RECOMMENDER SYSTEMS WITH TRUST-BASED SOCIAL NETWORKS

TU JIAYANG
XIN HAO

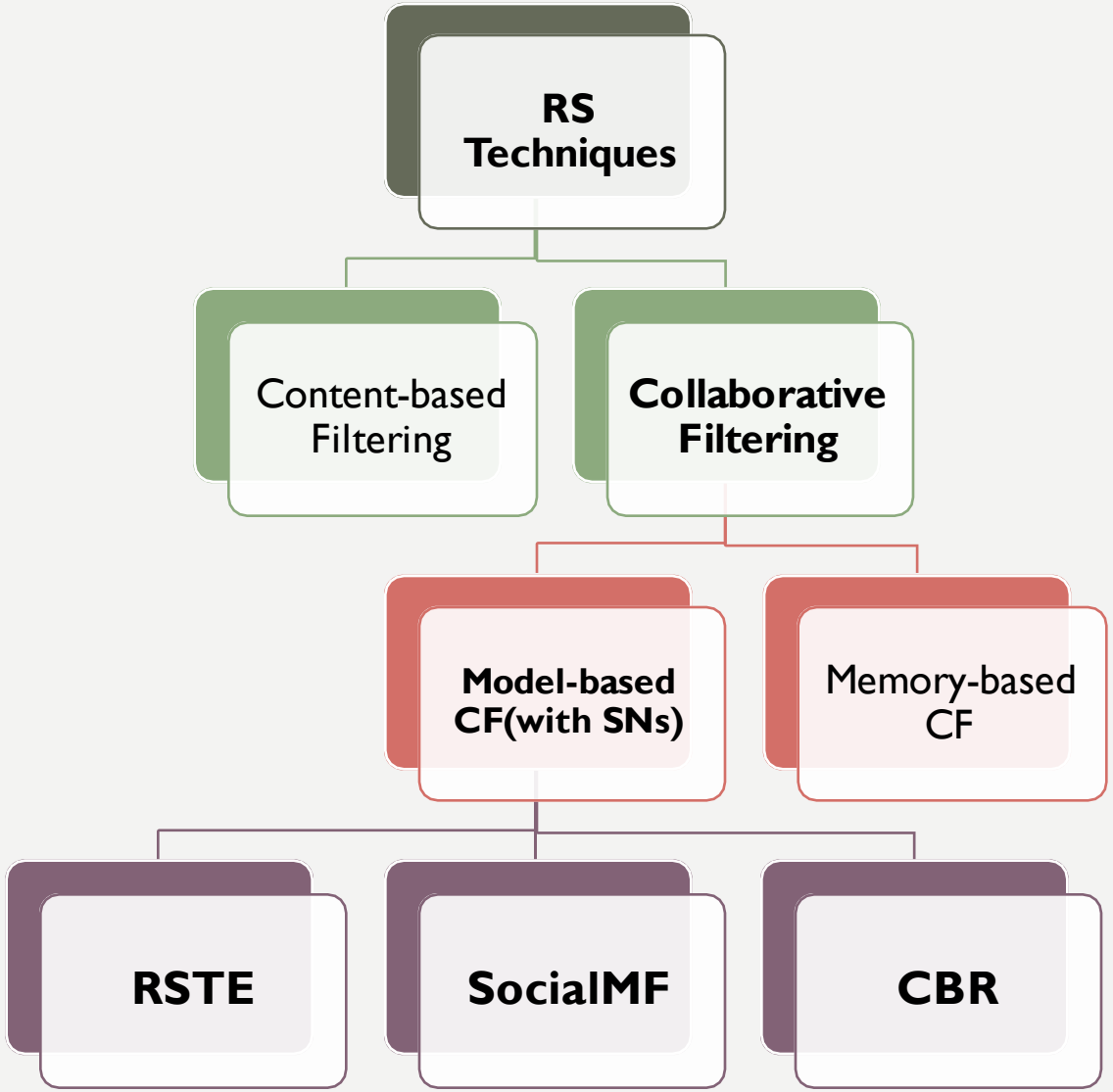
AGENDA

- **Recommender Systems (RS)**
  - Background
  - Collaborative Filtering (CF) Memory-based Approach
- **Recommendation with Trust-based Social Network**
  - Recommendation with Social Trust Ensemble (RSTE)
  - Social Recommendation with Trust Propagation (SocialMF)
  - Circle-based Recommendation (CBR)
- **Implementations**
  - Experiment Preliminaries
  - Result Analysis

RECOMMENDER SYSTEMS

- Widely applied in
  - E-commerce sites(Amazon, Alibaba)
  - Social networks (Facebook, Instagram, and Linkedin)
  - Personalized search(Google)
- Benefit users
  - Narrow down the set of choices
  - Discover interesting/new things
- Benefit providers
  - Enhance user trust and loyalty
  - Increase sales and opportunities for promotion

AN OVERVIEW OF TRUST-BASED RS



MEMORY-BASED CF

- **Assumptions:**
  - Users will give ratings to items
  - Customers who have similar tastes in the past, will have similar tastes in the future
- 1st step: **similarity assessment** between users
  - A popular and effective **similarity measure: Pearson Correlation**

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\ \sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

- a, b: users
- $r_{a,p}$: rating of user a for item p
- P: set of items, rated both by a and b
- $\overline{r_a}, \overline{r_b}$ = user's average ratings

- 2nd step: **rating prediction**

$$pred(a,p) = \overline{r_a} + \frac{\sum_{b \in N} sim(a,b) * (r_{b,p} - \overline{r_b})}{\sum_{b \in N} sim(a,b)}$$

MEMORY-BASED CF (CONT.)

- **Data sparsity problems**
  - Tiny overlapping ratings in massive item sets -> a highly sparse rating (E.g.. User 0 and User 1)
  - Fail in producing recommendations for **cold start users** (E.g. User 2)
- **Easily attacked** by creating ad hoc user profiles with additional ratings
- **Poor scalability**

Items / Users	0	1	2	…	6,000	6,001	6,002	…	
0	0.9	0.7	-	-	-	???	0.3	-	
1			0.6	-	-	0.4	-	-	
2	???	???	???	???	???	???	???	???	New users(cold starters)
…									

TRUST-BASED CF

- **Motivation**
  - To solve the data sparsity problem of memory-based CF
  - Users trust the tastes of their friends more than strangers
  - Social networks facilitate the simulation and formation of trust network

- **Problem definition**
  - Given user **u,** item **i,** rating matrix **R,** trust network **T,** predict the rating $\mathbf{R_{ui}}$

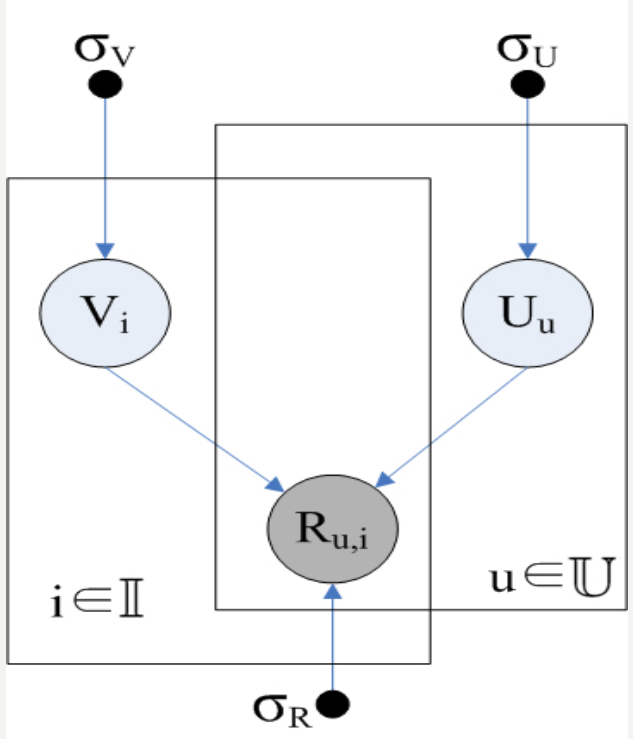
TRUST-BASED CF (CONT.)

- **Matrix Factorization**
  - Latent user features

    $$U \in \mathbb{R}^{K \times N}$$
  - Latent item features

    $$V \in \mathbb{R}^{K \times M}$$
  - Conditional probability for observed ratings

    $$p(R|U,V,\sigma_R^2) = \prod_{u=1}^{N} \prod_{i=1}^{M} \left[\mathcal{N}\left(R_{u,i} | g(U_u^T V_i), \sigma_r^2 \right) \right]^{I_{u,i}^R}$$
- **Recent work**
  - Recommendation with Social Trust Ensemble (RSTE)
  - Social Recommendation with Trust Propagation (SocialMF)
  - Circle-based Recommendation (CBR)

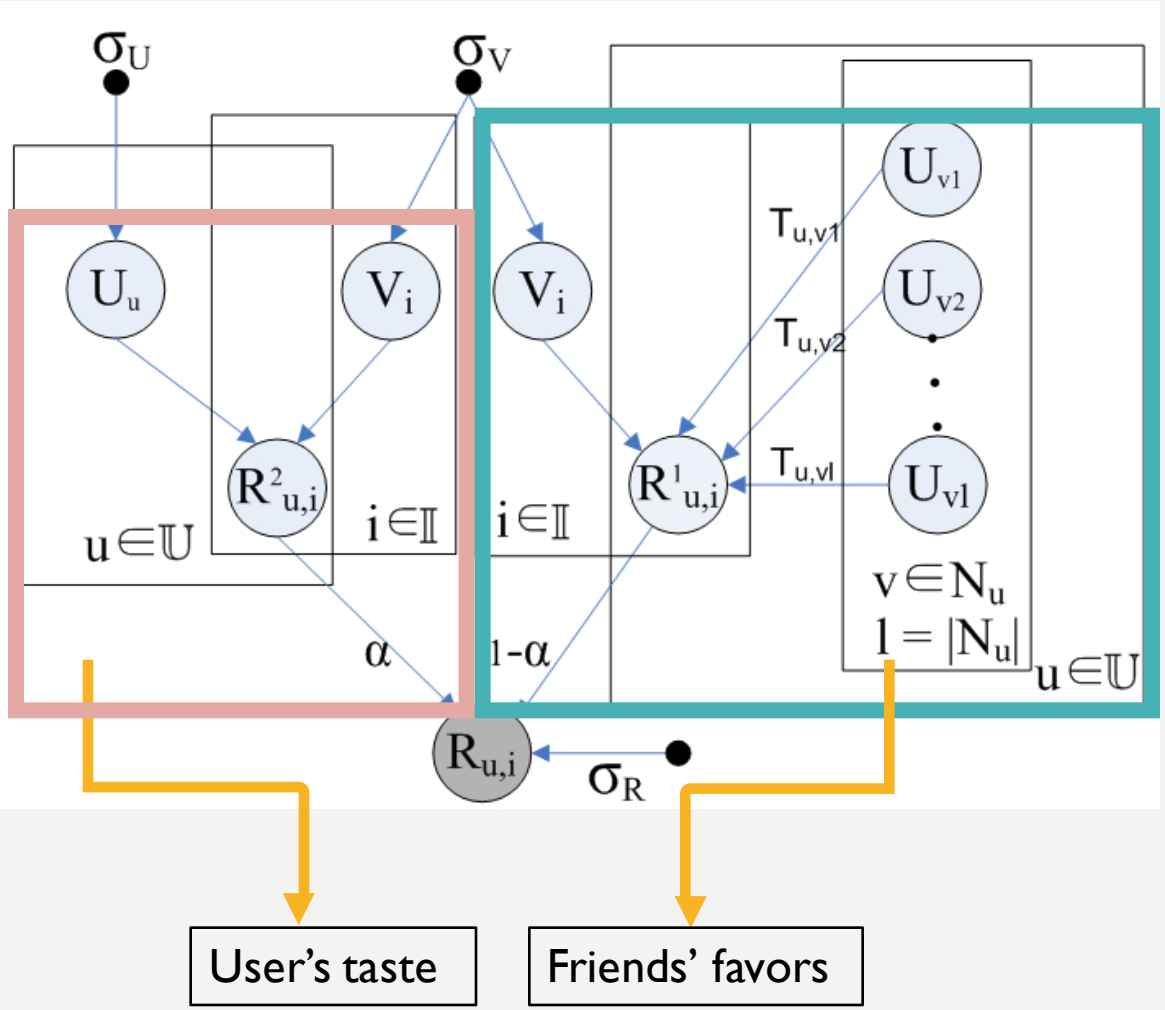


RSTE

- Fuse users' tastes and their trusted friends' favors together
- Predicted ratings:

$$\hat{R}_{u,i} = g(\alpha U_u^T V_i + (1-\alpha) \sum_{v \in N_u} T_{u,v} U_v^T V_i)$$

- g: logistics function to scale the results to [0,1]
- α: determine the contributing weights of user taste and friends' favors to final prediction



RSTE (CONT.)

- Through a **Bayesian inference**

$$p(U,V|R,S,\sigma_S^2,\sigma_U^2,\sigma_V^2) \propto p(R|S,U,V,\sigma_S^2)p(U|\sigma_U^2)p(V|\sigma_V^2),$$
$$= \prod_{i=1}^{m}\prod_{j=1}^{n}\left[\mathcal{N}\left(R_{ij}|g(\alpha U_i^T V_j + (1-\alpha)\sum_{k\in\mathcal{T}(i)} S_{ik}U_k^T V_j), \sigma^2\right)\right]^{I_{ij}^R}$$
$$\times \prod_{i=1}^{m}\mathcal{N}(U_i|0,\sigma_U^2\mathbf{I}) \times \prod_{j=1}^{n}\mathcal{N}(V_j|0,\sigma_V^2\mathbf{I}).$$

Take the log, and it is equivalent to minimize the following function

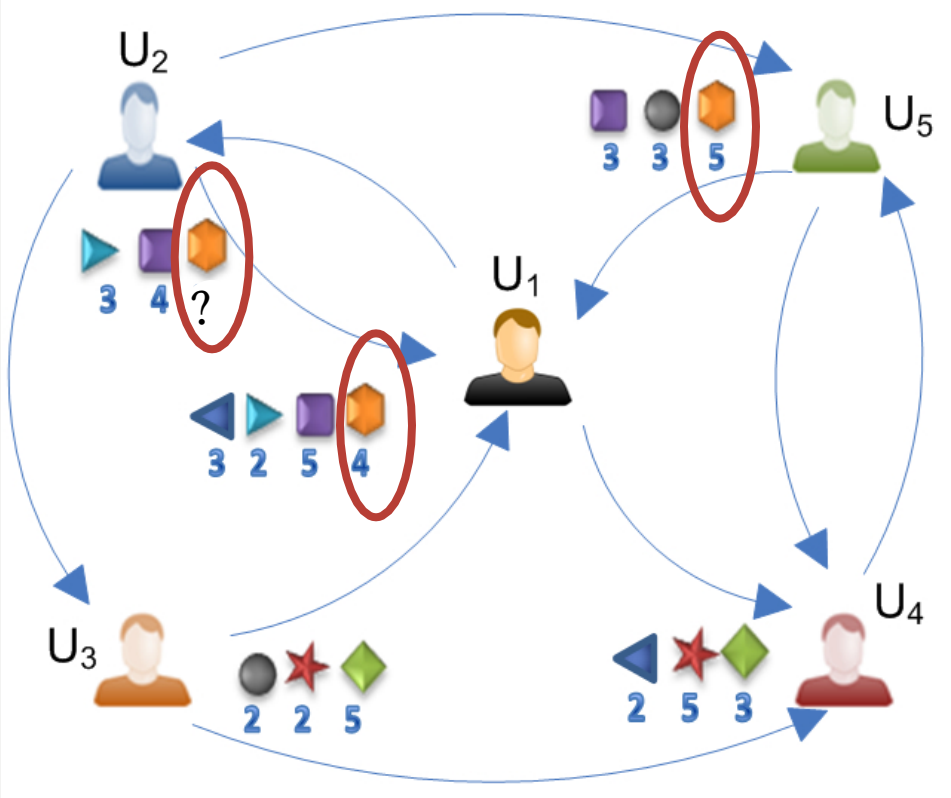
$$\mathcal{L}(R,S,U,V)$$
$$= \boxed{\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}^R (R_{ij} - g(\alpha U_i^T V_j + (1-\alpha)\sum_{k\in\mathcal{T}(i)} S_{ik}U_k^T V_j))^2}$$
$$+ \boxed{\frac{\lambda_U}{2}\|U\|_F^2 + \frac{\lambda_V}{2}\|V\|_F^2}, \qquad (13)$$

Sum-of-squared-error function

Regularization terms

ISSUES OF RSTE

- Do not handle the trust diffusion process
  - E.g. U5 can influence U2 when rating the orange item by influencing his friend U1
- Learning only depends on the observed ratings

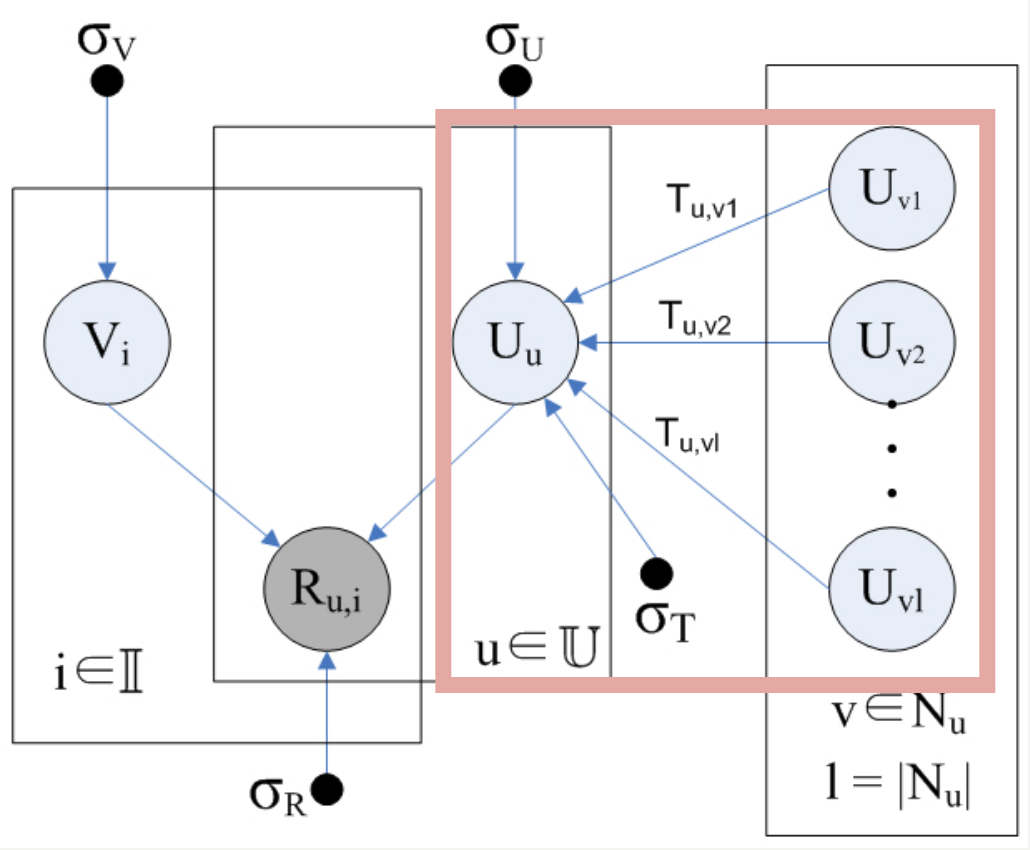


SOCIALMF

- Differentiate from RSTE:
  - User rating only depends on his own latent feature
  - The latent feature vectors of users are influenced by their neighbors

$$\widehat{U}_u = \sum_{v \in N_u} T_{u,v} U_v$$

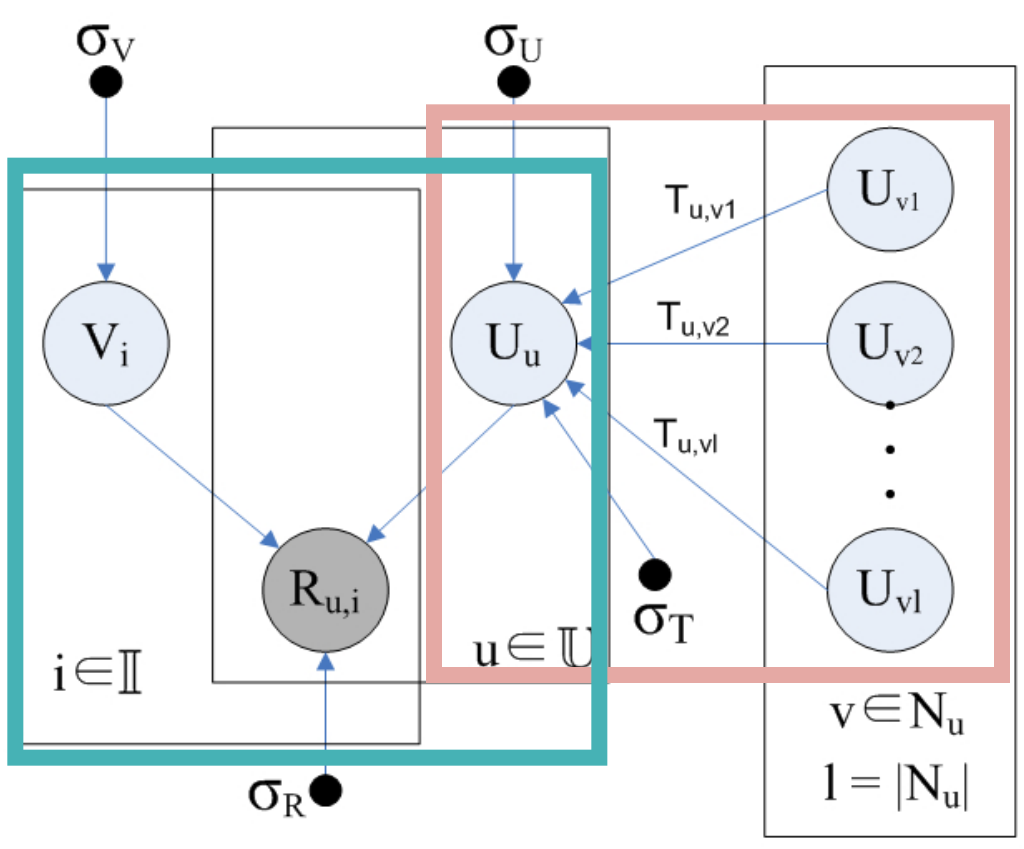
  - $T_{u,v}$ is the normalized trust value.



SOCIALMF (CONT.)

- Again, apply Bayesian inference:

$$p(U,V|R,T,\sigma_R^2,\sigma_T^2,\sigma_U^2,\sigma_V^2) \propto$$
$$p(R|U,V,\sigma_R^2)p(U|T,\sigma_U^2,\sigma_T^2)p(V|\sigma_V^2)$$
$$= \prod_{u=1}^{N}\prod_{i=1}^{M}\left[\mathcal{N}\left(R_{u,i}|g(U_u^T V_i),\sigma_r^2\right)\right]^{I_{u,i}^R}$$
$$\times \prod_{u=1}^{N}\mathcal{N}\left(U_u|\sum_{v\in N_u} T_{u,v}U_v,\sigma_T^2\mathbf{I}\right)$$
$$\times \prod_{u=1}^{N}\mathcal{N}\left(U_u|0,\sigma_U^2\mathbf{I}\right)\times\prod_{i=1}^{M}\mathcal{N}\left(V_i|0,\sigma_V^2\mathbf{I}\right)$$



Take the log function, and is equivalent to minimize the right

$$\mathcal{L}(R,T,U,V) = \frac{1}{2}\sum_{u=1}^{N}\sum_{i=1}^{M} I_{u,i}^R(R_{u,i}-g(U_u^T V_i))^2$$
$$+\frac{\lambda_U}{2}\sum_{u=1}^{N}U_u^T U_u + \frac{\lambda_V}{2}\sum_{i=1}^{M}V_i^T V_i$$
$$+\frac{\lambda_T}{2}\sum_{u=1}^{N}\left((U_u-\sum_{v\in N_u}T_{u,v}U_v)^T(U_u-\sum_{v\in N_u}T_{u,v}U_v)\right)$$

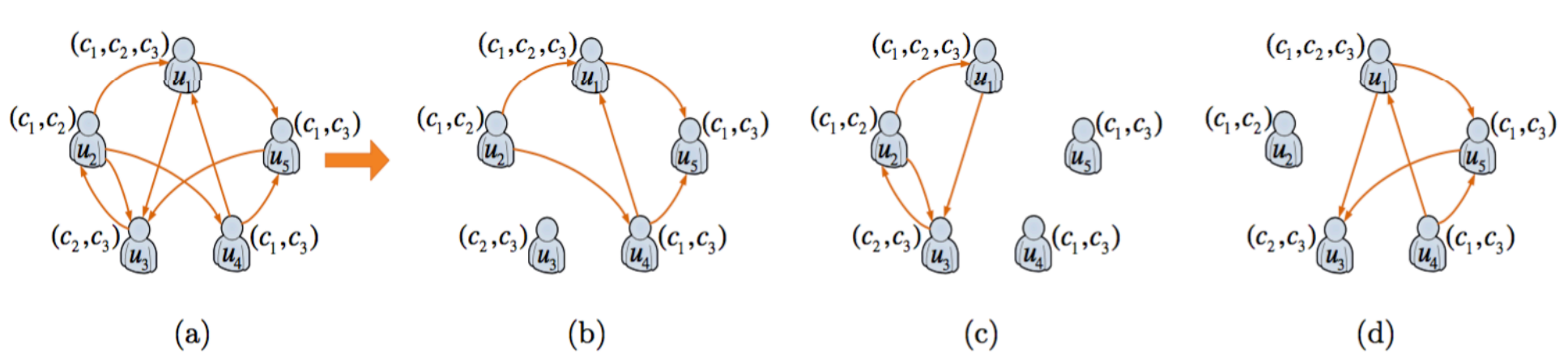
SOCIALMF (CONT.)

- **Trust propagation**
- **Learning is possible with the existence of the social network**
  - Partially depends on the observed ratings
  - Appropriate for cold start users
- **Issues**:
  - User may trust different users in different domains
  - Existing datasets mix the trust values of different categories together

CBR

- **Extend SocialMF**
  - Additional assumption: two users have different trust values concerning different item categories
  - Goal: train U^c and V^c, where c is the item category
- **1st step: trust circle inference**

  Regarding category c, user v is in the circle of user u, iff
  - In original social matrix: $S_{uv} > 0$
  - User u and v both have rated items in category c



CBR (CONT.)

- **2nd step: trust value assignment**
  - Equal trust (trust is averaged to #friends)
  - Expertise-based trust (an expert -> have given more ratings)
  - Trust splitting (trust proportional to #ratings in each category)
- **3rd step: model training**

To minimize the following

$$\mathcal{L}^{(c)}(R^{(c)}, Q^{(c)}, P^{(c)}, S^{(c)*}) = \frac{1}{2} \sum_{(u,i)\text{obs.}} \left(R^{(c)}_{u,i} - \hat{R}^{(c)}_{u,i}\right)^2 + \frac{\beta}{2} \sum_{\text{all } u} \left((Q^{(c)}_u - \sum_v S^{(c)*}_{u,v} Q^{(c)}_v)(Q^{(c)}_u - \sum_v S^{(c)*}_{u,v} Q^{(c)}_v)^\top \right) + \frac{\lambda}{2} \left(||P^{(c)}||^2_F + ||Q^{(c)}||^2_F \right), \quad (6)$$

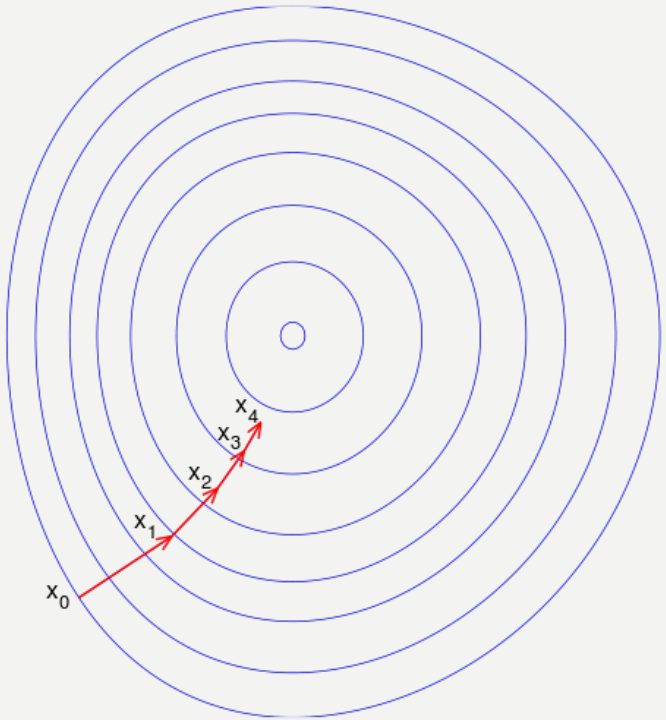
User feature vector w.r.t category c

GRADIENT DESCENT

- An optimization algorithm to find out the **local minimum** (or maximization)
- F(x) decreases fastest from x_n to x_{n+1} if it goes down in the direction of the negative gradient of F at x_n
- One starts with a guess of x_0 for a local minimum F, and considers the sequence $x_0, x_1, x_2 \ldots$, s.t.

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \gamma_n \nabla F(\mathbf{x}_n),\ n \geq 0.$$

- A particular choice of γ_n can ensure the convergence to a lcoal minimum



GRADIENT DESCENT (CONT.)

- Apply to find **latent user feature U** and **latent item feature V** to minimize the objective cost function L
- Update Ui and Vj in iteration n
  - $Ui = Ui - \gamma_n * \frac{\partial L}{\partial Ui}$
  - $Vj = Vj - \gamma_n * \frac{\partial L}{\partial Vj}$
- For example, in RSTE, the first derivatives of L w.r.t Ui and Vj have the RHS formula

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial U_i} = & \alpha \sum_{j=1}^{n} I_{ij}^{R} g'(\alpha U_i^T V_j + (1-\alpha) \sum_{k \in \mathcal{T}(i)} S_{ik} U_k^T V_j) V_j \\
& \times (g(\alpha U_i^T V_j + (1-\alpha) \sum_{k \in \mathcal{T}(i)} S_{ik} U_k^T V_j) - R_{ij}) \\
& + (1-\alpha) \sum_{p \in \mathcal{B}(i)} \sum_{j=1}^{n} I_{pj}^{R} g'(\alpha U_p^T V_j + (1-\alpha) \sum_{k \in \mathcal{T}(p)} S_{pk} U_k^T V_j) \\
& \times (g(\alpha U_p^T V_j + (1-\alpha) \sum_{k \in \mathcal{T}(p)} S_{pk} U_k^T V_j) - R_{pj}) S_{pi} V_j + \lambda_U U_i, \\
\frac{\partial \mathcal{L}}{\partial V_j} = & \sum_{i=1}^{m} I_{ij}^{R} g'(\alpha U_i^T V_j + (1-\alpha) \sum_{k \in \mathcal{T}(i)} S_{ik} U_k^T V_j) \\
& \times (g(\alpha U_i^T V_j + (1-\alpha) \sum_{k \in \mathcal{T}(i)} S_{ik} U_k^T V_j) - R_{ij}) \\
& \times (\alpha U_i + (1-\alpha) \sum_{k \in \mathcal{T}(i)} S_{ik} U_k^T) + \lambda_V V_j,
\end{aligned}
$$

IMPLEMENTATION SET UP

- **Datasets**
  - Yelp: crowd-sourced ratings and reviews of local business
  - #Users: 8351
  - #Items: 84653
  - Training: 80% Testing: 20%

- **Evaluation Metrics**
  - Mean Absolute Error (MAE)
  - Root Mean Square Error (RMSE)
  - Baseline: PMF

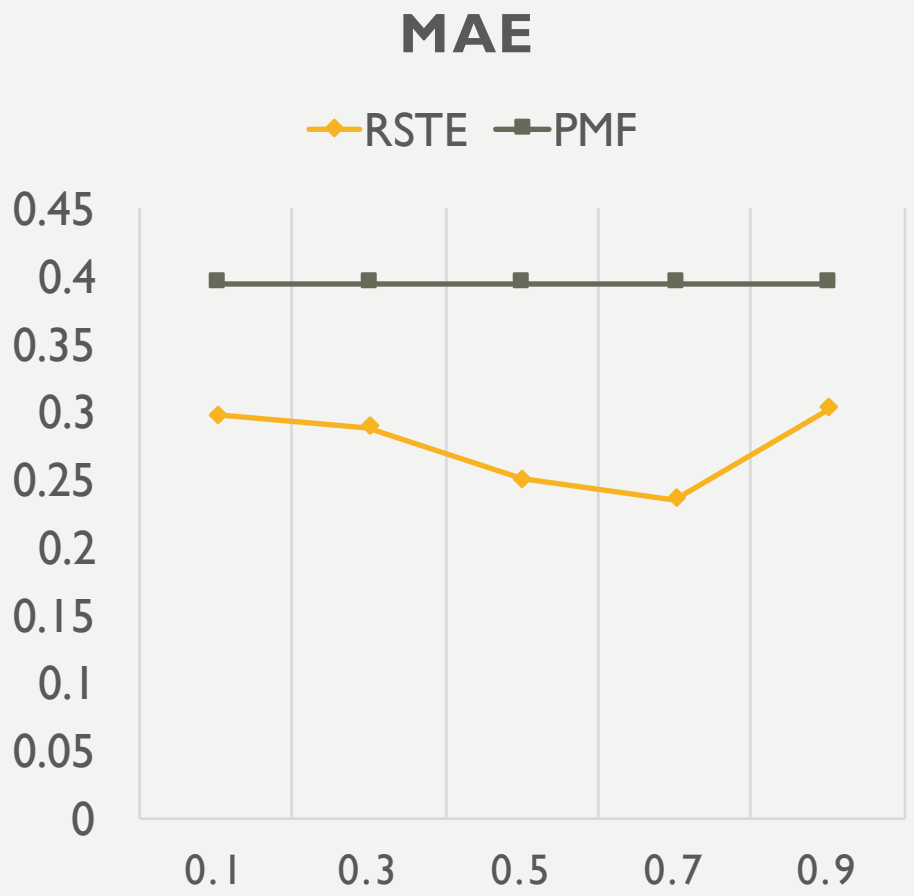
$$MAE = \frac{\sum_{i,j} |r_{i,j} - \widehat{r}_{i,j}|}{N}$$

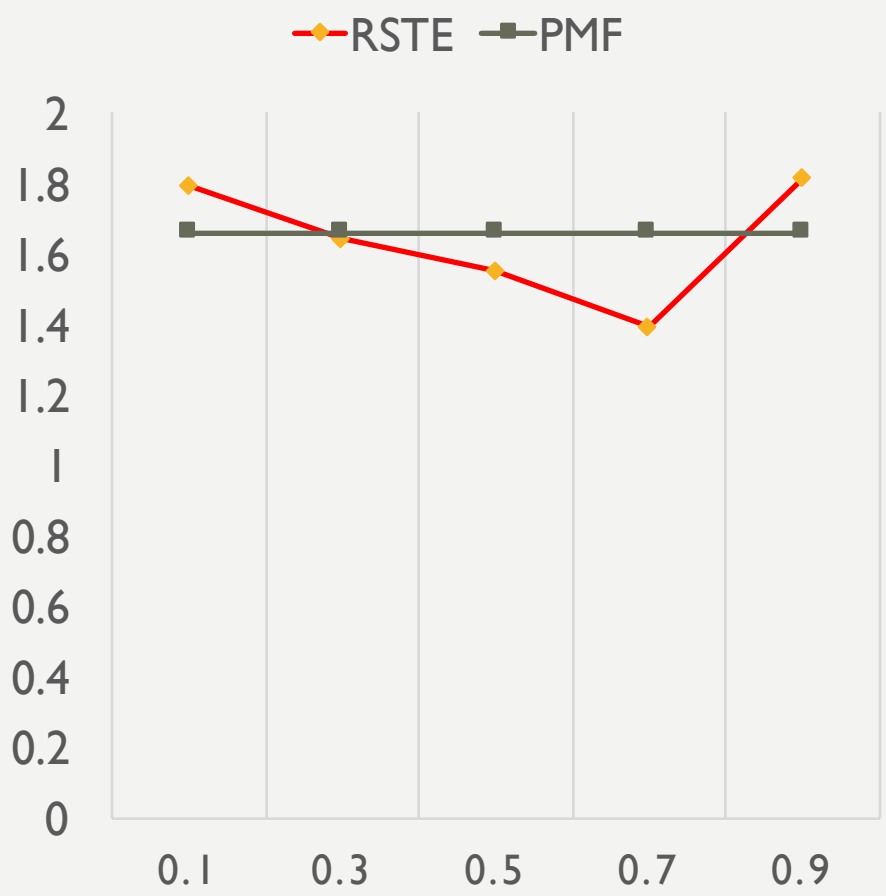
$$RMSE = \sqrt{\frac{\sum_{i,j} (r_{i,j} - \widehat{r}_{i,j})^2}{N}}.$$

- **Model parameters**
  - RSTE: α
  - SocialMF/CBR: λ_T
  - Latent feature dimension K

RESULTS ANALYSIS

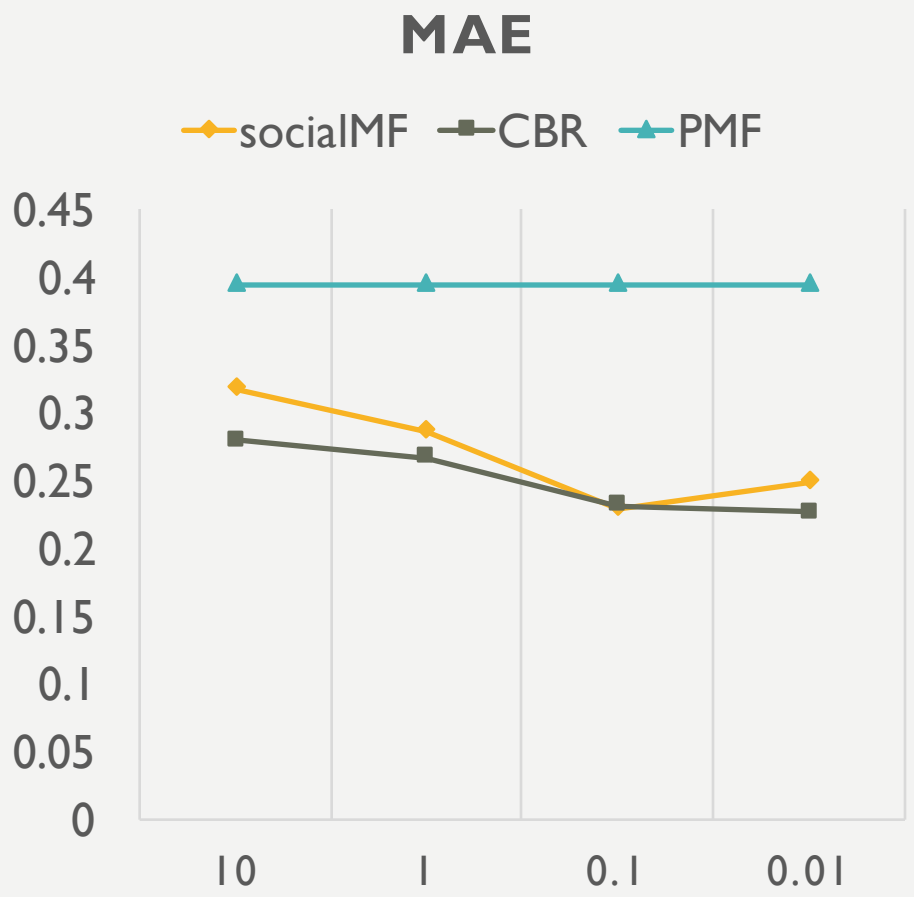
- Impact of trust weight α $(\lambda_U = \lambda_V = 0.01, K = 5)$

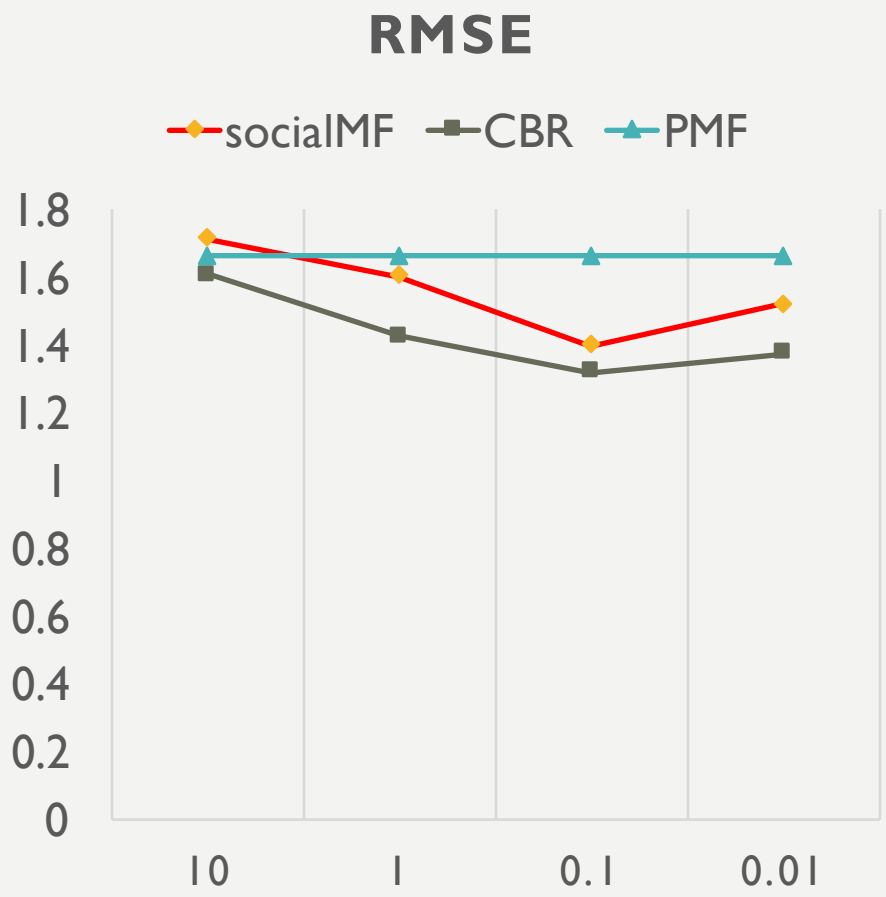




RESULTS ANALYSIS

- Impact of trust weight $\lambda_T (\lambda_U = \lambda_V = 0.01, K = 5)$





RESULTS ANALYSIS

- Impact of latent feature dimension K ($\lambda_U = \lambda_V = 1$)

	K = 3				K = 5			
Models	PMF	RSTE	SocialMF	CBR	PMF	RSTE	SocialMF	CBR
MAE	0.412	0.238	0.232	0.230	0.39	0.234	0.229	0.231
RMSE	1.671	1.42	1.367	1.223	1.658	1.39	1.364	1.216
TIME(s)	3.73	5.16	3.78	9.34	3.74	5.21	3.81	9.74

REFERENCES

[1] Ma, Hao, Irwin King, and Michael R. Lyu. "Learning to recommend with social trust ensemble." Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2009.

[2] Jamali, Mohsen, and Martin Ester. "A matrix factorization technique with trust propagation for recommendation in social networks." Proceedings of the fourth ACM conference on Recommender systems. ACM, 2010.

[3] Yang, Xiwang, Harald Steck, and Yong Liu. "Circle-based recommendation in online social networks." Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012.

THANK YOU !