recommender System Based on Matrix Decomposition

I. Abstract— Recommender System applies statistical and knowledge discovery techniques to the problem of making product recommendations. In this experiment, we will implement recommender system based on Matrix Decomposition using alternate least squares optimization(ALS).

II. Introduction

The motivations of the experiment are as follows. Firstly, explore the construction of recommended system. Secondly, understand the principle of matrix decomposition. Thirdly, be familiar to the use of gradient descent. Fourthly, construct a recommendation system under small-scale dataset and cultivate engineering ability.

III. Methods and Theory

Model-based CF Algorithms builds a model from the rating data and uses this model to predict missing ratings. Matrix Decomposition is one of Model-based CF Algorithms and is the most widely used algorithm.

In this experiment, we implement recommender system based on Matrix Decomposition using alternate least squares optimization(ALS).

Give a rating matrix $R_{n_{users}, n_{items}}$, rating matrix $R_{n_{users}, n_{items}}$ can be factorized into the multiplication of two low-rank feature matrices $P_{n_{users}, K}$ and $Q_{n_{item}, K}$'s transpose.

Objective function

ALS is to minimize the following objective function:

$$\mathrm{L} = \sum_{u,i} (r_{u,i} - p_u{}^T q_i)^2 + \lambda \left(\sum_u n_{p_u} ||p_u||^2 + \sum_i n_{q_i} ||q_i||^2 \right)$$

r_{ui} denotes the actual rating of user u for item i and $\hat{r}_{ui}$ denotes the prediction.

λ is regularization parameter to avoid overtting

$P_{n_{users}, K} = [p_1, p_2, \ldots p_{nusers}]^T \in R^{n_{users}\ X\ K}$

$Q_{n_{item}, K} = [q_1, q_2, \ldots q_{nitems}]^T \in R^{\ n_{items}\ X\ K}$

n_{p_u} and n_{q_i} denote the number of total ratings on user u and item i, respectively.

Optimize $P_{n_{users}, K}$ while fixing $Q_{n_{item}, K}$

$$\frac{\partial L}{\partial p_u} = 0$$

$$\sum_{r_{u,i} \neq 0} (q_i q_i{}^T + \lambda\, n_{p_u} I) \cdot p_u = Q_{n_{item}, K}{}^T \cdot R_{u*}{}^T$$

$$p_u = (q_i q_i{}^T + \lambda\, n_{p_u} I)^{-1} \cdot Q_{n_{item}, K}{}^T \cdot R_{u*}^T$$

R_{u*} denotes the u-th row of rating matrix $R_{n_{users}, n_{items}}$

Optimize $Q_{n_{item}, K}$ while fixing $P_{n_{users}, K}$

$$\frac{\partial L}{\partial q_i} = 0$$

$$\sum_{r_{u,i} \neq 0} (p_u p_u{}^T + \lambda\, n_{q_i} I) \cdot q_i = P_{n_{users}, K}{}^T \cdot R_{*i}{}^T$$

$$q_i = (p_u p_u{}^T + \lambda\, n_{q_i} I)^{-1} \cdot P_{n_{users}, K}{}^T \cdot R_{*i}$$

R_{*i} denotes the i-th column of rating matrix $R_{n_{users}, n_{items}}$

ALS for MF's steps

Step1

Require rating matrix R, feature matrixs $P_{n_{users}, K}$, $Q_{n_{item}, K}$ and regularization parameter λ

Step2

Optimize $P_{n_{users}, K}$ while fixing $Q_{n_{item}, K}$:

$$p_u = (q_i q_i{}^T + \lambda\, n_{p_u} I)^{-1} \cdot Q_{n_{item}, K}{}^T \cdot R_{u*}^T$$

Step3

Optime $Q_{n_{item}, K}$ while fixing $P_{n_{users}, K}$

$$q_i = (p_u p_u{}^T + \lambda\, n_{q_i} I)^{-1} \cdot P_{n_{users}, K}{}^T \cdot R_{*i}$$

Step4

Repeat the above process util convergence

IV. Experiment

A. Dataset

Utilizing MovieLens-100k dataset.

u.data -- Consisting 10,000 comments from 943 users out of 1682 movies. At least, each user comment 20 videos. Users and movies are numbered consecutively from number 1 respectively. The data is sorted randomly.

u1.base / u1.test are train set and validation set respectively, seperated from dataset u.data with proportion of 80% and 20%. It also make sense to train set and validation set from u1.base / u1.test to u5.base / u5.test.

B. Implementation

Experiment Step

1. Read the data set and divide it. Populate the original scoring matrix $R_{n_{users}, n_{items}}$ against the raw data, and fill 0 for null values.
2. Initialize the user factor matrix $P_{n_{users}, K}$ and the item (movie) factor matrix $Q_{n_{item}, K}$, where K is the number of potential features. In our experiment, **K** is **40.**
3. Determine the loss function and the hyperparameter penalty factor λ. In our experiment, $\boldsymbol{\lambda}$ is **0.1.**

4. Use alternate least squares optimization method to decompose the sparse user score matrix, get the user factor matrix and item (movie) factor matrix:

4.1 With fixd item factor matrix, find the loss partial derivative of each row of the user factor matrices, ask the partial derivative to be zero and update the user factor matrices.

4.2 With fixd user factor matrix, find the loss partial derivative of each row of the item factor matrices, ask the partial derivative to be zero and update the item

4.3 Calculate the $L_{validation}$ on the validation set, comparing with the $L_{validation}$ of the previous iteration to determine if it has converged.

5. Repeat step 4 several times. In our experiment, the **max iteration** is **100**. If the times is over 100 or rmse is smaller than **ε(0.00001)**, stop repeat step4. Then get a satisfactory user factor matrix P and an item factor matrix Q, Draw a $L_{validation}$ curve with varying iterations.

6. The final score prediction matrix $\check{R}_{n_{users},n_{items}}$ is obtained by multiplying the user factor matrix $P_{n_{users},K}$ and the transpose of the item factor matrix $Q_{n_{item},K}$.

Result

Use u.data, and divide u.data into training data and validation data.

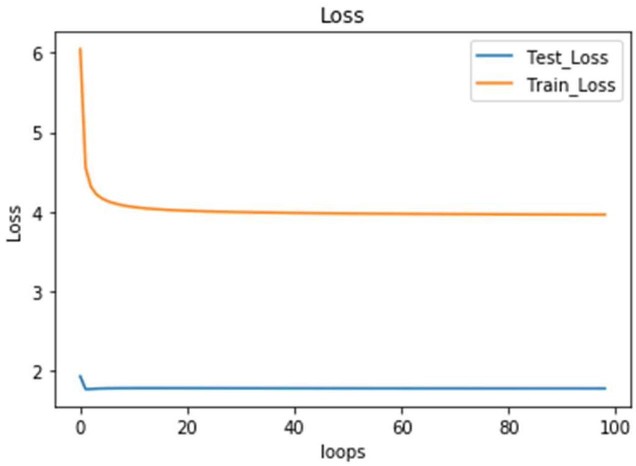

Fig.1.Use u.data

Use use u1.base / u1.test to u5.base / u5.test, and train five model.

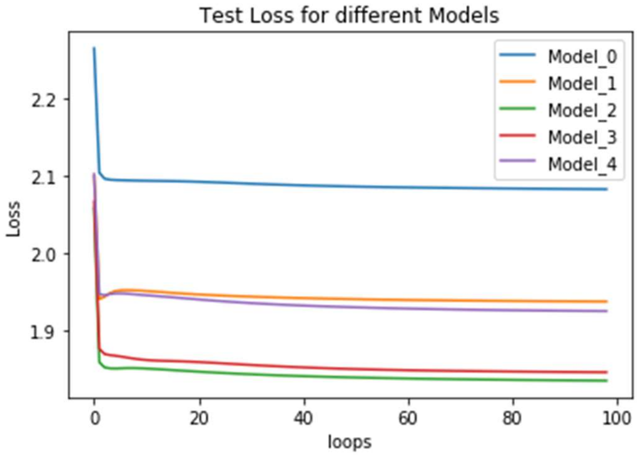

Fig.2. use the five divided dataset

V. CONCLUSION

Through this experiment, we learn a lot. We explore the construction of recommended system and understand the principle of matrix decomposition. What's more, by constructing a recommendation system under small-scale dataset, we impove engineering ability.