Matrix Factorization Recommender Systems and Cold Start
Aaron Tuor and Brian Hutchinson, Computer Science Department, Western Washington University

Overview

Recommender Systems
Created to direct users to items of interest, given a dataset, e.g., user ratings or product descriptions.

Cold Start Problem
How to give ratings for new users and new items for which no ratings are recorded in the dataset?

Outline
Survey state of the art Matrix Factorization solutions to the Cold Start problem.
Suggest avenues of further research.

User-Item Utility Matrix
Ratings for m users and n items in \( m \times n \) matrix, \( R \).

Ratings for \( R_{u,i} = u \) 's rating of item \( i \).

\[ R = \begin{pmatrix}
R_{1,1} & R_{1,2} & \cdots & R_{1,n} \\
R_{2,1} & R_{2,2} & \cdots & R_{2,n} \\
\vdots & \vdots & \ddots & \vdots \\
R_{m,1} & R_{m,2} & \cdots & R_{m,n}
\end{pmatrix} \]

Ratings for \( R_{u,i} = u \) 's rating of item \( i \).

Modeling Global Behavior

- \( R'_{u,i} = R_{u,i} - b_{u,i} \)
- \( b_{u,i} = \mu + \sigma_u + \sigma_i \), where \( \mu \) is mean rating, \( \sigma_u, \sigma_i \) are observed deviations of user \( u \) and item \( i \) ratings from \( \mu \).

Matrix Factorization Model

Assume low rank to complete \( R' \). \( R' \) may be factored into two matrices, \( P \) and \( Q^T \), so that \( PQ^T \approx R' \) where

- \( P_{u} \) is user 'u' association with latent factor \( x \)
- \( Q_{i} \) is item 'i' association with latent factor \( x \).

Objective Function

\[ L(Q,P) = \sum_{(u,i)} (R'_{u,i} - q_u^T P_i)^2 + \lambda (\|q_u\|^2 + \|P_i\|^2) \]

Gradient Descent Step

\[ \nabla L(Q,P) = \frac{1}{2} (R'_{u,i} - q_u^T P_i) Q_i + \lambda q_u + \lambda P_i \]

Find Optimal Profiles for users in Child Nodes

- \( h_u \) be a variable ranging over \( L, D, N \).
- \( h_u = \arg \min_{h} \sum_{u,i} (R'_{u,i} - q_u^T P_i + h_i) \)
- \( \min \sum_{u,i} (R'_{u,i} - q_u^T P_i + h_i) \) over \( h_u \) and \( h_i \)
- \( \lambda \) is a penalty parameter.

User Cold Start Problem

A basic MF model can't predict ratings for new users.

Functional Matrix Factorization

- Hybrid MF-Decision Tree Recommender System
- Uses MF to find best queries to estimate user profiles

Recommender Decision Trees

- New users are asked about their preference for certain items. Possible responses: Like, Dislike, Unknown.
- \( P(T) \) is function for approx. user profiles.

Objective Function

\[ \min_{T} \sum_{(u,i)} (R'_{u,i} - q_u^T T(a_{(u,i)})^2 + \lambda \|q_u\|^2) \]

Find T and Optimize w.r.t. \( Q \)

- Starting from root, partition users into sets by query \( h \).
- \( U_L(h) = \{u | a_{(h,u)} = \text{Like} \} \)
- \( U_D(h) = \{u | a_{(h,u)} = \text{Dislike} \} \)
- \( U_N(h) = \{u | a_{(h,u)} = \text{Unknown} \} \)

Find Q and Optimize w.r.t. \( T \)

- Exploiting Locality
- Measurement of local smoothness for factor \( Q \):

\[ S = \frac{1}{2} \|q_i - q_j\|^2 + \frac{d_i - d_j}{\|d_i\| \|d_j\|} \]

Objective Function

\[ L = \min_{T} \sum_{(u,i)} (R'_{u,i} - q_u^T T(a_{(u,i)})^2 + \lambda \|q_u\|^2 \]

Collective Factorization

\[ D_{ij} = \text{TF-IDF}(t_j, d_i) \]

Local Collective Embeddings

- Uses collective factorization to link user ratings to item text.
- TF-IDF
- \( t \) is a corpus term, \( d \) is a corpus document
- \( f_{t,d} = \# \times t \) occurs in \( d \)
- \( T_I = \{ t : t \text{ occurs in document } d \} \)

Collective Factorization

\[ R_{t,d} = \text{TF-IDF}(t_j, d_i) \]

Amazon Dataset

- \( 34,686,770 \) reviews (text & ratings) & 6,643,669 users
- 28 product categories & descriptions for all products

User Cold Start Problem

- How to recommend new unrated items in MF framework?

Item Cold Start Problem

- How to give ratings for new users and new items for which no ratings are recorded in the dataset?

Further Research

- Benchmark state of the art methods using Amazon dataset
- Explore cold start metric which leverages performance of both user and item cold start scenarios
- Develop method which addresses user and item cold start
- Use product descriptions to generate topic factorization and topic keywords to generate decision tree

References


Copyright © Stanford University. Learning from the Online News.