Week 12-2

Recommender System II

Big Data

Prof. Hwanjo Yu

Matrix-factorization based recommendation

Model-Based Recommendation

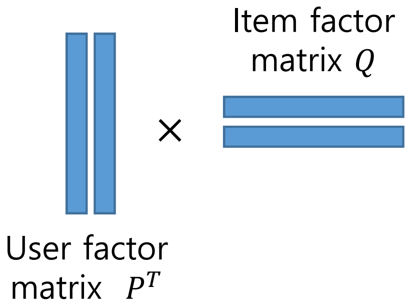
Big Data

Formal description

- Latent Model

$$\begin{array}{c} \\ Alice \\ Bob \\ Charlie \end{array} \begin{array}{c} \begin{array}{ccc} Avatar & The\ Matrix & Up \end{array} \\ \begin{pmatrix} ? & 4 & 2 \\ 3 & 2 & ? \\ 5 & ? & 3 \end{pmatrix} \end{array} \approx$$

Original matrix R



  - $r_{ij} \approx \widehat{r_{ij}} = [P^T Q]_{ij}$

- Goal: Find P and Q which minimize the error (RMSE)

  - $\underset{P,Q}{\operatorname{argmin}} RMSE(R, P^T Q)$

Transform to an optimization problem

- RMSE
  - $\sqrt{\frac{1}{|R|}\sum_{(i,j)\in R}\left(r_{ij}-\widehat{r_{ij}}\right)^2}$
- Minimizing RMSE is equal to minimizing unnormalized MSE
  - $\sqrt{\frac{1}{|R|}\sum_{(i,j)\in R}\left(r_{ij}-\widehat{r_{ij}}\right)^2} \Rightarrow \sum_{(i,j)\in R}\left(r_{ij}-[P^TQ]_{ij}\right)^2$
- The final objective function
  - $\underset{P,Q}{\operatorname{argmin}} \sum_{(i,j)\in R}\left(r_{ij}-[P^TQ]_{ij}\right)^2$
  - $\underset{P,Q}{\operatorname{argmin}} \left\|R-P^TQ\right\|_2^2$

An illustrating example for relationship between 2-norm and trace

- 2-norm
  - $\left\| \begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \end{bmatrix} \right\|^2 = x_1^2 + x_2^2 + x_3^2 + x_4^2$
- Trace
  - $Tr\left(\begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \end{bmatrix} \begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \end{bmatrix}^T \right) = Tr\left(\begin{bmatrix} x_1^2 + x_2^2 & x_1 x_3 + x_2 x_4 \\ x_3 x_1 + x_4 x_2 & x_3^2 + x_4^2 \end{bmatrix} \right) = x_1^2 + x_2^2 + x_3^2 + x_4^2$

Transform to an optimization problem

- RMSE
  - $\sqrt{\frac{1}{|R|}\sum_{(i,j)\in R}\left(r_{ij}-\widehat{r_{ij}}\right)^2}$
- Minimizing RMSE is equal to minimizing unnormalized MSE
  - $\sqrt{\frac{1}{|R|}\sum_{(i,j)\in R}\left(r_{ij}-\widehat{r_{ij}}\right)^2} \Rightarrow \sum_{(i,j)\in R}\left(r_{ij}-[P^TQ]_{ij}\right)^2$
- The final objective function
  - $\underset{P,Q}{\operatorname{argmin}} \sum_{(i,j)\in R}\left(r_{ij}-[P^TQ]_{ij}\right)^2$
  - $\underset{P,Q}{\operatorname{argmin}} \left\|R-P^TQ\right\|_2^2 = \underset{P,Q}{\operatorname{argmin}} \operatorname{Tr}\left(\left(R-P^T\cdot Q\right)\cdot\left(R-P^T\cdot Q\right)^T\right)$

Gradient Descent (GD)

- First-order optimization algorithm, which takes steps proportional to the negative of the gradient of the function at the current point
  - $x_{n+1} = x_n - \eta \nabla f(x_n)$
    - η: step size
    - f: objective function

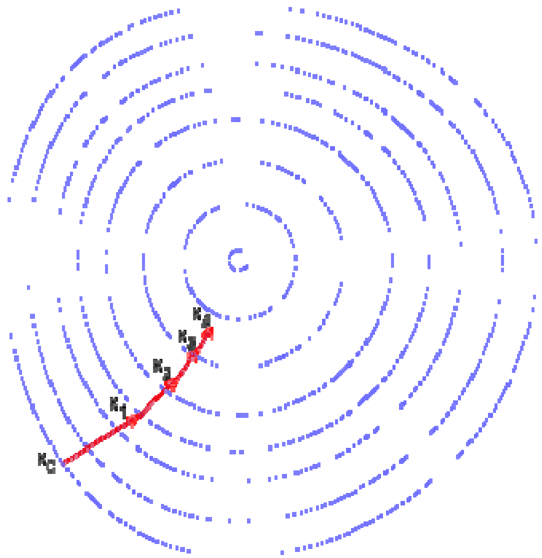


Figure. Illustration of gradient descent

GD for MF (1)

- Objective function for MF
  - $\|R - P^T \cdot Q\|_2^2 = \mathrm{Tr}\left(\left(R - P^T \cdot Q\right) \cdot \left(R - P^T \cdot Q\right)^T\right)$
- Derivative rule
  - $\frac{\partial}{\partial X} Tr[(C + AXB)(C + AXB)^T] = 2A^T(C + AXB)B^T$
- Gradient for the objective function
  - $\frac{\partial f(P,Q)}{\partial P} = -2\left(R - P^T \cdot Q\right)Q^T$
  - $\frac{\partial f(P,Q)}{\partial Q} = -2P\left(R - P^T \cdot Q\right)$

GD for MF (2)

- Updating rule for gradient descent
  - $x_{n+1} = x_n - \eta \nabla F(x_n)$
- Gradient for the objective function
  - $\frac{\partial f(P,Q)}{\partial P} = -2\left(R - P^T \cdot Q\right)Q^T$
  - $\frac{\partial f(P,Q)}{\partial Q} = -2P\left(R - P^T \cdot Q\right)$
- Updating rule
  - $E = R - P^T \cdot Q$
  - $P^T = P^T - \eta \frac{\partial f(P,Q)}{\partial P} = P^T + \eta' E Q^T$
  - $Q = Q - \eta \frac{\partial f(P,Q)}{\partial Q} = Q + \eta' P E$

Dimensionality check

$$R = m \times n$$
$$P = k \times m$$
$$Q = k \times n$$
$$P^T = m \times k$$
$$Q^T = n \times k$$
$$R - P^T \cdot Q = m \times n$$
$$\left(R - P^T \cdot Q\right)Q^T = m \times k$$
$$P^T\left(R - P^T \cdot Q\right) = k \times n$$

Stochastic Gradient Descent (SGD)

- Computing the standard GD is expensive
- SGD is a lightweight algorithm which repeats following procedure

   1. *Randomly* pick a data instance
   2. Calculate a gradient associated with the instance
   3. Update the solution by the gradient of the instance

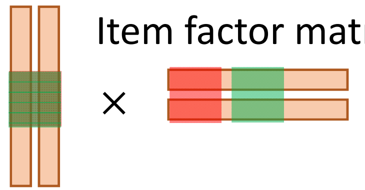
SGD for MF

- SGD gradually updates factor matrices by repeating following two steps

  1. Randomly pick a rating data

  2. Update corresponding vectors in factor matrices

$$
\begin{array}{l} \\ \textit{Alice} \\ \textit{Bob} \\ \textit{Charlie} \end{array}
\begin{array}{c} \begin{array}{ccc} \textit{Avatar} & \textit{The Matrix} & \textit{Up} \end{array} \\ \begin{pmatrix} ? & 4 & 2 \\ \boxed{3} & \boxed{2} & ? \\ 5 & ? & 3 \end{pmatrix} \end{array}
\approx
$$



Item factor matrix Q

Rating matrix R User factor matrix P^T

SGD for MF (2)

- objective function with the regularizers

$$\min_{P,Q} \sum_{(i,j)\in R} \underbrace{\left\{\left(r_{ij} - \boldsymbol{p}_i^T \cdot \boldsymbol{q}_j\right)^2 + \lambda_P \|\boldsymbol{p}_i\|_2^2 + \lambda_Q \left\|\boldsymbol{q}_j\right\|_2^2\right\}}_{f(r_{ij},\boldsymbol{p_i},\boldsymbol{q_j})}$$

- Gradient
  - $\frac{\partial f(r_{ij},\boldsymbol{p}_i,\boldsymbol{q}_j)}{\partial \boldsymbol{p}_i} = -2\left(r_{ij} - \boldsymbol{p}_i^T \cdot \boldsymbol{q}_j\right)\boldsymbol{q}_j + 2\lambda_p \boldsymbol{p}_i$
  - $\frac{\partial f(r_{ij},\boldsymbol{p}_i,\boldsymbol{q}_j)}{\partial \boldsymbol{q}_i} = -2\left(r_{ij} - \boldsymbol{p}_i^T \cdot \boldsymbol{q}_j\right)\boldsymbol{p}_i + 2\lambda_q \boldsymbol{q}_j$

- Updating rules

$$e_{ij} = r_{ij} - \boldsymbol{p}_i^T \cdot \boldsymbol{q}_j$$
$$p_i \leftarrow p_i + \eta'(e_{ij} \cdot \boldsymbol{q}_j - \lambda_P \cdot \boldsymbol{p}_i)$$
$$q_j \leftarrow q_j + \eta'(e_{ij} \cdot \boldsymbol{p}_i - \lambda_Q \cdot \boldsymbol{q}_j)$$

Another Approaches

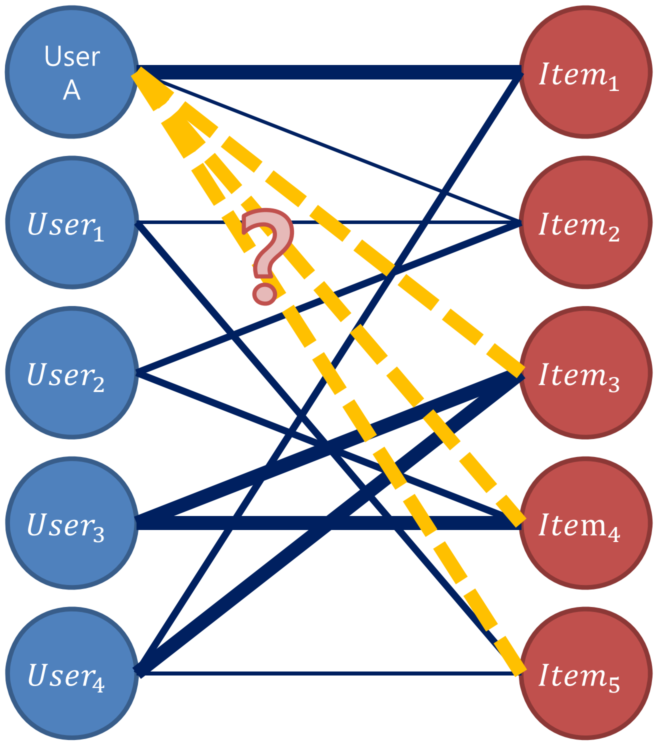
Big Data

Recommendation via random walk

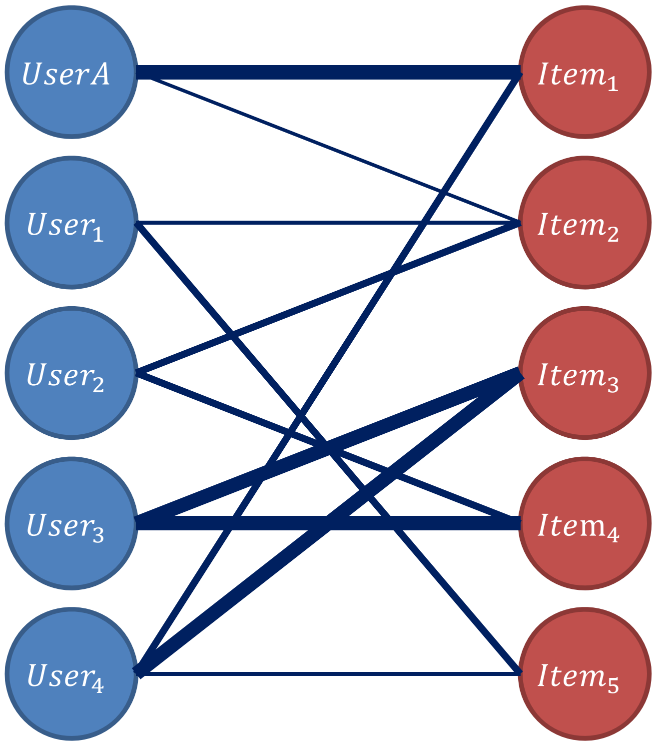
- Model users and items are a bipartite graph
  - Or a correlation graph on items [Gori07]
- Approximate similarity between users and items via **random walk** on the bipartite graph
- This technique is closely related to the concept of PageRank

Recommendation via random walk

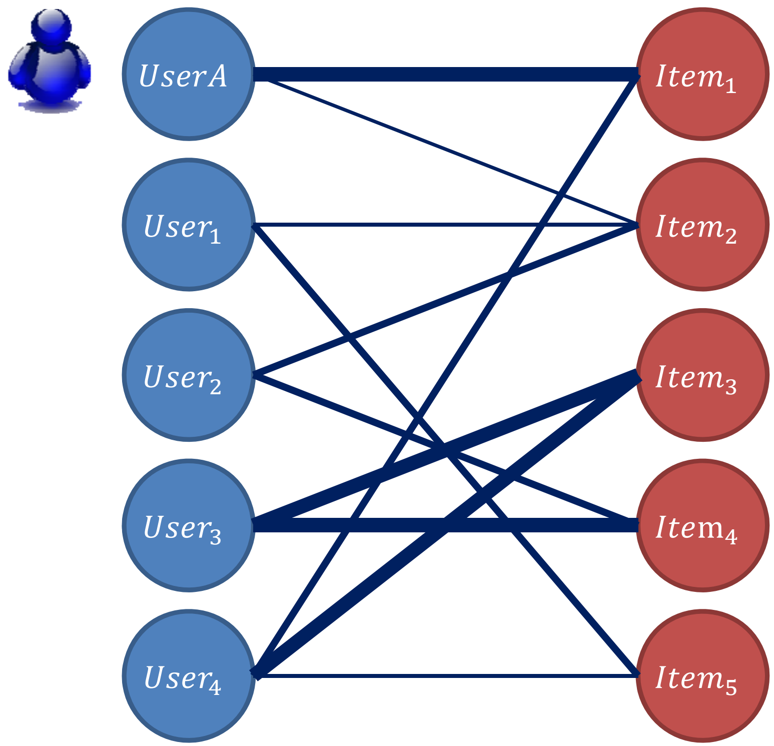
	Item 1	Item 2	Item 3	Item 4	Item 5
UserA	5	1	?	?	?
User 1		1			3
User 2		3		3	
User 3			5	5	
User 4	3		5		1



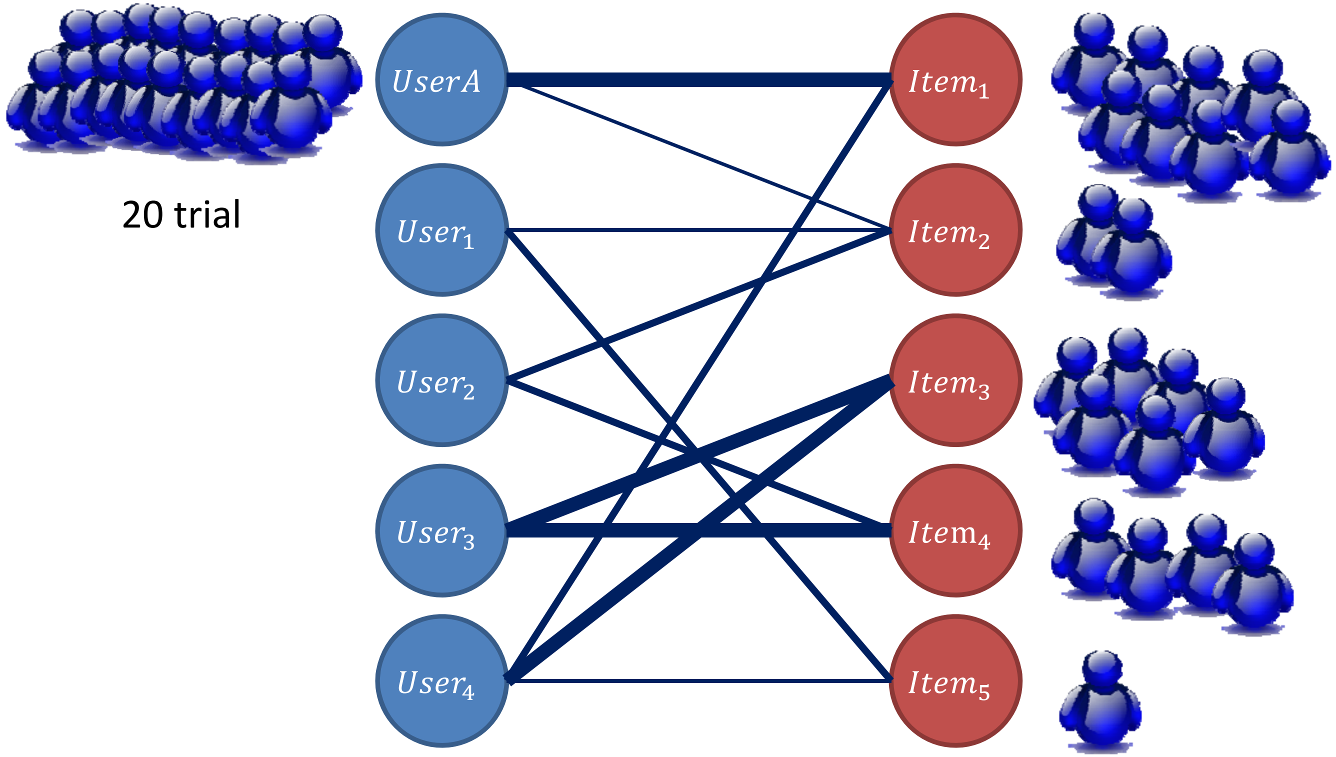
Example of random walk



Example of random walk



Example of random walk



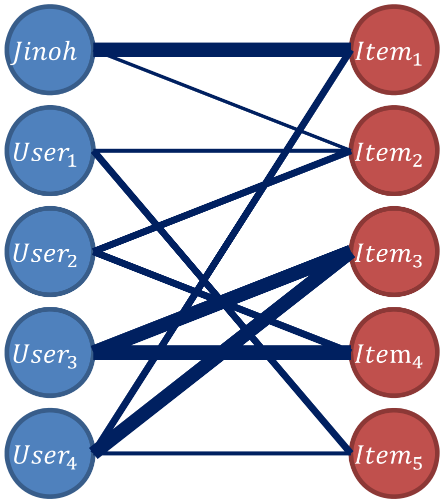
Recommendation via random walk

- The $s(u, i_{unseen})$ can be proportionally obtained by the fraction of trials reaching i_{unseen} via random walk

POSTECH
POHANG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Analytic representation of random walk

- A graph can be represented as an adjacency matrix



	$Item_1$	$Item_2$	$Item_3$	$Item_4$	$Item_5$
$Jinoh$	5	1	-	-	-
$User_1$	-	1	-	-	3
$User_2$	-	3	-	3	-
$User_3$	-	-	5	5	-
$User_4$	3	-	5	-	1

$$= A$$

Analytic representation of random walk

- A propagation can be represented as a matrix-vector multiplication

$$
\begin{array}{c}
\\ Jinoh \\ User_1 \\ User_2 \\ User_3 \\ User_4
\end{array}
\begin{array}{c}
\begin{array}{ccccc} Item_1 & Item_2 & Item_3 & Item_4 & Item_5 \end{array} \\
\begin{pmatrix}
5/6 & 1/6 & - & - & - \\
- & 1/4 & - & - & 3/4 \\
- & 1/2 & - & 1/2 & - \\
- & - & 1/2 & 1/2 & - \\
3/9 & - & 5/9 & - & 1/9
\end{pmatrix}
\end{array}
\times
\begin{array}{c}
\#walk_0@user \\
\begin{bmatrix} 20 \\ - \\ - \\ - \\ - \end{bmatrix}^T
\end{array}
=
\begin{array}{c}
\#walk_1@item \\
\begin{bmatrix} - \\ 100/6 \\ 20/6 \\ - \\ - \end{bmatrix}^T
\end{array}
$$

Analytic representation of random walk

- A propagation can be represented as a matrix-vector multiplication

$$
\begin{array}{c}
\begin{array}{cc}
& \begin{array}{ccccc} Item_1 & Item_2 & Item_3 & Item_4 & Item_5 \end{array} \\
\begin{array}{c} Jinoh \\ User_1 \\ User_2 \\ User_3 \\ User_4 \end{array} &
\begin{pmatrix}
5/6 & 1/6 & - & - & - \\
- & 1/4 & - & - & 3/4 \\
- & 1/2 & - & 1/2 & - \\
- & - & 1/2 & 1/2 & - \\
3/9 & - & 5/9 & - & 1/9
\end{pmatrix}
\end{array}
\times
\begin{array}{c} \#walk_0@user \\ \begin{bmatrix} - \\ 100/6 \\ 20/6 \\ - \\ - \end{bmatrix}^T \end{array}
=
\begin{array}{c} \#walk_1@item \\ \begin{bmatrix} 20/6 \\ 20/6 \\ 60/6 \\ 10/6 \\ 10/6 \end{bmatrix}^T \end{array}
\end{array}
$$

Summary

- Traditional problem formulation for recommender system
  - Find a function f that minimize error between ground truth and predicted rating in terms of RMSE
- Three major methods
  - KNN recommendation
  - Matrix factorization recommendation
    - Latent model
    - Stochastic gradient descents
- Random walk based recommendation