TrustWalker: A Random Walk Model for Combining Trust-based & Item-based Recommendation

Mohsen Jamali & Martin Ester,
Simon Fraser University, Canada, KDD-09

Present by: Menghong Han, Xiaofan Sun, Jie Tang

Outline

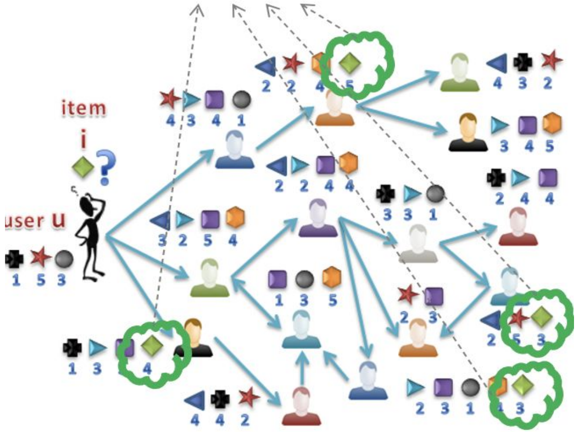
1. Problem Definition
2. Challenge
3. TrustWalker Model
4. Results & Evaluation
5. Comments & Improvement

PART THREE 1

Problem Definition

Problem Definition

$$\hat{r}_{u,i} = \frac{\sum_{v\in N_{u,i}} t_{u,v} r_{v,i}}{\sum_{v\in N_{u,i}} t_{u,v}}$$



Recommendation Problem:

Given user u and target item i, Predict the rating $r_{u,i}$

- **Collaborative Filtering (Item-based CF)**
  - Considers Users with Similar Rating Patterns
  - Aggregates the ratings of Similar Users

Problems with CF

- Requires Enough Ratings (Cold Start Users)
- Vulnerable to Attack Profiles

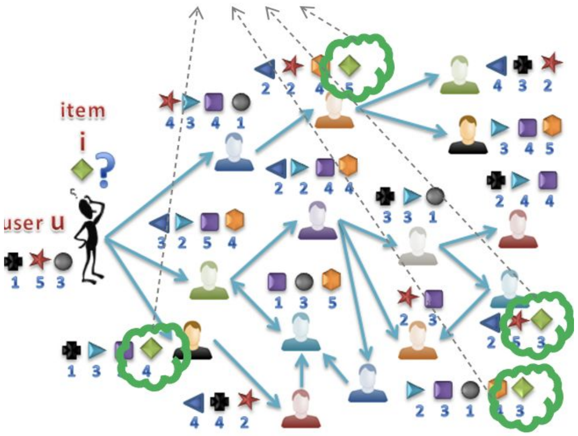
- **Trust-based Recommendation System**
  - Explores the trust network to find Raters
  - Aggregate the ratings from raters
  - Different weights for users

Motivations

- *Social Networks :*
  Independent source of information
- *Motivations of Trust-based Social Influence:*
  Users adopt the behavior of their friends

Problem Definition

$$\hat{r}_{u,i} = \frac{\sum_{v\in N_{u,i}} t_{u,v} r_{v,i}}{\sum_{v\in N_{u,i}} t_{u,v}}$$



❑ Trust-based Recommendation System (Cont.)

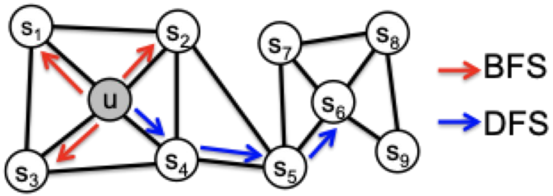
- **Tidal Trust**
  BFS to find raters at the closest distance
- **Mole Trust**
  BFS to find raters up to depth *max-depth*

Advantages

- Improving the coverage, percentage of < user, item > pairs in the test set
- Sovle cold start problems
- Attack resistance

Problems in Trust-based Recommendation

- Noisy data in far distances
- Low probability of finding rater at close distances



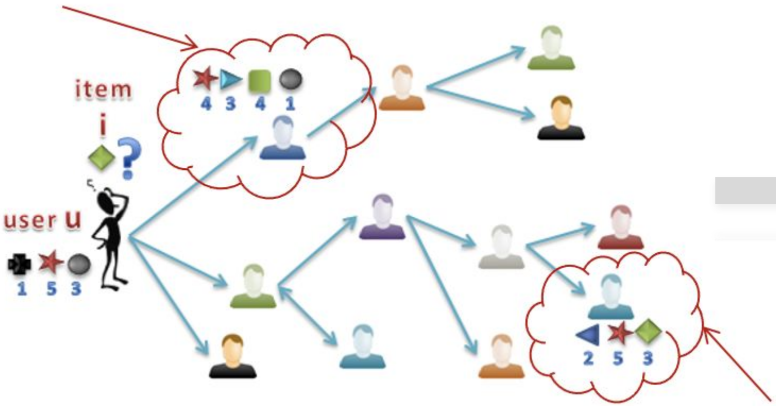
PART THREE 2

Challenge

Challenge

How Far to Go into Network?

Tradeoff between Precision and Recall (Coverage)



Trusted friends on similar items?

OR

Far neighbors on the exact target item?

TrustWalker Model

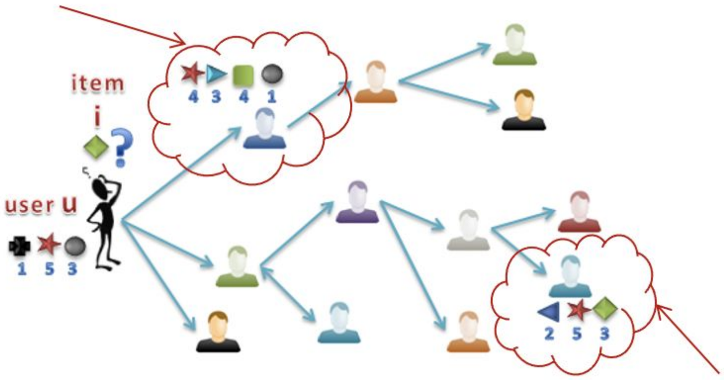
- ❑ Random Walk Model
- ❑ Combines Item-based and Trust-based Recommendation

PART THREE 3

Trustwalker Model Explained

Model Explained-- a single random walk

- ❖ starts from Source user u_0
- ❖ at step k, at node u:
  - if u has rated item i, return $r_{u,j}$
  - With probability $\phi_{u,i,k}$, we don't continue the random walk. We stay at node u and randomly select one of the items (j) similar to i rated by u and return $r_{u,j}$.
  - With probability $1 - \phi_{u,i,k}$, we continue our random walk and walk to another user v who is one of u's direct trusted neighbors ($v \in TU_u$).



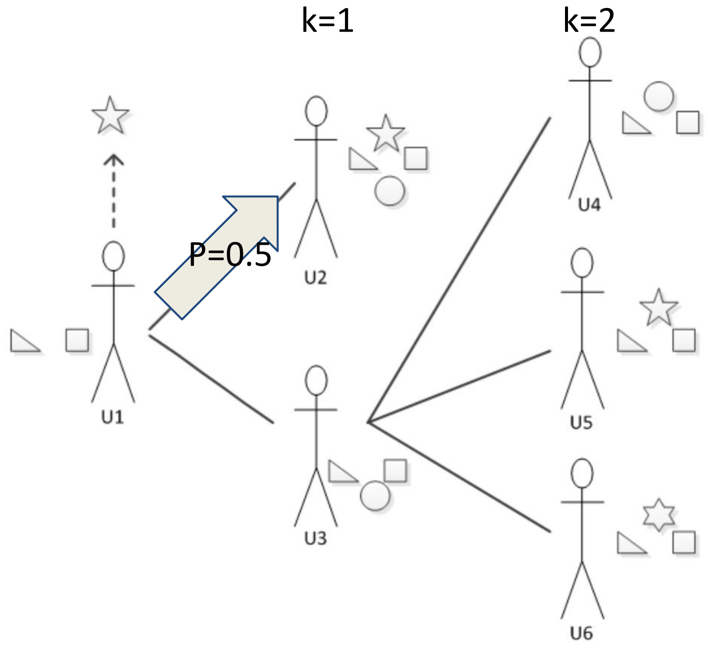
$$\phi_{u,i,k} = \max_{j \in RI_u} sim(i,j) \times \frac{1}{1 + e^{-\frac{k}{2}}}$$

Key Idea:

Based on the trust network, to find a rating on the exact item or a similar item and return the rating as our prediction

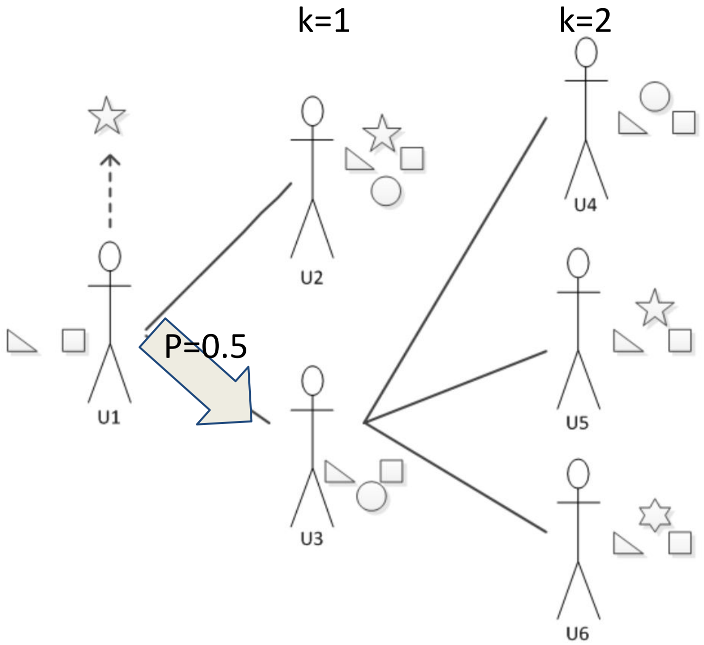
Model Explained -- equation sum up

prediction = Expected value of rating returned by random walk



Model Explained -- equation sum up

prediction = Expected value of rating returned by random walk



With probability $\phi_{u,i,k}$,
stay at node U3 and randomly choose one item(j) similar to item i rated by U3 and return r(u3,j)

And

With probability $1 - \phi_{u,i,k}$, continue to walk

set $max\text{-}depth = 6$.

Model Explained -- equation sum up

prediction = Expected value of rating returned by random walk

$$\widehat{r}_{u,i} = \sum_{\{(v,j)|R_{v,j}\}} P(XY_{u,i} = (v,j))\; r_{v,j}$$

$$P(XY_{u,i} = (v,j)) = \begin{cases} P(X_{u,i} = v)\phi_{u,i} P(Y_{v,i} = j) & v \neq u; i \neq j \\ P(X_{u,i} = v) & v \neq u; i = j \\ \phi_{v,i,1} P(Y_{v,i} = j), & v = u; i \neq j \end{cases}$$

Model Explained -- item similarity

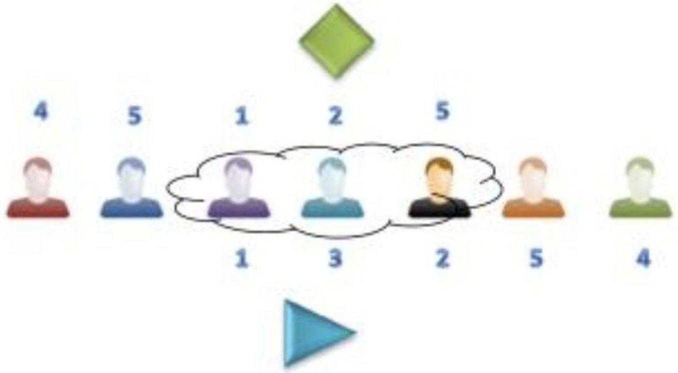
Using Pearson Correlation

Item Similarity---

$$sim(i,j) = \frac{1}{1 + e^{-\frac{|UC_{i,j}|}{2}}} \times corr(i,j)$$

set of common users who have rated items i and j;

Probability of having high correlation for pairs of items with few users in common is high.



PART THREE 4

Experiments

Experiments

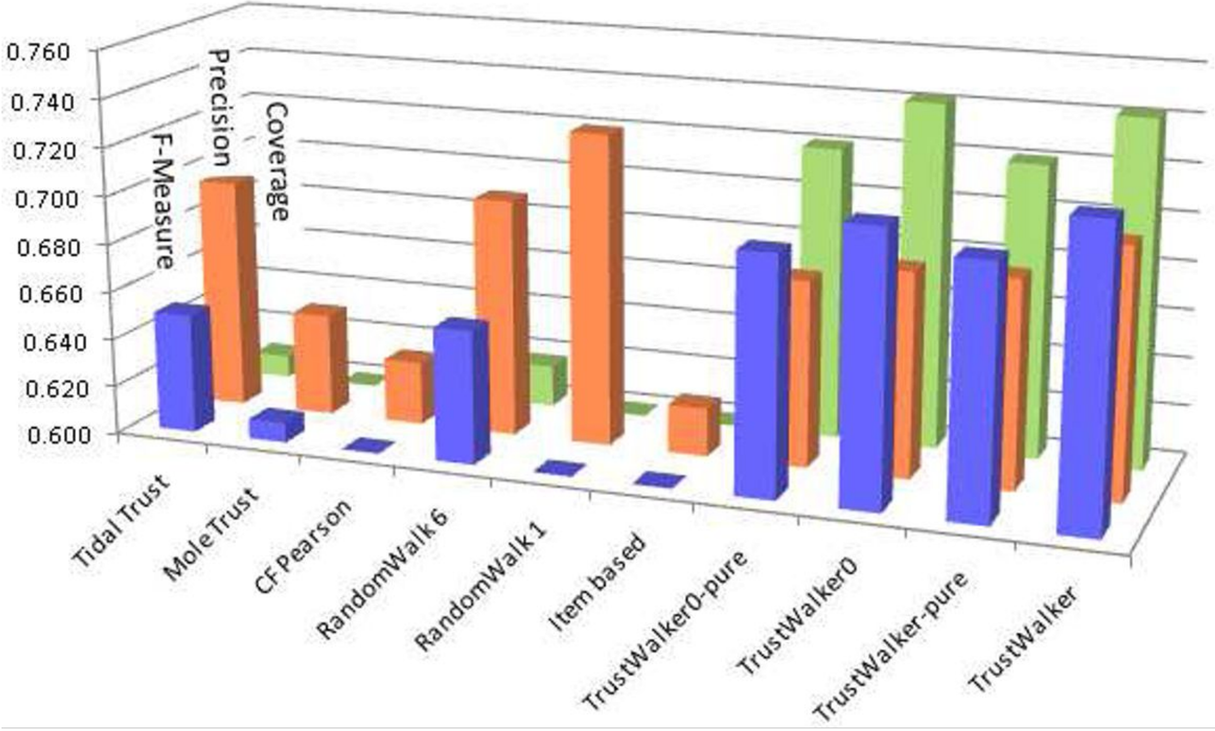
- **Epinions.com Data Set**
  – 49K total users, 24K cold start users (users with less than 5 ratings)
  –104K items, 575K ratings, 508K trust expressions
  – Binary trust, ratings in [1,5]

- **Evaluation Metrics**
  – RMSE
  – Coverage (percentage of pairs that we can predict)
  – Precision = 1- RMSE/4
  – FMeasure

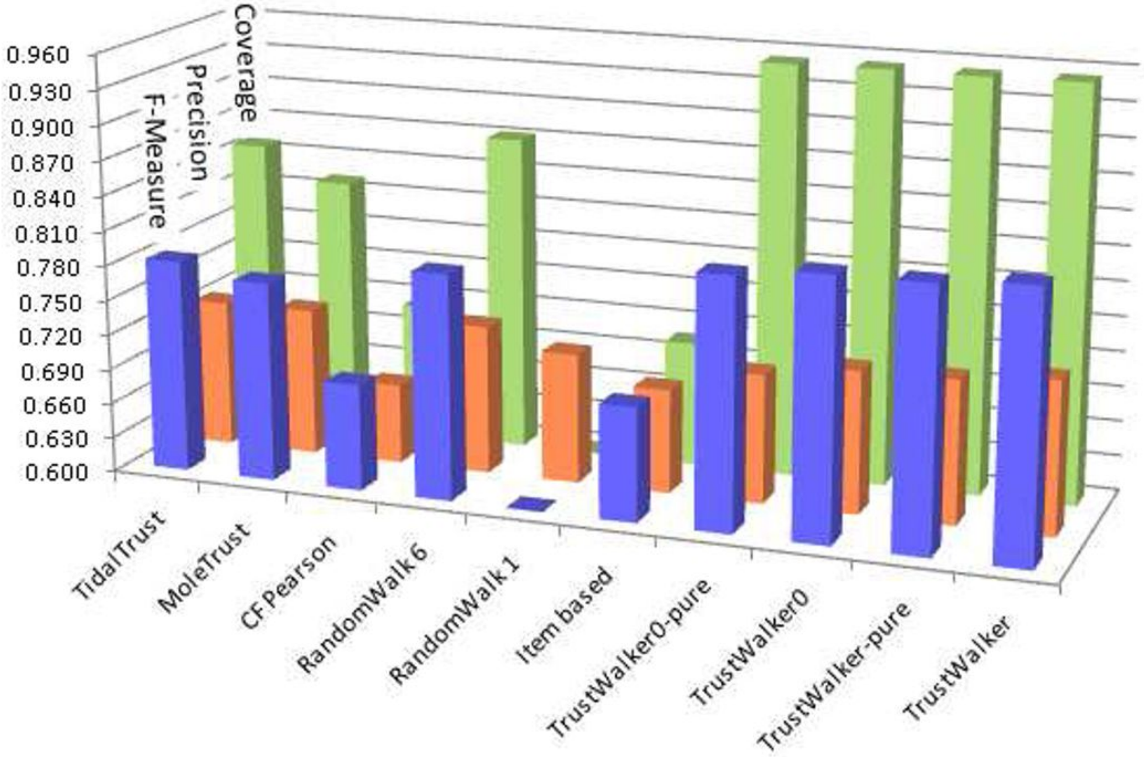
$$RMSE = \sqrt{\frac{\sum_{(u,i)|R_{u,i}} (r_{u,i} - \widehat{r}_{u,i})^2}{|\{(u,i)|R_{u,i}\}|}}$$

$$FMeasure = \frac{2 \times Precision \times Coverage}{Precision + Coverage}$$

Experiments - Cold Start Users



Experiments - All Users

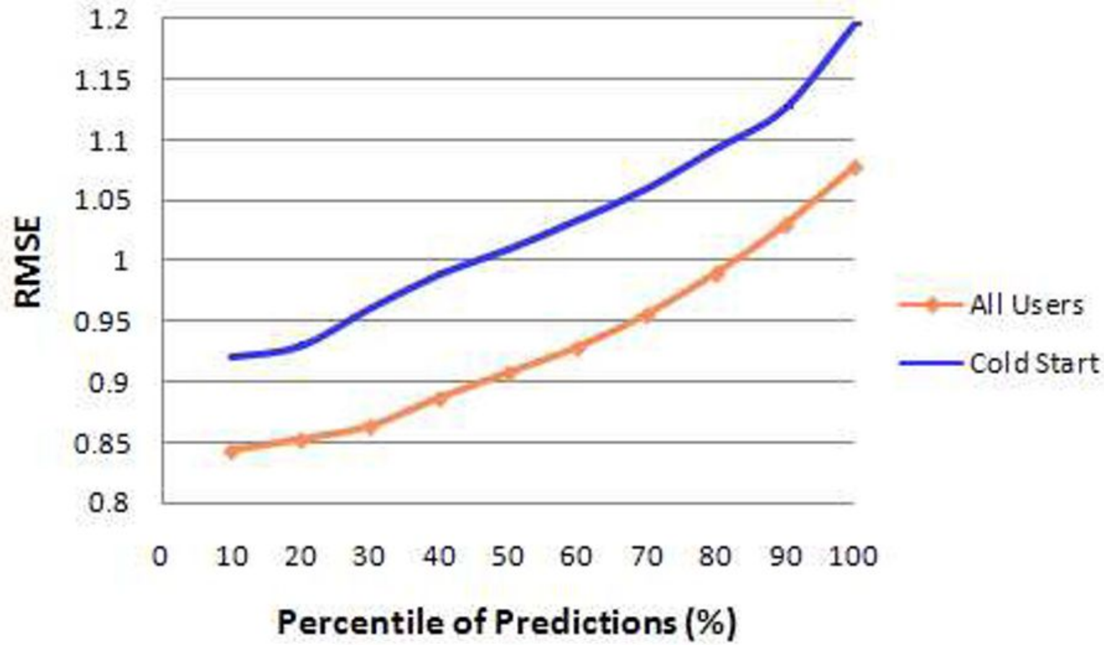


Experiments - Confidence

variance = $\sigma^2 = \frac{\sum_{i=1}^{T}(r_i - \overline{r})^2}{T}$

$$confidence = 1 - \frac{\sigma^2}{max\sigma^2}$$

- More confident Predictions have lower error



PART THREE 5

Optimization

Opitimization

- **Paper's Proposal**
  - – Top-N recommendation [RecSys'09]
  - – Distributed Recommender
  - – Context dependent trust

- **Our Proposal`**
  - – Item similarity measurement (*Pearson Correlation* might not be the best)
  - – With $\Phi_{u,i,k}$, the random walk stops, randomly select item *j* rated by *u* and return $r_{u,j}$

    1) select item *j* based on similarity scores

    2) if 2 more items are highly similar, set up a threshold, take average all the highly similar ones

    3) if all items have low similarity scores, exclude this random walk result (threshold, ex < 0.3)

Thank You