Recommendation Based on Influence Sets

Jian CHEN (ellachen@scut.edu.cn)

@ South China University of Technology

Jian YIN

@ Sun-Yat Sen University, P.R.China

Presentation:

Jiangtao REN, @Sun-Yat Sen Univ. P.R.China

Presentation Outline

- Backgrounds
- Existing work and limitations
- Our proposal: *RIS*
- Experimental results
- Conclusions

Presentation Outline

- Backgrounds
- Existing work and limitations
- Our proposal: *RIS*
- Experimental results
- Conclusions

Backgrounds

- *Recommender Systems*
  - Apply knowledge discovery techniques to dramatically reduce the useless information and help us to find the information which is most valuable to us, usually during a live interaction.
- *Collaborative Filtering*
  - "Based on the premise that people looking for information should be able to make use of what others have already found and evaluated." (Maltz & Ehrlich, 1995)

Collaborative Filtering

- User-based CF Algorithm
  - try to predict the opinion the user will have on the different items and be able to recommend the "best" items to each user based on the user's previous likings and the opinions of other like minded users.
  - limitations:
    - Sparsity – evaluation of large item sets, users purchases are under 1%.
    - Scalability - nearest neighbor require computation that grows with both the number of users and the number of items.
    - New item problem - new or recently added items lacking enough visited or rated records by a sufficient number of users

Collaborative Filtering

- Item-based CF Algorithm
  - looks into the set of items the target user has rated & computes how similar they are to the target item and then selects k most similar items.
  - prediction is computed by taking a weighted average on the target user’s ratings on the most similar items.
  - limitations:
    - avoids the *scalability* problems by first exploring the relatively static relationships between the items rather than the users.
    - still suffers from the problems associated with data *sparsity*, and it still lack the ability to provide recommendations for *newly added items*.

Presentation Outline

- Backgrounds
- Existing work and limitations
- Our proposal: *RIS*
- Experimental results
- Conclusions

Problem Description

- List of *m* users and a list of *n* Items .

$m \times n$ user-item ratings matrix M

	$\mathbf{i_1}$	…	$\mathbf{i_k}$	…	$\mathbf{i_n}$
$\mathbf{u_1}$	$W_{1,1}$	…	$W_{1,k}$	…	*N/A*
…	…	…	…	…	…
$\mathbf{u_j}$	$W_{j,1}$	…	?	…	$W_{j,n}$
…	…	…	…	…	…
$\mathbf{u_m}$	*N/A*	…	$W_{m,k}$	…	$W_{m,n}$

The CF Process

- Step 1: find the *k* nearest neighbor for each item among the columns of matrix *M* and store the results as a similarity table.

There are three measures to calculate the similarity between items i & j in the ratings matrix M:

1. Standard cosine value

2. Adjusted Cosine Similarity

3. Correlation coefficient

Item Similarity Table

	***k*=1**	***k*=2**	***k*=3**
$\mathbf{i_1}$	i_2	i_3	i_4
$\mathbf{i_2}$	i_1	i_3	i_4
$\mathbf{i_3}$	i_2	i_1	i_4
$\mathbf{i_4}$	i_2	i_1	i_3

The CF Process

- Step 2: perform *kNN* query to find the *k* most similar items for computing the probable ratings $W_{a,t}$ of *it* as the opinion values provided by the active user u_a.

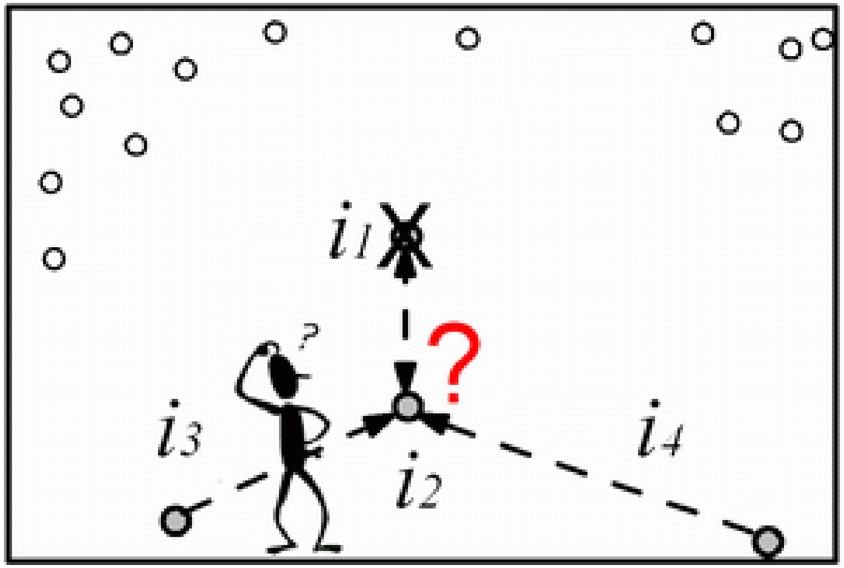
$$W_{a,t} = \frac{\sum_{j=1}^{k}\left(\underline{W_{a,j}} \times sim\left(i_j, i_t\right)\right)}{\sum_{j=1}^{k} sim\left(i_j, i_t\right)}$$

Limitations

- If target user u_a has no ratings for all k nearest neighbors of some item (such as i_2), then the traditional item-based CF cannot produce prediction ratings for this item, shown as:

For i_2 and $k=1$

	i_1	i_2	i_3	i_4
u_2	*N/A*	?	*4*	*5*



Presentation Outline

- Backgrounds
- Existing work and limitations
- Our proposal: *RIS*
- Experimental results
- Conclusions

Our proposal: *RIS*

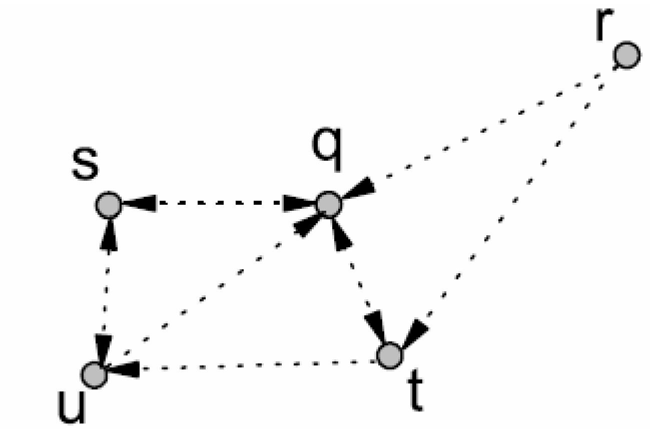
- In essential, item-based CF approach utilizes the relationships among items.
- The traditional CF approaches are mono-directional (just find the *k* most influential items for target item i_t).
- They ignore target item also has ***influence*** for other items.
  - For example, the appearance of mobile phones with camera certainly will influence the prices and sales of those without camera.

Our proposal: *RIS*

- a novel item-based CF approach
- based on the concept of *Influence Sets*
- combine the effects of *k* nearest neighbors with *reverse k’ nearest neighbors* for a target item
- alleviate the dataset sparsity problem effectively
- achieve better prediction accuracy than traditional item-based CF approach

The Concept of Influence Set

- The *influence set* of a given data point *q*
  - all the data points that have *q* as one of their *k* nearest neighbors in a data set. $RkNN(q) = \{p \in S \mid q \in kNN(p)\}$
  - Notice *k* nearest neighbors and reverse *k* nearest neighbors are *NOT SYMMETRIC.*



$2NN(q) = \{ s, t \}$
$R2NN(q) = \{ r, s, t, u \}$

2*NN* and *R*2*NN* examples

(An arrow from point *i* to point *j* indicates that *j* is nearest neighbors of *i*)

Example

- For item i_2 and k = 1

Item Similarity Table

	***k*=1**	***k*=2**	***k*=3**
$\mathbf{i_1}$	i_2	i_3	i_4
$\mathbf{i_2}$	i_1	i_3	i_4
$\mathbf{i_3}$	i_2	i_1	i_4
$\mathbf{i_4}$	i_2	i_1	i_3

$NN(i_2)=\{ i_1 \}$

$RNN(i_2)=\{ i_1, i_3, i_4 \}$

$|NN(i_2)|=1$

$|RNN(i_2)|=3$

enhances the density of item's information

Example

- For item i_2 and k = 1, if target user u_a has no ratings for i_1 (k nearest neighbors of i_2), then RIS can use the influence set of i_2 (i_3 and i_4) to compute the prediction value of i_2.

	i_1	i_2	i_3	i_4
u_2	*N/A*	**?**	*4*	*5*

The *RIS* Process

- Step 1: same to traditional CF
- Step 2: *RIS* defines two different prediction generation formulas, named *RIS1* and *RIS2*, integrating the effects of *k* nearest neighbors with reverse *k'* nearest neighbors of item i_t

Prediction Generation

- *RIS1:*

$$W_{a,t} = \frac{\sum_{i_j \in kNN(i_t)} \left(W_{a,j} \times sim\left(i_j, i_t\right)\right) + \boxed{\sum_{i_{j'} \in Rk'NN(i_t)} \left(W_{a,j'} \times sim\left(i_{j'}, i_t\right)\right)}}{\sum_{i_j \in kNN(i_t)} sim\left(i_j, i_t\right) + \boxed{\sum_{i_{j'} \in Rk'NN(i_t)} sim\left(i_{j'}, i_t\right)}}$$

Prediction Generation

- *RIS2:*

$$W_{a,t} = \underline{\alpha} \times \frac{\sum\limits_{i_j \in kNN(i_t)} \left(W_{a,j} \times sim\left(i_j, i_t\right)\right)}{\sum\limits_{i_j \in kNN(i_t)} sim\left(i_j, i_t\right) + \sum\limits_{i_{j'} \in Rk'NN(i_t)} sim\left(i_{j'}, i_t\right)}$$

$$+ \left(\underline{1 - \alpha}\right) \times \frac{\sum\limits_{i_{j'} \in Rk'NN(i_t)} \left(W_{a,j'} \times sim\left(i_{j'}, i_t\right)\right)}{\sum\limits_{i_j \in kNN(i_t)} sim\left(i_j, i_t\right) + \sum\limits_{i_{j'} \in Rk'NN(i_t)} sim\left(i_{j'}, i_t\right)}$$

Presentation Outline

- Backgrounds
- Existing work and limitations
- Our proposal: *RIS*
- Experimental results
- Conclusions

The Data Sets

- MovieLens -- 100K dataset from the MovieLens recommendation systems (www.movielens.org)
- Each user in this dataset have rated at least 20 movies.
- The dataset contains over 100,000 ratings, converted into a user-movie matrix *R* that had 943 rows and 1682 columns using a 1(bad)-5(excellent) numerical scale .

The Data Sets

- a variable called *x* that determines what percentage of data is used as training and test sets. *x=0.8* means:
  - 80% of the data - training set.
  - 20% 0f the data - test set.

- the *sparsity level* of the data set
  = 1- (nonzero entries/total entries)
  = 0.93695. *high*

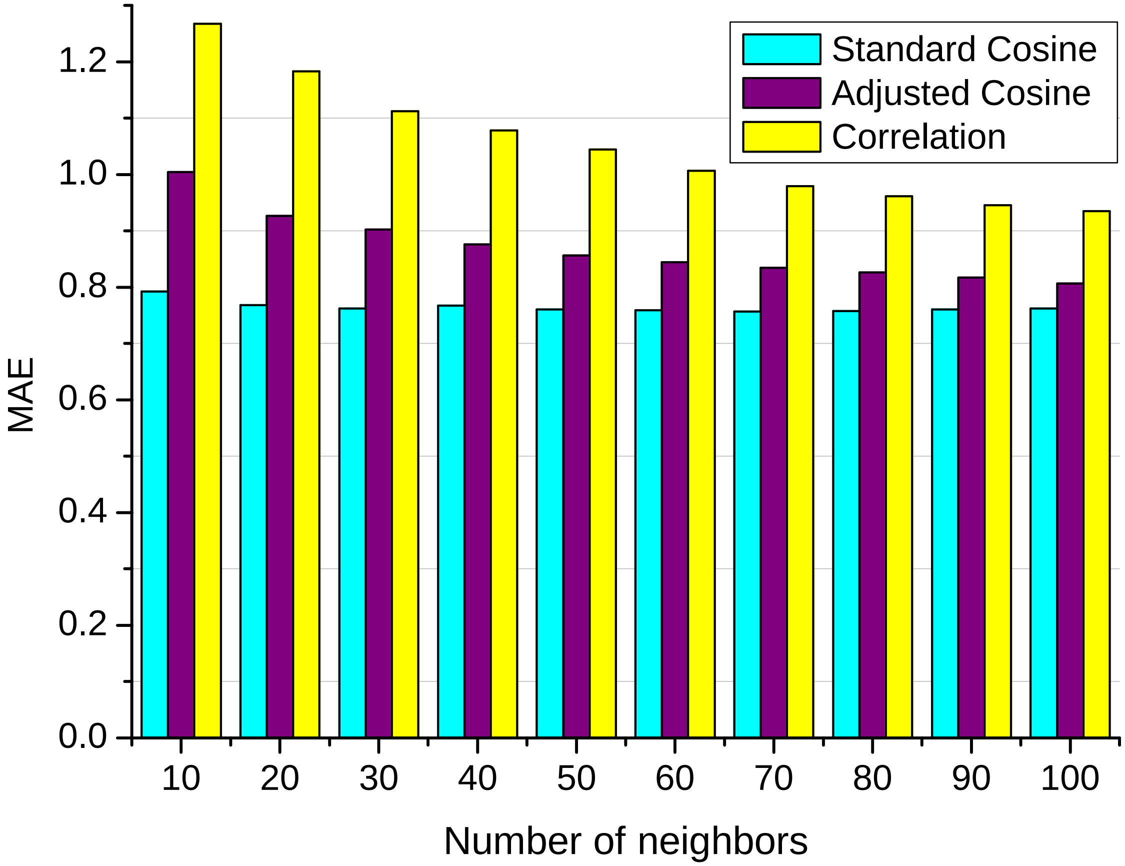
Evaluation Metrics

- MAE – Mean Absolute Error
  - deviation of recommendations from their true user-specified ratings.

$$MAE = \frac{\sum_{i=1}^{N} \left| p_i - r_i \right|}{N}$$

  - The lower the *MAE*, the more accurately the recommendation algorithm predicts user ratings.

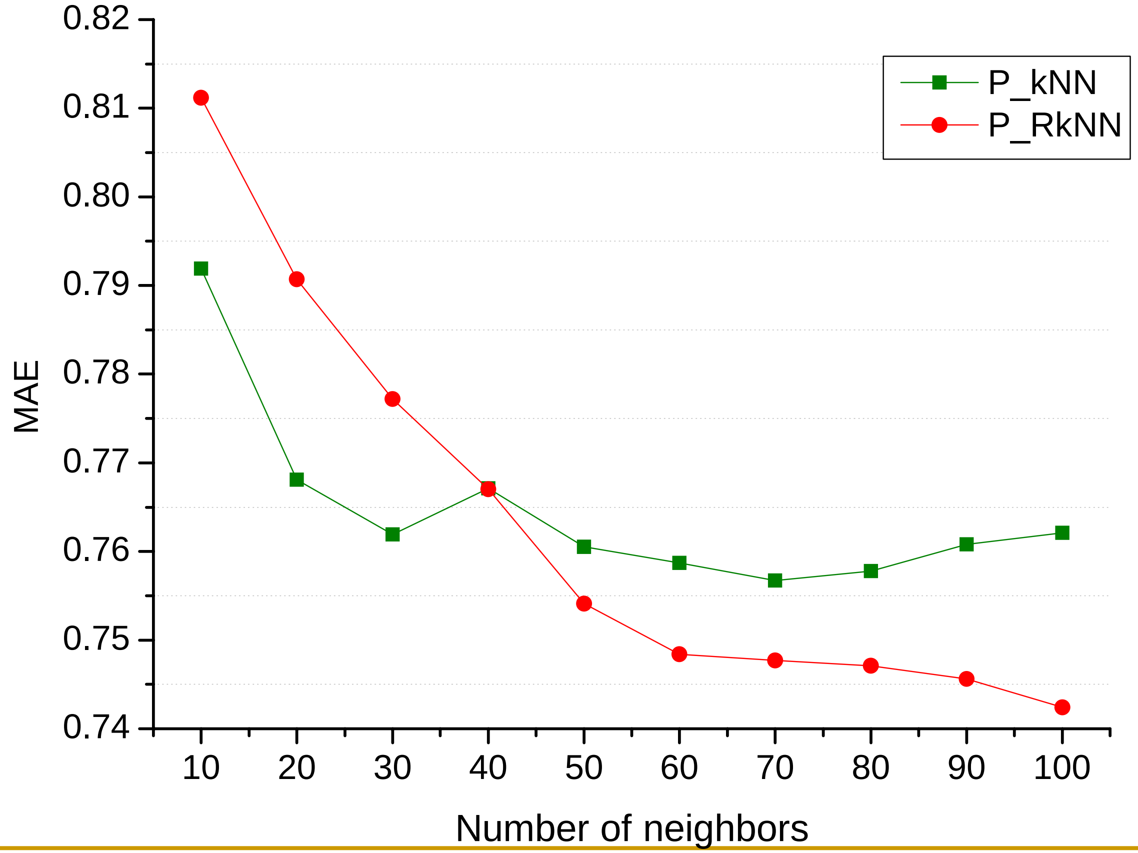
Comparison of Similarity Measure Method



Select *standard cosine similarity* for the rest of our experiments.

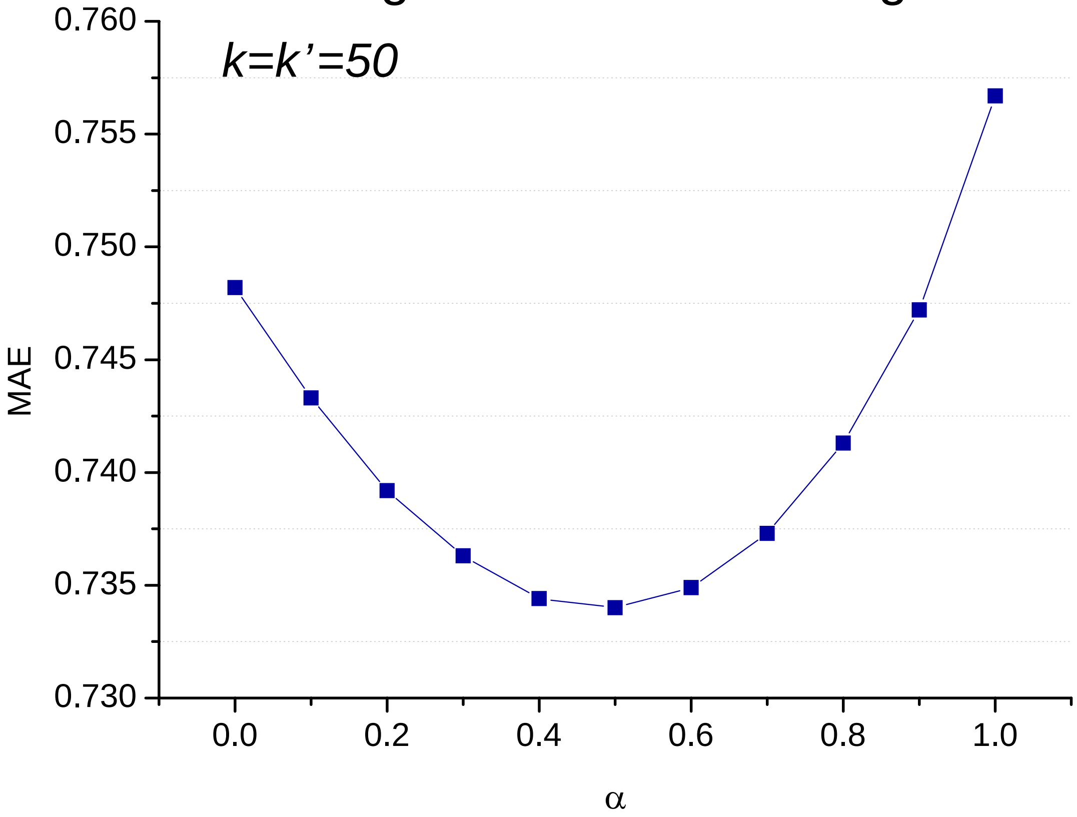
Sensitivity of *k*NN/*Rk*NN Ratio (1)

- P_kNN -- totally based on *kNN*
- P_RkNN -- totally based on *RkNN*



Sensitivity of *k*NN/R*k*NN Ratio (2)

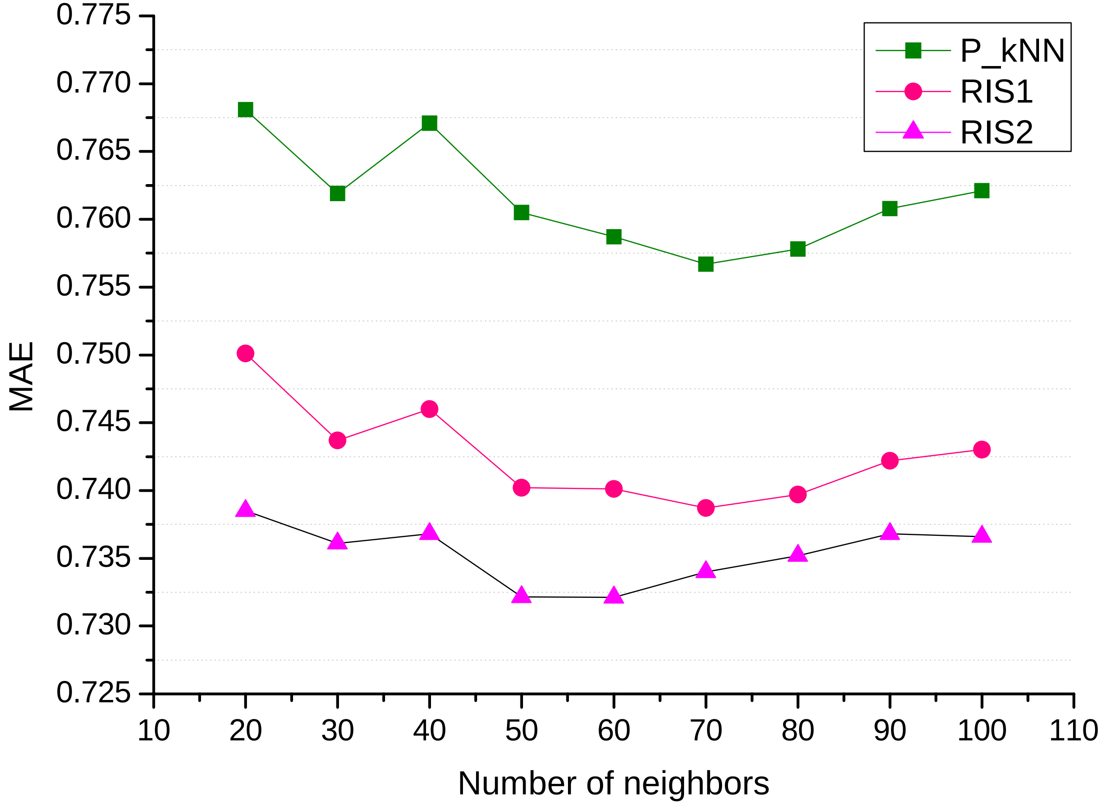
- A value of α = 0.7 would indicate that 70% *k* nearest neighbors and 30% reverse *k'* nearest neighbors were used to generate prediction.



Select α *=0.5* as an optimum value for our subsequent experiments.

Comparison of prediction quality of RIS and item-based CF algorithms

- P_kNN- the traditional item-based CF algorithm which totally based on *kNN*



RIS can achieve better recommendation quality.

Presentation Outline

- Backgrounds
- Existing work and limitations
- Our proposal: *RIS*
- Experimental results
- Conclusions

Conclusions

- *RIS* is a novel item-based CF approach based on the concept of Influence Sets
- *RIS* can alleviate the dataset sparsity problem effectively
- *RIS* achieves better prediction accuracy than traditional item-based CF approach
- *RIS* has the capability to deal with "new item problems" based on full integration of domain ontology and usage patterns.

THANK YOU!

☺