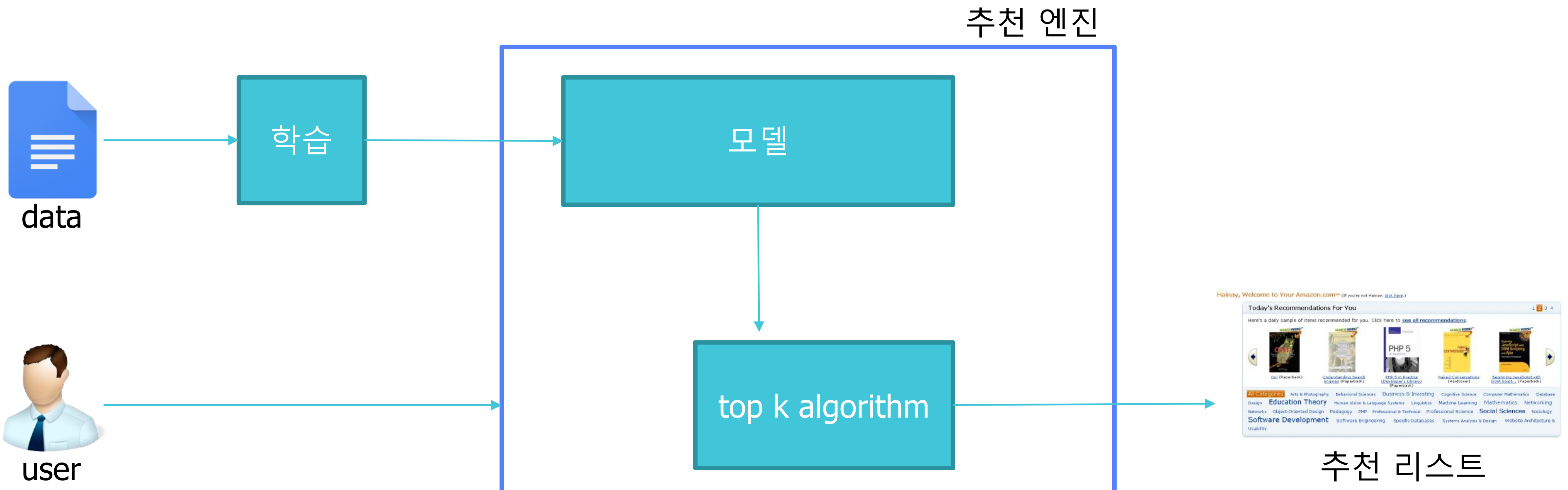
추천 시스템에서 matrix completion 문제 해결 방법

양영욱

Contents

- 추천 시스템 구조
- 추천 시스템의 문제
- Matrix completion Problem
- Collaborative Filtering
- Sparsity Problem
- Matrix factorization

추천 시스템 구조



추천 시스템의 문제

- 추천 문제란?
  - 사용자-상품
    - 사용자 u가 아이템 i를 얼마나 좋아하나? r_{ui}
    - 즉, 사용자 u가 아이템 i를 좋아할 것인가 하지 않을 것인가를 예측하는 모델을 찾는 문제

Matrix Completion Problem

- Matrix의 빈 곳을 채우는 문제
  - R은 완전한 matrix를 의미
  - R에는 결함이 없다고 가정

$$\min_{\hat{R}} \|\hat{R} - R\|_F^2$$

$$\begin{array}{cc} & \begin{array}{cccccccc} movie.1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \left\{\begin{array}{c} user1 \\ user2 \\ user3 \\ user4 \\ user5 \\ user6 \\ user7 \\ user8 \end{array}\right. & \begin{bmatrix} 3 & 5 & * & 4 & 1 & * & * & 2 \\ * & 3 & 5 & 1 & 2 & * & * & 3 \\ 4 & 1 & * & 4 & 1 & * & 3 & 2 \\ 5 & 2 & * & * & 2 & 3 & * & * \\ * & 2 & 4 & 2 & * & * & 1 & 2 \\ 5 & * & * & 5 & 4 & * & * & 4 \\ 1 & * & 5 & 2 & 3 & 1 & 5 & 3 \\ * & 3 & 2 & 1 & 4 & * & * & * \end{bmatrix} \end{array}$$

Collaborative Filtering

- List of **m Users** and a list of **n Items**
- Each user has a **list of items** with associated **opinion**
  - **Explicit opinion** - a rating score
  - Sometime the rating is **implicitly** – purchase records or listen to tracks
- **Active user** for whom the CF prediction task is performed
- **Metric** for measuring **similarity between users**
- Method for selecting a subset of **neighbors**
- Method for **predicting a rating** for items not currently rated by the active user.

Collaborative Filtering

- The basic steps:
  - 1. Identify set of ratings for the **target/active user**
  - 2. Identify set of users most similar to the target/active user according to a similarity function (**neighborhood** formation)
  - 3. Identify the products these similar users liked
  - 4. **Generate a prediction** - rating that would be given by the target user to the product - for each one of these products
  - 5. Based on this predicted rating recommend a set of top N products

User-base CF(1)

						sim(u,v)
2		2	4	5		NA
5		4			1	
		5		2		
	1		5		4	
		4			2	
4	5		1			NA

User-base CF(2)

						sim(u,v)
2		2	4	5		NA
5		4			1	0.87
		5		2		
	1		5		4	
		4			2	
4	5		1			NA

User-base CF(3)

						sim(u,v)
2		2	4	5		NA
5		4			1	0.87
		5		2		1
	1		5		4	
		4			2	
4	5		1			NA

User-base CF(4)

						sim(u,v)
2		2	4	5		NA
5		4			1	0.87
		5		2		1
	1		5		4	-1
		4			2	
4	5		1			NA

User-base CF(5)

						sim(u,v)
2		2	4	5		NA
5		4			1	0.87
		5		2		1
	1		5		4	-1
3.51*	3.81*	4	2.42*	2.48*	2	
4	5		1			NA

Sparsity problem

- If you represent the Netflix Prize rating data in a User/Movie matrix you get...
  - 500,000 x 17,000 = 8,500 M positions
  - Out of which only 100M are not 0's!
- Methods of dimensionality reduction
  - Matrix factorization

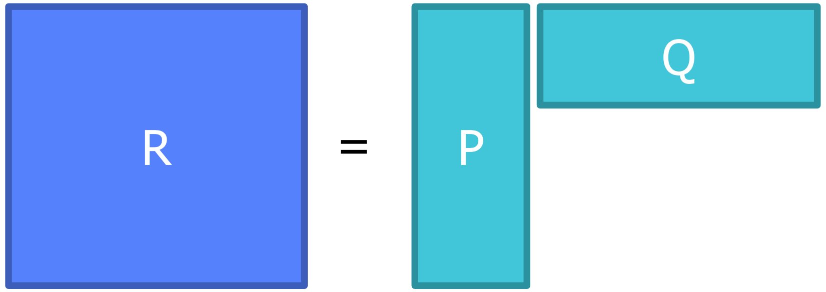
Matrix factorization

- R에 가까운 $\hat{R}$을 찾는 문제
  - Optimization problem

$$\min_{\hat{R}} \|\hat{R} - R\|_F^2$$

$$\min_{P,Q} \sum_{u,i \in \kappa} (r_{ui} - p_u \cdot q_i)^2$$

$$\min_{P,Q} \sum_{u,i \in \kappa} (r_{ui} - p_u \cdot q_i)^2 + \lambda(\|p_u\|_2^2 + \|q_i\|_2^2).$$



Regularization term

Non-convex function(local optimum)

- gradient descent method

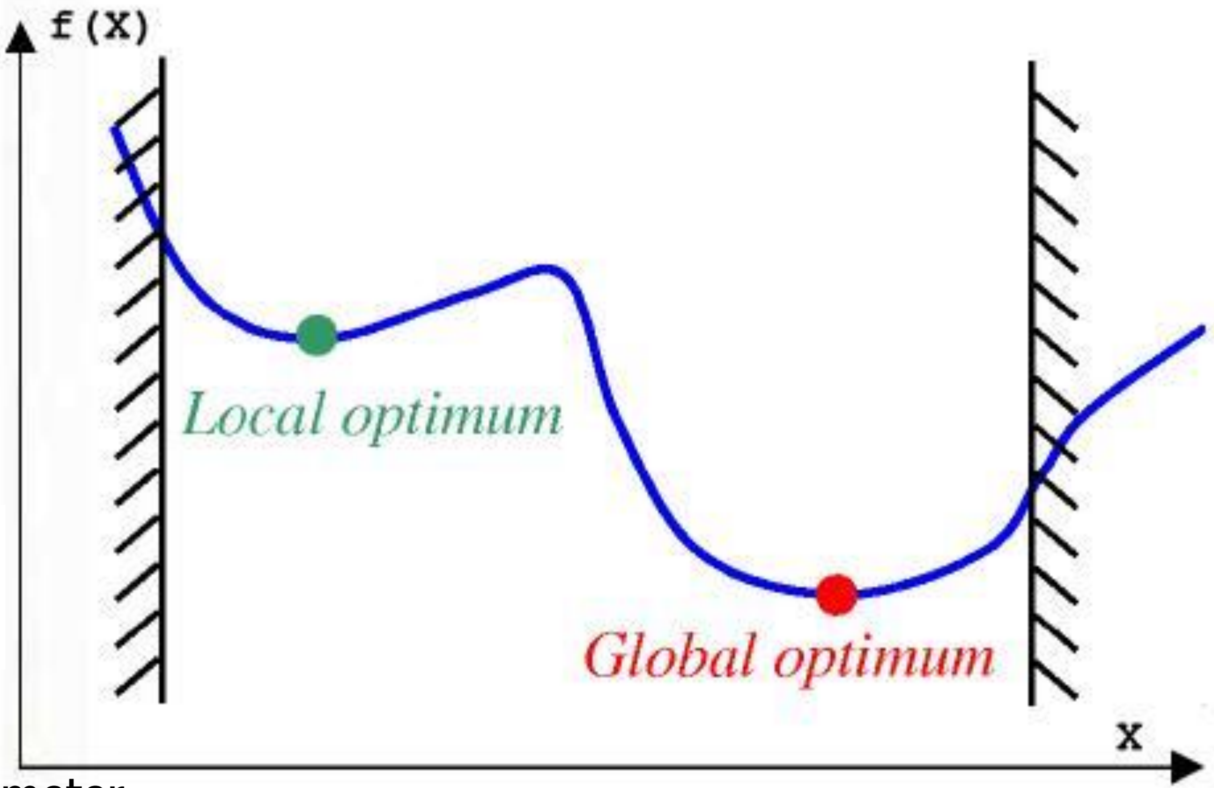
$$p'_{ik} = p_{ik} + \alpha\frac{\partial}{\partial p_{ik}}e^2_{ij} = p_{ik} + 2\alpha e_{ij}q_{kj}$$
$$q'_{kj} = q_{kj} + \alpha\frac{\partial}{\partial q_{kj}}e^2_{ij} = q_{kj} + 2\alpha e_{ij}p_{ik}$$

Learning rate

Regularization term
추가

$$p'_{ik} = p_{ik} + \alpha\frac{\partial}{\partial p_{ik}}e^2_{ij} = p_{ik} + \alpha(2e_{ij}q_{kj} - \beta p_{ik})$$
$$q'_{kj} = q_{kj} + \alpha\frac{\partial}{\partial q_{kj}}e^2_{ij} = q_{kj} + \alpha(2e_{ij}p_{ik} - \beta q_{kj})$$

Regularization parmeter



example

Rating Matrix (User × Item)

User	W	X	Y	Z
A		4.5	2.0	
B	4.0		3.5	
C		5.0		2.0
D		3.5	4.0	1.0

=

User Matrix

A	1.2	0.8
B	1.4	0.9
C	1.5	1.0
D	1.2	0.8

X

Item Matrix

W	X	Y	Z
1.5	1.2	1.0	0.8
1.7	0.6	1.1	0.4