CONTEXT-RICH RECOMMENDATION: INTEGRATING LINKS, TEXT, AND SPATIO-TEMPORAL DIMENSIONS

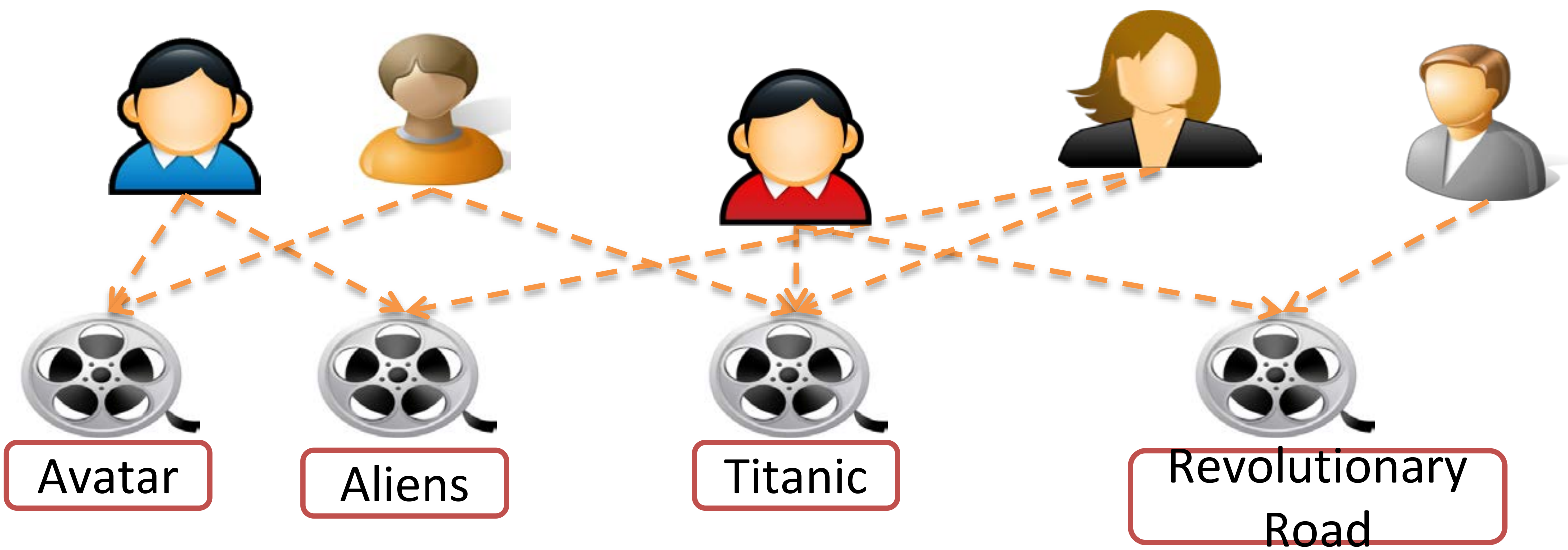
Yizhou Sun (University of California, Los Angeles)
Xiang Ren (University of Southern California)
Hongzhi Yin (The University of Queensland, Australia)

August 11, 2017

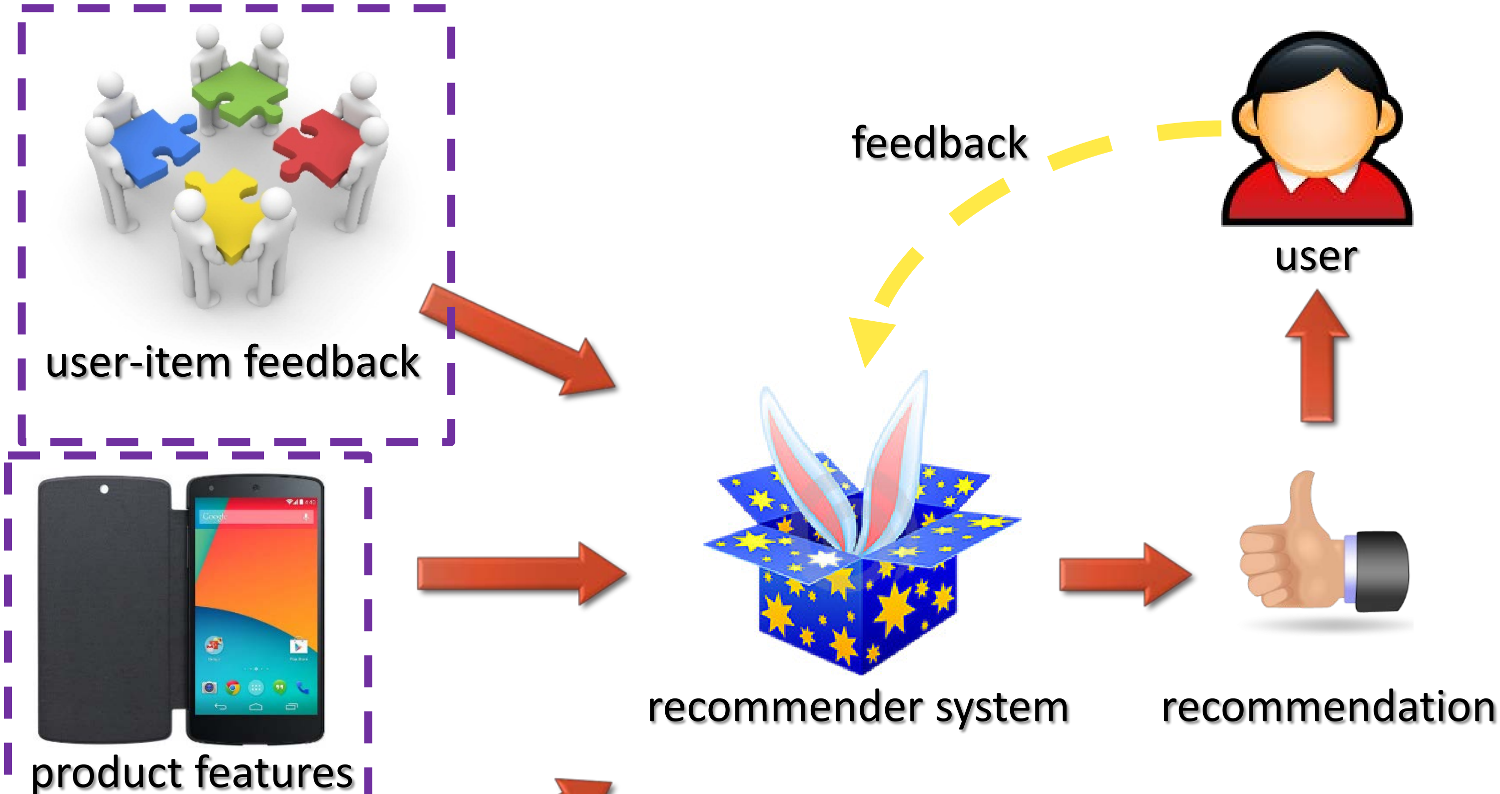
Outline

- Part I: Introduction and Preliminaries
- Part II: Recommendation in Heterogeneous Information Networks
- Part III: Recommendation in a Text-Rich Setting
- Part IV: Recommendation with Spatio-Temporal Information
- Part V: Research Frontiers and Summary

Traditional View of Recommendation



Recommendation Paradigm



Collaborative Filtering

E.g., K-Nearest Neighbor (Sarwar WWW'01), Matrix Factorization (Hu ICDM'08, Koren IEEE CS'09), Probabilistic Model (Hofmann SIGIR'03)

Content-Based Methods

Hybrid Methods

E.g., (Balabanovic Comm. ACM'97, ... SIGIR'02)

E.g., Content-Based CF (Antonopoulus ...), External Knowledge CF (Ma WSDM'11)

An Example of Traditional Method: Matrix Factorization

R: Rating Matrix

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	5	2		3		4		
u_2	4	3			5			
u_3	4		2				2	4
u_4								
u_5	5	1	2		4	3		
u_6	4	3		2	4		3	5

$\hat{R}$: Estimated Rating Matrix

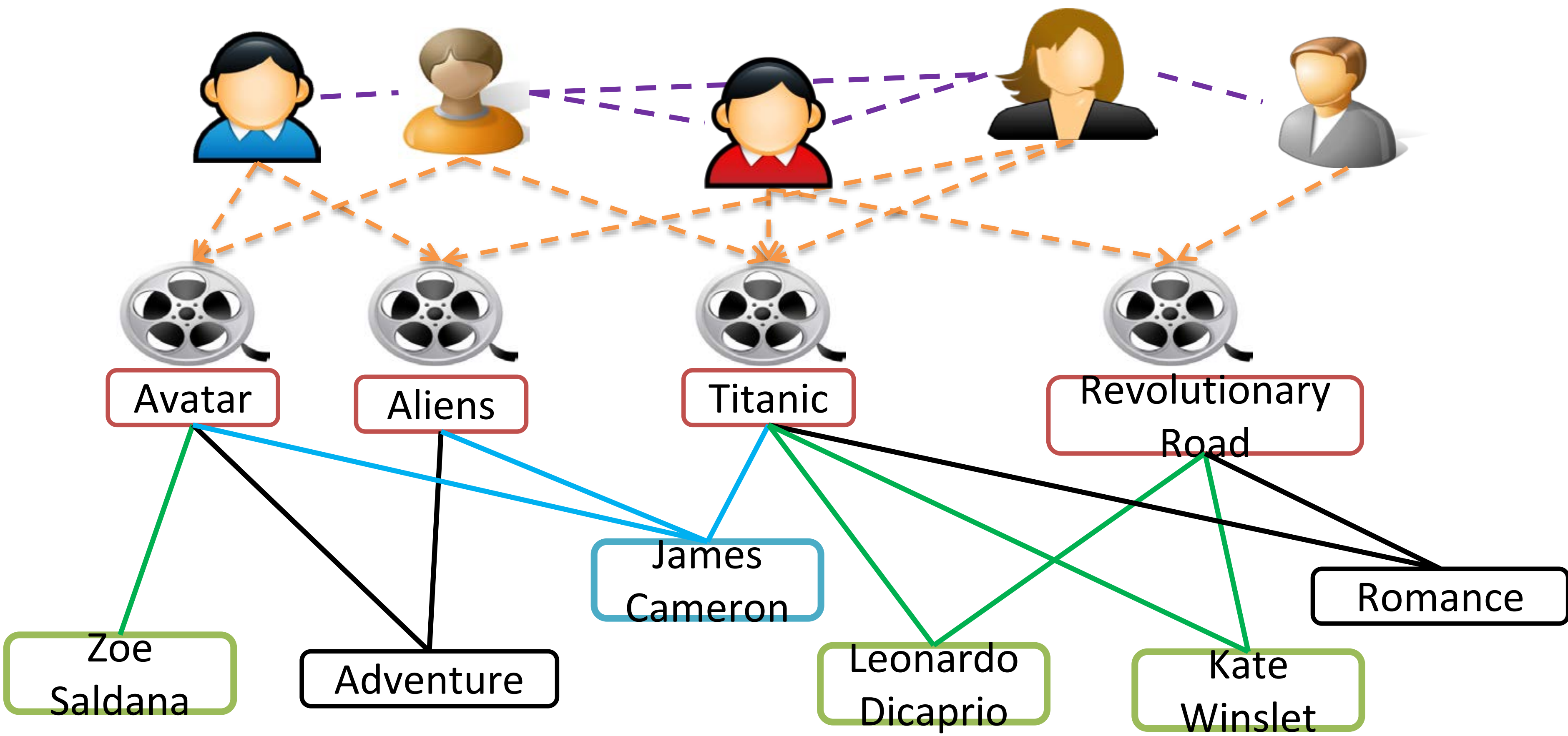
	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	5	2	**2.5**	3	**4.8**	4	**2.2**	**4.8**
u_2	4	3	**2.4**	**2.9**	5	**4.1**	**2.6**	**4.7**
u_3	4	**1.7**	2	**3.2**	**3.9**	**3.0**	2	4
u_4	**4.8**	**2.1**	**2.7**	**2.6**	**4.7**	**3.8**	**2.4**	**4.9**
u_5	5	1	2	**3.4**	4	3	**1.5**	**4.6**
u_6	4	3	**2.9**	2	4	**3.4**	3	5

$$U = \begin{bmatrix} 1.55 & 1.22 & 0.37 & 0.81 & 0.62 & -0.01 \\ 0.36 & 0.91 & 1.21 & 0.39 & 1.10 & 0.25 \\ 0.59 & 0.20 & 0.14 & 0.83 & 0.27 & 1.51 \\ 0.39 & 1.33 & -0.43 & 0.70 & -0.90 & 0.68 \\ 1.05 & 0.11 & 0.17 & 1.18 & 1.81 & 0.40 \end{bmatrix} \quad V = \begin{bmatrix} 1.00 & -0.05 & -0.24 & 0.26 & 1.28 & 0.54 & -0.31 & 0.52 \\ 0.19 & -0.86 & -0.72 & 0.05 & 0.68 & 0.02 & -0.61 & 0.70 \\ 0.49 & 0.09 & -0.05 & -0.62 & 0.12 & 0.08 & 0.02 & 1.60 \\ -0.40 & 0.70 & 0.27 & -0.27 & 0.99 & 0.44 & 0.39 & 0.74 \\ 1.49 & -1.00 & 0.06 & 0.05 & 0.23 & 0.01 & -0.36 & 0.80 \end{bmatrix}$$

Challenges

- How to address the data sparsity and cold start issues?
- How to integrate content information, such as text, into the recommendation?
- How to do spatio-temporal recommendation?

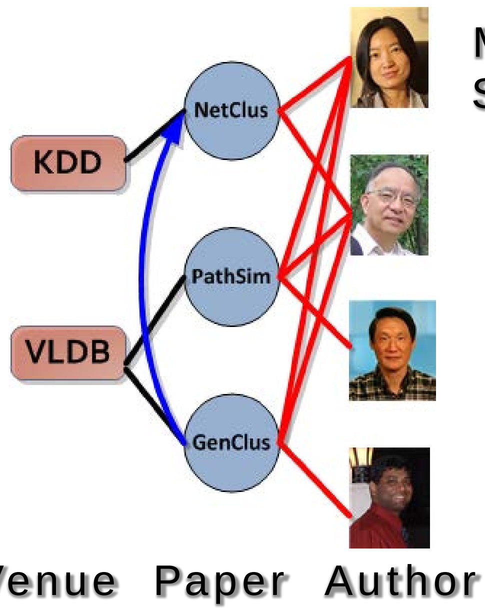
Solution: A Heterogeneous Information Network View of Recommendation



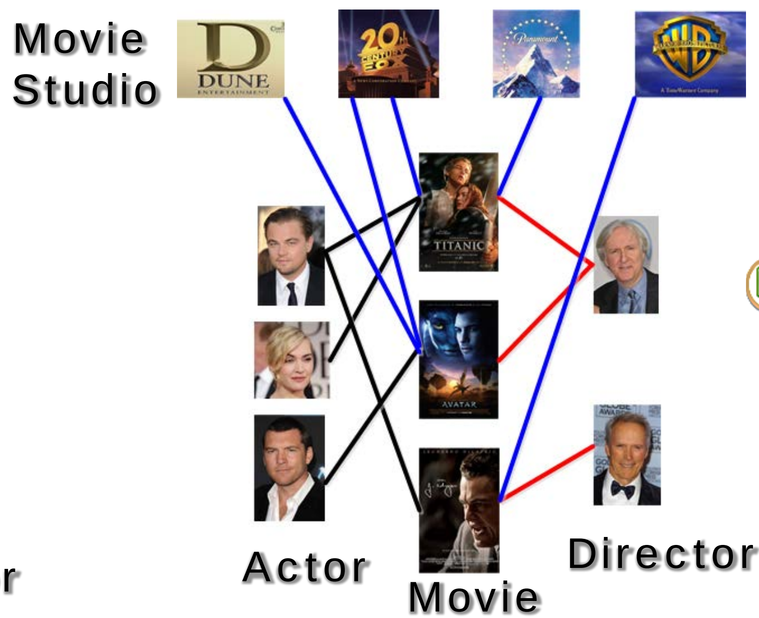
What Are Information Networks?

- A network where each node represents an entity (e.g., user in a social network) and each link (e.g., friendship) a relationship between entities.
  - Nodes/links may have attributes, labels, and weights.
  - Links may carry rich semantic information.

Heterogeneous Information Networks



DBLP Bibliographic Network

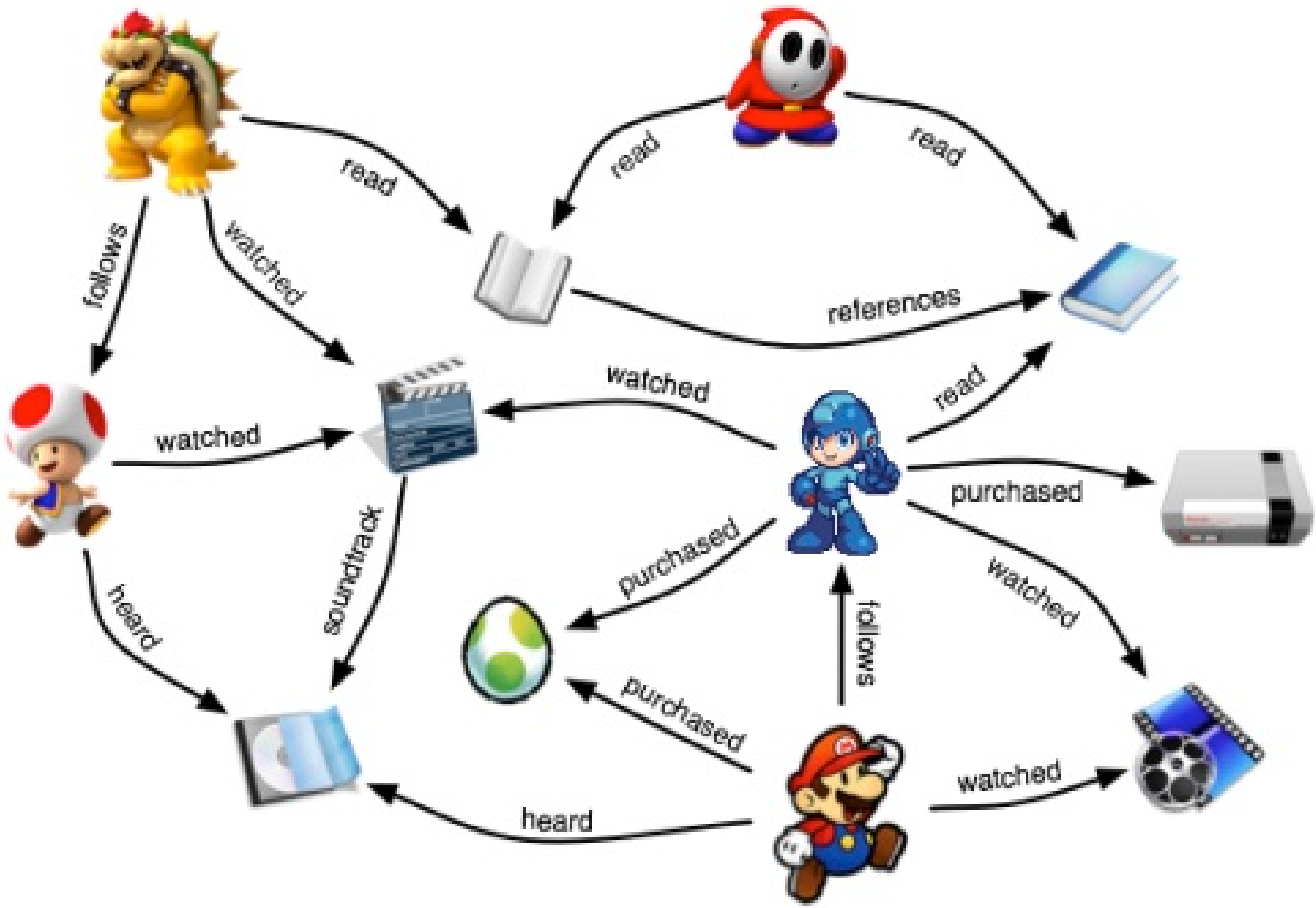


The IMDb Movie Network

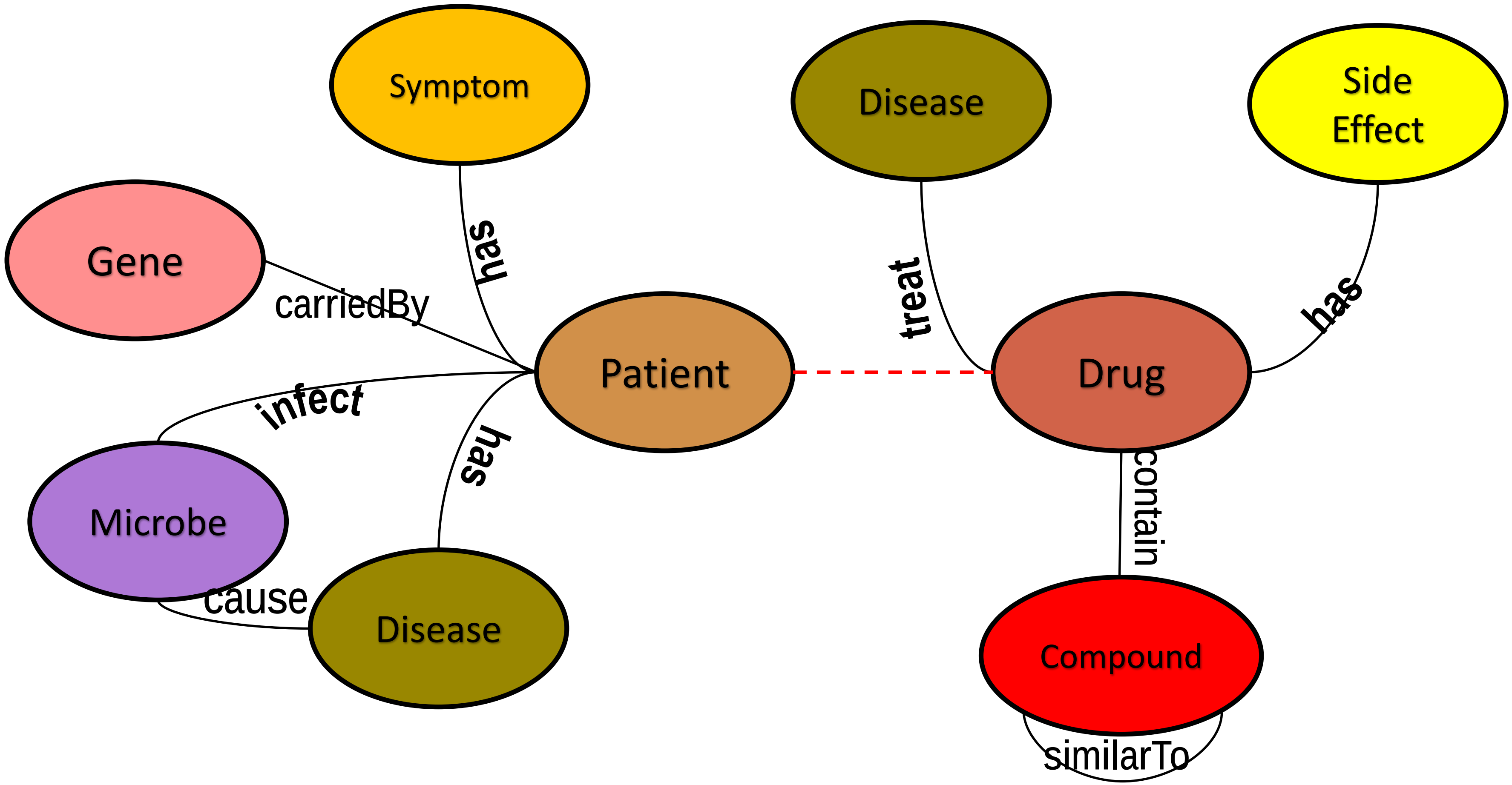

The Facebook Network

1. **Multiple entity types and link types**
2. ***New problems*** **are emerging in heterogeneous networks!**

We are living in a connected world!



Even in Biomedical Domain



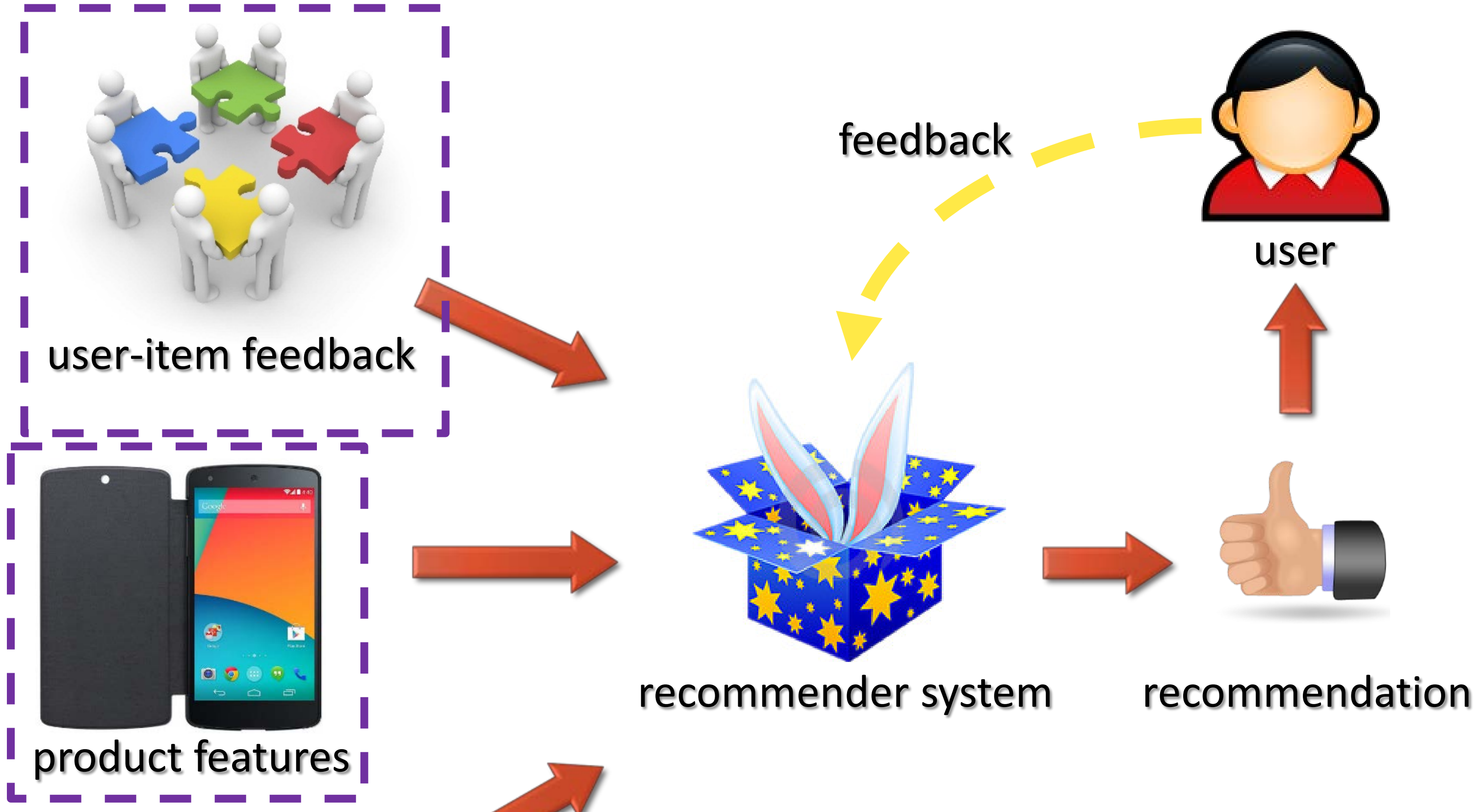
Outline

- Part I: Introduction and Preliminaries
- Part II: Recommendation in Heterogeneous Information Networks
- Part III: Recommendation in a Text-Rich Setting
- Part IV: Recommendation with Spatio-Temporal Information
- Part V: Research Frontiers and Summary

Part II: Recommendation in Heterogeneous Information Networks

- Hybrid Collaborative Filtering with Information Networks
- Graph Regularization for Recommendation
- Network Embedding-based Entity Recommendation
- Neural Network-based Collaborative Filtering

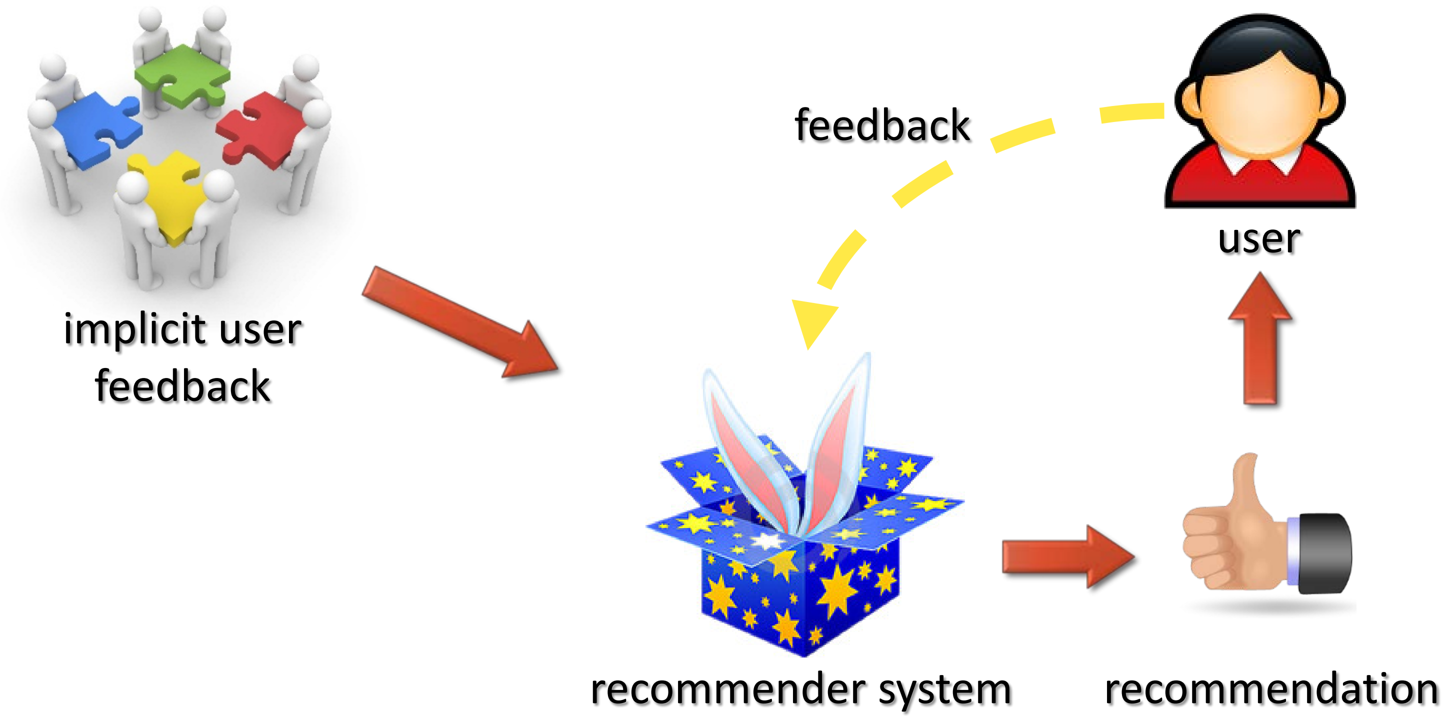
Recommendation Paradigm



Collaborative Filtering

E.g., K-Nearest Neighbor (Sarwar WWW'01), Matrix Factorization (Hu ICDM'08, Koren IEEE-CS'09), Probabilistic Model (Hofmann SIGIR'03)

Problem Definition

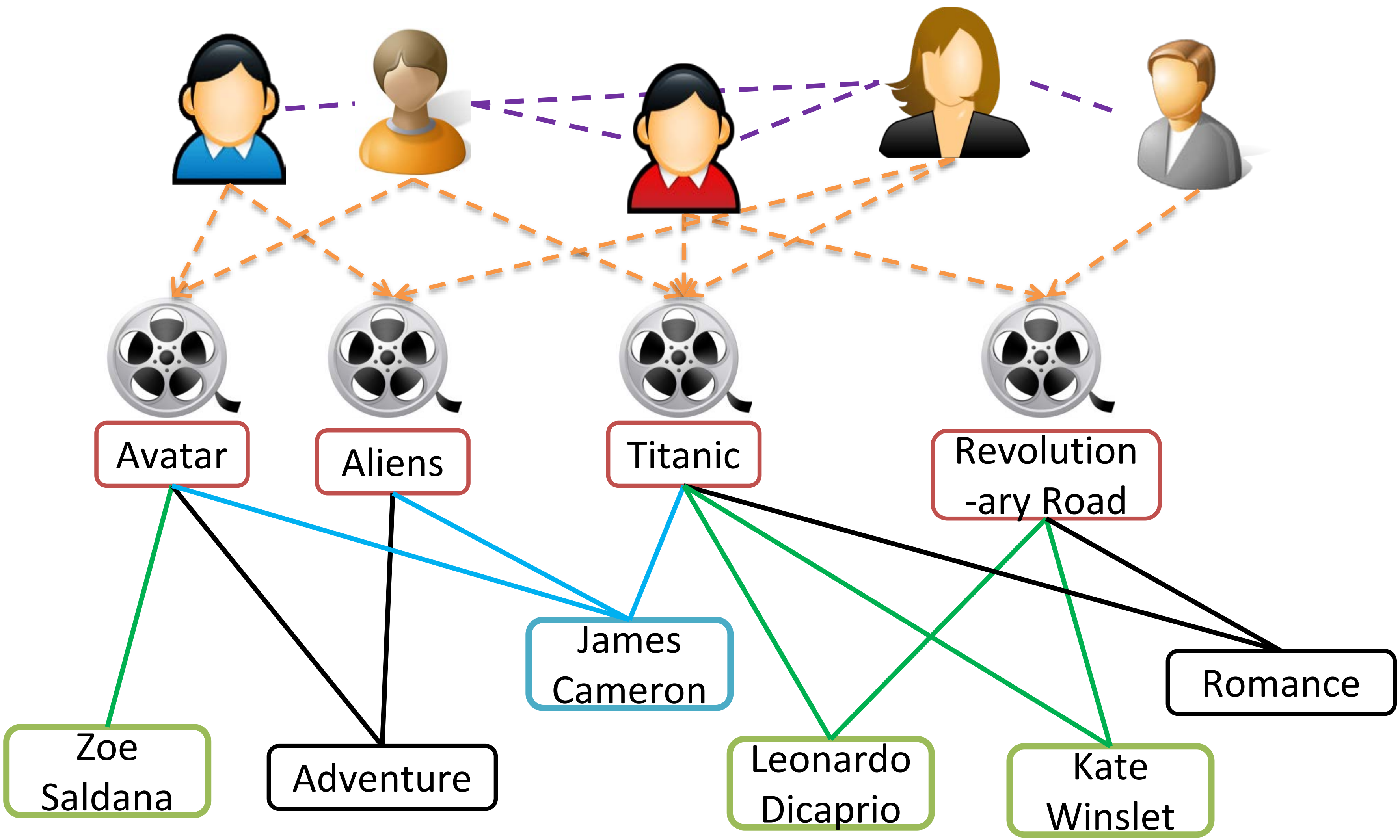


hybrid collaborative filtering with information networks

Hybrid Collaborative Filtering with Networks

- Utilizing network relationship information can enhance the recommendation quality
- However, most of the previous studies only use single type of relationship between users or items (e.g., social network Ma,WSDM'11, trust relationship Ester, KDD'10, service membership Yuan, RecSys'11)

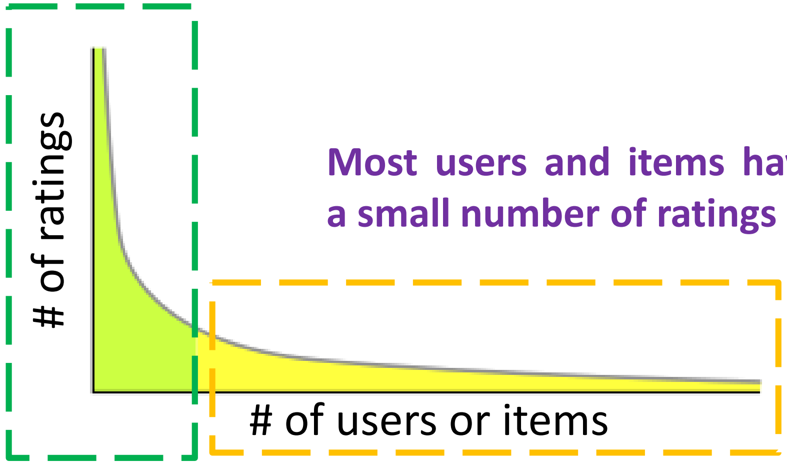
The Heterogeneous Information Network View of Recommender System



Relationship Heterogeneity Alleviates Data Sparsity

Collaborative filtering methods suffer from data sparsity issue

A small number of users and items have a large number of ratings

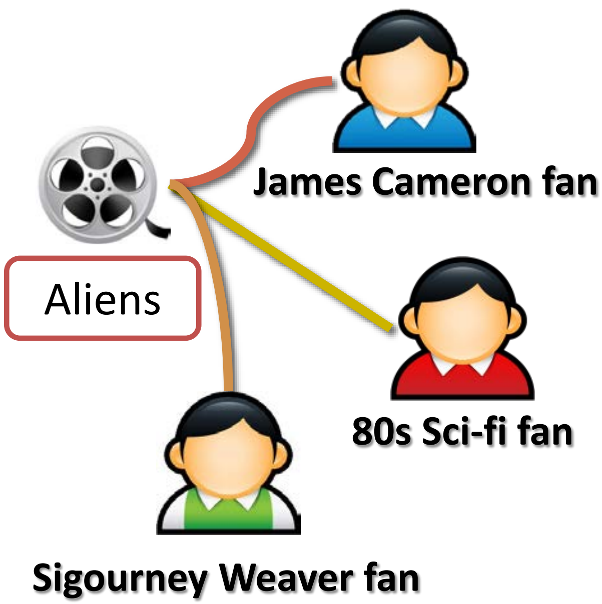


Most users and items have a small number of ratings

- Heterogeneous relationships complement each other
- Users and items with limited feedback can be connected to the network by different types of paths
  - Connect new users or items (cold start) in the information network

Relationship Heterogeneity Based Personalized Recommendation Models (Yu et al., WSDM'14)

Different users may have different behaviors or preferences



Different users may be interested in the same movie for different reasons

Two levels of personalization

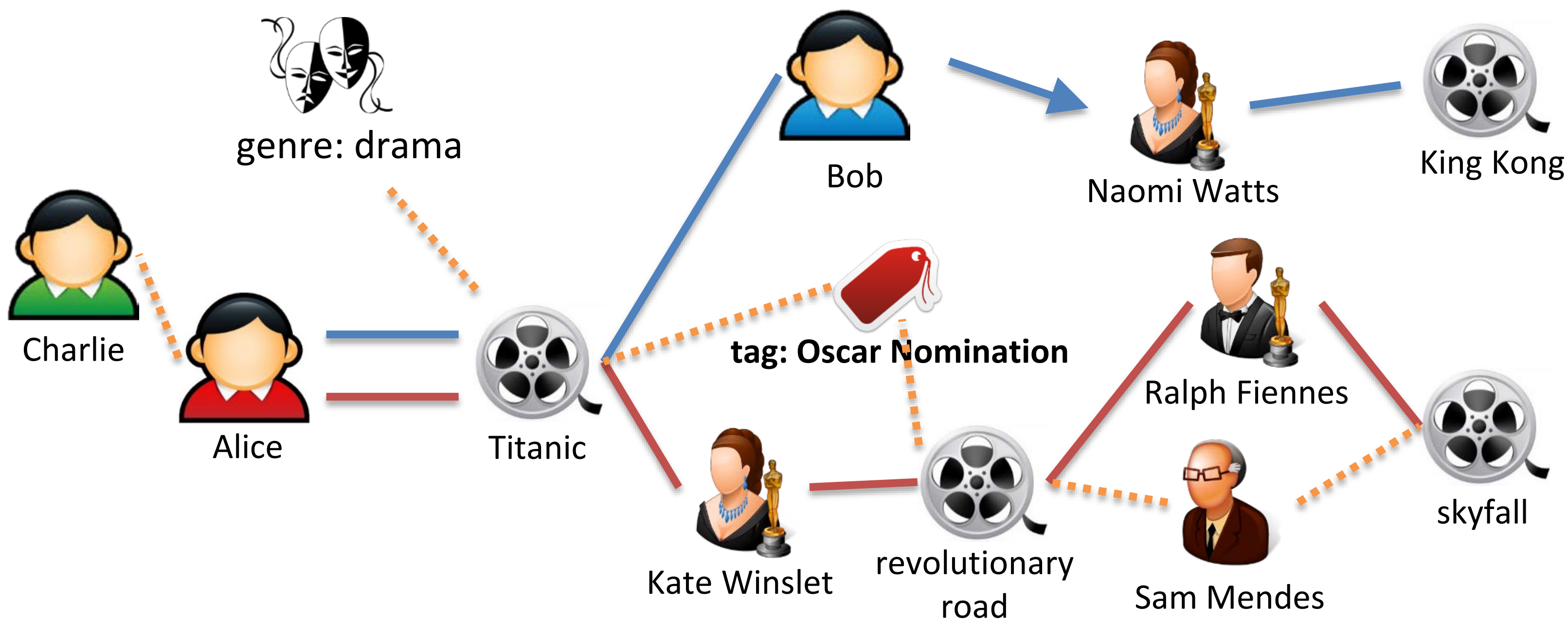
Data level

- Most recommendation methods use one model for all users and rely on personal feedback to achieve personalization

Model level

- With different entity relationships, we can learn personalized models for different users to further distinguish their differences

Preference Propagation-Based Latent Features



Generate *L* different meta-path (path types) **connecting users and items** → **Propagate user implicit feedback along each meta-path** → **Calculate latent-features for users and items for each meta-path with NMF related method**

Recommendation Models

Observation 1: Different meta-paths may have different importance

Global Recommendation Model

ranking score

features for user *i* and item *j*

$$\boxed{\hat{r}(u_i, e_j)} = \sum_{q=1}^{L} \theta_q \cdot \boxed{\hat{U}_i^{(q)} \hat{V}_j^{(q)T}} \quad (1)$$

the q-th meta-path

Observation 2: Different users may require different models

Personalized Recommendation Model

user-cluster similarity

$$\hat{r}_p(u_i, e_j) = \sum_{k=1}^{c} \boxed{sim(C_k, u_i)} \sum_{q=1}^{L} \theta_q^{\{k\}} \cdot \hat{U}_i^{(q)} \hat{V}_j^{(q)T} \quad (2)$$

c total soft user clusters

Parameter Estimation

- Bayesian personalized ranking (Rendle UAI'09)
- Objective function

sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$.

$$\min_{\Theta} \; -\sum_{u_i \in \mathcal{U}} \boxed{\sum_{(e_a > e_b) \in R_i}} \ln \sigma(\hat{r}(u_i, e_a) - \hat{r}(u_i, e_b)) + \frac{\lambda}{2} \|\Theta\|_2^2 \qquad (3)$$

for each correctly ranked item pair
i.e., u_i gave feedback to e_a but not e_b

Soft cluster users with NMF + k-means → **For each user cluster, learn one model with Eq. (3)** → **Generate personalized model for each user on the fly with Eq. (2)**

Learning Personalized Recommendation Model

Experiment Setup

- Datasets

Name	#Items	#Users	#Ratings	#Entities	#Links
IM100K	943	1360	89,626	60,905	146,013
Yelp	11,537	43,873	229,907	285,317	570,634

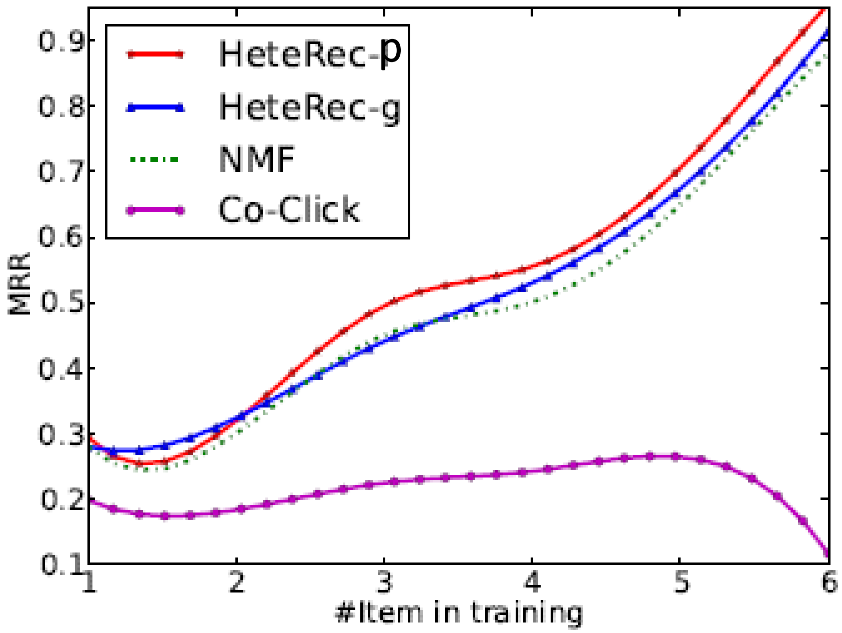
- Comparison methods:
  - Popularity: recommend the most popular items to users
  - Co-click: conditional probabilities between items
  - NMF: non-negative matrix factorization on user feedback
  - Hybrid-SVM: use Rank-SVM with plain features (utilize both user feedback and information network)

Performance Comparison

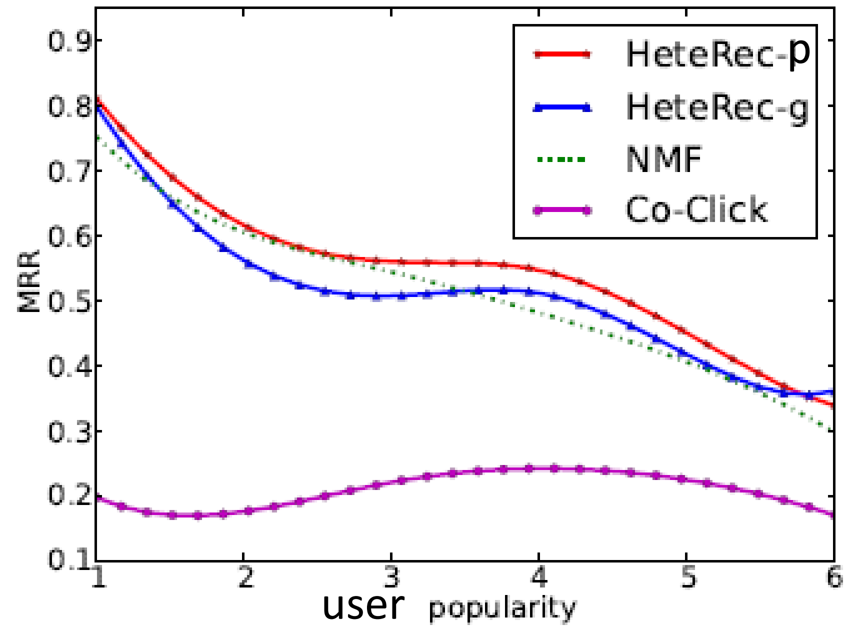
Method	IM100K				Yelp			
	Prec1	Prec5	Prec10	MRR	Prec1	Prec5	Prec10	MRR
Popularity	0.0731	0.0513	0.0489	0.1923	0.00747	0.00825	0.00780	0.0228
Co-Click	0.0668	0.0558	0.0538	0.2041	0.0147	0.0126	0.01132	0.0371
NMF	0.2064	0.1661	0.1491	0.4938	0.0162	0.0131	0.0110	0.0382
Hybrid-SVM	0.2087	0.1441	0.1241	0.4493	0.0122	0.0121	0.0110	0.0337
HeteRec-g	0.2094	0.1791	0.1614	0.5249	0.0165	0.0144	0.0129	0.0422
HeteRec-p	**0.2121**	**0.1932**	**0.1681**	**0.5530**	**0.0213**	**0.0171**	**0.0150**	**0.0513**

HeteRec personalized recommendation (HeteRec-p) provides the best recommendation results

Performance under Different Scenarios



(a) Performance Change with User Feedback Number



(b) Performance Change with User Feedback Popularity

HeteRec–p consistently outperform other methods in different scenarios

better recommendation results if users provide more feedback

better recommendation for users who like less popular items

Part II: Recommendation in Heterogeneous Information Networks

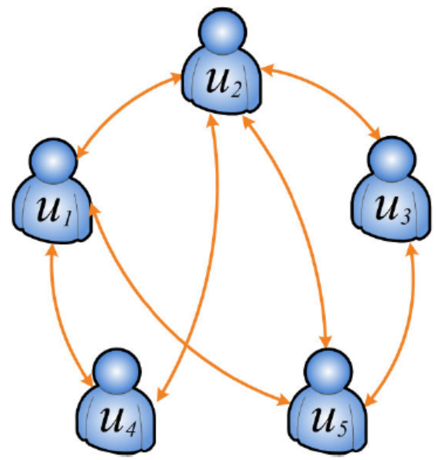
- Hybrid Collaborative Filtering with Information Networks
- Graph Regularization for Recommendation
- Network Embedding-based Entity Recommendation
- Neural Network-based Collaborative Filtering

From Graph Regularization Point of View

- Why additional links help?
  - They define new similarity metrics between users or items.
- How to integrate this assumption into recommendation?
  - Use graph regularization to force two entities to be similar in latent space, if they are similar in graph
- The original form of graph regularization
  - $\frac{1}{2}\sum w_{ij}\left(f_i - f_j\right)^2 = f'Lf$
    - w_{ij} : *similarity of node i and j*
    - f_i: some latent representation for node i
    - L: Laplacian matrix of W, i.e., $L = D - W$,
      - *where D is a diagonal matrix and* $D_{ii} = \sum_j w_{ij}$

Recommender Systems with Social Regularization [Ma et al., WSDM'11]

- Input: Social Relation + Rating Matrix



	v_1	v_2	v_3	v_4	v_5
u_1	1		2	3	
u_2		3			1
u_3		4		5	
u_4	5			4	
u_5		2	5		4

(a) Real World Social Recommendation (b) Social Network (c) User-Item Rating Matrix

Two Regularization Forms

- Model 1: Average-based Regularization
  - We are similar to the average of our friends

$$\min_{U,V} \mathcal{L}_1(R,U,V) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}(R_{ij} - U_i^T V_j)^2 + \frac{\alpha}{2}\sum_{i=1}^{m}\|U_i - \frac{1}{|\mathcal{F}^+(i)|}\sum_{f\in\mathcal{F}^+(i)} U_f\|_F^2 + \frac{\lambda_1}{2}\|U\|_F^2 + \frac{\lambda_2}{2}\|V\|_F^2, \quad (5)$$

- Model2: Individual-based Regularization
  - We are similar to each of our friends

$$\min_{U,V} \mathcal{L}_2(R,U,V) = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}(R_{ij} - U_i^T V_j)^2 + \frac{\beta}{2}\sum_{i=1}^{m}\sum_{f\in\mathcal{F}^+(i)} Sim(i,f)\|U_i - U_f\|_F^2 + \lambda_1\|U\|_F^2 + \lambda_2\|V\|_F^2. \quad (11)$$

Similarity can be propagated via friends: transitivity!

How to compute similarity between two users?

- Cosine similarity (VSS)

$$Sim(i,f) = \frac{\sum_{j \in I(i) \cap I(f)} R_{ij} \cdot R_{fj}}{\sqrt{\sum_{j \in I(i) \cap I(f)} R_{ij}^2} \cdot \sqrt{\sum_{j \in I(i) \cap I(f)} R_{fj}^2}}$$

- Pearson correlation coefficient (PCC)

$$Sim(i,f) = \frac{\sum_{j \in I(i) \cap I(f)} (R_{ij} - \overline{R}_i) \cdot (R_{fj} - \overline{R}_f)}{\sqrt{\sum_{j \in I(i) \cap I(f)} (R_{ij} - \overline{R}_i)^2} \cdot \sqrt{\sum_{j \in I(i) \cap I(f)} (R_{fj} - \overline{R}_f)^2}}, \tag{14}$$

Results

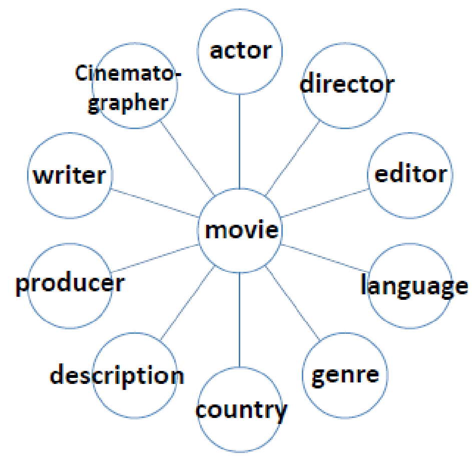
Table 5: Performance Comparisons (Dimensionality = 10)

Dataset	Training	Metrics	UserMean	ItemMean	NMF	PMF	RSTE	$SR1_{vss}$	$SR1_{pcc}$	$SR2_{vss}$	$SR2_{pcc}$
Douban	80%	MAE	0.6809	0.6288	0.5732	0.5693	0.5643	0.5579	0.5576	0.5548	**0.5543**
		Improve	18.59%	11.85%	3.30%	2.63%	1.77%				
		RMSE	0.8480	0.7898	0.7225	0.7200	0.7144	0.7026	0.7022	0.6992	**0.6988**
		Improve	17.59%	11.52%	3.28%	2.94%	2.18%				
	60%	MAE	0.6823	0.6300	0.5768	0.5737	0.5698	0.5627	0.5623	0.5597	**0.5593**
		Improve	18.02%	11.22%	3.03%	2.51%	1.84%				
		RMSE	0.8505	0.7926	0.7351	0.7290	0.7207	0.7081	0.7078	0.7046	**0.7042**
		Improve	17.20%	11.15%	4.20%	3.40%	2.29%				
	40%	MAE	0.6854	0.6317	0.5899	0.5868	0.5767	0.5706	0.5702	0.5690	**0.5685**
		Improve	17.06%	10.00%	3.63%	3.12%	1.42%				
		RMSE	0.8567	0.7971	0.7482	0.7411	0.7295	0.7172	0.7169	0.7129	**0.7125**
		Improve	16.83%	10.61%	4.77%	3.86%	2.33%				
Epinions	90%	MAE	0.9134	0.9768	0.8712	0.8651	0.8367	0.8290	0.8287	0.8258	**0.8256**
		Improve	9.61%	15.48%	5.23%	4.57%	1.33%				
		RMSE	1.1688	1.2375	1.1621	1.1544	1.1094	1.0792	1.0790	1.0744	**1.0739**
		Improve	8.12%	13.22%	7.59%	6.97%	3.20%				
	80%	MAE	0.9285	0.9913	0.8951	0.8886	0.8537	0.8493	0.8491	0.8447	**0.8443**
		Improve	9.07%	14.83%	5.68%	4.99%	1.10%				
		RMSE	1.1817	1.2584	1.1832	1.1760	1.1256	1.1016	1.1013	1.0958	**1.0954**
		Improve	7.30%	12.95%	7.42%	6.85%	2.68%				

Meta-Path-based Regularization [Yu et al., IJCAI-HINA'13]

- What if it is more than one type of relation?

	E1	e2	…	em
u1	0	0	0	1
u2	0	2	0	5
…	0	0	0	0
un	3	4	0	0



Rating Data

Heterogeneous Information Network

- Solution:
  - Use meta-path to generate similarity relation between items, e.g., movie-director-movie
  - Learn the importance score for each meta-path

Notations

- We have *n* users and *m* items.
  - $\mathcal{U} = \{u_1, ..., u_n\}$ $\mathcal{I} = \{e_1, ..., e_m\}$
- By computing similarity scores of all item pairs along certain meta-path, we can get a similarity matrix
  - $S^{(l)} \in \mathbb{R}^{n \times n}$
- With *L* different meta-paths, we can calculate *L* similarity matrices as
  - $S^{(1)}, S^{(2)}, ..., S^{(L)}$

Objective Function

Approximate R with U V product

Regularization on U V

$$\min_{U,V,\boldsymbol{\theta}} \quad \boxed{\|Y \odot (R - UV^T)\|_F^2} + \boxed{\lambda_0(\|U\|_F^2 + \|V\|_F^2)} +$$

$$\boxed{\frac{\lambda_1}{2} \cdot \sum_{i,j} \sum_{l=1}^{L} \theta_l S_{ij}^{(l)} \|V_i - V_j\|_2^2} + \boxed{\lambda_2 \|\boldsymbol{\theta}\|_2^2,}$$

Similar items measured from HIN should have similar low-rank representations

Regularization on *ϑ, which is the importance score for each meta-path*

$$\text{s.t.} \quad U \geq 0, \ \ V \geq 0, \ \ \boldsymbol{\theta} \geq 0, \ \ \text{and} \ \sum_{l=1}^{L} \theta_l = 1,$$

Equivalent Objective Function Using Graph Laplacian

$$D_{ii}^{(l)} = \sum_{j=1}^{n} S_{ij}^{(l)} \qquad L^{(l)} = D^{(l)} - S^{(l)}$$

$$\min_{U,V,\boldsymbol{\theta}} \quad \|Y \odot (R - UV^T)\|_F^2 + \lambda_0(\|U\|_F^2 + \|V\|_F^2) +$$

$$\boxed{\lambda_1 \cdot \mathrm{Tr}\left(V^T(\sum_l \theta_l L^{(l)})V\right)} + \lambda_2\|\boldsymbol{\theta}\|_F^2,$$

Similar items measured from HIN should have similar low-rank representations

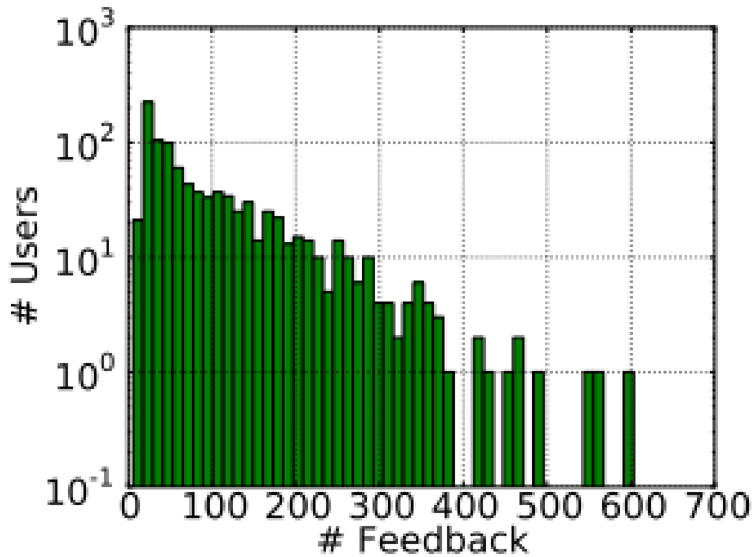
$$\text{s.t.} \quad U \geq 0, \;\; V \geq 0, , \;\; \boldsymbol{\theta} \geq 0, \;\; \text{and} \;\; \sum_{l=1}^{L} \theta_l = 1.$$

Dataset

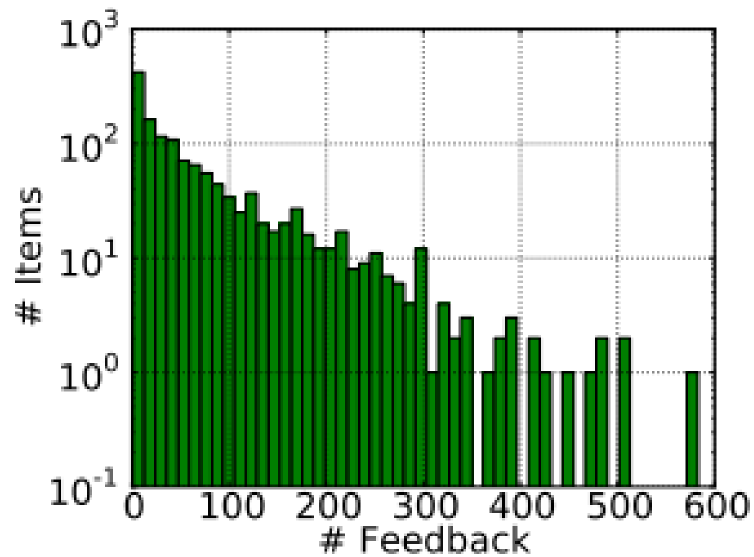
- We combine IMDb + MovieLens100K

Name	#Items	#Users	#Ratings	#Entities	#Links
IM100K	943	1360	89,626	60,905	146,013

(a) Datasets Description



(b) #Ratings vs. #Users



(c) #Ratings vs. Item Popularity

We random sample training datasets of different sizes (0.4, 0.6, and 0.8)

Results

Metric	MAE			RMSE		
Training Size	40%	60%	80%	40%	60%	80%
UserMean	0.8400	0.8409	0.8324	1.0479	1.0482	1.0407
ItemMean	0.8167	0.8237	0.8130	1.0281	1.0354	1.0235
NMF (d=40)	2.1944	2.1862	2.0162	2.4459	2.4391	2.2915
WNMF (d=10)	0.7919	0.7879	0.7589	1.0055	1.0028	0.9677
WNMF (d=20)	0.7917	0.7875	0.7591	1.0060	1.0026	0.9681
WNMF (d=40)	0.7886	0.7833	0.7569	1.0027	0.9991	0.9655
Hete-MF (d=10)	0.7838	0.7800	0.7530	0.9950	0.9931	0.9683
Hete-MF (d=20)	0.7818	0.7802	0.7528	0.9941	0.9938	0.9593
Hete-MF (d=40)	**0.7780**	**0.7772**	**0.7400**	**0.9900**	**0.9905**	**0.9503**

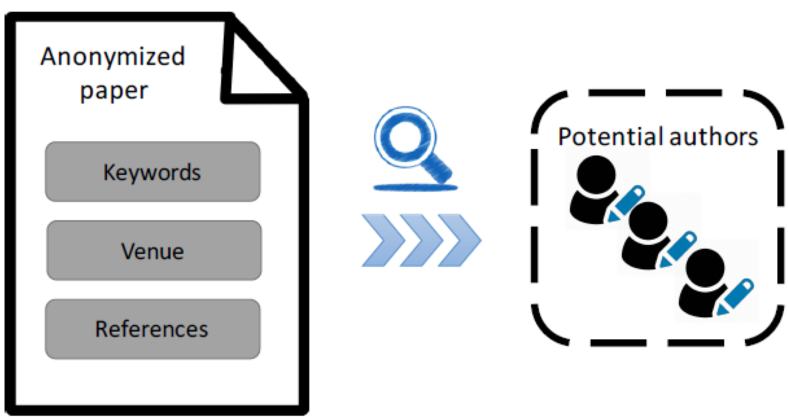
Part II: Recommendation in Heterogeneous Information Networks

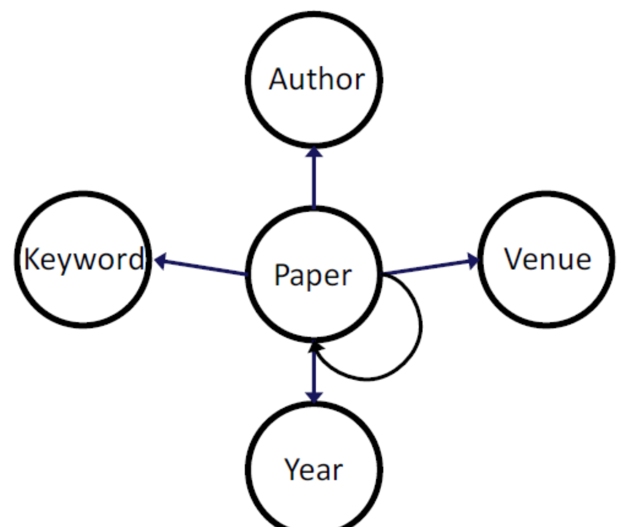
- Hybrid Collaborative Filtering with Information Networks
- Graph Regularization for Recommendation
- Network Embedding-based Entity Recommendation
- Neural Network-based Collaborative Filtering

Task-Guided Meta-Path Augmented Embedding [Chen et al., WSDM'17]

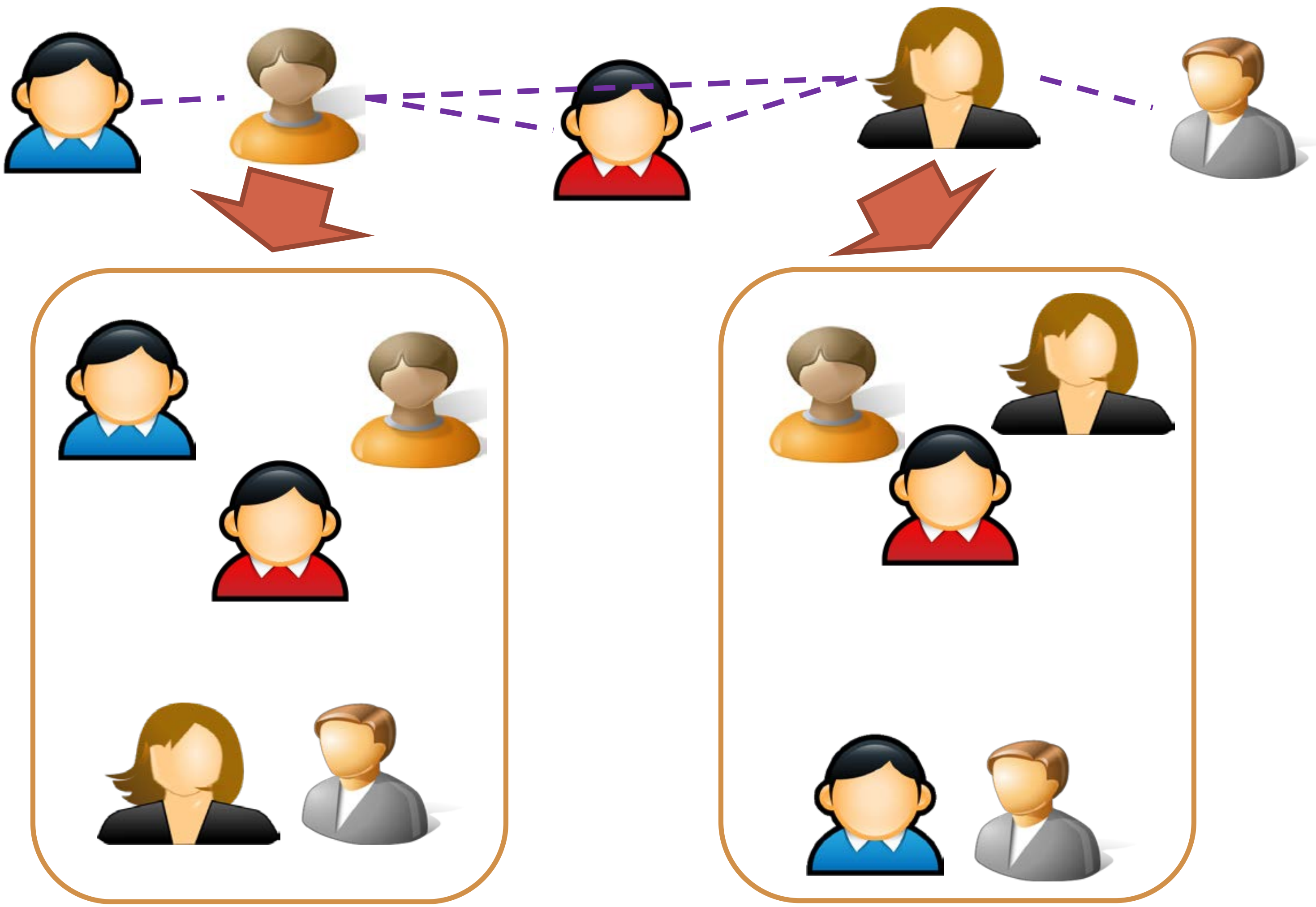
- Given an anonymized paper, with
  - Venue (e.g., WSDM)
  - Year (e.g., 2017)
  - Keywords (e.g., "heterogeneous network embedding")
  - References (e.g., [Chen et al., IJCAI'16])

  - Can we predict its authors?





Challenge 1: Task Guided Embedding

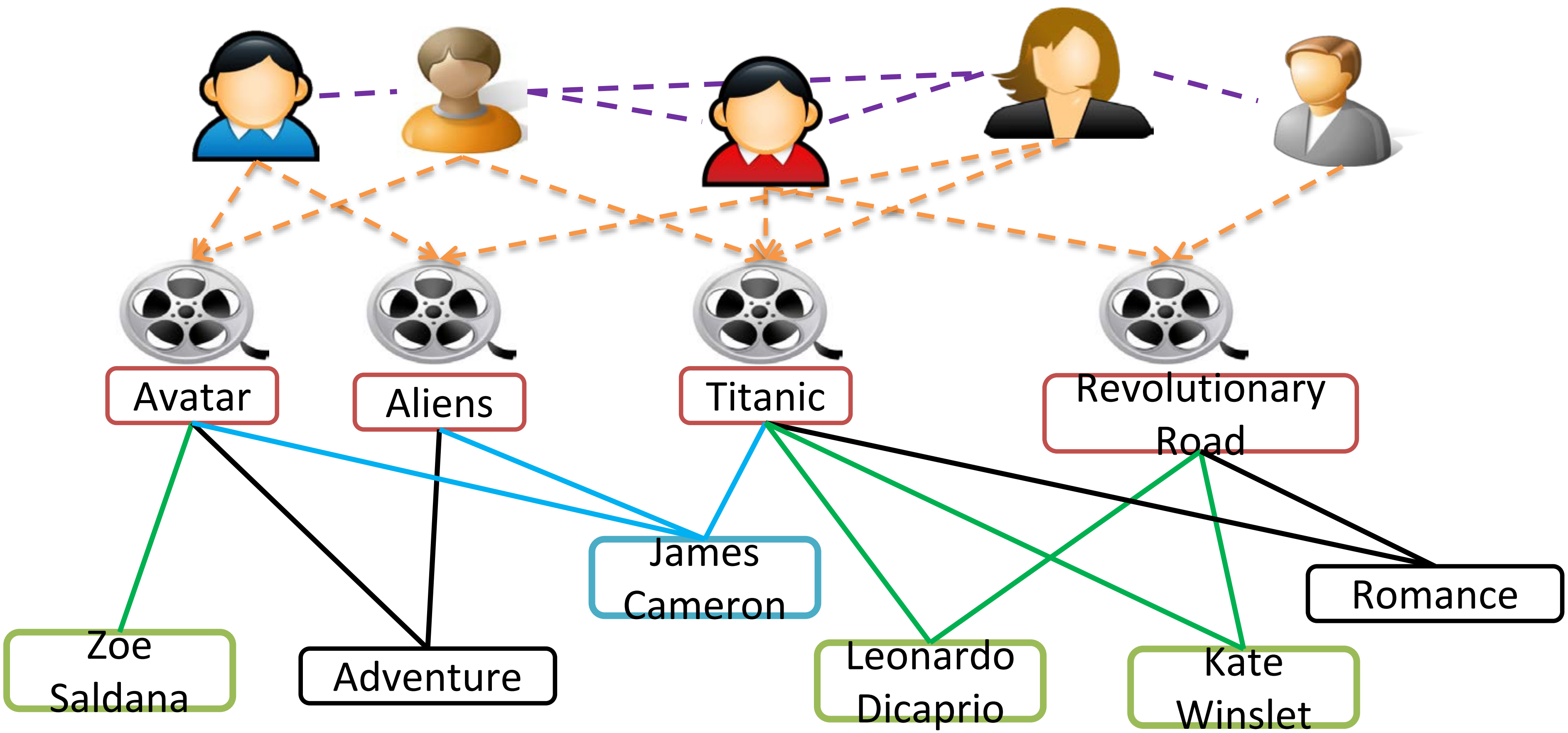


Embedding for Movie Recommendation

Embedding for Voting Prediction

Challenge 2: Heterogeneous Network Embedding

- How to utilize links belonging to different types with different semantic meanings?



Our Solution

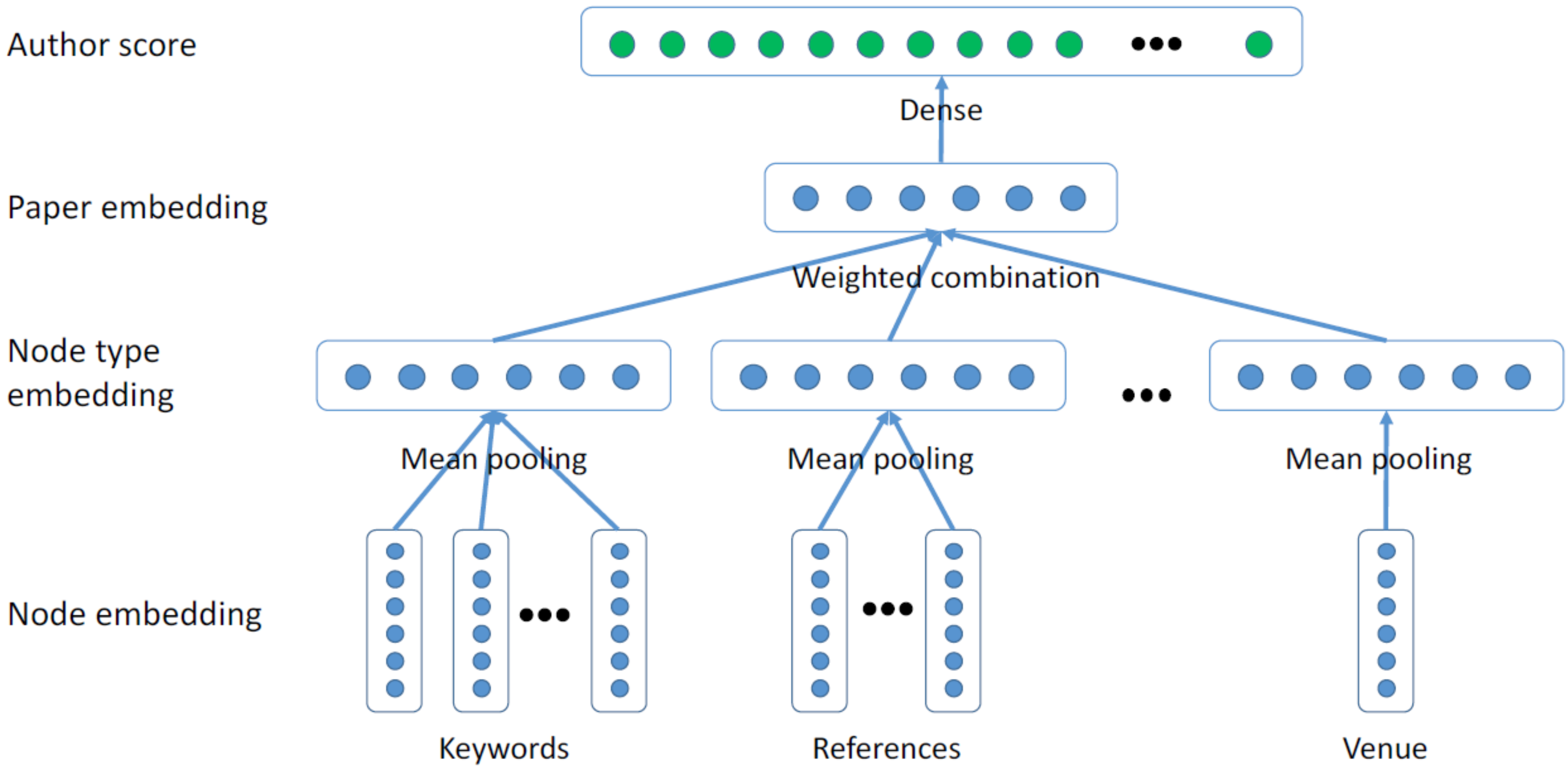
- Task-guided and path-augmented embedding: A Semi-Supervised framework
  - Task-guided embedding takes care of supervised labels
    - E.g., **Author** "**Ting Chen**" should be close to **Keyword** "**Heterogeneous network embedding**"
  - Path-augmented embedding takes care of the global structure of networks (Path-augmented network regularization)
    - E.g., **Keyword** "**heterogeneous network embedding**" should be close to **Keyword** "**node representation**"
      - meta-path: Keyword-Paper->Paper-Keyword

The Combined Model

- Joint training of two types of embedding
- Path selection is performed to pick most informative meta-paths for network embedding.

Component 1: Task-Guided Embedding

- The embedding architecture for author identification



Formally

- Consider the ego-network of p: $X_p = (X_p^1, X_p^2, \ldots, X_p^T)$,
  - T: the number of types of nodes associated with paper type
  - X_p^t: the set of nodes with type *t* associated with paper p
- u_a: embedding of author *a*
- u_n: embedding of node *n*
- V_p: embedding of paper *p*
  - *Weighted average of all the neighbors*
- The score function between *p* and *a* is defined as:

$$f(p,a) = u_a^T V_p = u_a^T \left(\sum_t w_t V_p^{(t)} \right)$$

$$= u_a^T \left(\sum_t w_t \sum_{n \in X_p^{(t)}} u_n / |X_p^{(t)}| \right)$$

Ranking-based Objective

- Given a paper *p*, author *a* that is an author of *p*, and author *b* that is not an author of *p*
  - $f(p, \boldsymbol{a}) > f(p, \boldsymbol{b})$

- A hinge loss function with margin is used as objective function
  - $$\max\left(0, f(p, b) - f(p, a) + \xi\right)$$

Component 2: Path-Augmented Embedding

- Limitations of task-guided embedding
  - Supervised labels expensive to obtain
  - The rich structure information of heterogeneous information networks is not fully explored

- Path-Augmented Embedding
  - Prepare meta-paths that are potentially related to the task
    - author-paper-author
    - author-paper->paper
    - author-paper
  - Apply general purpose embedding

Formally

- For each meta-path-based relation
  - Define the probability of reaching node j from node i via meta-path r via their embeddings

$$P(j|i;r) = \frac{\exp(u_i^T u_j)}{\sum_{j' \in DST(r)} \exp(u_i^T u_{j'})}$$

  - Use negative sampling to approximate the distribution
  - Extend LINE [Tang et al., 2015]

- The goal is to maximize the likelihood to observing all the paths under each meta-path

The Joint Model

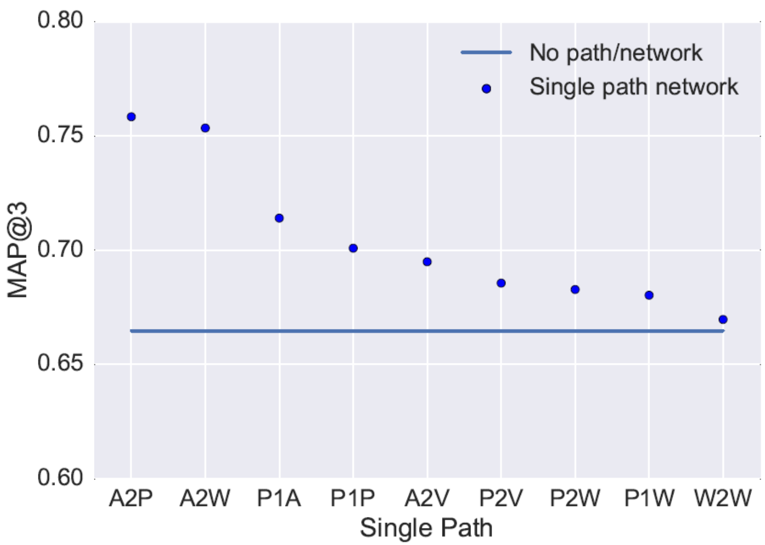
- Objective function

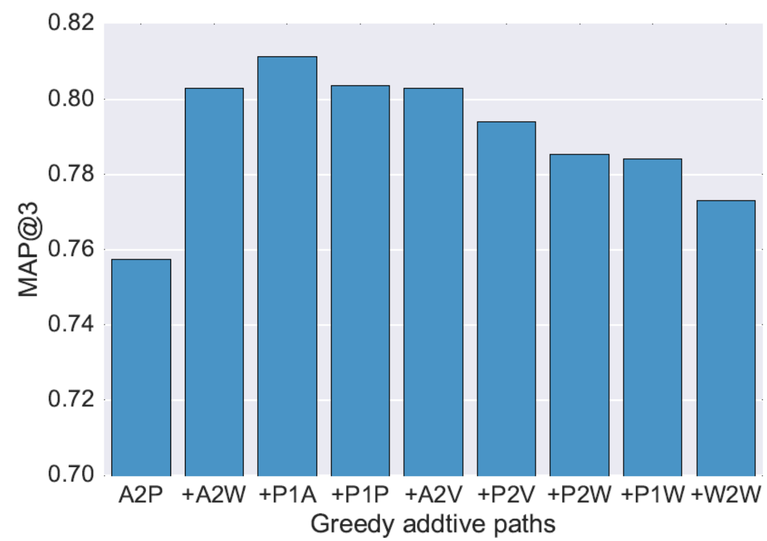
$$
\begin{aligned}
\mathcal{L} =&(1-\omega)\mathcal{L}_{task-specific} + \omega\mathcal{L}_{network-general} + \Omega(\mathcal{M}) \\
=&(1-\omega)\mathbb{E}_{(p,a,b)}\left[\max\Big(0, f(p,b) - f(p,a) + \xi\Big)\right] \\
&+ \omega\mathbb{E}_{(r,i,j)}\left[-\log \hat{P}(j|i;r)\right] + \lambda\sum_{i}\|u_i\|_2^2
\end{aligned}
$$

How to select meta-paths?

- A greedy strategy is used to select meta-paths
  - Step 1: Rank single meta-path according to their performance
  - Step 2: Greedily add the current best meta-path into current pool, stop until the performance deteriorates

- Different meta-paths will be selected for different tasks





Experiments

- Dataset:
  - AMiner Citation data set.
  - Papers before 2012 are used in training, and papers on and after 2012 are used as test.

Table 1 : Node statistics

	Paper	Author	keyword	Venue	Year
Train	1.6M	1M	4M	7K	60
Test	34K	62K	42K	1K	2

Table 3 : Length-2 link statistics

A-P-A	A-P-P	A-P-V	A-P-W	A-P-Y	P-P-V	P-P-W	V-P-W	W-P-W	Y-P-W
17M	18M	4M	38M	4M	3M	27M	12M	118M	12M

Baselines

- Supervised feature-based baselines (i.e. LR, SVM, RF, LambdaMart).
  - Manually crafted features
- Task-specific embedding.
- Network-general embedding.
- Pre-training + Task-specific embedding.
  - Take general embedding as initialization of task-specific embedding

Comparison

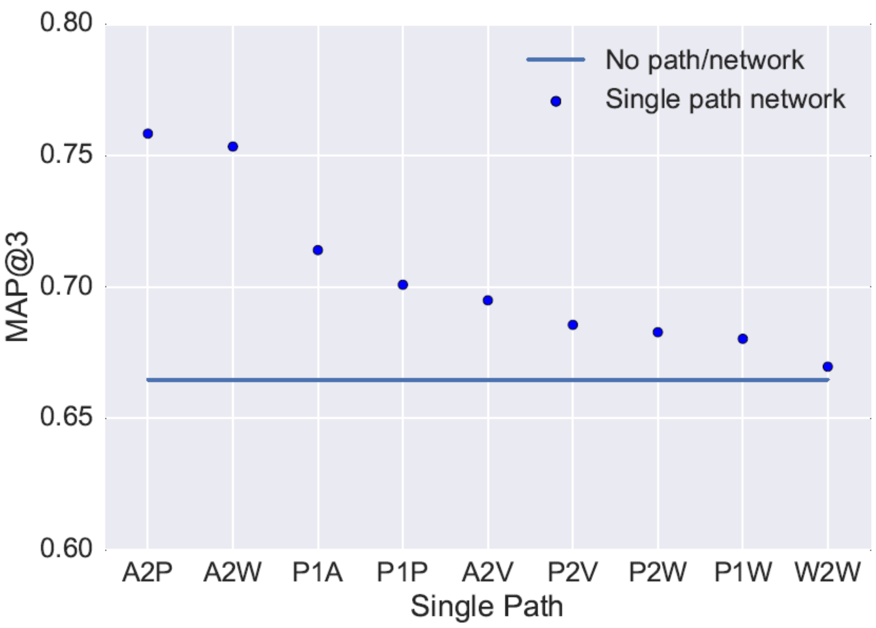
- Easy task: choose author candidate as true authors + negative authors

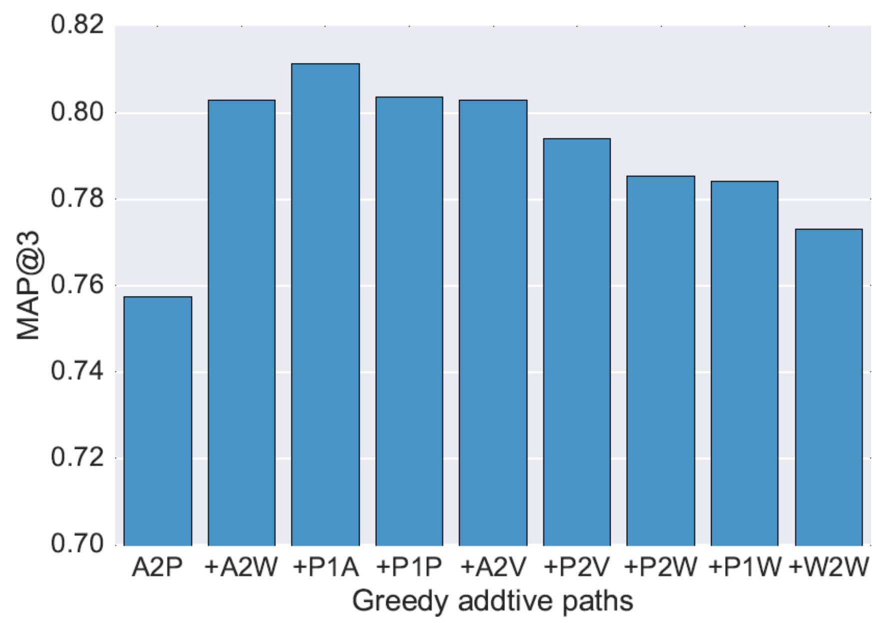
Table 5 : Author identification performance comparison.

Models	MAP@3	MAP@10	Recall@3	Recall@10
LR	0.7289	0.7321	0.6721	0.8209
SVM	0.7332	0.7365	0.6748	0.8267
RF	0.7509	0.7543	0.6921	0.8381
LambdaMart	0.7511	0.7420	0.6869	0.8026
Task-specific	0.6876	0.7088	0.6523	0.8298
Pre-train+Task.	0.7722	0.7962	0.7234	0.9014
Network-general	0.7563	0.7817	0.7105	0.8903
Combined	**0.8113**	**0.8309**	**0.7548**	**0.9215**

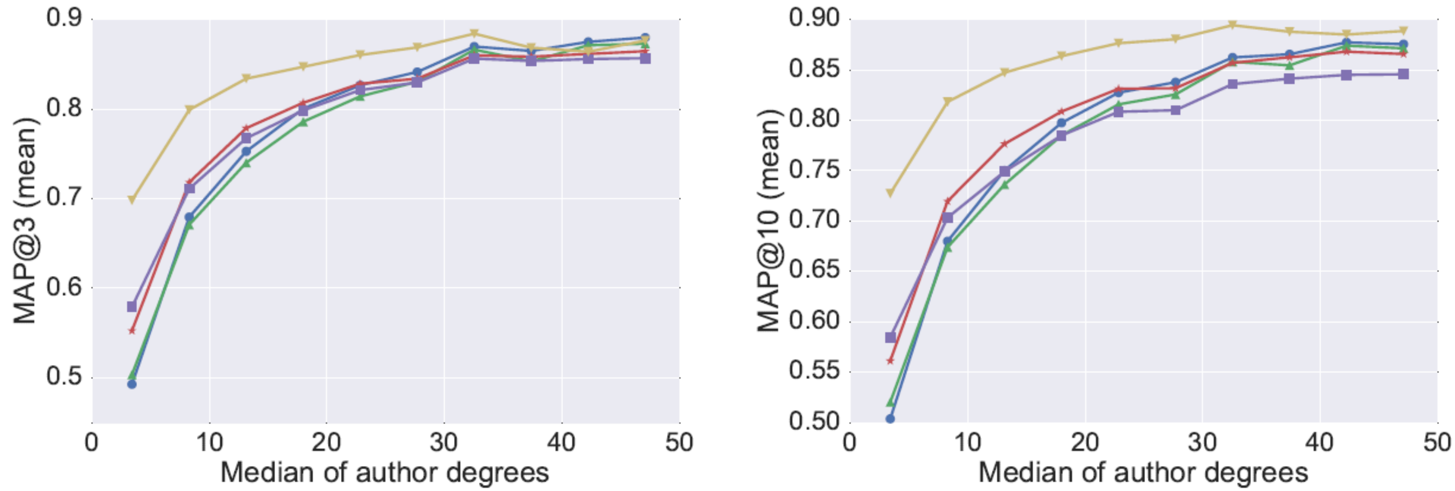
Which meta-paths are selected?

- A-P->P: author *write* paper *cite* paper
- A-P-W: author *write* paper *contain* keyword
- P-A: paper *written-by* author



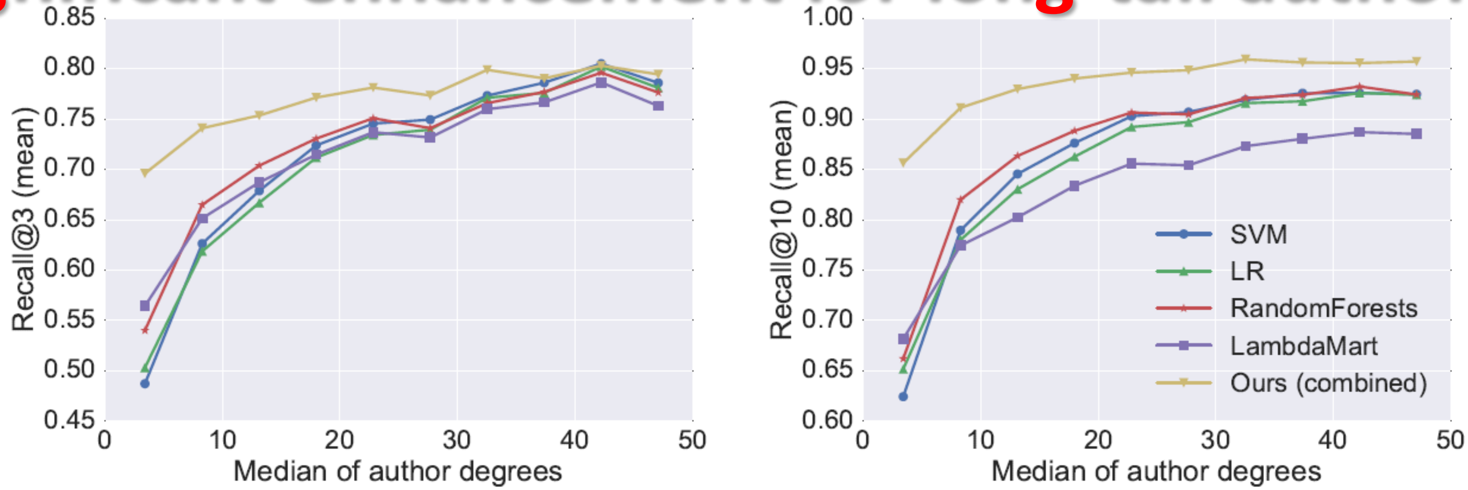


Performance over Different Groups of Authors



(c) MAP@3 (d) MAP@10

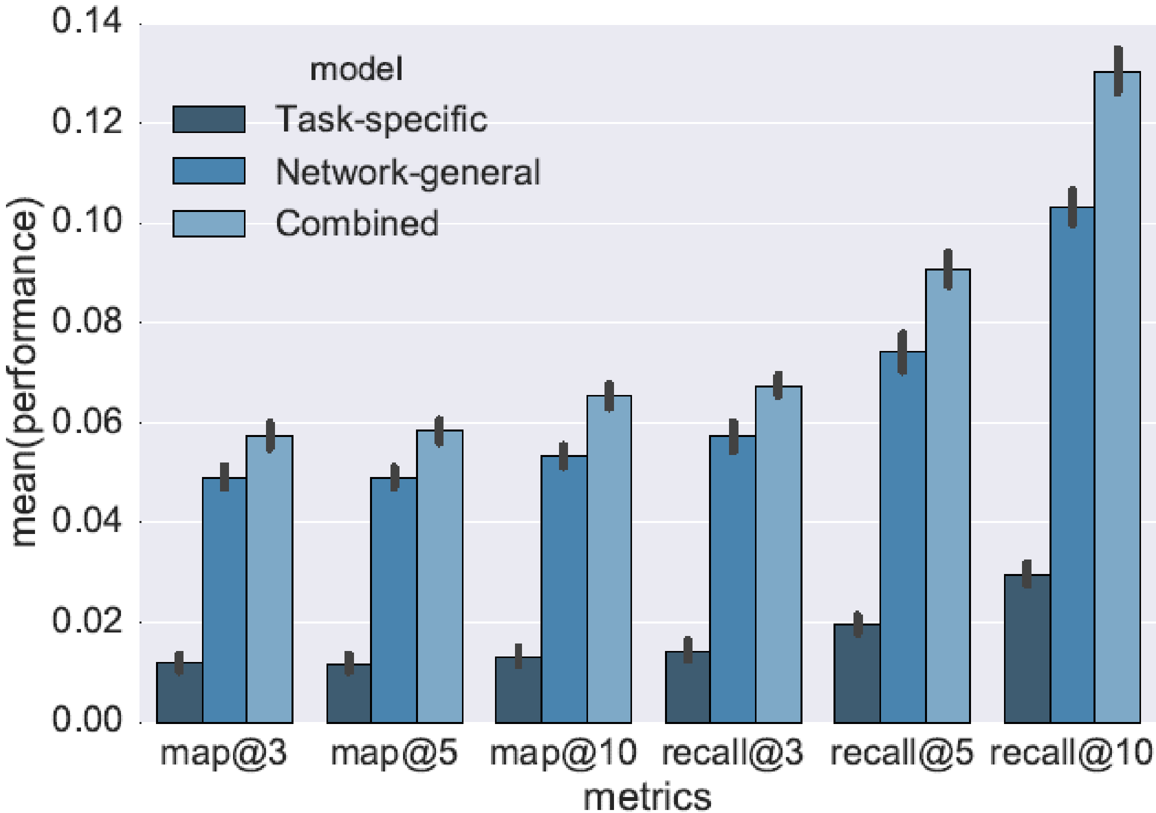
Significant enhancement for long-tail authors!



(e) Recall@3 (f) Recall@10

The Real Game

- Treat all the authors as candidates



- Future work: full text analysis

Case Study

Top ranked authors for **Keyword "variational inference"**

Task-specific	Network-general	Combined
Chong Wang	Yee Whye Teh	Michael I. Jordan
Qiang Liu	Mohammad E. Khan	Yee Whye Teh
Sheng Gao	Edward Challis	Zoubin Ghahramani
Song Li	Ruslan Salakhutdinov	John William Paisley
Donglai Zhu	Michael I. Jordan	David M. Blei
Neil D. Lawrence	Zoubin Ghahramani	Max Welling
Sotirios Chatzis	Matthias Seeger	Alexander T. Ihler
Si Wu	David B. Dunson	Eric P. Xing
Huan Wang	Dae Il Kim	Ryan Prescott Adams
Weimin Liu	Pradeep D. Ravikumar	Thomas L. Griffiths

Part II: Recommendation in Heterogeneous Information Networks

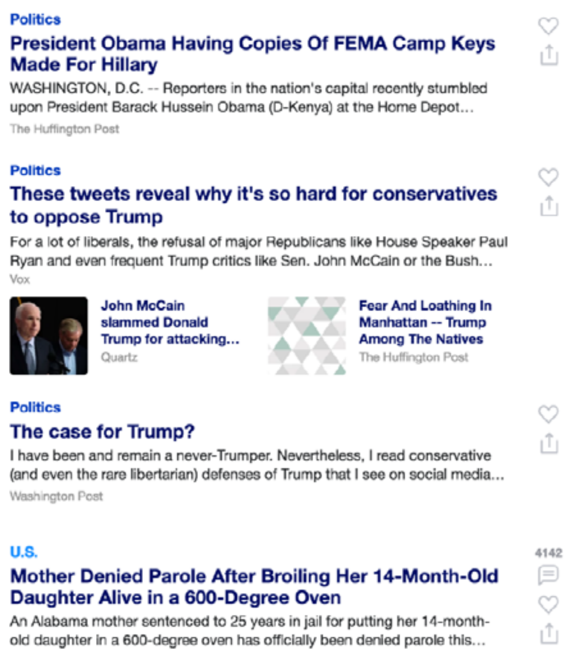
- Hybrid Collaborative Filtering with Information Networks
- Graph Regularization for Recommendation
- Network Embedding-based Entity Recommendation
- Neural Network-based Collaborative Filtering

Application: News Recommendation

- Chen et al., “On Sampling Strategies for Neural Network-based Collaborative Filtering,” KDD’17

- Use the news recommendation as a running example

Yahoo news feed

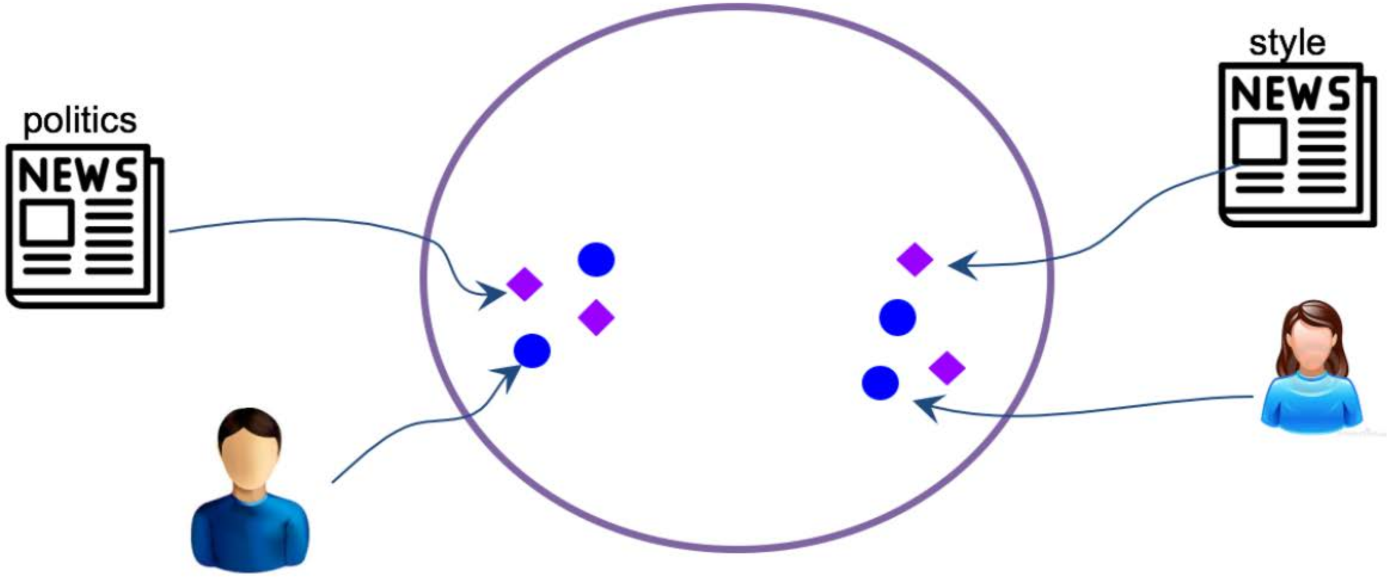


Which news will a user click?

Goal: learning to predict a user’s interests on items (news, articles) based on their text content.

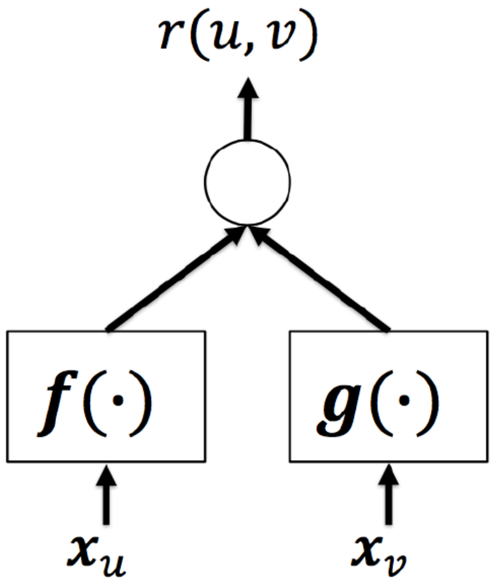
Solution

- Neural Network-based Collaborative Filtering



Subsumes several existing work, e.g., Bansal et al., RecSys'16, Van den Oord et al., NIPS'13

More Generally: Functional Embedding



Interaction function

Embedding functions

Figure 1: Computational flow of the proposed framework.

$$r_{uv} = \mathbf{f}(x_u)^T \mathbf{g}(x_v)$$

Embeddings: $\mathbf{f}_u, \mathbf{g}_v \in \mathbb{R}^d$

Goal: Minimize the loss function between predicted and observed rating or ranking

Challenge

- Computational cost is very heavy when embedding functions are complex multi-layer non-linear transformations, such as RNN and CNN

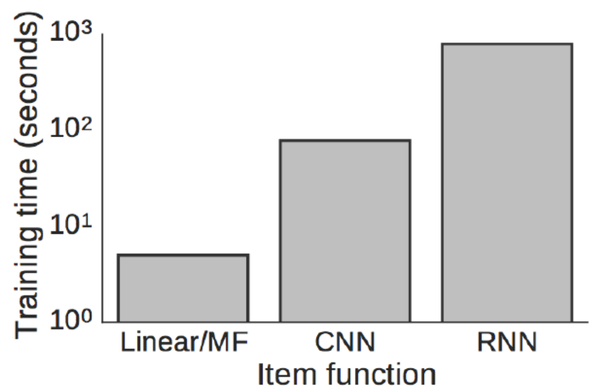
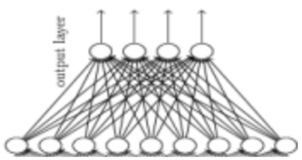


Figure 2: Model training time per epoch with different types of item functions (in log-scale).

Three Types of Computations

Major Computation Cost Breakdown

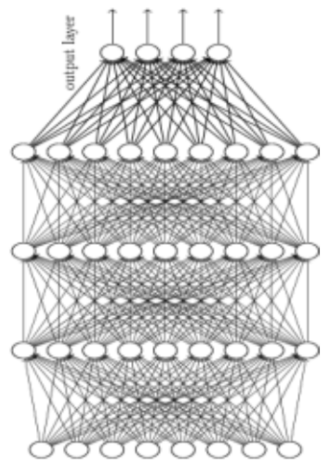
(both forward/backward)



User function computation

t_f

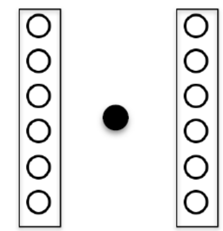
10



Item function computation

t_g

100



Interaction function (dot product) computation

t_i

1

Very rough order of magnitude estimate of **time units** (depending on specific configurations)

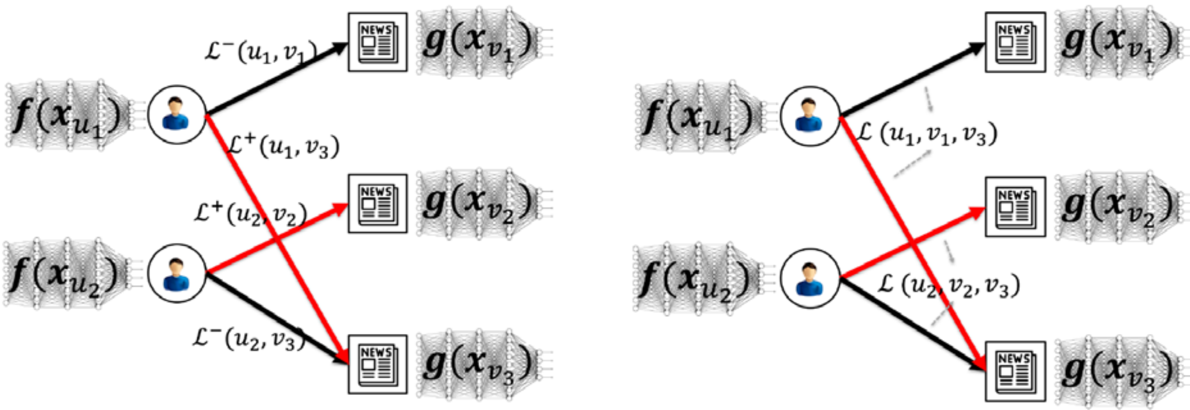
The Cost Model in a Mini-Batch

- Assume in a Mini-Batch, we have
  - $\# \, users$
  - $\# \, items$
  - $\# \, user - item \; interactions$
- Computation cost:
  - $t_f * \#users + \boxed{t_g} * \#items + t_i * \#interactions$

Most expensive!

Solution to Speedup Computation

- Design sampling strategies that can share the computational costs on the node type that are expensive
  - Data to sample here: Links between users and items
  - Major computational costs: on nodes, esp. on items that involves rich text

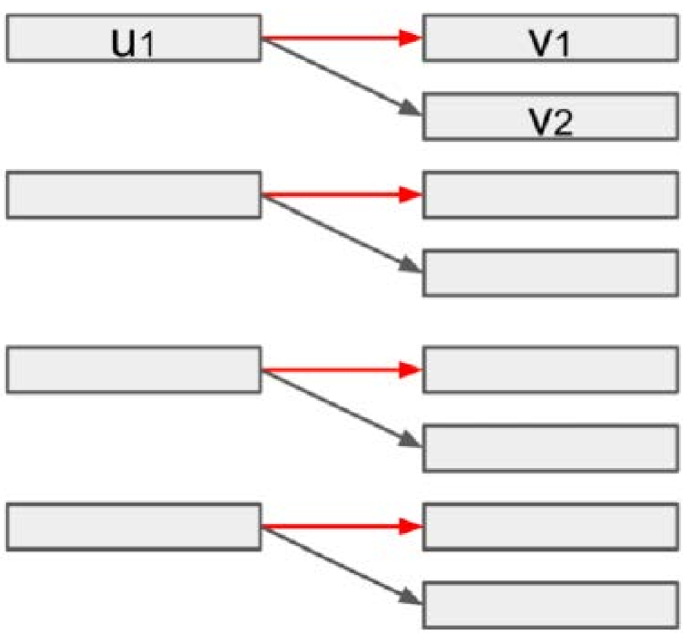


Pointwise Loss

Pairwise Loss

Existing Sampling Strategies

- Negative Sampling: b positive links and k negative links for each positive link
  - In each mini-batch: no items are shared!

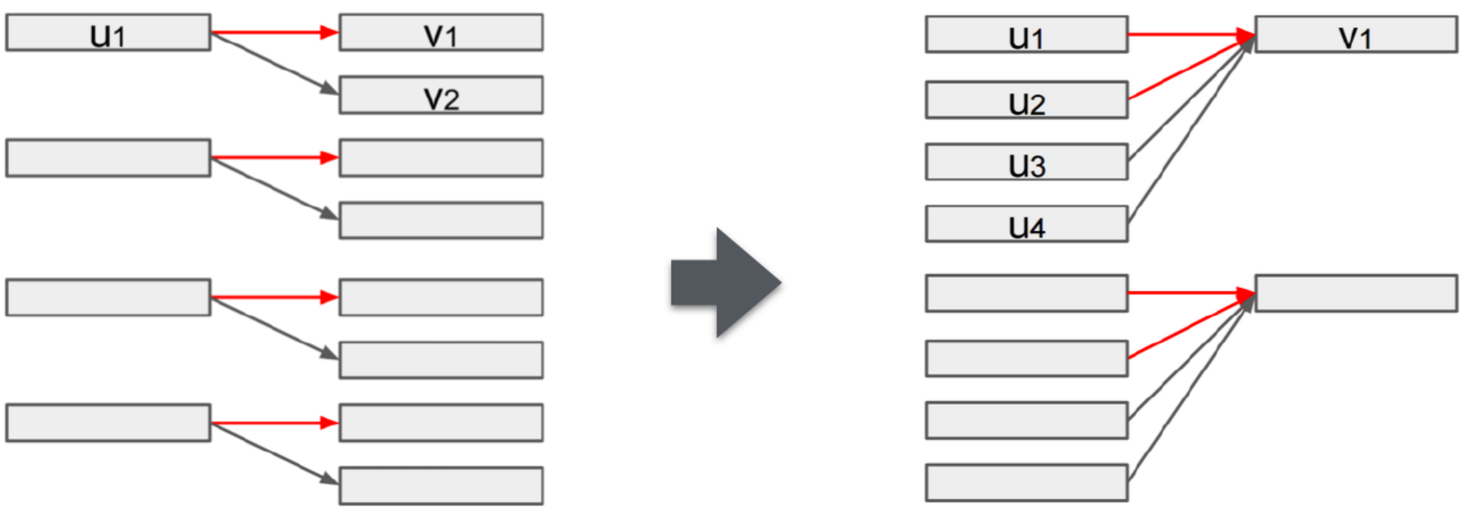


- $\# \, users: b$
- $\# \, items: (1+k)b$
- $\# \, interactions: (1+k)b$

- **Main cost: $t_g * (1+k)b$**

Proposed Strategy 1: Stratified Sampling

- Share items: b positive links and k negative links for each positive link; # of positive link per item: s



(a) Negative

- $\# \, users: b$
- $\# \, items: (1+k)b$
- $\# \, interactions: (1+k)b$

- **Main cost: $t_g * (1+k)b$**

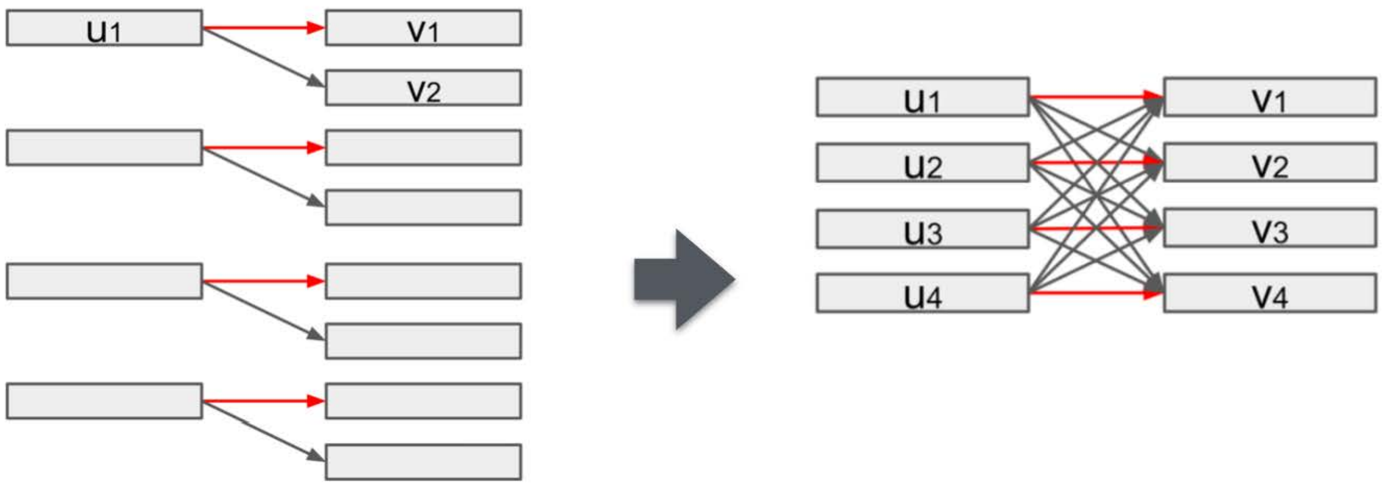
(b) Stratified (by Items)

- $\# \, users: (1+k)b$
- $\# \, items: b/s$
- $\# \, interactions: (1+k)b$

- **Main cost: $t_g * b/s$**

Proposed Strategy 2: Negative Sharing

- Treat all the non-links as negative links, again share items: still b positive links



(a) Negative

(c) Negative Sharing

- $\# \, users: b$
- $\# \, items: (1+k)b$
- $\# \, interactions: (1+k)b$

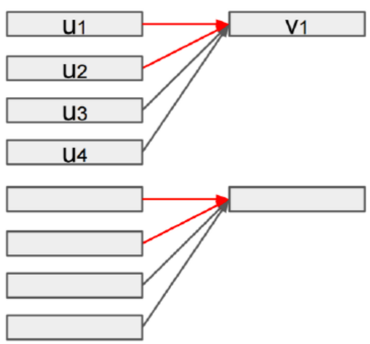
- **Main cost: $t_g * (1+k)b$**

- $\# \, users: b$
- $\# \, items: b$
- $\# \, interactions: b^2$

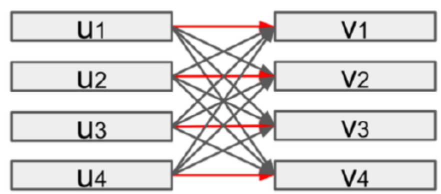
- **Main cost: $t_g * b$**

Combine Two Strategies

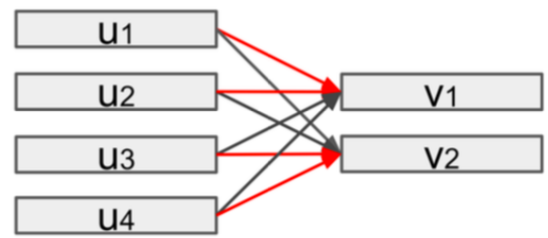
- Stratified sampling only: Cannot deal with ranking-based loss functions
- Negative sharing only: Too many negative links used, diminishing return



(b) Stratified (by Items)



(c) Negative Sharing



Still b positive links, s positive links per item

(d) Stratified with N.S.

- $\# \, users: b$
- $\# \, items: b/s$
- $\# \, interactions: b * b/s$

- **Main cost: $t_g * b/s$**

Cost Summary

Sampling	# pos. links	# neg. links	# t_f	# t_g	# t_i	pointwise	pairwise
IID [3]	b	bk	$b(1+k)$	$b(1+k)$	$b(1+k)$ vec	✓	×
Negative [1, 21, 29]	b	bk	b	$b(1+k)$	$b(1+k)$ vec	✓	✓
Stratified (by Items)	b	bk	$b(1+k)$	$\frac{b}{s}$	$b(1+k)$ vec	✓	×
Negative Sharing	b	$b(b-1)$	b	b	$b \times b$ mat	✓	✓
Stratified with N.S.	b	$\frac{b(b-1)}{s}$	b	$\frac{b}{s}$	$b \times \frac{b}{s}$ mat	✓	✓

- Computation cost estimation (using b=256, k=20, t_f=10, t_g=100, t_i=1, s=2)
  - IID sampling: 597k
  - Negative sampling: 546k
  - Stratified sampling (by item): 72k
  - Negative Sharing: 28k
  - Stratified sampling with negative sharing: 16k

(all in time units)

Experimental Results

- Speedup up to 30 times with even performance improvement
- Datasets

Table 1: Data statistics for user, items and their interactions.

	# of user	# of item	# of interaction
Citeulike	5,551	16,980	204,986
News	10,000	58,579	515,503

Table 2: Data statistics for text content.

	voc. size	max	min	mean	median
Citeulike (title)	4,777	15	2	9	9
Citeulike (title&abs.)	23,011	300	22	194	186
News (title)	16,589	20	1	11	11
News (title&sum.)	41,537	200	2	89	90

Running Time

Total speedup = speedup per iter * speedup of # iter

Table 3: Comparisons of speedup for different sampling strategies against IID Sampling.

Model	Sampling	CiteULike Per it.	CiteULike # of it.	CiteULike Total	News Per it.	News # of it.	News Total
CNN	Negative	1.02	1.00	1.02	1.03	1.03	1.06
	Stratified	8.83	0.97	8.56	6.40	0.97	6.20
	N.S.	8.42	$\boxed{\textbf{2.31}}$	19.50	6.54	$\boxed{\textbf{2.21}}$	14.45
	Strat. w. N.S.	**15.53**	$\boxed{1.87}$	**29.12**	**11.49**	$\boxed{2.17}$	**24.98**
LSTM	Negative	0.99	0.96	0.95	1.0	1.25	1.25
	Stratified	3.1	0.77	2.38	3.12	1.03	3.22
	N.S.	2.87	$\boxed{\textbf{2.45}}$	7.03	2.78	$\boxed{\textbf{4.14}}$	**11.5**
	Strat. w. N.S.	**3.4**	$\boxed{2.22}$	**7.57**	**3.13**	$\boxed{3.32}$	10.41

Converge faster and fewer iterations are needed when more links are used!

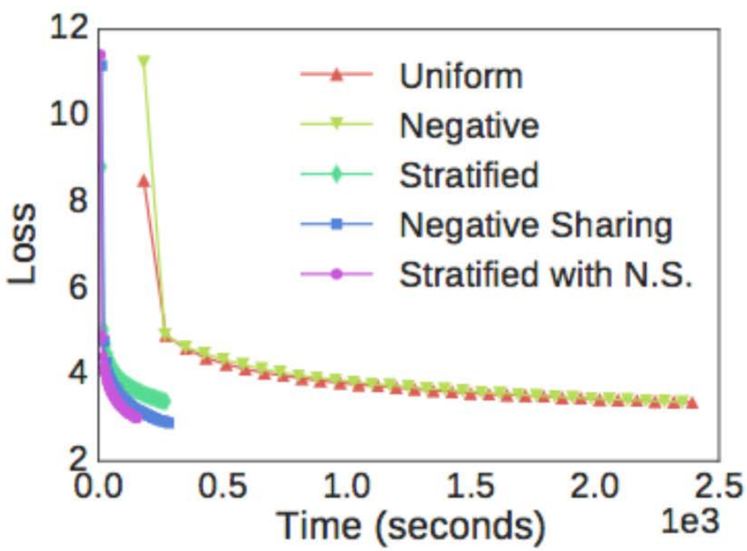
Performance

Table 4: Recall@50 for different sampling strategies under different models and losses.

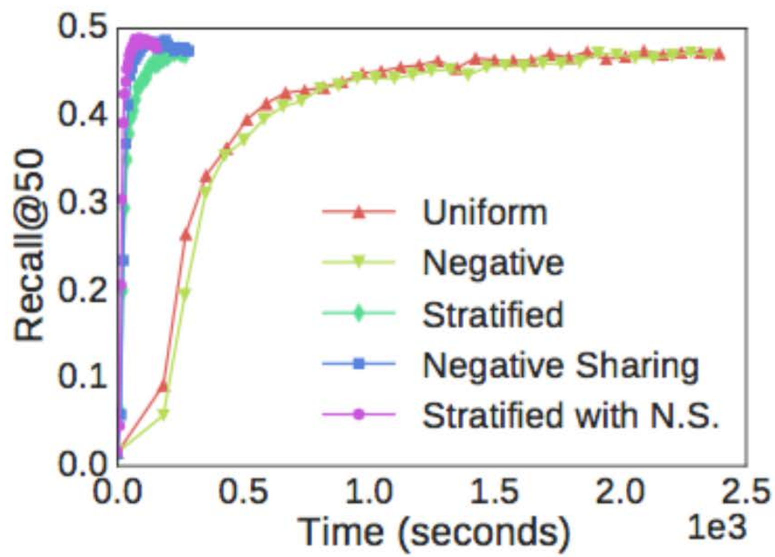
		CiteULike				News			
Model	Sampling	SG-loss	MSE-loss	Hinge-loss	Log-loss	SG-loss	MSE-loss	Hinge-loss	Log-loss
CNN	IID	0.4746	0.4437	-	-	0.1091	0.0929	-	-
	Negative	0.4725	0.4408	0.4729	0.4796	0.1083	0.0956	0.1013	0.1009
	Stratified	0.4761	0.4394	-	-	0.1090	0.0913	-	-
	Negative Sharing	0.4866	0.4423	**0.4794**	0.4769	0.1131	0.0968	0.0909	0.0932
	Stratified with N.S.	**0.4890**	**0.4535**	0.4790	**0.4884**	**0.1196**	**0.1043**	**0.1059**	**0.1100**
LSTM	IID	0.4479	0.4718	-	-	0.0971	0.0998	-	-
	Negative	0.4371	0.4668	0.4321	0.4540	0.0977	0.0977	0.0718	0.0711
	Stratified	0.4344	0.4685	-	-	0.0966	0.0996	-	-
	Negative Sharing	0.4629	0.4839	0.4605	0.4674	**0.1121**	0.0982	0.0806	0.0862
	Stratified with N.S.	**0.4742**	**0.4877**	**0.4703**	**0.4730**	0.1051	**0.1098**	**0.1017**	**0.1002**

Convergence Curves

- Convergence to a better place using much less time



training



test

Summary

- Information network view of recommendation task
  - Capture context-rich environment
- Information network mining approaches can help recommendation tasks
  - Better performance and better interpretability
- Meta-path is powerful in capturing different intentions and similarities
- Sampling strategy becomes important when dealing with neural network-based collaborative filtering

References

- T. Chen, Y. Sun, Y. Shi, and L. Hong. On Sampling Strategies for Neural Network-based Collaborative Filtering. In KDD, 2017.
- T. Chen and Y. Sun. Task-Guided and Path-Augmented Heterogeneous Network Embedding for Author Identification. In WSDM, 2017.
- G. Fu, Y. Ding, A. Seal, B. Chen, Y. Sun, and E. Bolton. Predicting drug target interactions using meta-path-based semantic network analysis In BMC Bioinformatics 17:160, 2016.
- Hao Ma, Michael R. Lyu, Irwin King. Learning to Recommend with Trust and Distrust Relationships. In RecSys, 2009.
- Hao Ma, Dengyong Zhou, Chao Liu, Michael R. Lyu, Irwin King. Recommender Systems with Social Regularization. In WSDM, 2011.
- Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu. PathSim: Meta path-based top-k similarity search in heterogeneous information networks. In VLDB'11/PVLDB, Seattle, WA, Aug. 2011.
- Y. Sun, Jiawei Han, Charu C. Aggarwal, and Nitesh V. Chawla. When Will It Happen? --- Relationship Prediction in Heterogeneous Information Networks. In WSDM, 2012.
- Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han, Co-Author Relationship Prediction in Heterogeneous Bibliographic Networks. In ASONAM, 2011.
- X. Yu, X. Ren, Y. Sun, B. Sturt, U. Khandelwal, Q. Gu, B. Norick, and J. Han. Personalized Entity Recommendation: A Heterogeneous Information Network Approach. In WSDM, 2014.
- X. Yu, X. Ren, Q. Gu, Y. Sun, and J. Han. Collaborative Filtering with Entity Similarity Regularization in Heterogeneous Information Networks. In IJCAI-HINA, 2013.
- X. Yu, X. Ren, Y. Sun, B. Sturt, U. Khandelwal, Q. Gu, B. Norick, and J. Han. HeteRec: Entity Recommendation in Heterogeneous Information Networks with Implicit User Feedback. In RecSys, 2013.

Outline

- Part I: Introduction and Preliminaries
- Part II: Recommendation in Heterogeneous Information Networks
- Part III: Recommendation in a Text-Rich Setting
- Part IV: Recommendation with Spatio-Temporal Information
- Part V: Research Frontiers and Summary

Break

PART III: RECOMMENDATION IN A TEXT-RICH SETTING

Xiang Ren
Department of Computer Science
University of Illinois at Urbana-Champaign / USC
xren7@illinois.edu

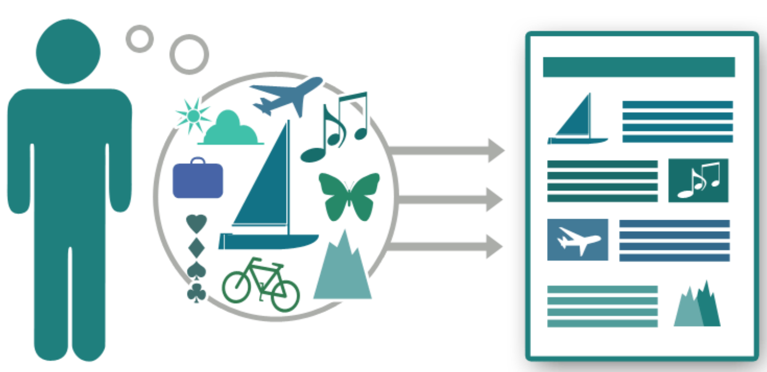
August 11, 2017

Outline

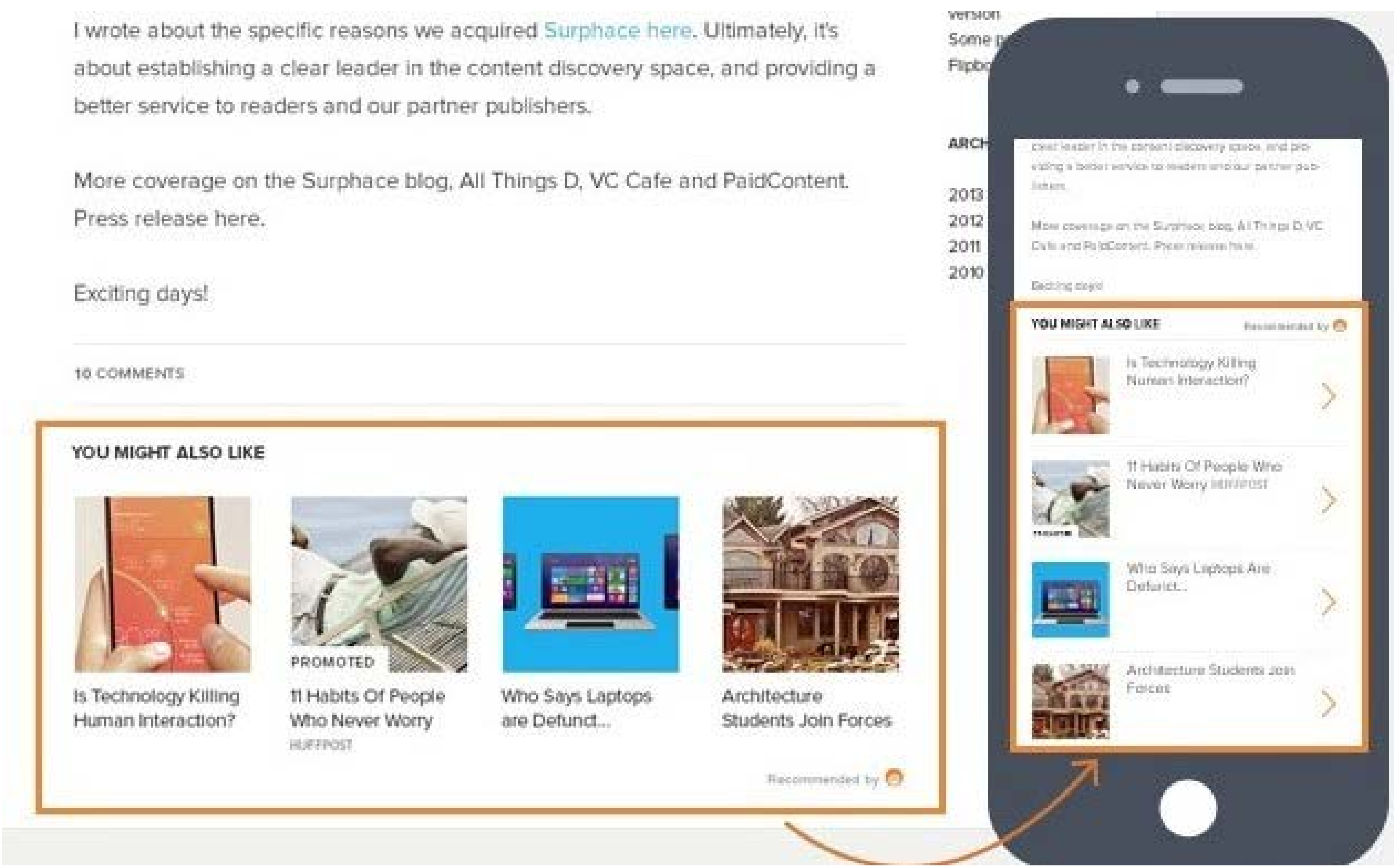
- Background
- Content-based Recommendation: An Overview
- Recommendation in Text-Rich Information Network
- Recommendation in Networks Constructed from Text
- Summary

Textual Information in Recommendation

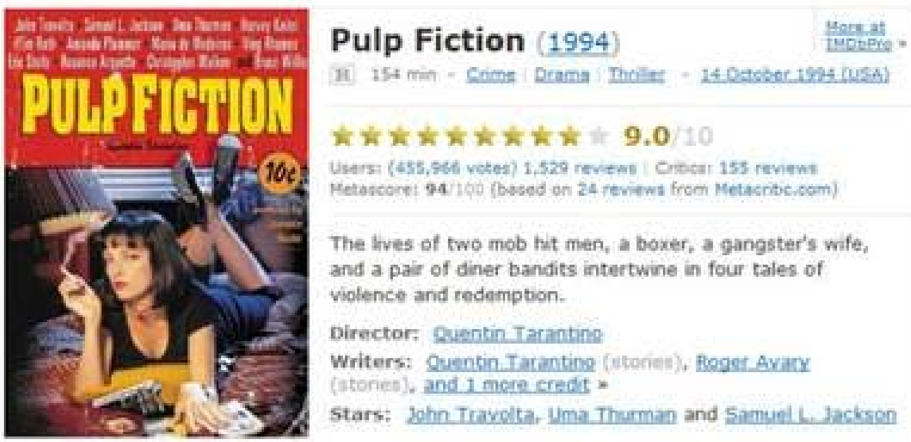
- Rich text information are associated with users and items
  - User → textual user profile
  - Product / Movie → description, review
  - News article → textual content
  - Scientific paper → textual content



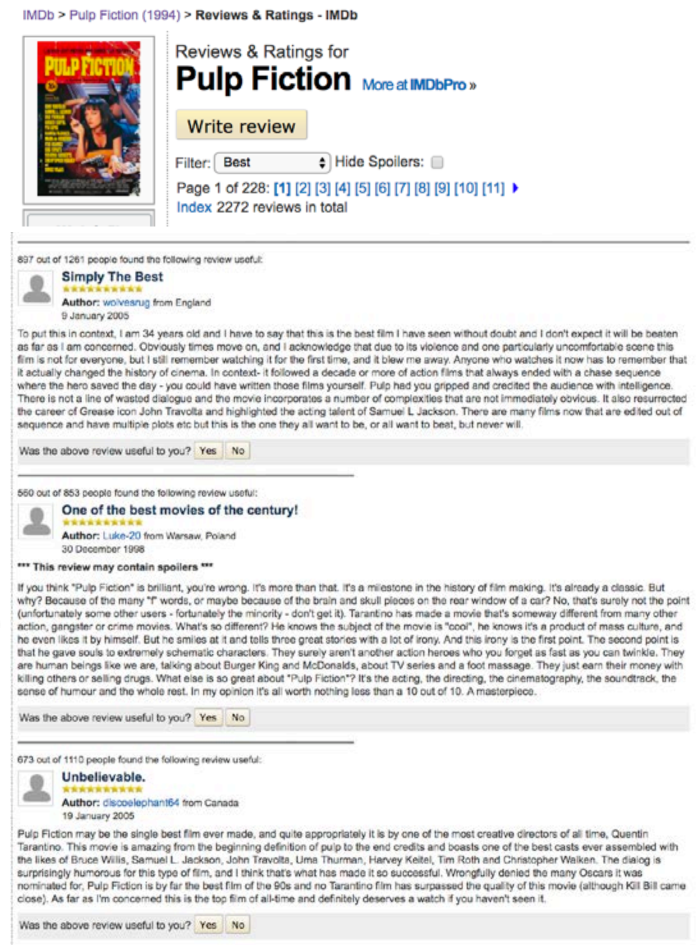
Use Case I: Recommending Related Articles



Use Case II: Movie Recommendation



More information about the movie.....

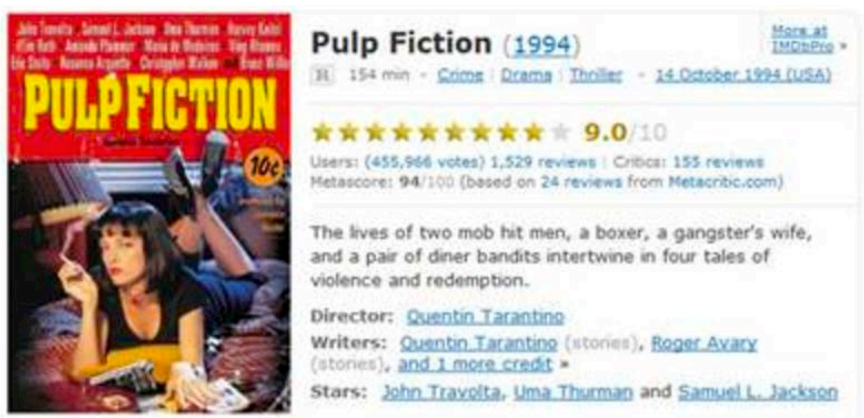


Challenge I

- Unstructured textual information → clean, structured representation?
  - What are the semantic units?
    - Word, n-gram, phrases, entities, …
  - Text is highly variable → data sparsity
    - Domains, genres, languages
  - How to aggregate for objects (user, item, etc.)
    - Weighting methods?

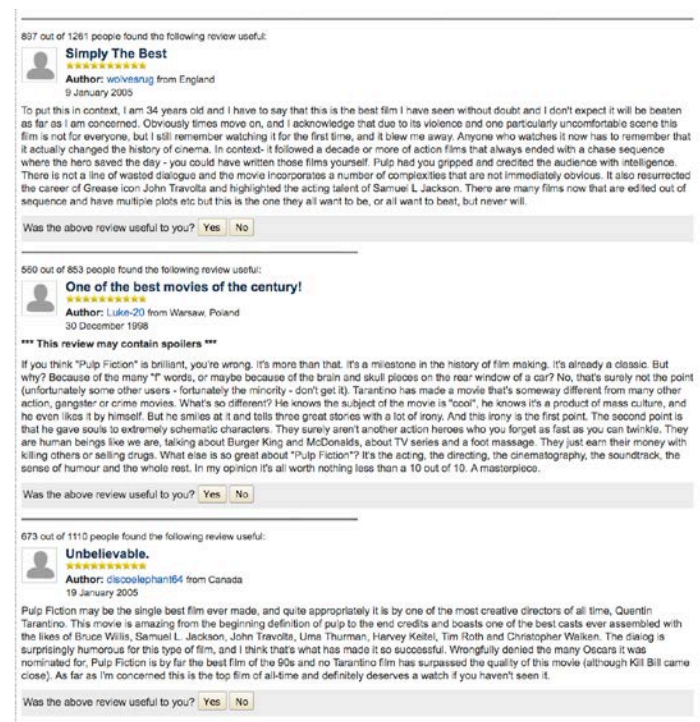
Challenge II

- How to unify <u>textual</u> information with existing <u>structured</u> information?



Structured attribute information

+



Unstructured review text

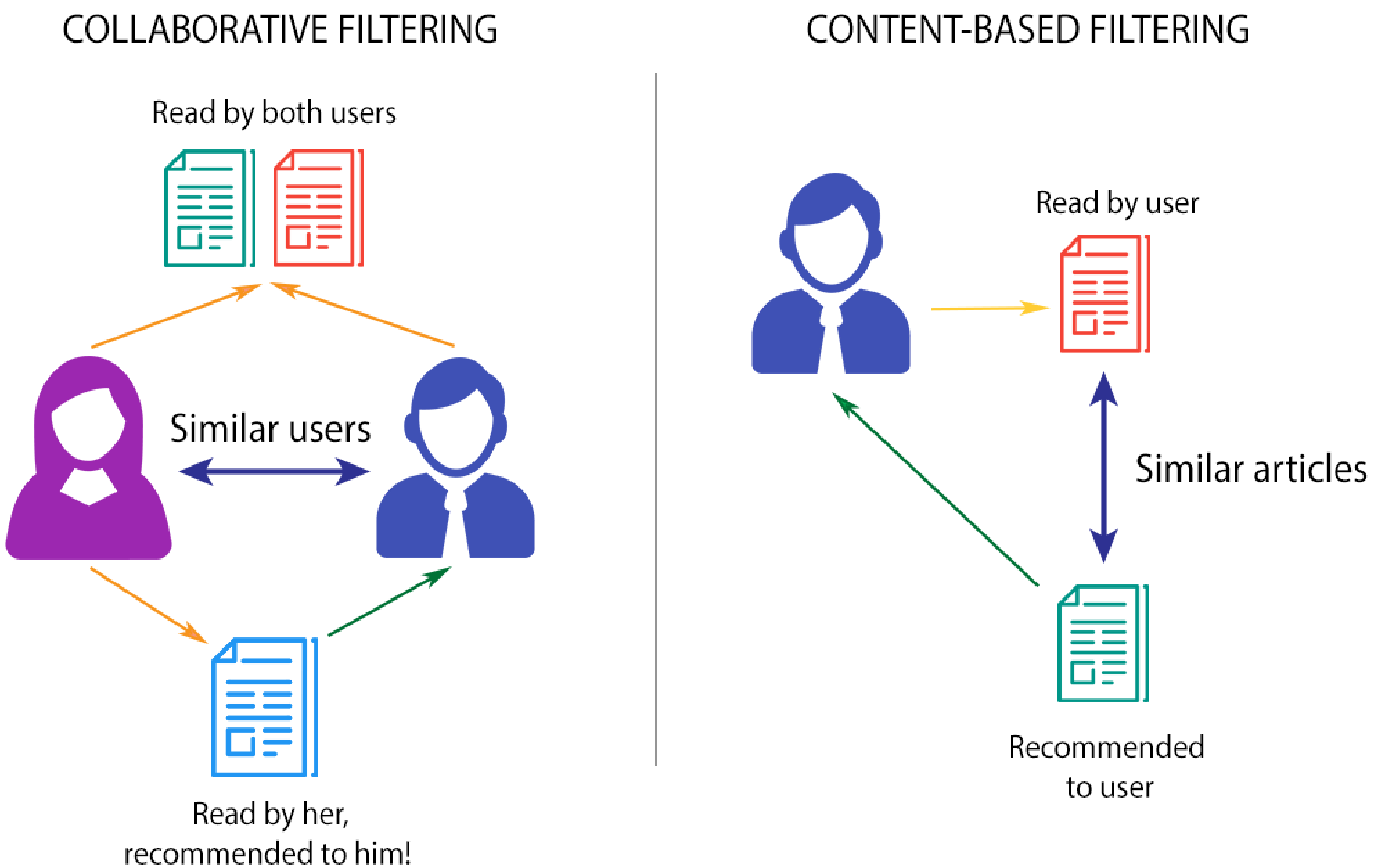
How to Leverage Textual Information?

- Feature-based Approach
  - Content-based recommendation
- Network-based Approach
  - Recommendation in Text-Rich Information Network
- Text-to-Network Approach
  - Recommendation in Networks Constructed from Text

Outline

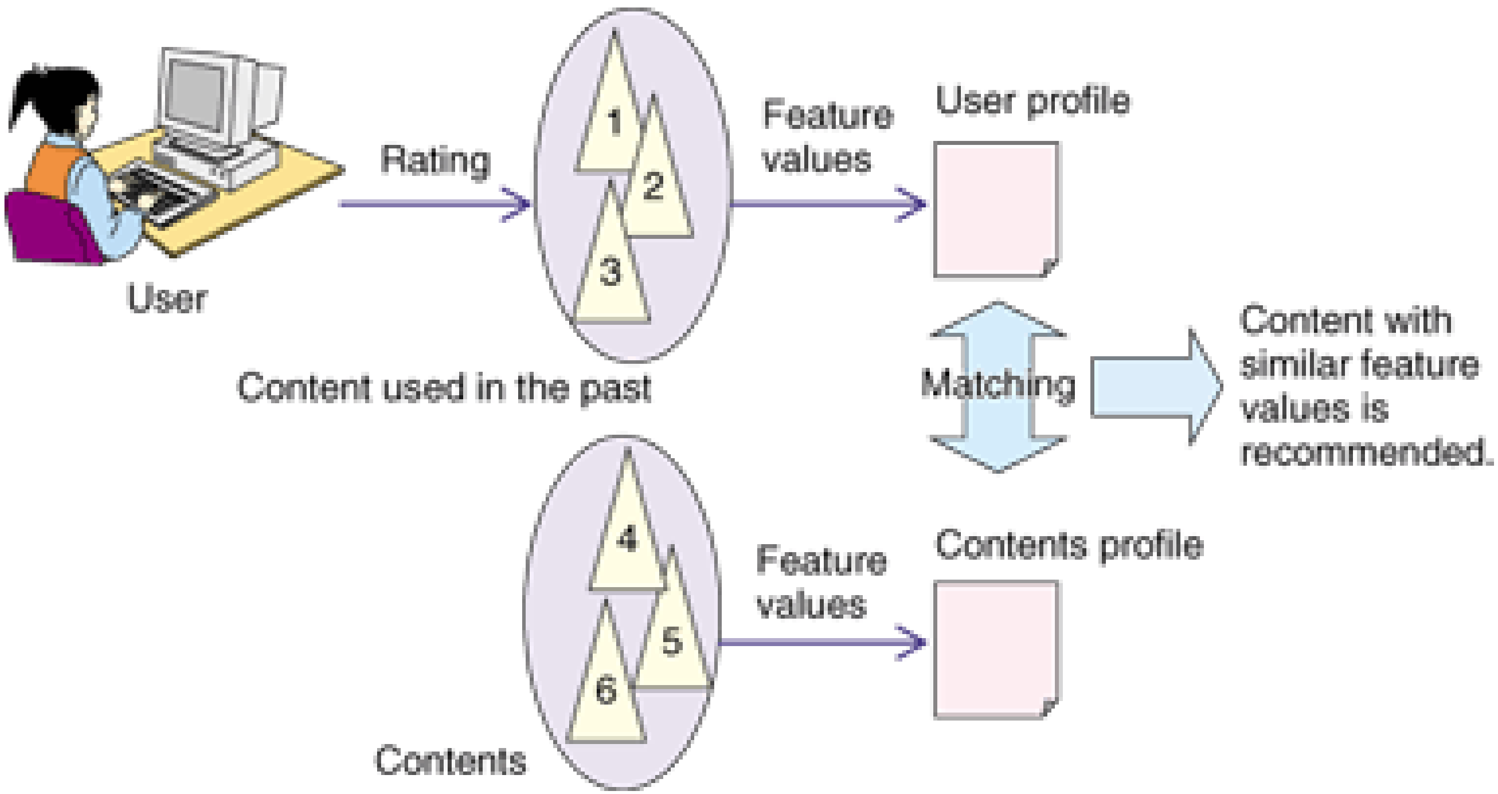
- Background
- Content-based Recommendation: An Overview
- Recommendation in Text-Rich Information Network
- Recommendation in Networks Constructed from Text
- Summary

Collaborative Filtering vs. Content-Based Recommendation



https://www.themarketingtechnologist.co/building-a-recommendation-engine-for-geek-setting-up-the-prerequisites-13

Content-Based Recommendation: Basic Idea



Advantages

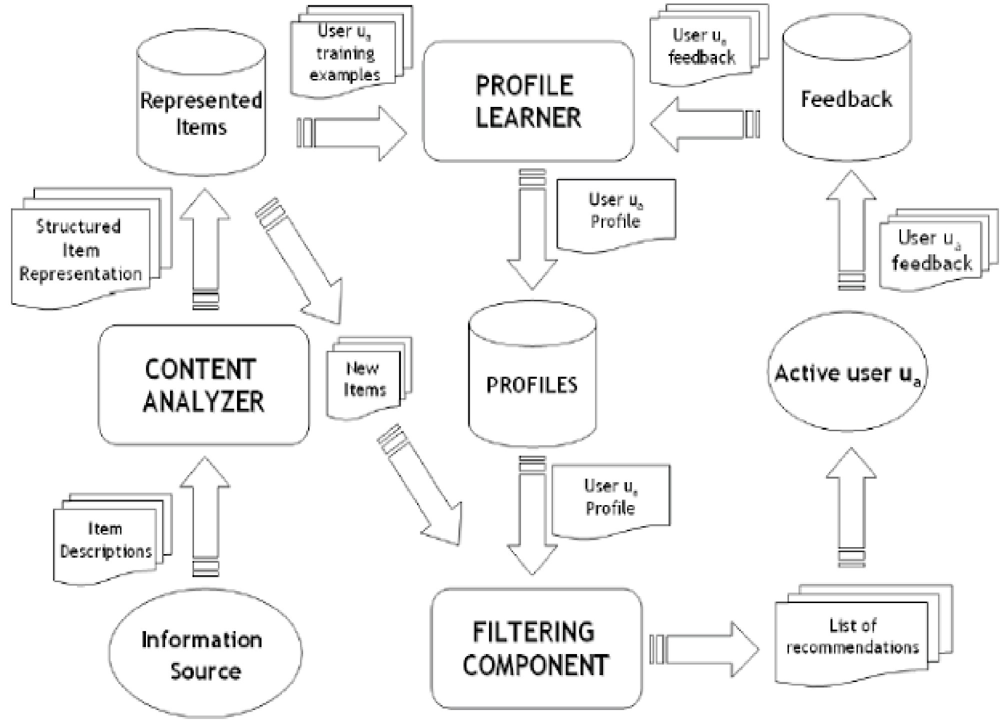
- User independence
  - Item profile & user profile
  - No need for other users' ratings (vs. CF)
- Transparency
  - Profile features → Why it is recommended?
  - CF: unknown users have similar tastes as yours
- No "cold start"
  - Effective on new items (if profile known)

Disadvantages

- Restriction on content analysis
  - Item profile is vague → low performance
- “No surprise”
  - Known features → no degree of “novelty”
- New users
  - Missing/incomplete user info → low performance

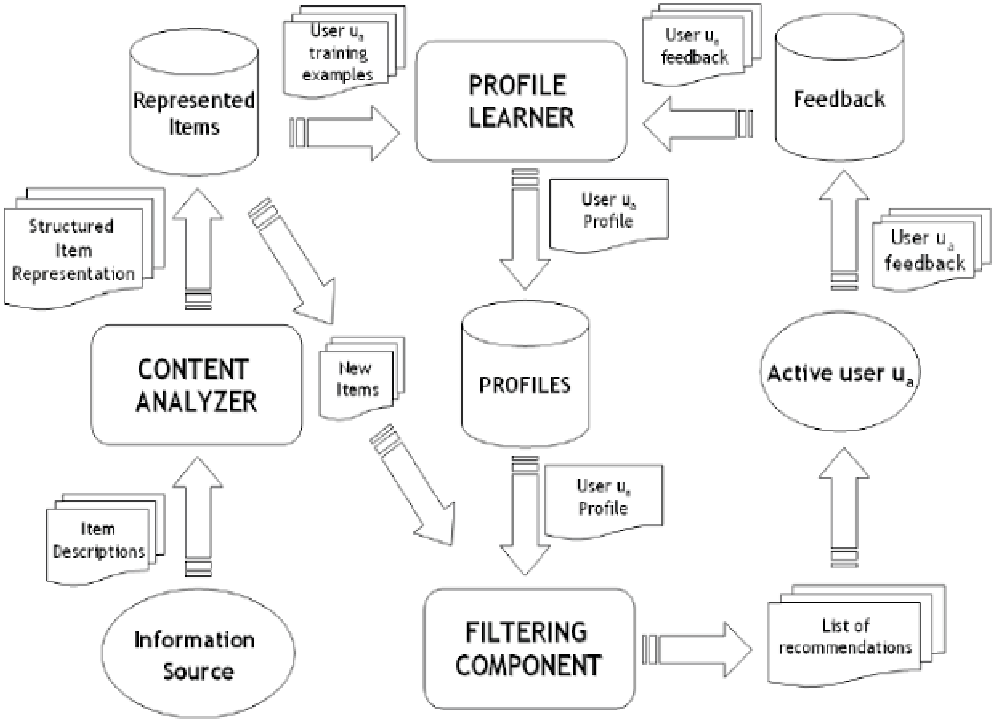
Major Components

- Content analyzer
  - Item → features
  - Feature engineering, information extraction



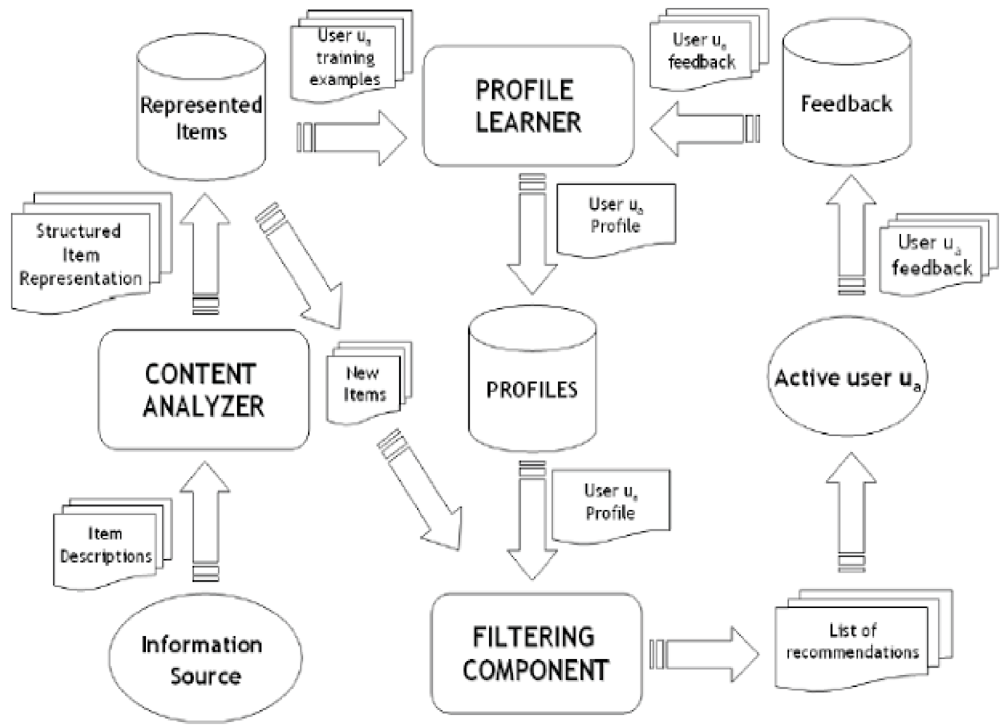
Major Components

- Content analyzer
  - Item → features
  - Feature engineering, information extraction
- Profiler learner
  - User → feature profile
  - Data integration, user modeling



Major Components

- Content analyzer
  - Item → features
  - Feature engineering, information extraction
- Profiler learner
  - User → feature profile
  - Data integration, user modeling
- Filtering component
  - User → item recommendation



Content Analyzer: Item Representation

- Items stored in a database table

ID	Name	Cuisine	Service	Cost
1001	Mike's Pizza	Italian	Counter	Low
1002	Chris's Café	French	Table	Medium
1003	Jacques Bistro	French	Table	High

- Structured data
  - Small number of attributes
  - Each item is described by the same set of attributes
  - Known set of values that the attributes may have

Content Analyzer: Item Representation

- Information about item could also be free text
  - text description, customer review, news articles
- Unstructured data
  - No attribute names with well-defined values
  - Natural language complexity
    - Same word with different meanings
    - Different words with same meaning

Item Representation: TF-IDF Weighting

- Compute a weight for each term that represents the importance or relevance of that term
  - The term with highest weight occur more often in that document than in other documents
    - → more central to the topic of the document

Item Representation: TF-IDF Weighting

- Limitations
  - This method does not capture the context in which a word is used
  - “This restaurant does not serve vegetarian dishes”

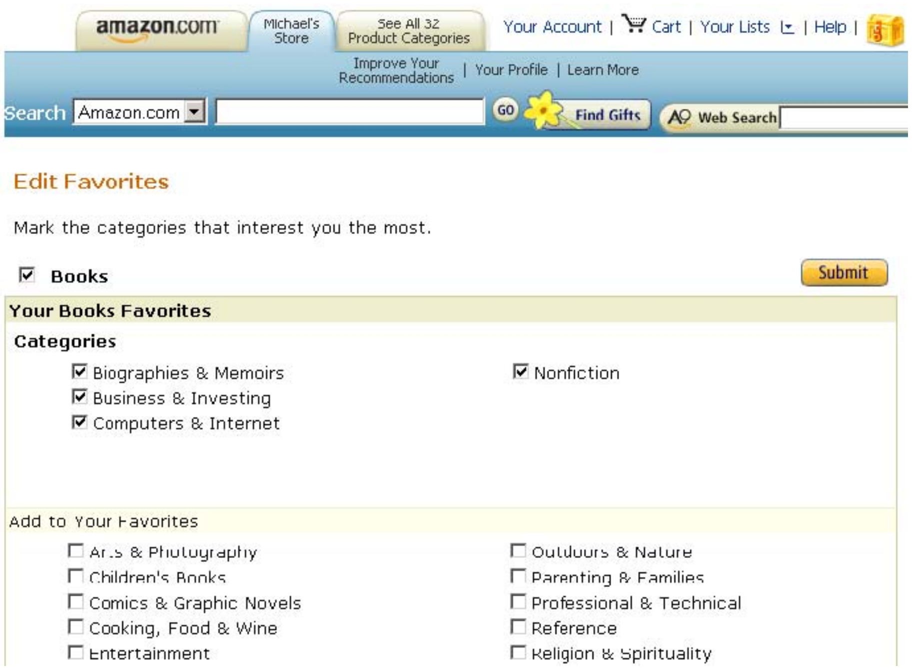
- Information extraction
  - turning text into machine-readable structures

User Profiles

- This profile consists of two main types of information
  - User's interaction history.
    - items viewed by a user, items purchased by a user, search queries, etc.
  - A model of the user's interests/preferences
    - $s_{ij} = f(U_i, I_j)$ where U_i is user representation and I_j is item representation
    - → How likely an user is interested in an item

User Interest Modeling

- "Manual" user interest modeling
  - User customization
    - Provide "check box" interface that let the users construct their own profiles of interests

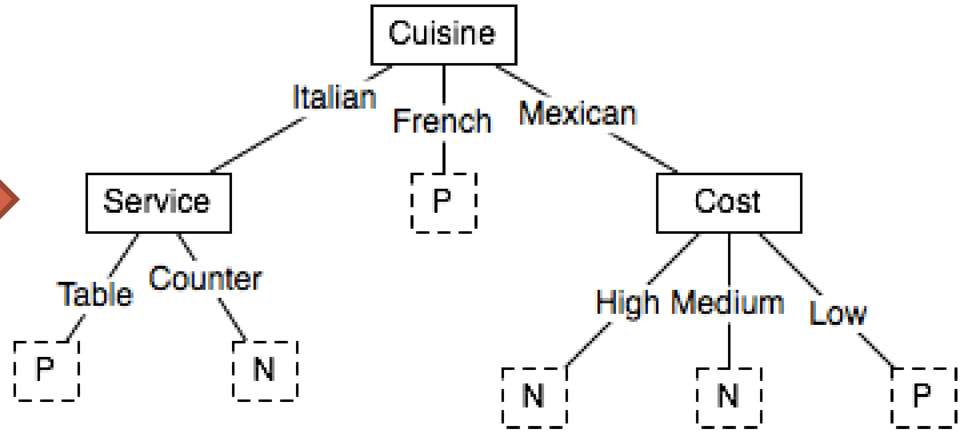


Limitations
- Efforts from user
- Cannot cope with changes in user interests
- …

User Interest Modeling

- Learning a user interest model
  - Learning a classifier
    - Decision tree, Naïve Bayes, SVM, NeuralNets
  - Training data: user-item interaction history
    - Explicit ratings, implicit feedbacks
    - Feature space ←→ features for user/item representations

Cuisine	Service	Cost	Rating
Italian	Counter	Low	Negative
French	Table	Med	Positive
French	Counter	Low	Positive
…	…	…	…



Content-Based Recommendation: Summary

- Content-based Recommendation
  - Basic Idea
  - Pros & cons
  - Major components
- Item Representation
- User Profiles
  - Manual interest crafting
  - Learning A User Interest Model

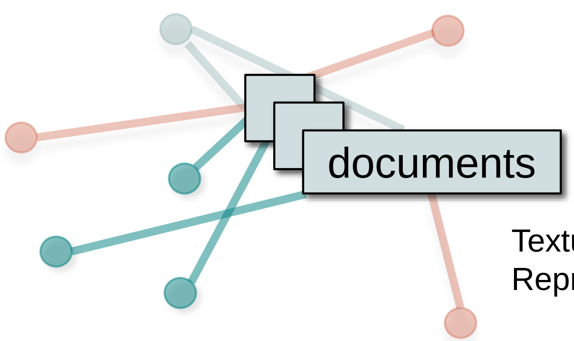
Outline

- Background
- Content-based Recommendation: An Overview
- Recommendation in Text-Rich Information Network
- Recommendation in Networks Constructed from Text
- Summary

Representing Text : Two Approaches

- Feature-based Approach
  - Object (user, item, etc.) → feature representation
  - → Content-based recommendation models

"DBSCAN is a method for clustering in process of knowledge discovery."



Textual Feature Representation

Words:
dbscan, methods, clustering, process, …

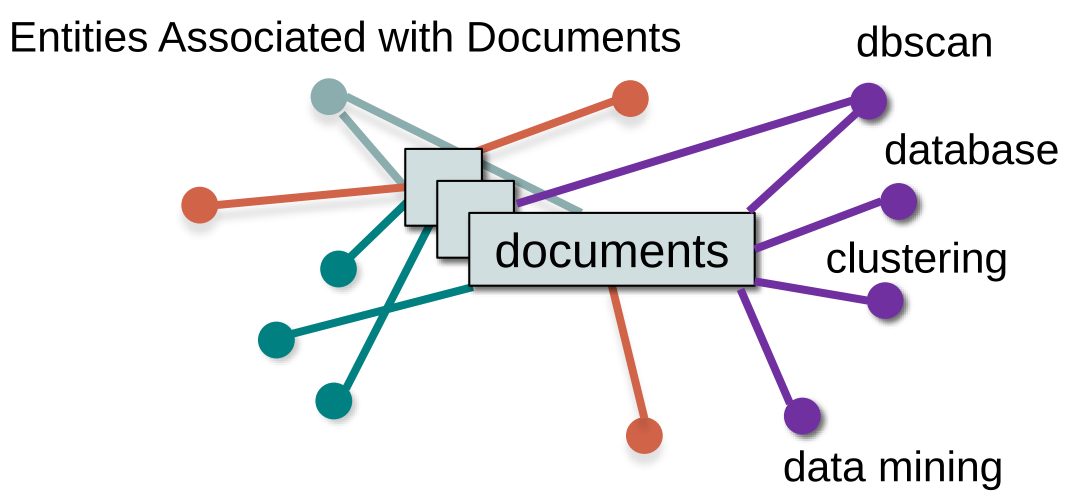
Topics:
[k-means, clustering, clusters, dbsca, …]
[clusters, density, dbscan, clustering, …]
[machine, learning, knowledge, mining, …]

Knowledge base concepts:
data mining: /m/0blvg
clustering analysis: /m/031f5p
dbscan: /m/03cg_k1

Document Keyphrase:
dbscan: [dbscan, density, clustering, ...]
clustering: [clustering, clusters, partition, ...]
data mining: [data mining, knowledge, ...]

Representing Text : Two Approaches

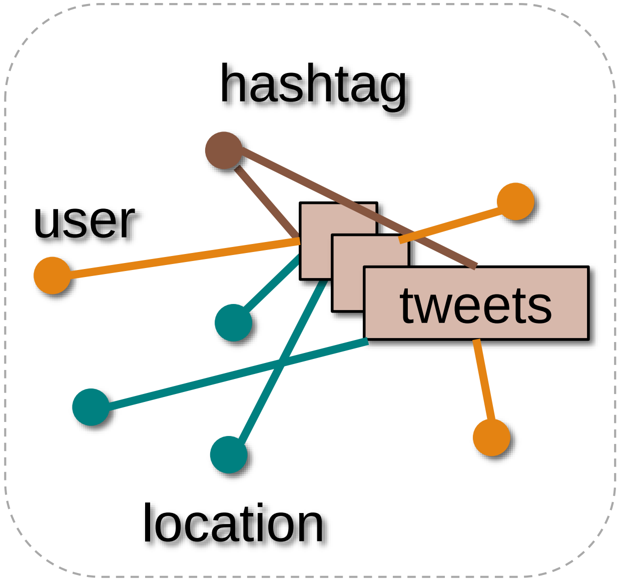
- Network-based Approach
  - Information network as an unified data model
    - objects & text units → nodes
    - object-text unit relationships → edges
  - → Recommendation in Text-Rich Information Network

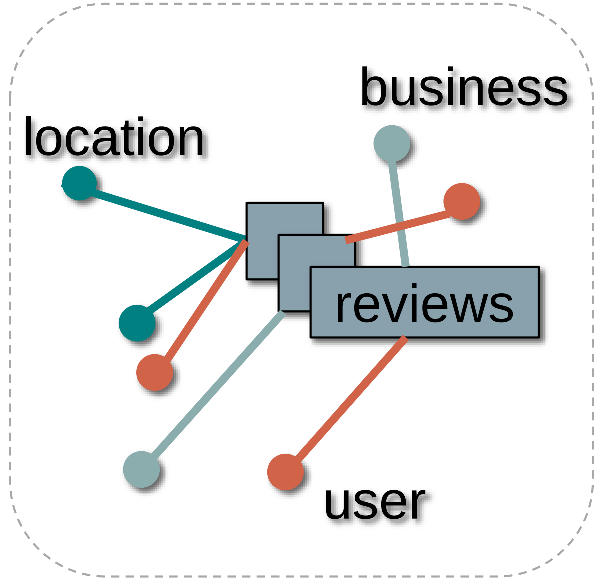


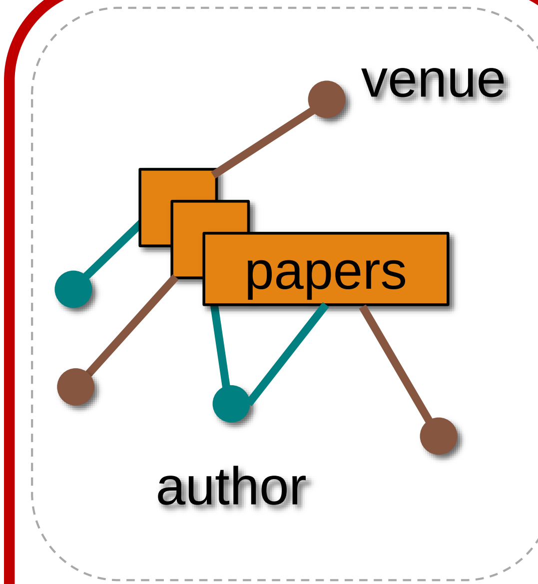
Network-Based Approach: Advantages

- Unified representation
  - Structured & unstructured information
- Richer semantics
  - Capture relationships between textual units
- Collective inference
  - Model objects jointly

Examples







Article / post Recommendation

Point-of-Interest Recommendation

Citation Recommendation

Citation Recommendation: Motivation

citation recommendation clustering

Scholar

About 65,800 results (**0.09** sec)

Articles

Case law

My library

Any time

Since 2014

Since 2013

Since 2010

Custom range...

Sort by relevance

Sort by date

include patents

include citations

[Personalized **recommendation** in social tagging systems using hierarchical **clustering**]

A Shepitsen, J Gemmell, B Mobasher... - Proceedings of the 2008 ..., 2008 - dl.acm.org

... for profit or commercial advantage and that copies bear this notice and the full **citation** on the ... Other collaborative tagging applications focus on blogs, **citations** and wikis. ... if the personalized approach moves the resource further down the ranking in the **recommendation** set, the ...

Cited by 334 Related articles All 11 versions Cite Save

[Context-aware **citation recommendation**]

Q He, J Pei, D Kifer, P Mitra, L Giles - Proceedings of the 19th ..., 2010 - dl.acm.org

... The bibliography candidates pro- vided by a global **recommendation** should collectively satisfy the **citation** information needs of all out-link ... Definition 3.3 (Local **Recommendation**). ... out-link local context c∗ with respect to d, a local recom- mendation is a ranked list of **citations** in a ...

Cited by 94 Related articles All 16 versions Cite Save

[**Citation recommendation** without author supervision]

Q He, D Kifer, J Pei, P Mitra, CL Giles - ... on Web search and data mining, 2011 - dl.acm.org

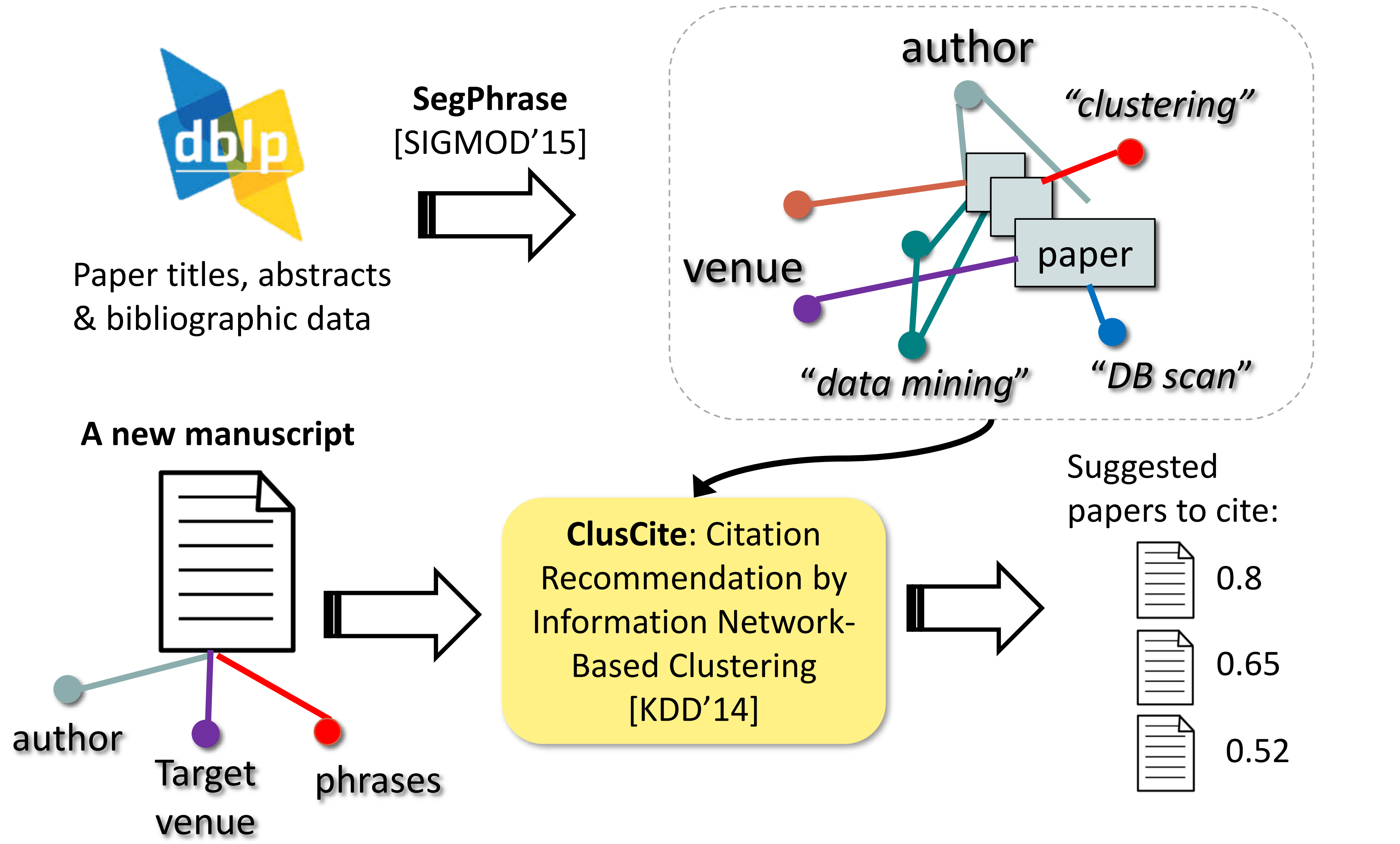
... SETUP In this section, we introduce notation and terminology, and describe the **citation recommendation** problem. ... it to try to recognize locations in the query manuscript d where **citations** should exist. ... goal is to clus- ter this bipartite graph to obtain **clusters** of **citation** contexts and ...

Cited by 28 Related articles All 10 versions Cite Save

Citation Recommendation: Motivation

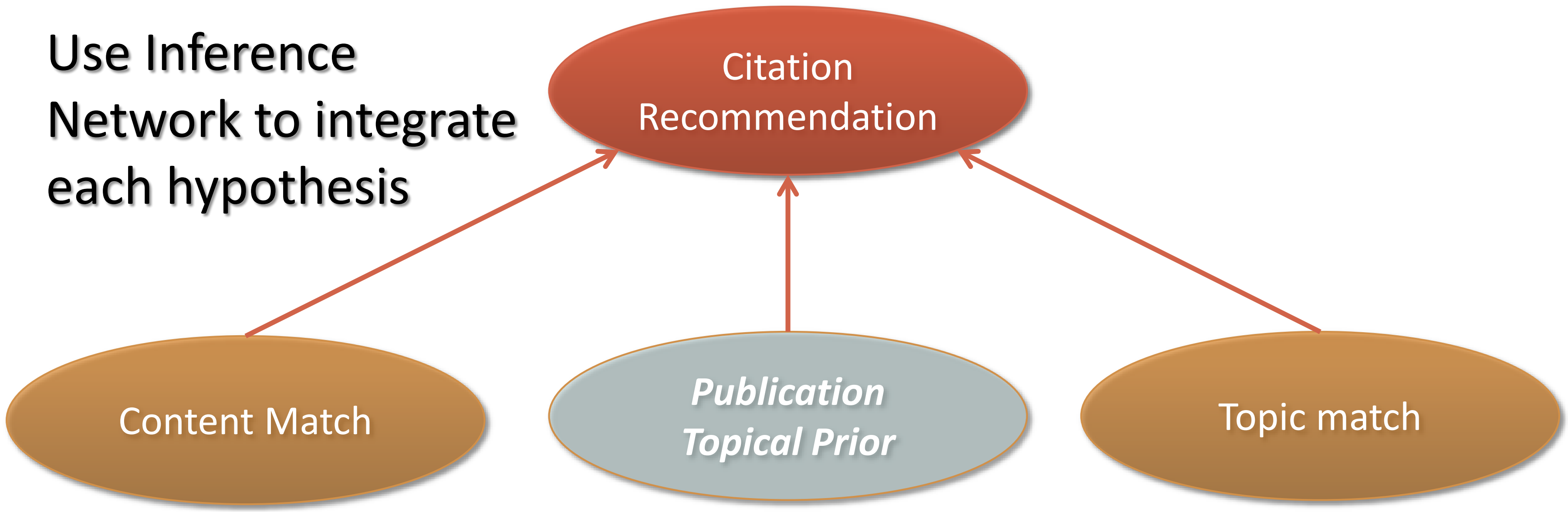
- Research papers need to cite relevant and important previous work
  - background, context and innovation
- Already large, rapidly growing body of scientific literature
  - automatic recommendations of high quality citations
- Traditional literature search systems
  - rich information needs → queries with a few keywords

Problem Statement



Prior Art: A Global Recommendation Model

Use Inference Network to integrate each hypothesis

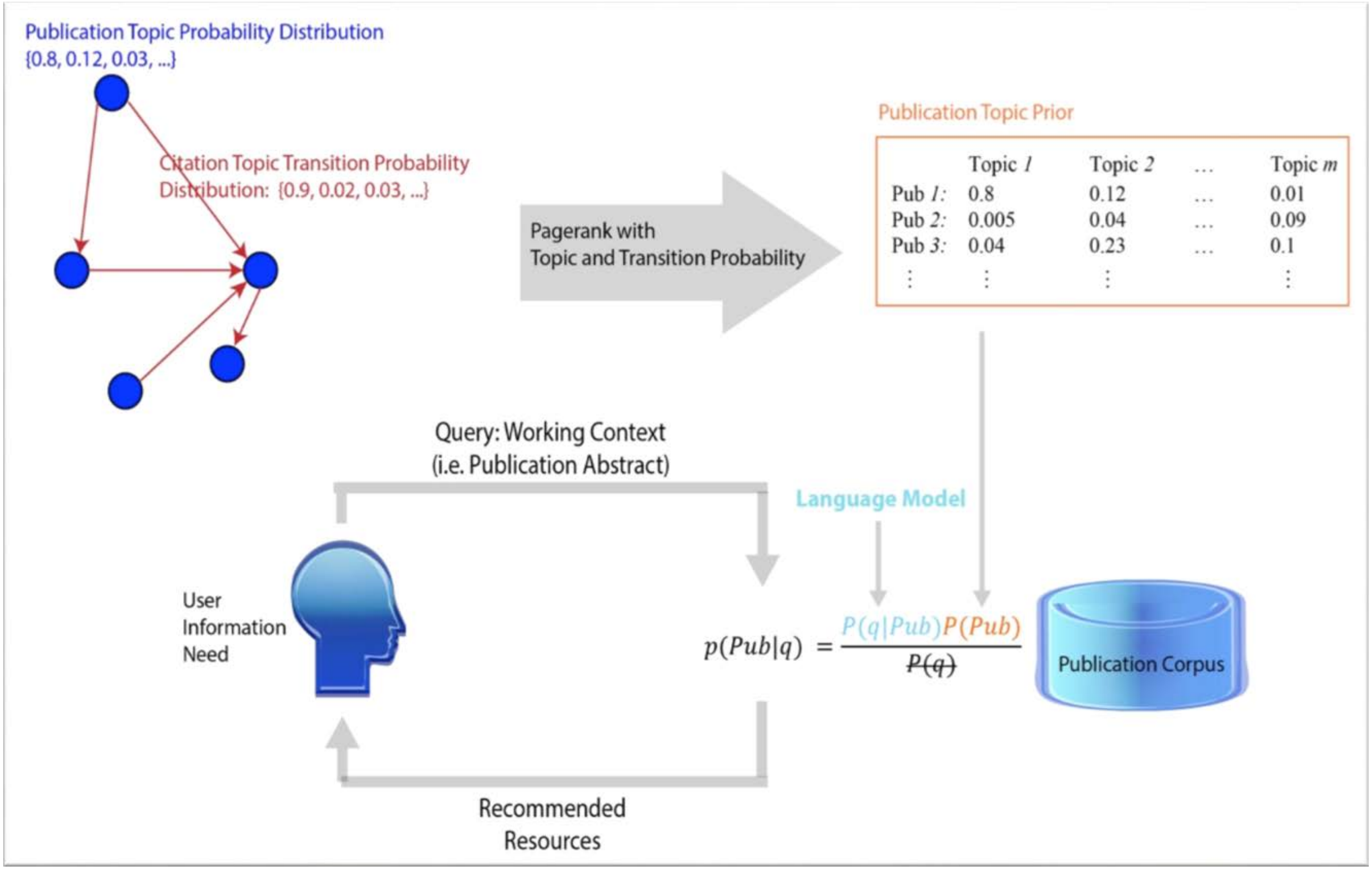


Given a paper abstract:

1. Word level match (language model)
2. Topic level match (KL-Divergence)
3. Topic importance

Liu et al. *Full-Text based Context-Rich Heterogeneous Network Mining Approach for Citation Recommendation*. JCDL, 2014.

Prior Art: A Global Recommendation Model



Liu et al. *Full-Text based Context-Rich Heterogeneous Network Mining Approach for Citation Recommendation*. JCDL, 2014.

Global Model: Limitations

- Global model:
  - all papers adopt same criterion and follow same behavioral pattern in citing other papers
    - e.g., equal importance between “content match” & “topic match” for every paper

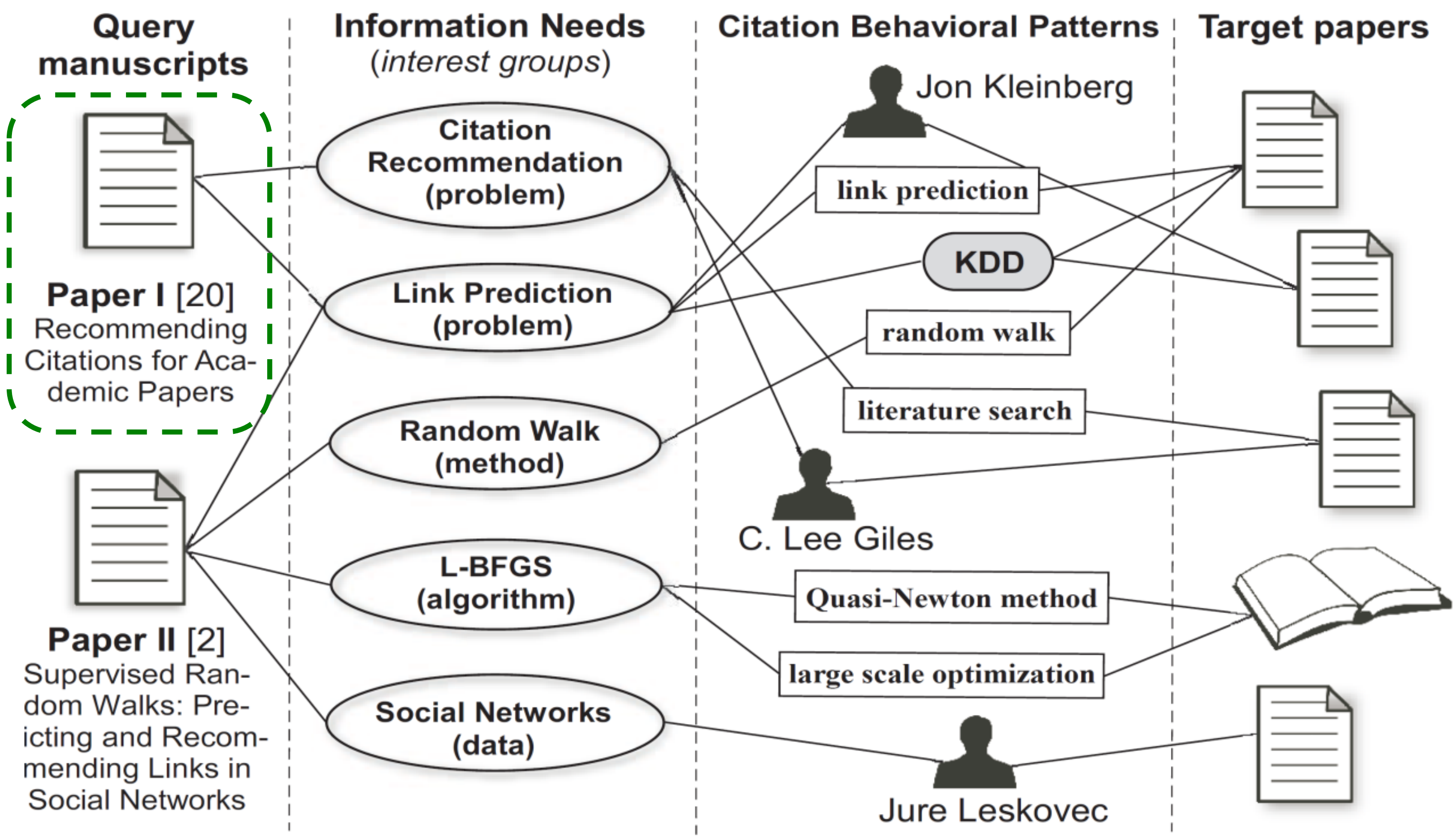
- Context-based [He et al., WWW’10; Huang et al., CIKM’12]
- Topical similarity-based [Nallapati et al., KDD’08; Tang et al., PAKDD’09]
- Structural similarity-based [Liben-Nowell et al., CIKM’03; Strohman et al., SIGIR’07]
- Hybrid methods [Bethard et al., CIKM’10; Yu et al., SDM’12]

From Global Model to Paper-Specific Model

- Global model:
  - all papers adopt same criterion and follow same behavioral pattern in citing other papers

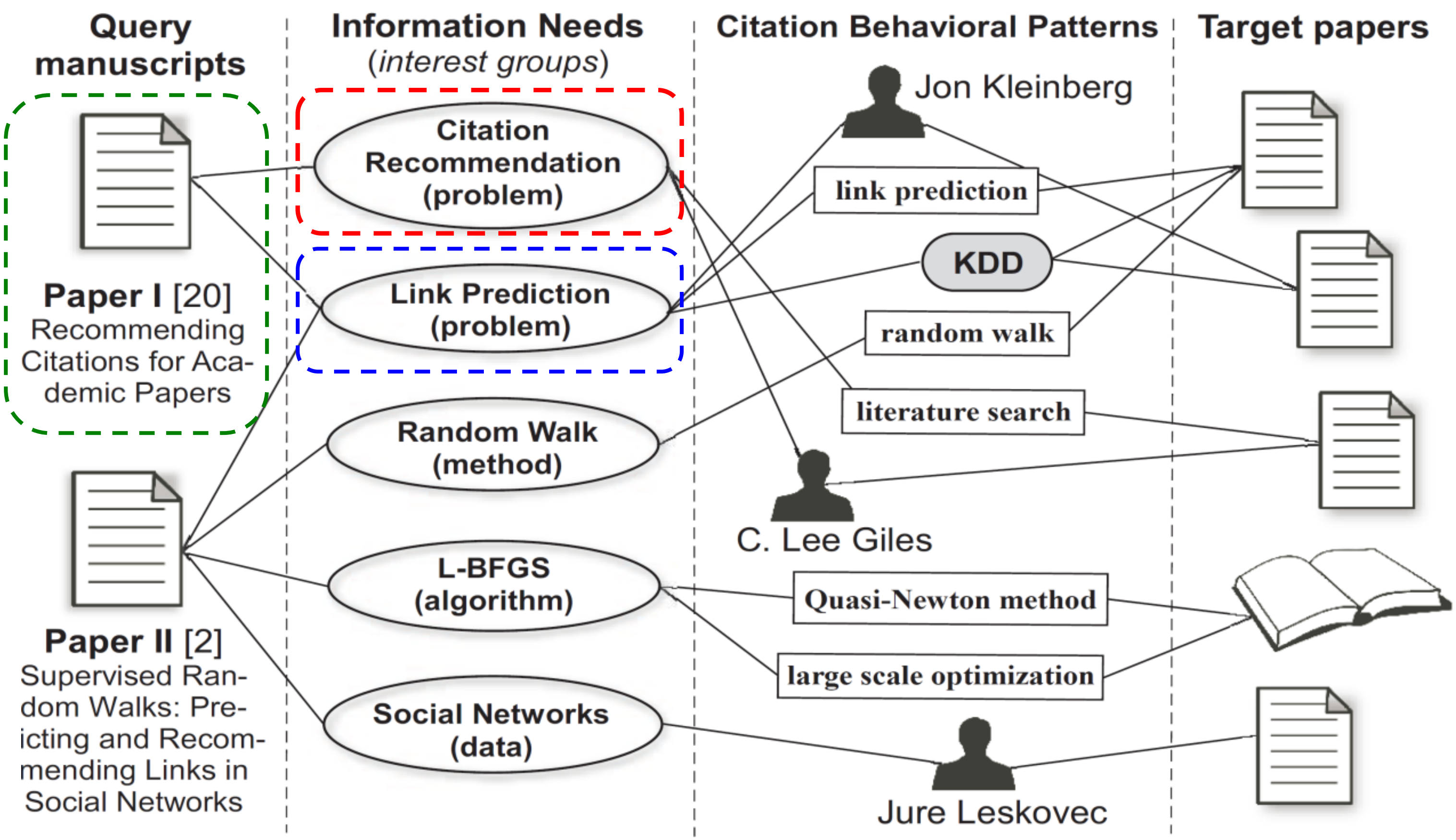
- Paper citations → different interests groups
  - Each group has its own behavioral pattern to identify references of interests

Distinctive Behavioral Pattern: Example



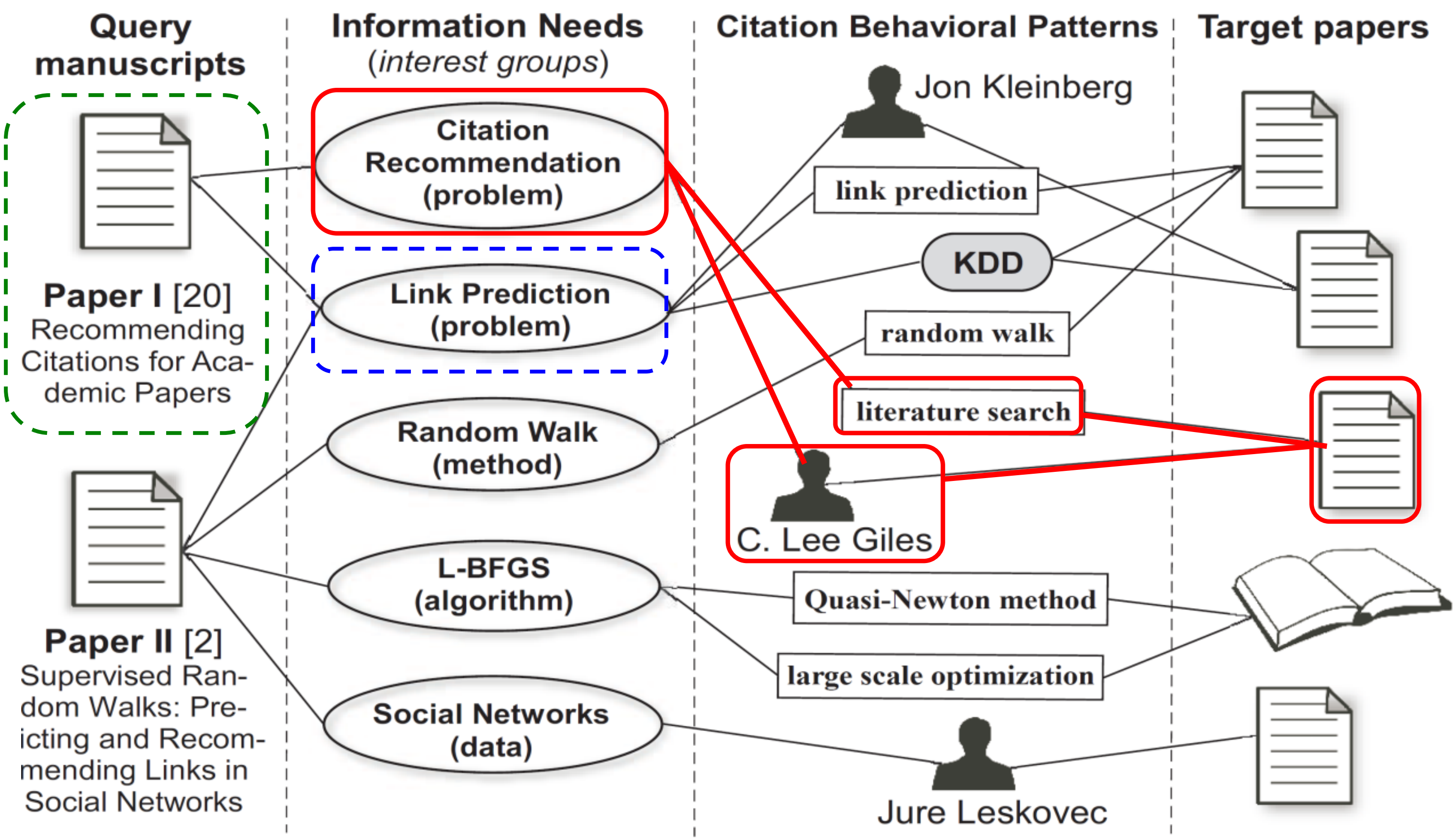
Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.

Distinctive Behavioral Pattern: Example



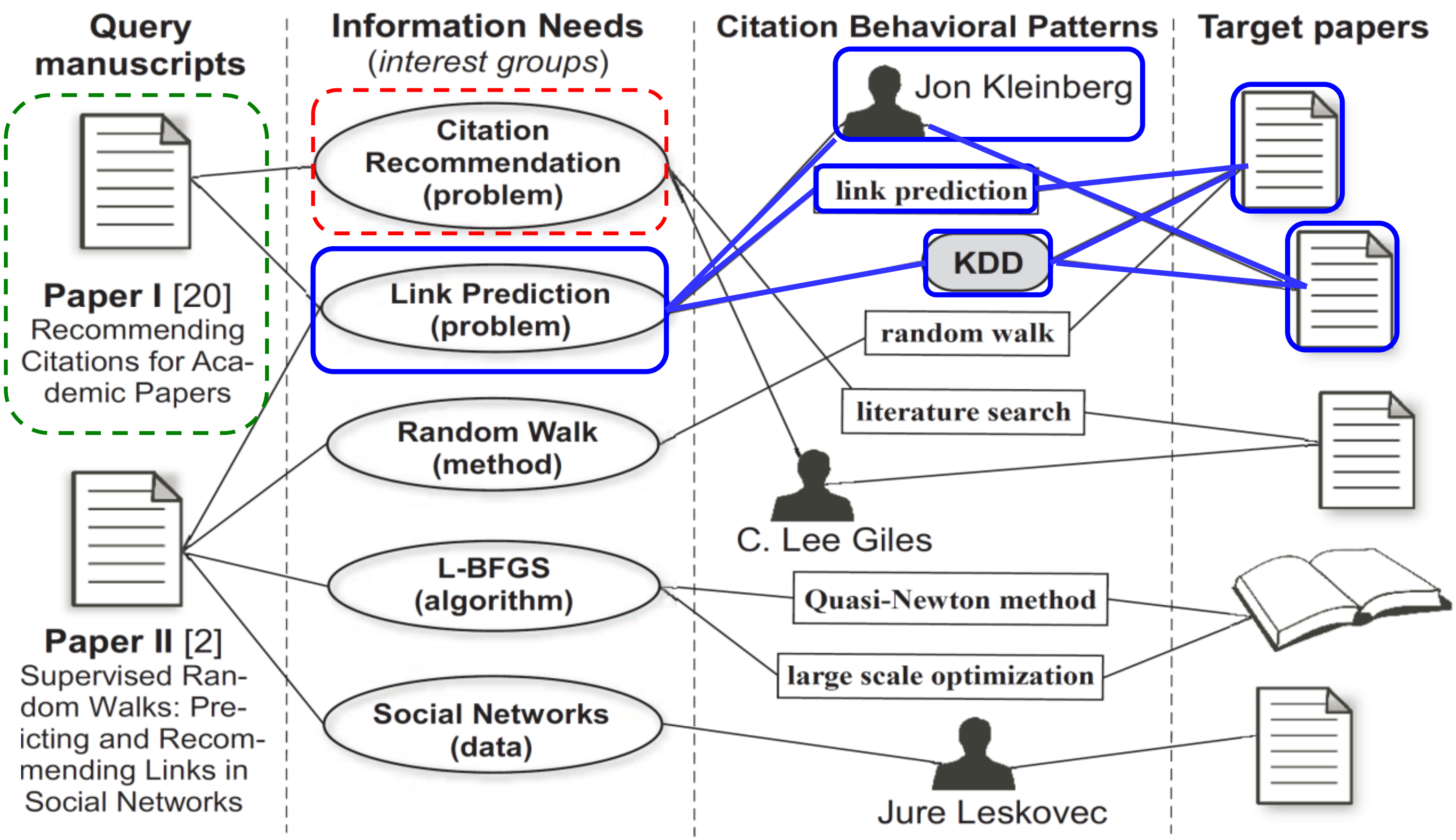
Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.

Distinctive Behavioral Pattern: Example



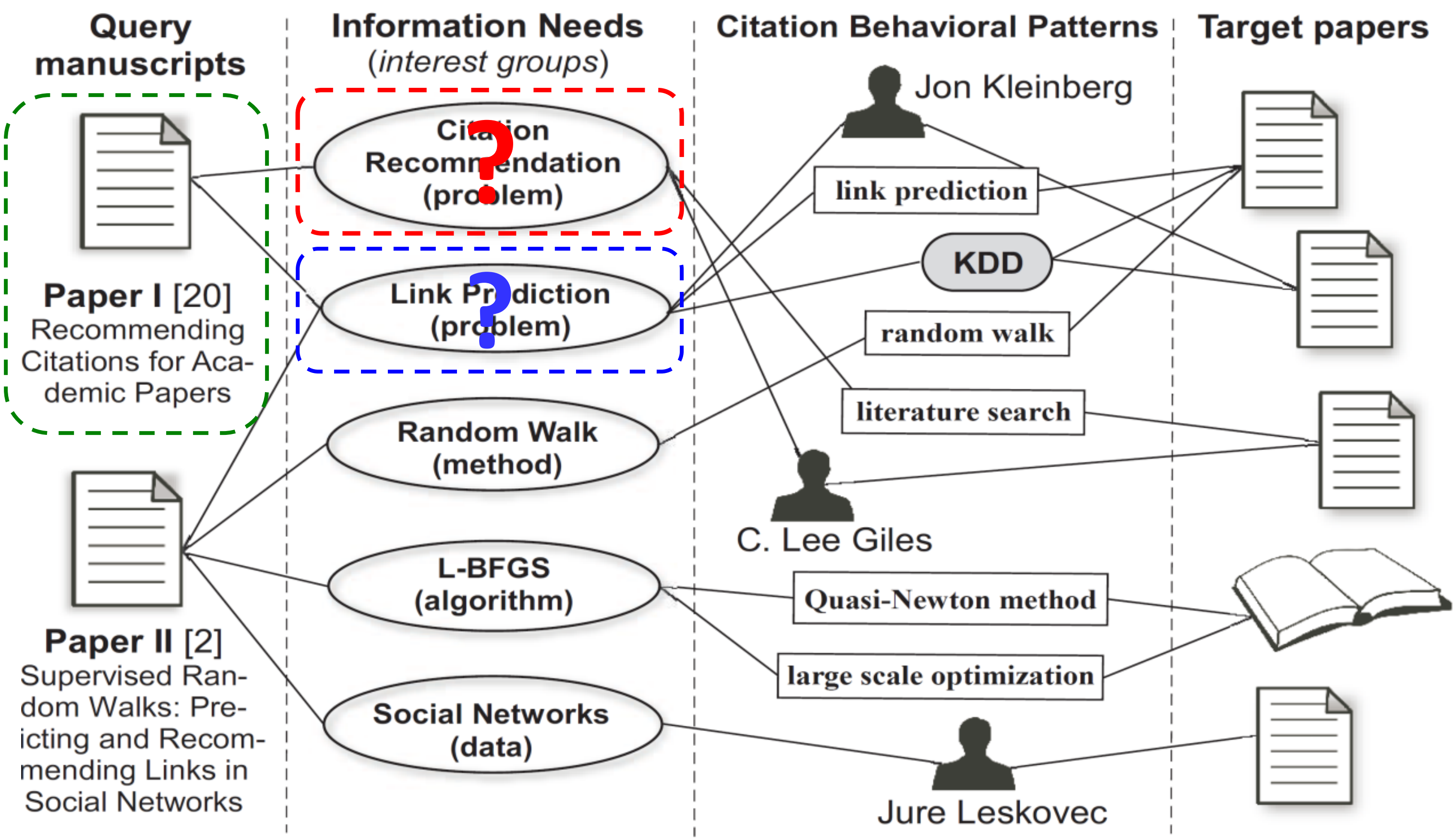
Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.

Distinctive Behavioral Pattern: Example



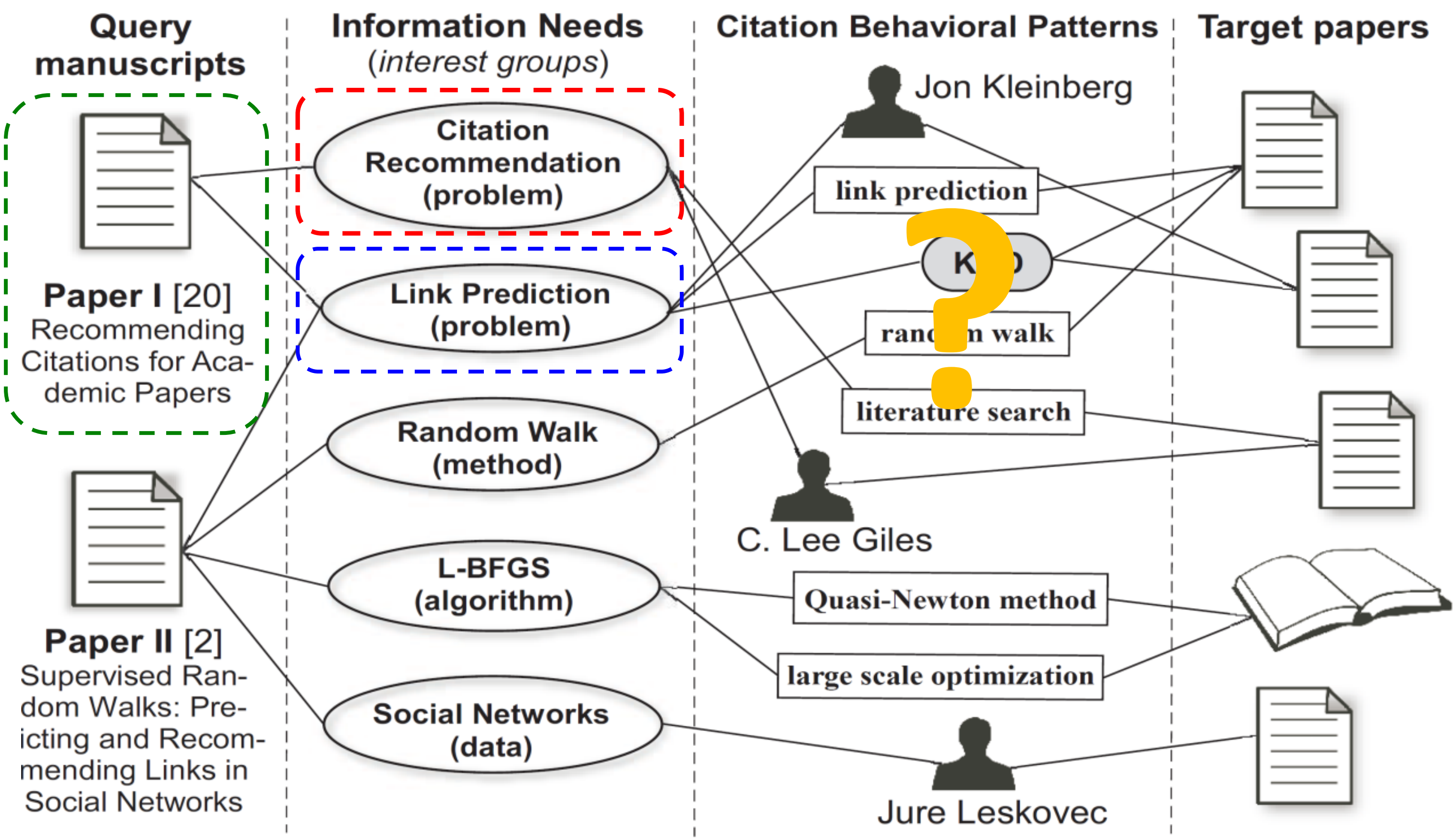
Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.

Distinctive Behavioral Pattern: Example



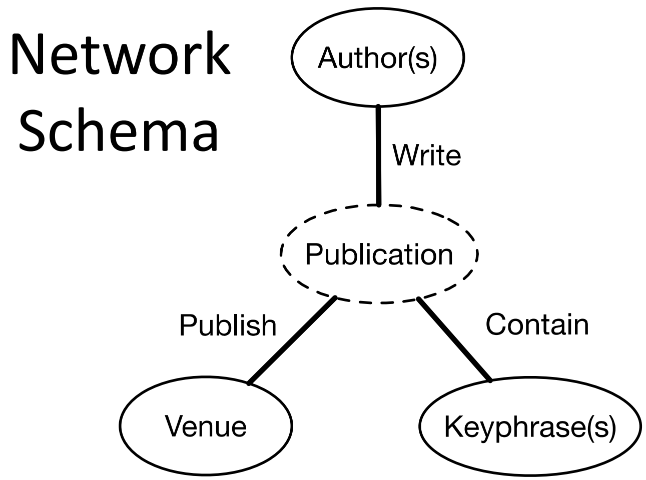
Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.

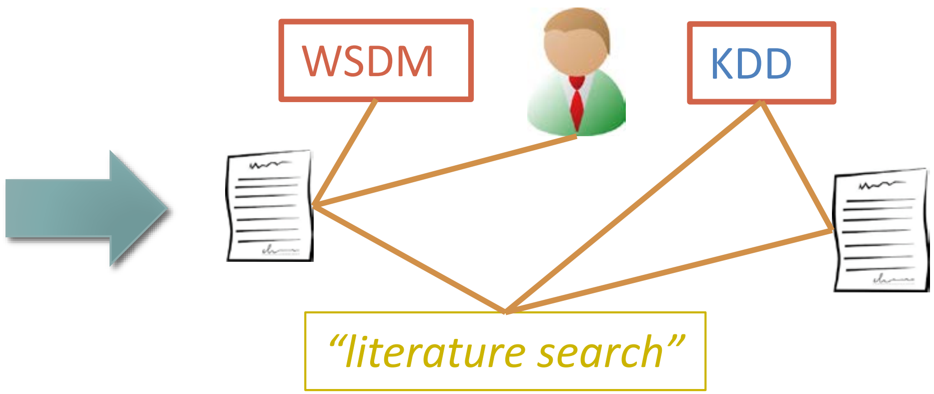
Distinctive Behavioral Pattern: Example



Each group follow distinct behavioral patterns and adopt different criterions in deciding relevance and authority of a candidate paper.

Heterogeneous Bibliographic Network





A unified graph representation for bibliographic dataset (papers and their attributes)

- Captures paper-paper relevance of different semantics
- Enables authority propagation between different types of objects

ClusCite: A Paper-specific Recommendation Model

Citations tend to be ***softly*** clustered into different ***interest groups***, based on the heterogeneous network structures

(Ren et al., KDD'15)

ClusCite: A Paper-specific Recommendation Model

Citations tend to be ***softly*** clustered into different ***interest groups***, based on the heterogeneous network structures

learn *distinct* models on finding relevant papers and judging authority of papers

Derive group membership for query manuscript

Paper-specific recommendation model: by integrating learned models of its related interest groups

Phrase I: Joint Learning (offline)

Phrase II: Recommendation (online)

(Ren et al., KDD'15)

Proposed Model: Overview

How likely a query manuscript ***q*** will cite a candidate paper ***p*** (suppose **K** interest groups):

$$s(q,p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ r^{(k)}(q,p) + f_{\mathcal{P}}^{(k)}(p) \right\}$$

Proposed Model: Overview

How likely a query manuscript ***q*** will cite a candidate paper ***p*** (suppose **K** interest groups):

$$s(q,p) = \sum_{k=1}^{K} \underbrace{\boxed{\theta_q^{(k)}}}_{\text{query's group membership}} \cdot \underbrace{\boxed{\left\{ r^{(k)}(q,p) + f_{\mathcal{P}}^{(k)}(p) \right\}}}_{\text{relative citation score (how likely } q \text{ will cite } p\text{) within each group}}$$

Proposed Model: Overview

How likely a query manuscript ***q*** will cite a candidate paper ***p*** (suppose **K** interest groups):

$$s(q,p) = \sum_{k=1}^{K} \underbrace{\boxed{\theta_q^{(k)}}}_{\text{query's group membership}} \cdot \left\{ \underbrace{\boxed{r^{(k)}(q,p)}}_{\text{paper relative relevance (query-candidate paper)}} + \underbrace{\boxed{f_{\mathcal{P}}^{(k)}(p)}}_{\text{paper relative authority (candidate paper)}} \right\}$$

Proposed Model: Overview

How likely a query manuscript ***q*** will cite a candidate paper ***p*** (suppose **K** interest groups):

$$s(q,p) = \sum_{k=1}^{K} \boxed{\theta_q^{(k)}} \cdot \left\{ \boxed{r^{(k)}(q,p)} + \boxed{f_{\mathcal{P}}^{(k)}(p)} \right\}$$

query's group membership

paper relative relevance (query-candidate paper)

paper relative authority (candidate paper)

It is desirable to suggest papers that have *high* relevance and authority scores across *multiple* related interest groups of the query manuscript

Proposed Model: Group Membership

$$s(q,p) = \sum_{k=1}^{K} \boxed{\theta_q^{(k)}} \cdot \left\{ r^{(k)}(q,p) + f_{\mathcal{P}}^{(k)}(p) \right\}$$

- **Learn each query's group membership: scalability & generalizability**
- **Leverage the group memberships of related attribute objects to *approximate* query's group membership**

$$\theta_q^{(k)} = \sum_{\mathcal{X} \in \{\mathcal{A},\mathcal{V},\mathcal{T}\}} \sum_{x \in N_{\mathcal{X}}(q)} \frac{\theta_x^{(k)}}{|N_{\mathcal{X}}(q)|}$$

Different types of attribute objects (***X*** = authors/venues/terms)

Query's related (linked) objects of type-***X***

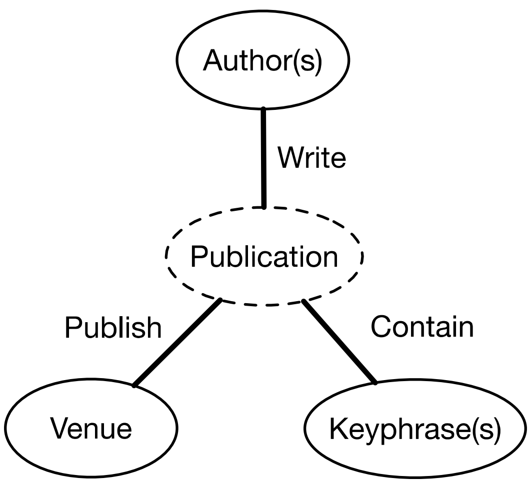
Attribute object's group membership (*to learn*)

Proposed Model: Paper Relevance

$$s(q,p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ \boxed{r^{(k)}(q,p)} + f_{\mathcal{P}}^{(k)}(p) \right\}$$

$$r^{(k)}(q,p) = \sum_{l=1}^{L} w_k^{(l)} \cdot \boxed{\phi^{(l)}(q,p)}$$

meta path-based relevance score (*l*-th feature)



Network schema

Table 1: Meta paths with different semantics.

Meta path	Semantic meaning of the relation
$P-A-P$	p_i and p_j share same author(s)
$P-T-P$	p_i and p_j contain same term(s)
$P-V-P$	p_i and p_j are in the same venue
$P-T-P \rightarrow P$	p_i share term(s) with the paper(s) that cite p_j
$P-A-P \leftarrow P$	p_i share the same author(s) with the paper(s) cited by p_j

Proposed Model: Paper Relevance

Relevance features play different roles in different interest groups

$$s(q,p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ \boxed{r^{(k)}(q,p)} + f_{\mathcal{P}}^{(k)}(p) \right\}$$

$$r^{(k)}(q,p) = \sum_{l=1}^{L} \boxed{w_k^{(l)}} \cdot \phi^{(l)}(q,p)$$

weights on different meta path-based features

Table 2: Learned weights on seven different meta paths for four mined interest groups ($K = 40$).

Meta path	Group 1	Group 2	Group 3	Group 4
$P-V-P$	0.0024	0.0113	0.0158	0.3076*
$P-A-P$	0.0054	0.0006	0.0192	0.1243
$P-A-P \rightarrow P$	0.6133**	0.2159*	0.2254	0.0213
$P-T-P$	0.1227	0.0947	0.1579	0.1095
$P-T-P \rightarrow P$	0.0442	0.5448**	0.3250*	0.0231
$P-T-P \leftarrow P$	0.1938*	0.0870	0.3578**	0.2409**

Proposed Model: Object Relative Authority

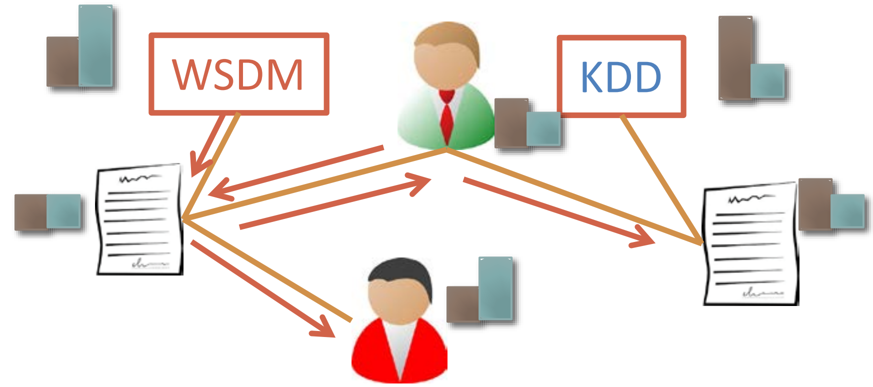
$$s(q,p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ r^{(k)}(q,p) + \boxed{f_{\mathcal{P}}^{(k)}(p)} \right\}$$

Paper relative authority: A paper may have quite different visibility/authority among different groups, even it is overall highly cited

Relative authority propagation over the network

Relative authority in Group A

Relative authority in Group B



Proposed Model: Object Relative Authority

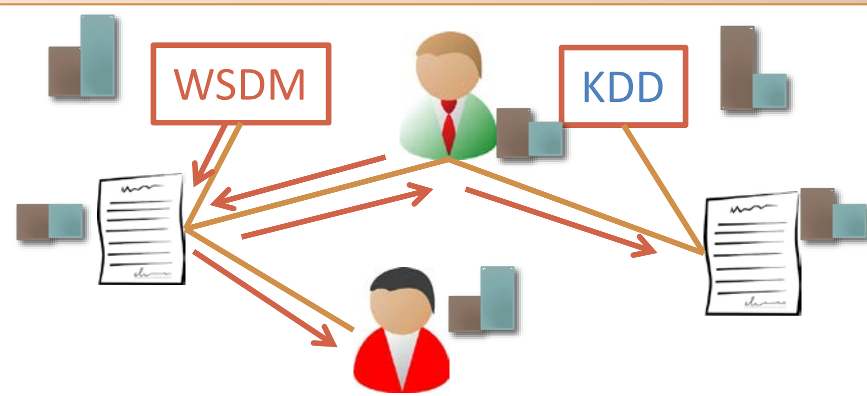
$$s(q,p) = \sum_{k=1}^{K} \theta_q^{(k)} \cdot \left\{ r^{(k)}(q,p) + \boxed{f_{\mathcal{P}}^{(k)}(p)} \right\}$$

Paper relative authority: A paper may have quite different visibility/authority among different groups, even it is overall highly cited

Table 6: Top-5 authority venues and authors from two example interest groups derived by ClusCite.

Rank	Venue		Author	
	Group I (database and information system)			
1	VLDB	0.0763	Hector Garcia-Molina	0.0202
2	SIGMOD	0.0653	Christos Faloutsos	0.0187
3	TKDE	0.0651	Elisa Bertino	0.0180
4	CIKM	0.0590	Dan Suciu	0.0179
5	SIGKDD	0.0488	H. V. Jagadish	0.0178
	Group II (computer vision and multimedia)			
1	TPAMI	0.0733	Richard Szeliski	0.0139
2	ACM MM	0.0533	Jitendra Malik	0.0122
3	ICCV	0.0403	Luc Van Gool	0.0121
4	CVPR	0.0401	Andrew Blake	0.0117
5	ECCV	0.0393	Alex Pentland	0.0114

Relative authority propagation over the network



Model Learning: Joint Optimization

A joint optimization problem:

$$\min_{\mathbf{P},\mathbf{W},\mathbf{F}_{\mathcal{P}},\mathbf{F}_{\mathcal{A}},\mathbf{F}_{\mathcal{V}}} \frac{1}{2}\mathcal{L} + \mathcal{R} + \frac{c_p}{2}\|\mathbf{P}\|_F^2 + \frac{c_w}{2}\|\mathbf{W}\|_F^2$$

$$\text{s.t.} \quad \mathbf{P} \geq 0; \quad \mathbf{W} \geq 0.$$

Weighted model prediction error

Graph regularization for encoding authority propagation

$$\mathcal{L} = \sum_{i,j=1}^{n} M_{ij}\Big(Y_{ij} - \sum_{k=1}^{K}\sum_{l=1}^{L}\theta_{p_i}^{(k)} w_k^{(l)} S_{jl}^{(i)} - \sum_{k=1}^{K}\theta_{p_i}^{(k)} F_{\mathcal{P},kj}\Big)^2$$
$$= \sum_{i=1}^{n}\Big\|\mathbf{M}_i \odot \Big(\mathbf{Y}_i - \mathbf{R}_i\mathbf{P}(\mathbf{W}\mathbf{S}^{(i)T} + \mathbf{F}_{\mathcal{P}})\Big)\Big\|_2^2.$$

$$\mathcal{R} = \frac{\lambda_{\mathcal{A}}}{2}\sum_{i=1}^{n}\sum_{j=1}^{|\mathcal{A}|} R_{ij}^{(\mathcal{A})}\Big\|\frac{\mathbf{F}_{\mathcal{P},i}}{D_{ii}^{(\mathcal{PA})}} - \frac{\mathbf{F}_{\mathcal{A},j}}{D_{jj}^{(\mathcal{AP})}}\Big\|_2^2 + \frac{\lambda_{\mathcal{V}}}{2}\sum_{i=1}^{n}\sum_{j=1}^{|\mathcal{V}|} R_{ij}^{(\mathcal{V})}\Big\|\frac{\mathbf{F}_{\mathcal{P},i}}{D_{ii}^{(\mathcal{PV})}} - \frac{\mathbf{F}_{\mathcal{V},j}}{D_{jj}^{(\mathcal{VP})}}\Big\|_2^2$$

Algorithm: alternating minimization (w.r.t. each variable)

Experimental Results

- **Datasets**
  - DBLP: 137k papers; ~2.3M relationships; Avg # citations/paper: 5.16
  - PubMed: 100k papers; ~3.6M relationships; Avg # citations/paper: 17.55

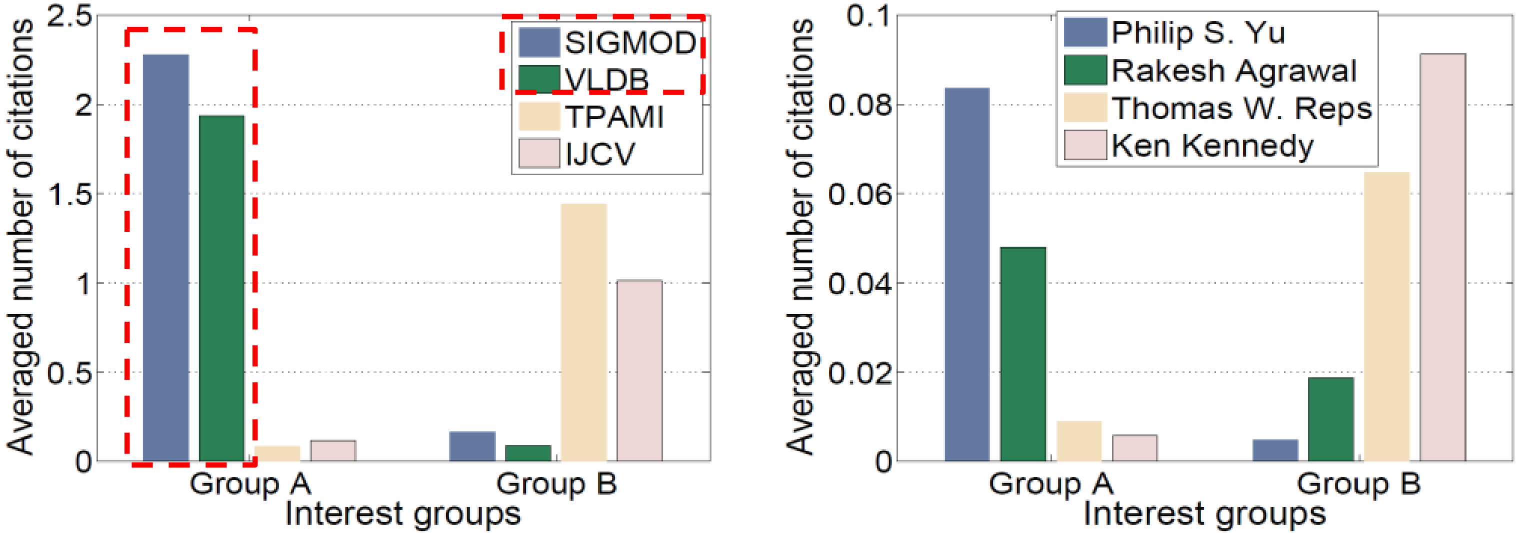
Experiment: Case Study I

- Example output of relative authority ranking

Rank	Venue		Author	
	Group I (database and information system)			
1	VLDB	0.0763	Hector Garcia-Molina	0.0202
2	SIGMOD	0.0653	Christos Faloutsos	0.0187
3	TKDE	0.0651	Elisa Bertino	0.0180
4	CIKM	0.0590	Dan Suciu	0.0179
5	SIGKDD	0.0488	H. V. Jagadish	0.0178
	Group II (computer vision and multimedia)			
1	TPAMI	0.0733	Richard Szeliski	0.0139
2	ACM MM	0.0533	Jitendra Malik	0.0122
3	ICCV	0.0403	Luc Van Gool	0.0121
4	CVPR	0.0401	Andrew Blake	0.0117
5	ECCV	0.0393	Alex Pentland	0.0114

Experiment: Case Study II

- Case study on citation behavioral patterns



(a) Citations on venues (b) Citations on authors

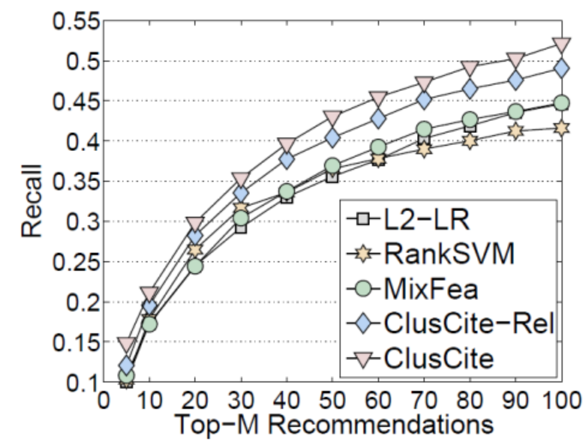
Each paper is assigned to the group with highest group membership score

Experiment: Comparing with State-of-the-Art Methods

❑ Performance Comparisons

– 17.68% improvement in Recall@50; 9.57% in MRR, on DBLP

Method	DBLP				
	P@10	P@20	R@20	R@50	MRR
BM25	0.1260	0.0902	0.1431	0.2146	0.4107
PopRank	0.0112	0.0098	0.0155	0.0308	0.0451
TopicSim	0.0328	0.0273	0.0432	0.0825	0.1161
Link-PLSA-LDA	0.1023	0.0893	0.1295	0.1823	0.3748
L2-LR	0.2274	0.1677	0.2471	0.3547	0.4866
RankSVM	0.2372	0.1799	0.2733	0.3621	0.4989
MixFea	0.2261	0.1689	0.2473	0.3636	0.5002
ClusCite-Rel	0.2402	0.1872	0.2856	0.4015	0.5156
ClusCite	**0.2429**	**0.1958**	**0.2993**	**0.4279**	**0.5481**



(a) Recall on DBLP

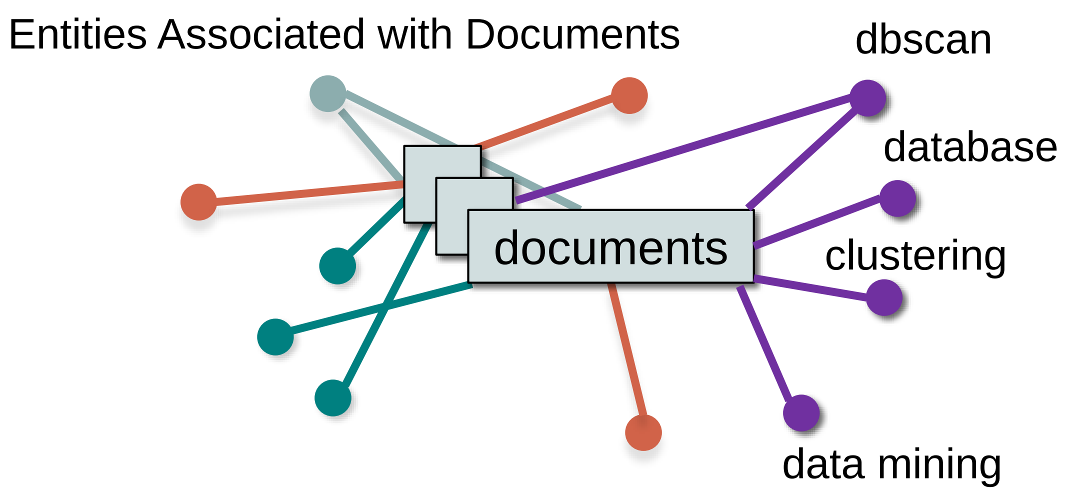
- BM25: content-based
- PopRank [WWW'05]: heterogeneous link-based authority
- TopicSim: topic-based similarity by LDA
- Link-PLSA-LDA [KDD'08]: topic and link relevance
- Meta-path based relevance:
  - L2-LR [SDM'12, WSDM'12]: logistics regression with L2 regularization
  - RankSVM [KDD'02]
- MixSim: relevance, topic distribution, PopRank scores, using RankSVM

Outline

- Background
- Content-based Recommendation: An Overview
- Recommendation in Text-Rich Information Network
- Recommendation in Networks Constructed from Text
- Summary

Text-Rich Information Network

- Nodes: phrases extracted from text
- Edges: relationship between phrase and document
  - How important is the phrase?
  - → TF-IDF weighting
  - → relationship strength (edge weight)



Network Construction from Text

- Can we construct the network from text?
  - First step: given nodes (phrases), can we learn the edge weights from data?

- Problem Statement
  - Joint learning of (1) recommendation model & (2) network edge weights for textual nodes

Example: Job Recommendation in LinkedIn

Given a LinkedIn member, we aim to find the jobs that he/she is most interested in.

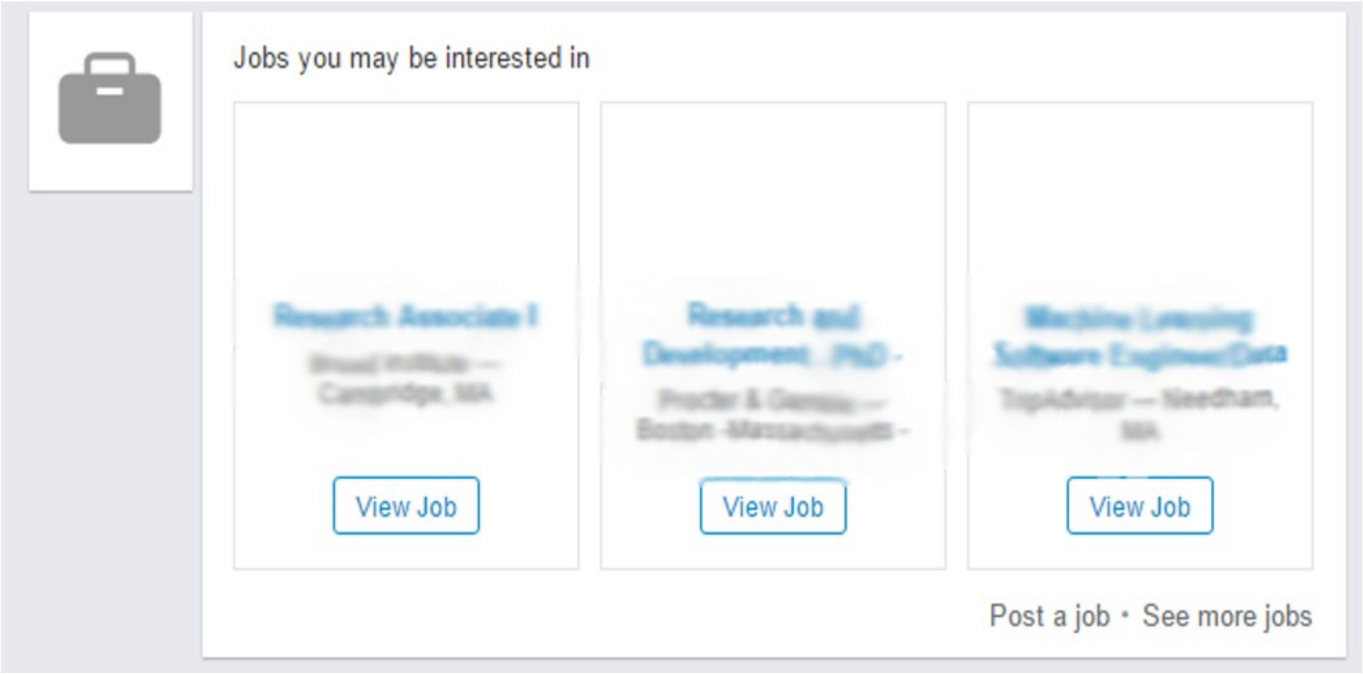


Figure :Job recommendation panel on www.linkedin.com

Example: Job Recommendation in LinkedIn

What information is available for members and jobs?

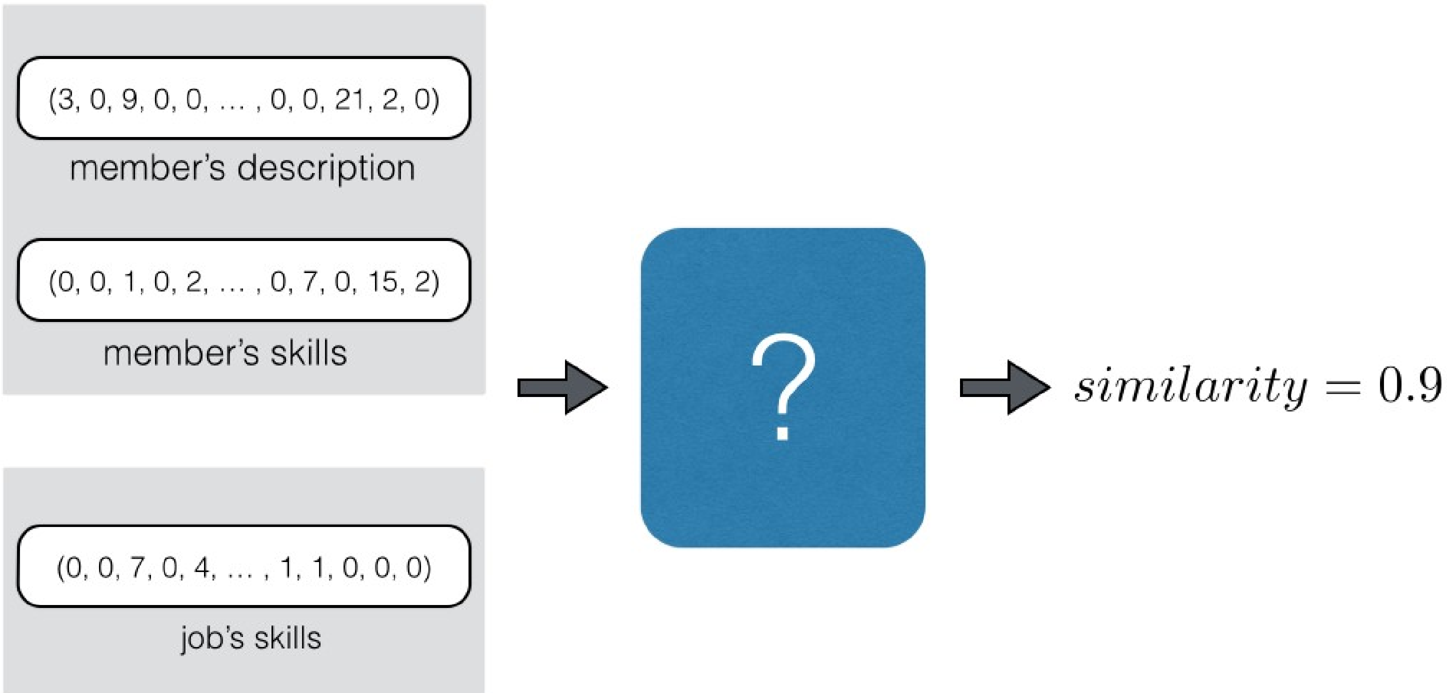
Content	Field
Recommender Systems for Talent Matching	Title
algorithms, python, machine learning, data mining, data analysis, linux, statistics	Skills
We use various data mining and information retrieval techniques to overcome the limitations of our sparse input data (member profiles and job descriptions)	Description

User

Field	Content
Title	Software Engineer Manager – Data Mining/Data Analysis/Machine Learning
Location	Bay Area, CA
Description	As a Software Engineer Data Manager, you will be responsible for leading a team of data scientists and relevance engineers that build and own recommendation algorithms, models, and systems.
Skills	java, C++, machine learning, data mining, information retrieval, big data, natural language processing, recommender systems

Job

A Simple Solution: TF-IDF Weighting



- For each (member field *s* , job field *t*), calculate the similarity score between two feature vectors.
- Aggregate the scores of all field pairs (*s*, *t*).

Issue: Lower idf Terms Can be Predictive

- High idf term: government
- Low idf term: machine learning

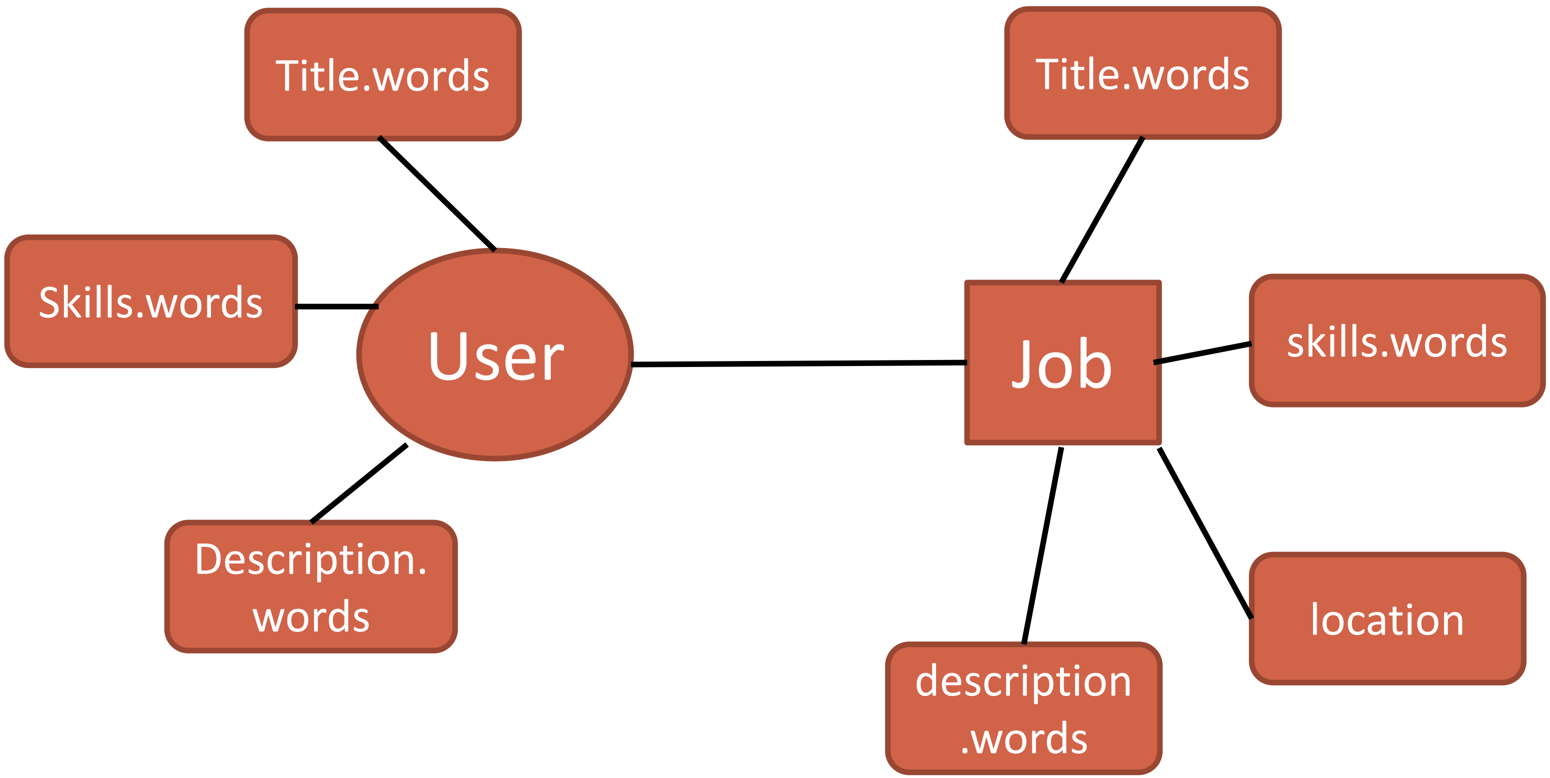
Member (description)	Recommended Jobs (description)
Member I have enrolled in a project which provides users with a visualization of government financial statistics using machine learning techniques ...	**Job 1** We are a managed services provider and we support many projects with government agencies and non-profit organizations. **Job 2** You will apply machine learning algorithms to analyze large data sets ...

Issue: Lower idf Terms Can be Predictive

- High idf term: government
- Low idf term: machine learning

- Limitations:
  - The feature for each word is defined by heuristic, not necessarily reasonable
  - Each field pair contributes equally

The Network View



Solution

- Learn a better representation for words
  - In particular, learn the optimal global term weights for each user text field and item text field
  - e.g., importance of “machine learning" in job skills

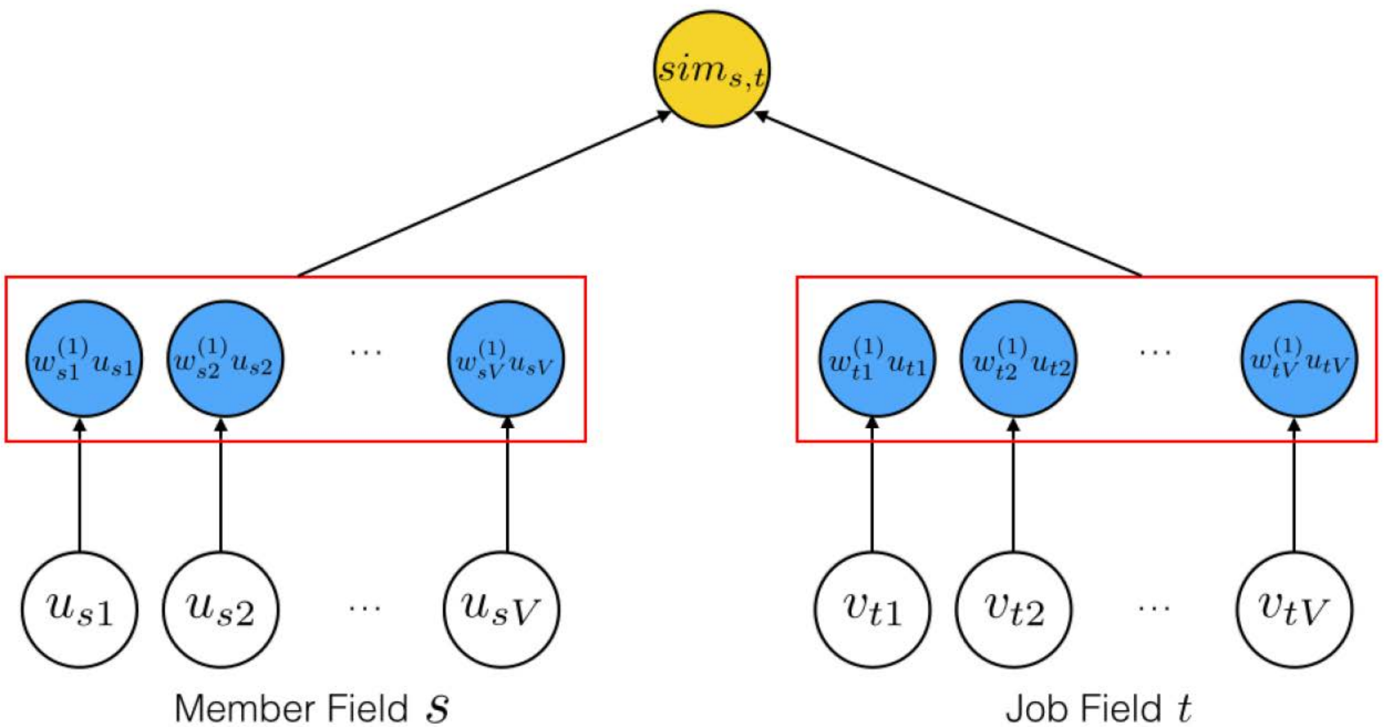
- Learn the weights of multiple content matching features between user and item profiles (field pairs)
  - e.g., user skills vs. job skills, user titles vs. job skills

A Two-layer Score Function Model

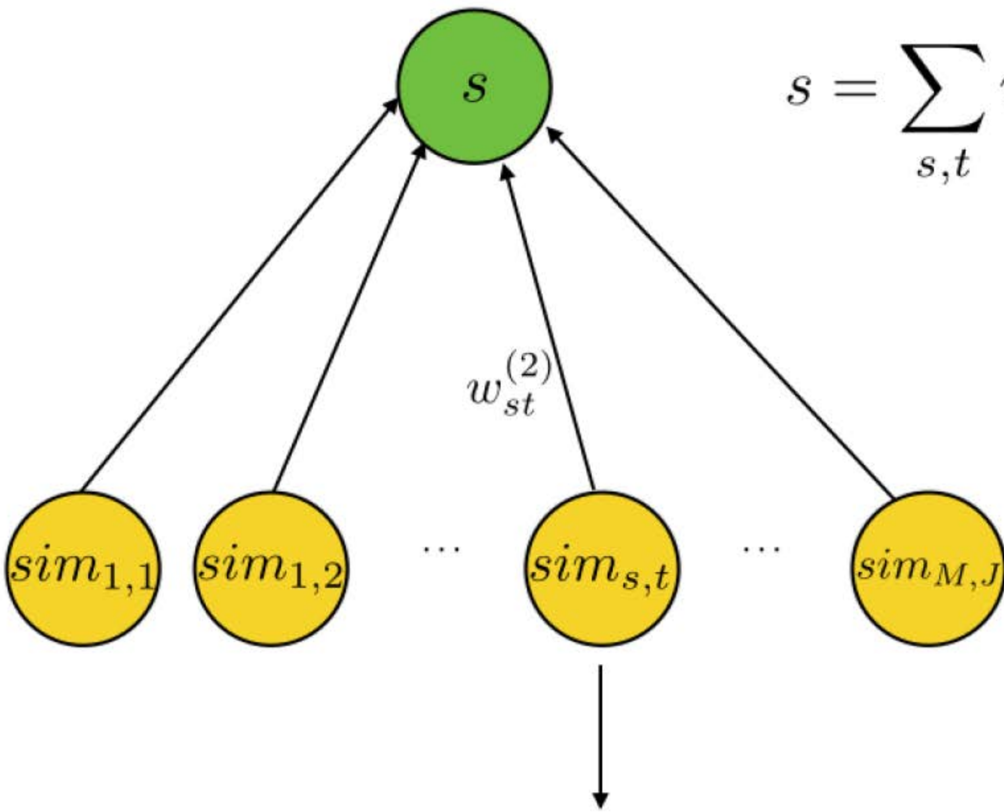
- First layer:
  - Map each original word feature into a scaled version
  - Calculate the cosine similar for each filed pair based on the weighted word feature
- Second layer:
  - Take the cosine similarity for each field pair as input, and take a weighted linear combination of these inputs

First Layer

- V: the size of the vocabulary



Second Layer



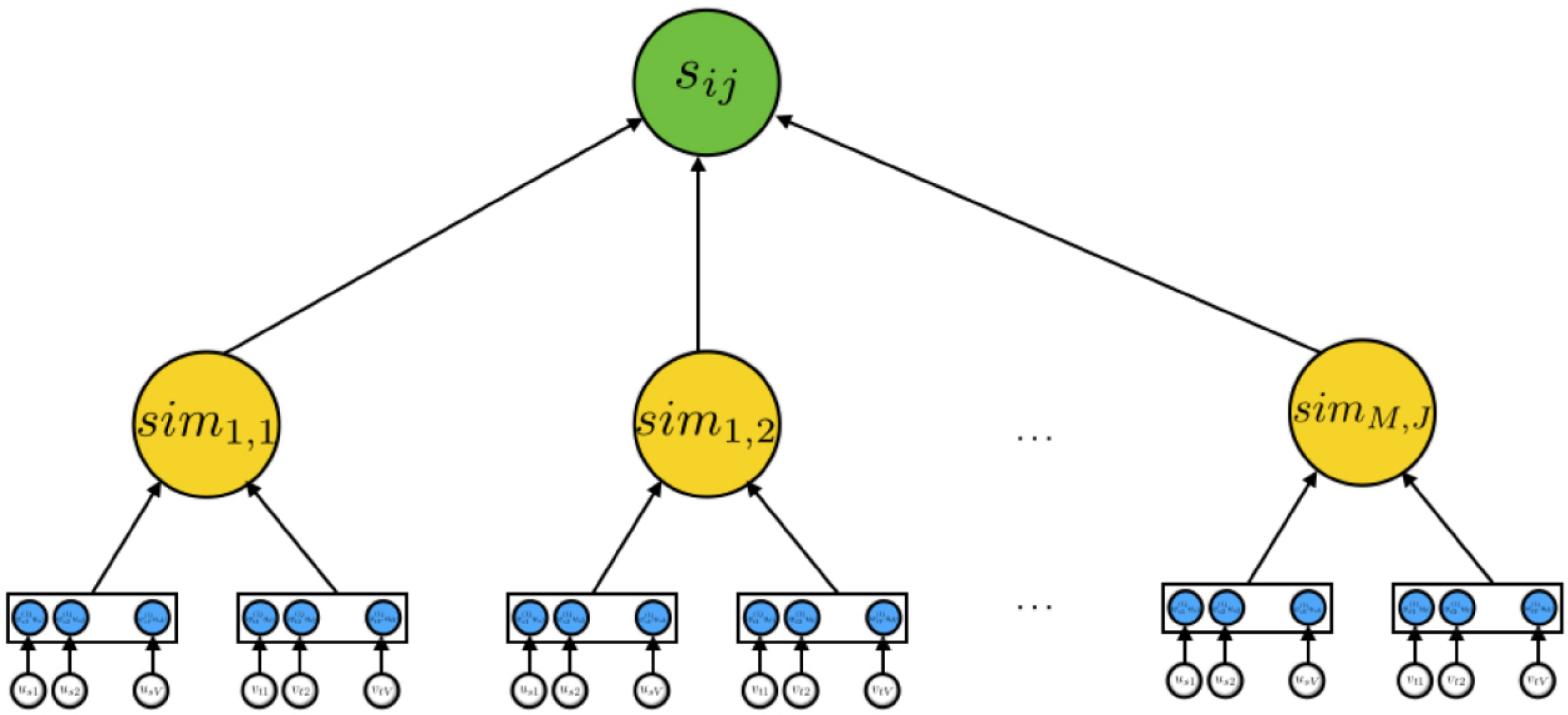
$$s = \sum_{s,t} w_{st}^{(2)} sim_{s,t} + w_0^{(2)}$$

Similarity between member field s and job field t

The Unified Model: Multi-Layer Regression Model (MLRM)

Objective: minimize the logit loss of all (member i, job j) pairs.

$$L(W^{(1)}, W^{(2)}) = \sum_{i,j} \log(1 + e^{-y_{ij} s_{ij}})$$



Experiments

- Data:
  - LinkedIn data
  - 490K unique terms and 75 fields in total
  - 3.1M (member, job) pairs

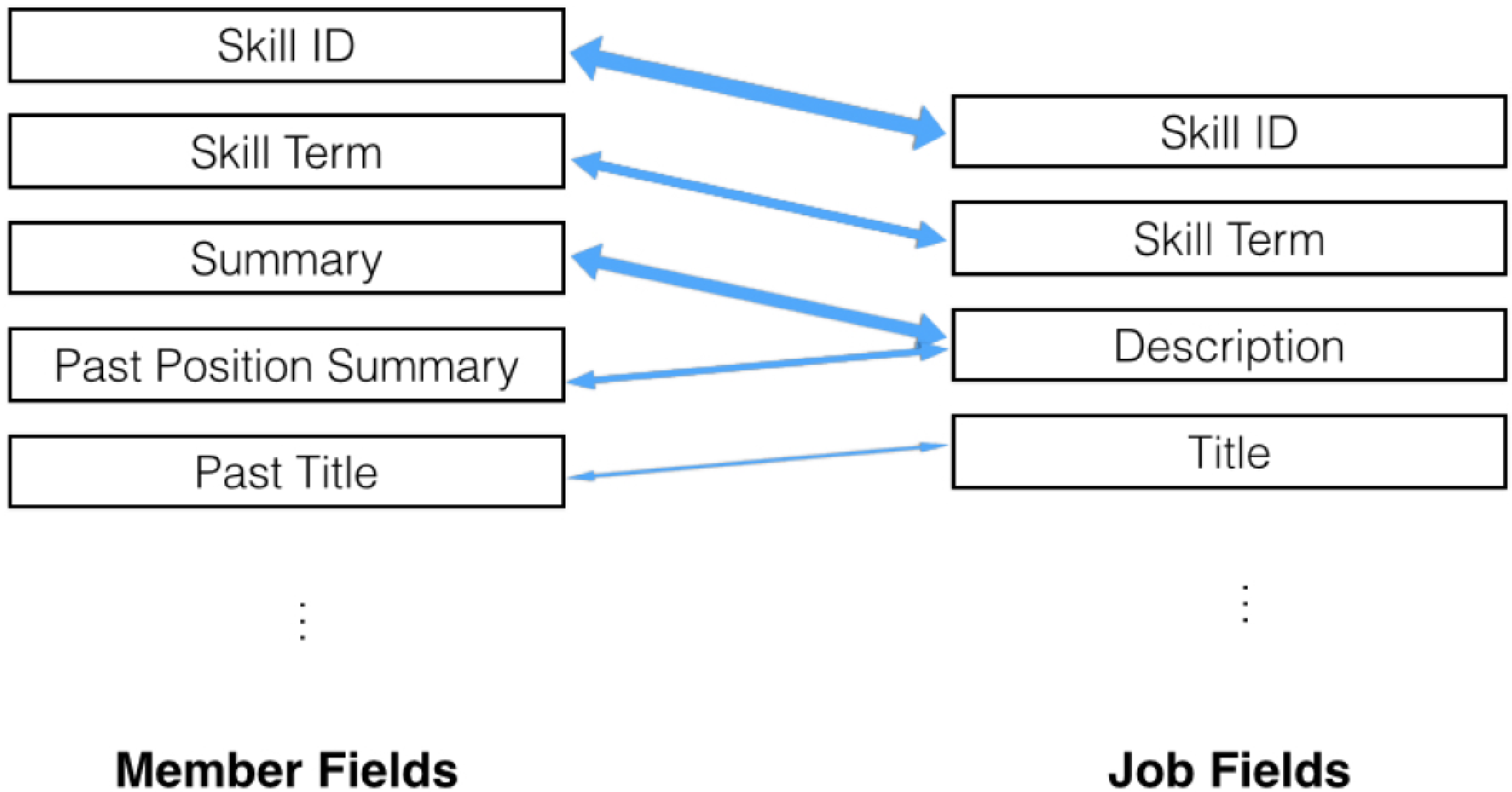
Positive	Feedback Negative	Random Negative
50%	25%	25%

  - 90% as training, and 10% held out as testing

Case Study: Top Terms in Job Skills

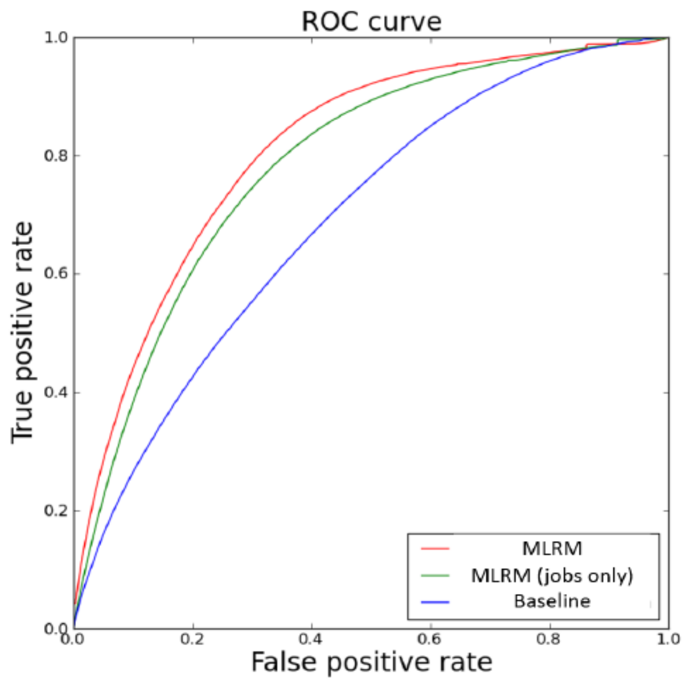
Case Study: Top Terms in Member Skills

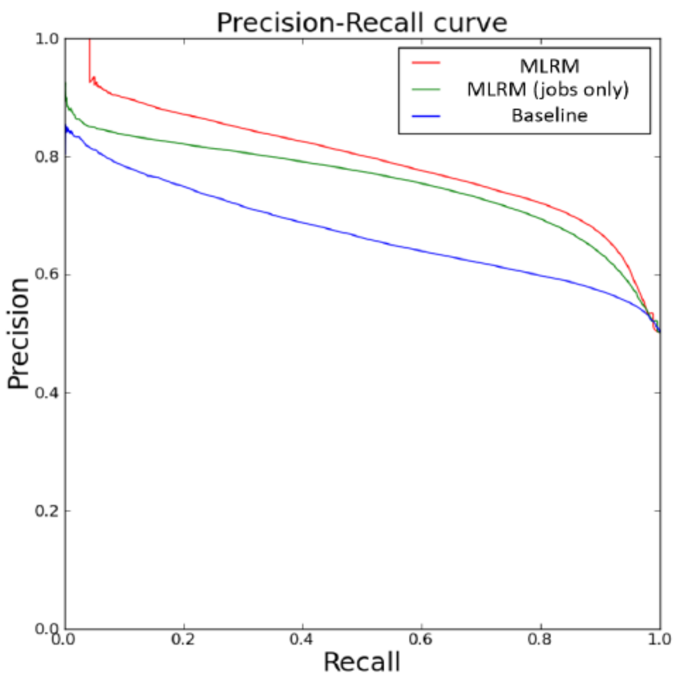
Case Study: Most Important Member-Job Field Pairs



AUC and AUPRC

Method	AUC	AUPRC
Baseline (tf-idf as feature)	0.692	0.671
Multi-layer Logistic Regression Model	**0.811 (+17.2%)**	**0.793 (+18.2%)**
Multi-layer Logistic Regression Model (jobs only)	0.792 (+14.5%)	0.771 (+14.9%)





Outline

- Background
- Content-based Recommendation: An Overview
- Recommendation in Text-Rich Information Network
- Recommendation in Networks Constructed from Text
- Summary

Summary

- Background
  - Textual info in recommendation
  - Challenge: unstructured data to structures
- Content-based Recommendation: An Overview
- Recommendation in Text-Rich Information Network
- Recommendation in Networks Constructed from Text

Summary

- Background
- Content-based Recommendation: An Overview
  - Basic idea, pros & cons
  - Major components, item/user representations
- Recommendation in Text-Rich Information Network
- Recommendation in Networks Constructed from Text

Summary

- Background
- Content-based Recommendation: An Overview
- Recommendation in Text-Rich Information Network
  - Global recommendation model
  - Paper-specific recommendation model
- Recommendation in Networks Constructed from Text

Summary

- Background
- Content-based Recommendation: An Overview
- Recommendation in Text-Rich Information Network
- Recommendation in Networks Constructed from Text
  - A joint term-weight learning framework for recommendation

Outline

- Part I: Introduction and Preliminaries
- Part II: Recommendation in Heterogeneous Information Networks
- Part III: Recommendation in a Text-Rich Setting
- Part IV: Recommendation with Spatio-Temporal Information
- Part V: Research Frontiers and Summary

Break

PART IV: SPATIAL-TEMPORAL RECOMMENDATION

Hongzhi Yin
School of ITEE
University of Queensland, Australia
h.yin1@uq.edu.edu

August 11, 2017

Outline

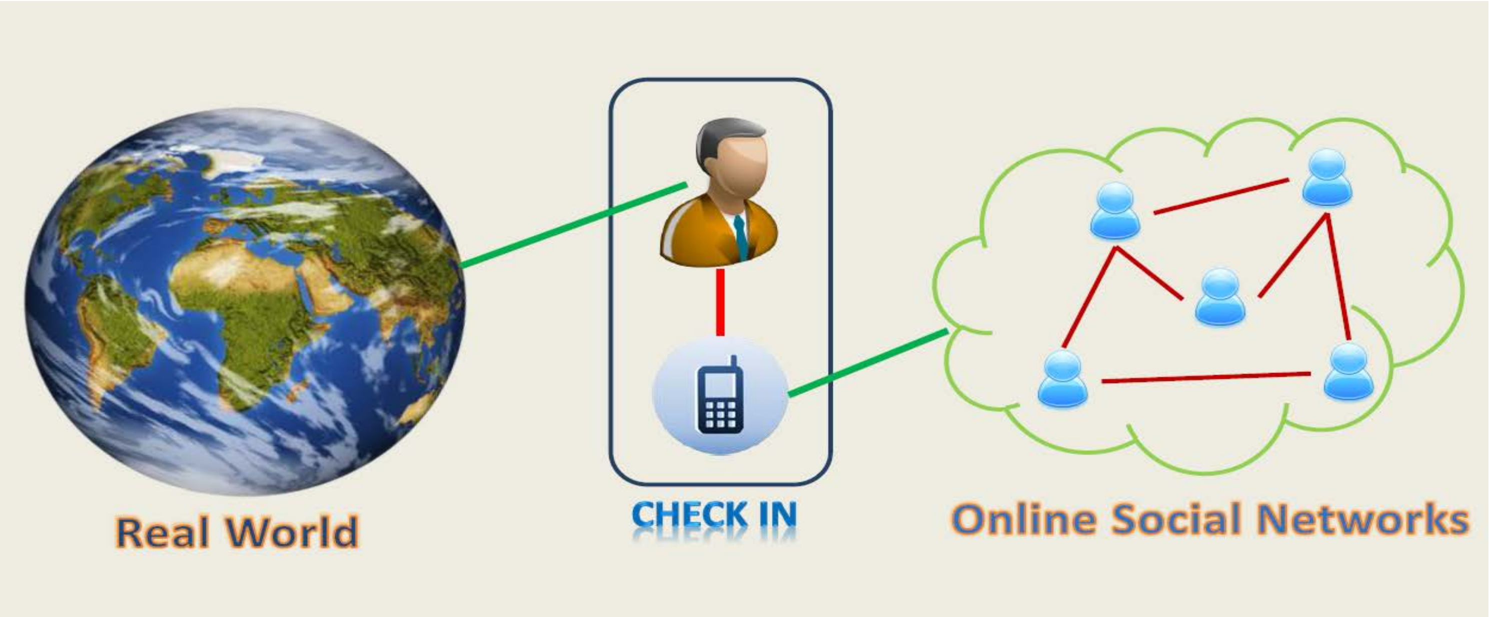
- Introduction
- Challenges of ST-Recommendation
- Effective Recommender Models

Geo-Social Networks

- Geo-social network (GSN) is very popular
  - Location-based Social Networks -LBSNs (e.g., Foursquare, Instagram, Yelp, Facebook Places, Google Places)
  - Event-based Social Networks - EBSNs (e.g., Meetup, Plancast, Douban-Event)
  - Traditional Social Networks enhanced by locations (e.g., Sina Weibo, Twitter and Wechat)

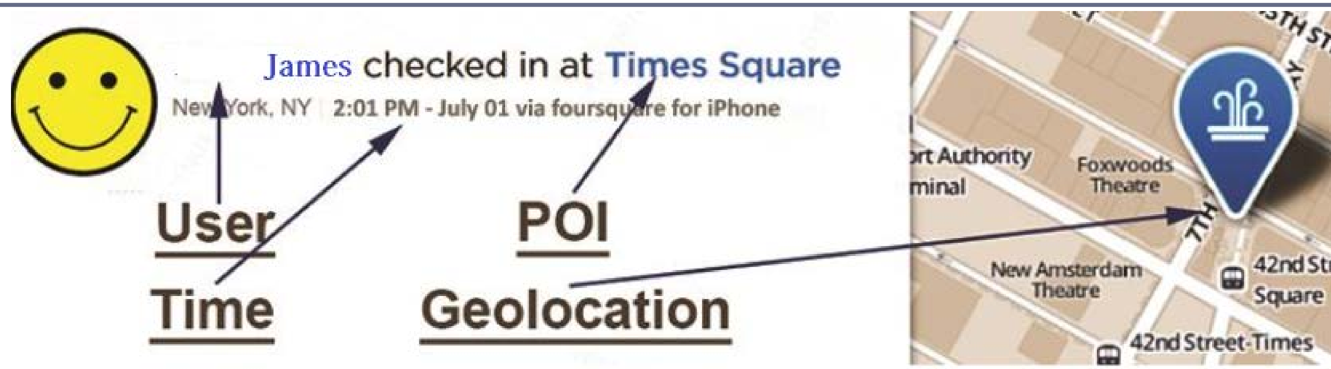
Check-in in Geo-Social Networks

- Users can post their physical locations or geo-tag information via "check-in" and share their visiting experiences with their friends in the social networks.
- Check-in bridges the gap between Real World and Online Social Networks.



Gao et al. Data Analysis on Location-Based Social Networks

User Check-in Behaviours in LBSNs



Check-in Contents

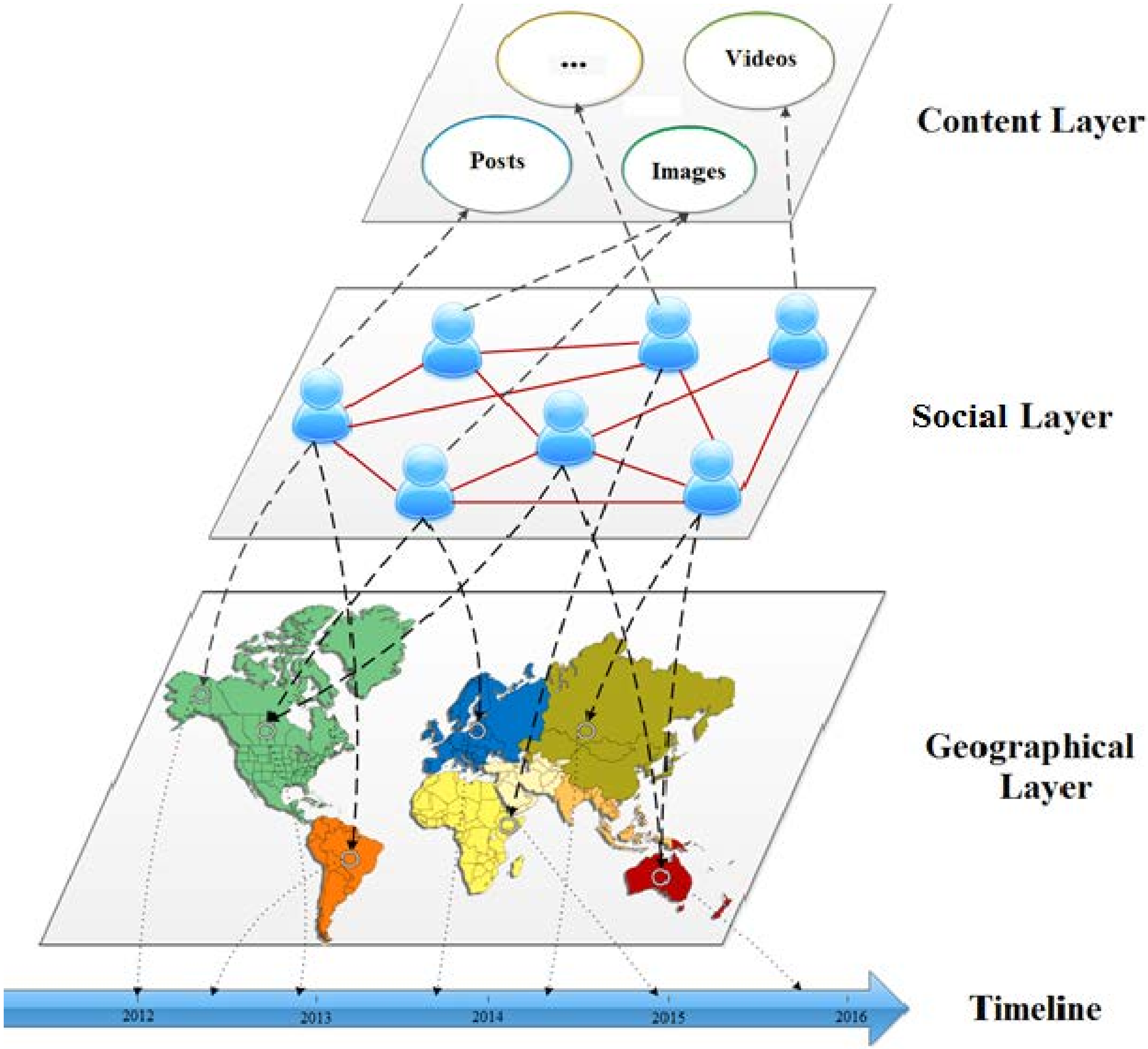
Great tourist visit if you never been to NYC. It's crowded, full of restaurants, Broadway theaters and stores that close late.

Mariana Pantalena · June 3

Save Like

A check-in record consist of four elements: user, POI, time and check-in content.

Information Networks in Geo-social Networks



Gao et al. Data Analysis on Location-Based Social Networks

Spatial Item Recommendation

- What to Recommend?
- Traditional Recommendation focuses on non-spatial items
  - Virtual Items, i.e., items that can be digitalized, such as movie, music, news, webpage, games, apps
  - Products on E-Commerce websites
- ST recommendation focuses on the recommendation of spatial items, i.e., items with geo-location attribute
  - Point of Interests, such as restaurants, hotels, shops, stations
  - Events or Activities, such as party, concerts, culture salons, conferences and outdoor

Online

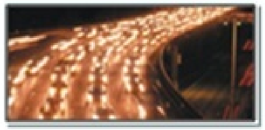
Offline

Spatial Item Recommendation

- Spatial item recommendation aims to provide users valuable suggestions and assist them make right decision in their daily routines and trip planning, by sensing and mining
  - User Activities in the offline world
  - User Generated Contents in the online world

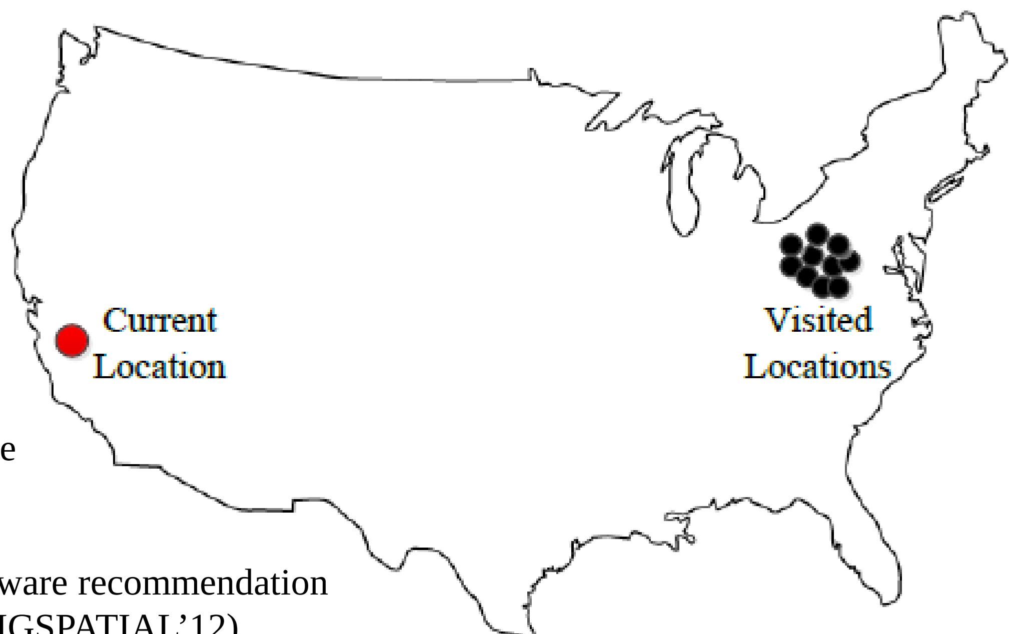
Geo-Social Networks can capture both.



Typical Recommendation Scenarios

- Home-town Recommendation
  - Make recommendations nearby users' hometown or familiar regions
  - Most studies focus on.
- Out-of-town Recommendation
  - Make recommendation when users travel out of town or unfamiliar regions
  - More useful.



Yin. et al. LCARS: A location-content-aware recommendation system. (KDD'13)

Bao. et al. Location-based and preference-aware recommendation using sparse geo-social networking data. (SIGSPATIAL'12)

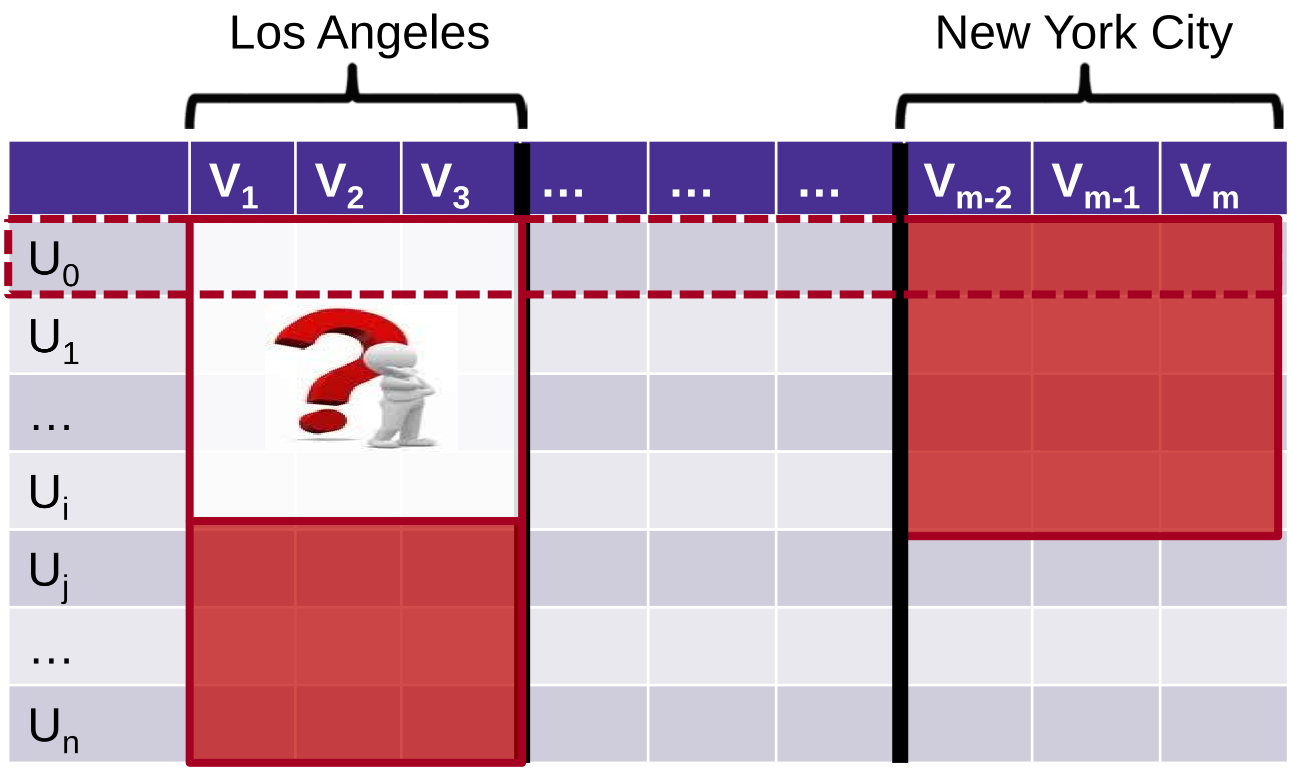
Outline

- Introduction
- Challenges of ST-Recommendation
- Effective Recommender Models

Data Sparsity and Travel Locality

- Data Sparsity
  - Millions of spatial items in the world
  - A user only check-ins a very small number of spatial items (less than 100), resulting in a very sparse user-item matrix.
- Travel Locality
  - Most of users' check-in records are generated in their living regions (e.g., home cities), since users tend to travel a limited distance when visiting venues and attending events. User check-in records out-of-town are extremely sparse.
  - An investigation shows that the check-in records generated by users in their non-home cities are very few and only take up 0.47% of the check-in records they left in their home cities.

Example



Millions of POIs around the world. *A* user checks-in less than **100 POIs.**

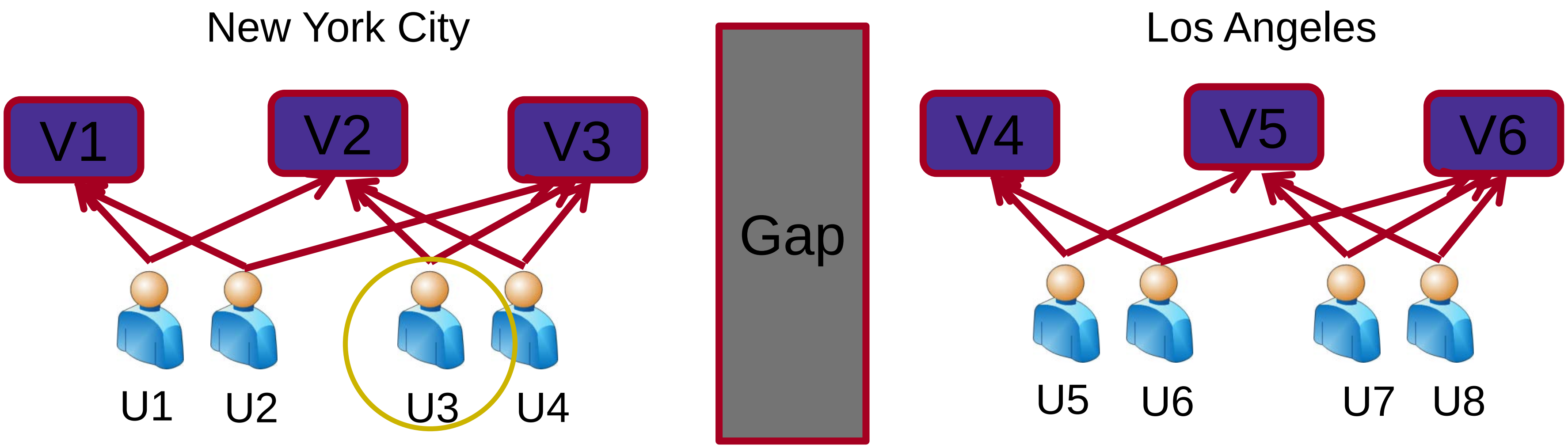
Most of POIs visited by users are located in **their hometowns** due to the locality of user travel.[1]

Problem: When users from New York City are traveling in Los Angeles,

how to make recommendations to them?

Yin. et al. LCARS: A location-content-aware recommendation system. (KDD'13)
Bao. et al. Location-based and preference-aware recommendation using sparse geo-social networking data. (SIGSPATIAL'12)

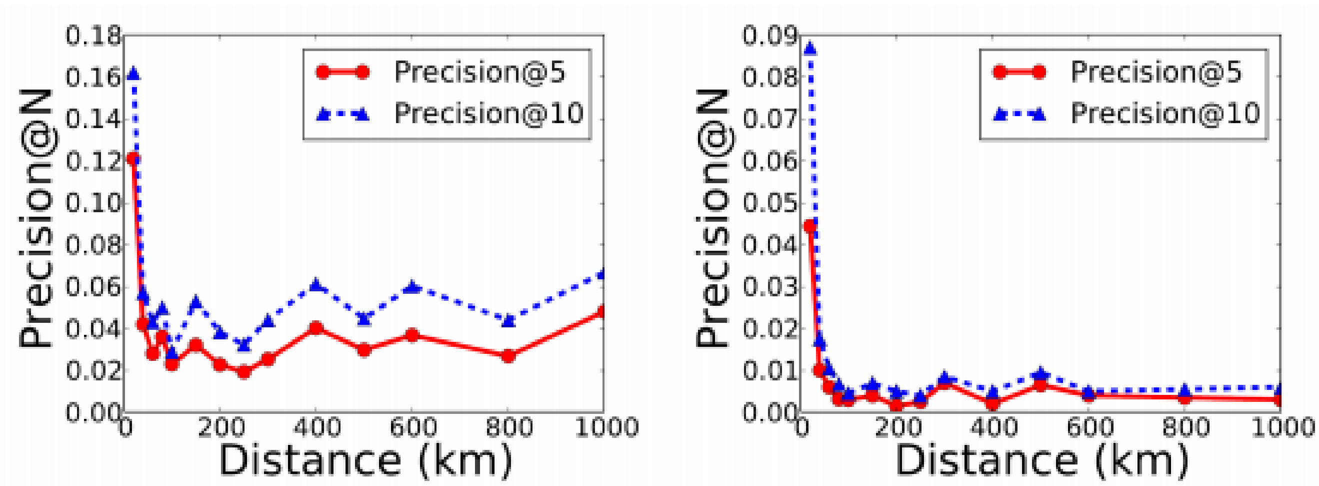
Analysis of CF-based Methods



Travel Locality: When U3 travels to Los Angeles that is new to her since she has no activity history there, how can we recommend spatial items to her? In other words, how to link the users in one side to the items in the other side?

Both Graph-based methods and Collaborative Filtering methods would fail in this scenario.

Performance of CF-based Methods



(a) Foursquare (b) Gowalla

1. CF performs well when the target locations are close to the home locations.
2. The precision degrades when the target locations are 100km away from their home locations.
3. The abrupt change at 100km can be explained by the fact that around 100 km is the typical human radius of "reach" as it takes about 1 to 2 hours to drive such distance.

[1] Ference *et al.* Location Recommendation for Out-of-Town Users in Location-Based Social Networks. In CIKM, 2013

Spatial Dynamics of User Interests

- Spatial Dynamics of User Interests
  - Users tend to have different preferences when they travel in different regions, especially which have different urban compositions and cultures.
  - For example, a user never goes gambling when she lives in Beijing, China, but when she travels in Macao or Las Vegas she is most likely to visit casinos.
  - User preferences learned from her check-ins at one region (e.g, home city) are not necessarily applicable to other regions.

Spatial Dynamics of User Interests

- Spatial Dynamics of User Interests
  - We derive top four categories of POIs visited by a group of users in three different cities.

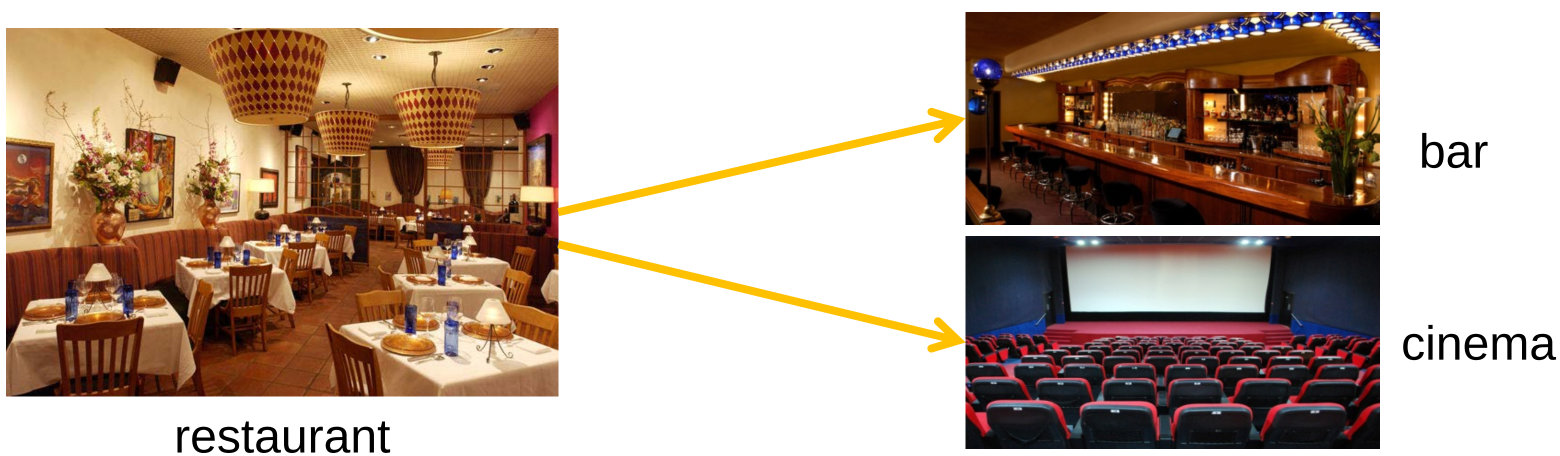
City	Top POI Types	Percentage of Check-ins(%)
Gold Coast (AU)	Beach	71.36%
	Surf Spot	14.82%
	Theme Park	9.60%
	Scenic Lookout	3.36%
Las Vegas (US)	Casino	80.32%
	Nightlife	10.61%
	Outlet	5.82%
	Hotel	3.23%
Istanbul (Turkey)	Mosque	68.32%
	Museum	15.45%
	Cafe	7.65%
	Art Gallery	5.83%

Yin. et al. Adapting to user interest drift for POI recommendation. TKDE, 2016

Sequential Influence

- Sequential Influence
  - Human movement exhibits sequential patterns.
  - Besides personal interests, we also need to consider the spatial items the user has visited recently.

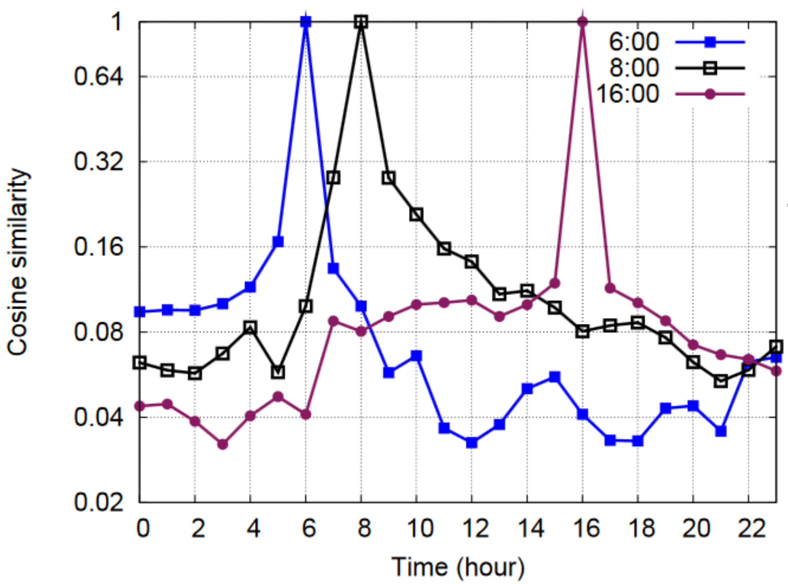
People usually go to cinemas or bars after restaurants since they would like to relax after dinner.



[1] C. Song, Z. Qu, N. Blumm, and A.-L. Barabasi, “Limits of predictability in human mobility,” Science, 2010.

Temporal Dynamics

- Temporal Dynamics of User Preferences
  - Generally, users tend to have different needs and preferences at different times.
  - A user is more likely to go to a restaurant rather than a bar for lunch at noon, and is more likely to go to a bar rather than a library at midnight.



User preference similarities between a given hour (6:00, 8:00, and 16:00) and other hours

Yuan et al. Time-aware Point-of-interest Recommendation. SIGIR-13

Temporal Dynamics

- A user's preferences change continuously over time, but exhibits temporal cyclic patterns.
  - A user may regularly arrive at the office around 9:00 am, go to a restaurant for lunch at 12:00 pm, and watch movies at night around 8:00 pm
- There are multiple types of temporal cyclic patterns
  - Daily effect
  - Weekly effect
  - Weekday-Weekend pattern
  - Seasonal effect
- How to automatically choose the proper time granularity?
- How to implement a multi-granularity temporal model to automatically adapt to different datasets?

Hosseini. et al. Jointly Modelling Heterogeneous Temporal Properties in Location Recommendation (DASFAA-17)

Summary of Challenges

- Related with Spatial Factor
  - Data Sparsity
  - Travel Locality
  - Spatial Dynamics of User Interests;
    - Also called the drift of user interest across geographical regions

Out-of-town recommendation
A hard task!!!

- Related with Temporal Factor
  - Sequential Influence
  - Multi-Granularity Temporal Cyclic Patterns

References

- Hongzhi Yin, Yizhou Sun, Bin Cui, Zhiting Hu, Ling Chen. "LCARS: A Location-Content-Aware Recommender System". KDD 2013.
- Hongzhi Yin, Bin Cui, Yizhou Sun, Zhiting Hu, Ling Chen. "LCARS: A Spatial Item Recommender System". TOIS 2014.
- Weiqing Wang, Hongzhi Yin, Ling Chen, Yizhou Sun, Shazia Sadiq, Xiaofang Zhou. "Geo-SAGE: A Geographical Sparse Additive Generative Model for Spatial Item Recommendation" . KDD 2015
- Weiqing Wang, Hongzhi Yin, Ling Chen, Yizhou Sun, Shazia Sadiq, Xiaofang Zhou. "ST-SAGE: A Spatial-Temporal Sparse Additive Generative Model for Spatial Item Recommendation". TIST 2017.
- Weiqing Wang, Hongzhi Yin, Shazia Sadiq, Ling Chen, Min Xie, Xiaofang Zhou. "SPORE: A Sequential Personalized Spatial Item Recommender System". ICDE 2016.
- Hongzhi Yin, Xiaofang Zhou, Bin Cui, Hao Wang, Kai Zheng, Quoc Viet Hung Nguyen. "Adapting to User Interest Drift for POI Recommendation". TKDE 2016.

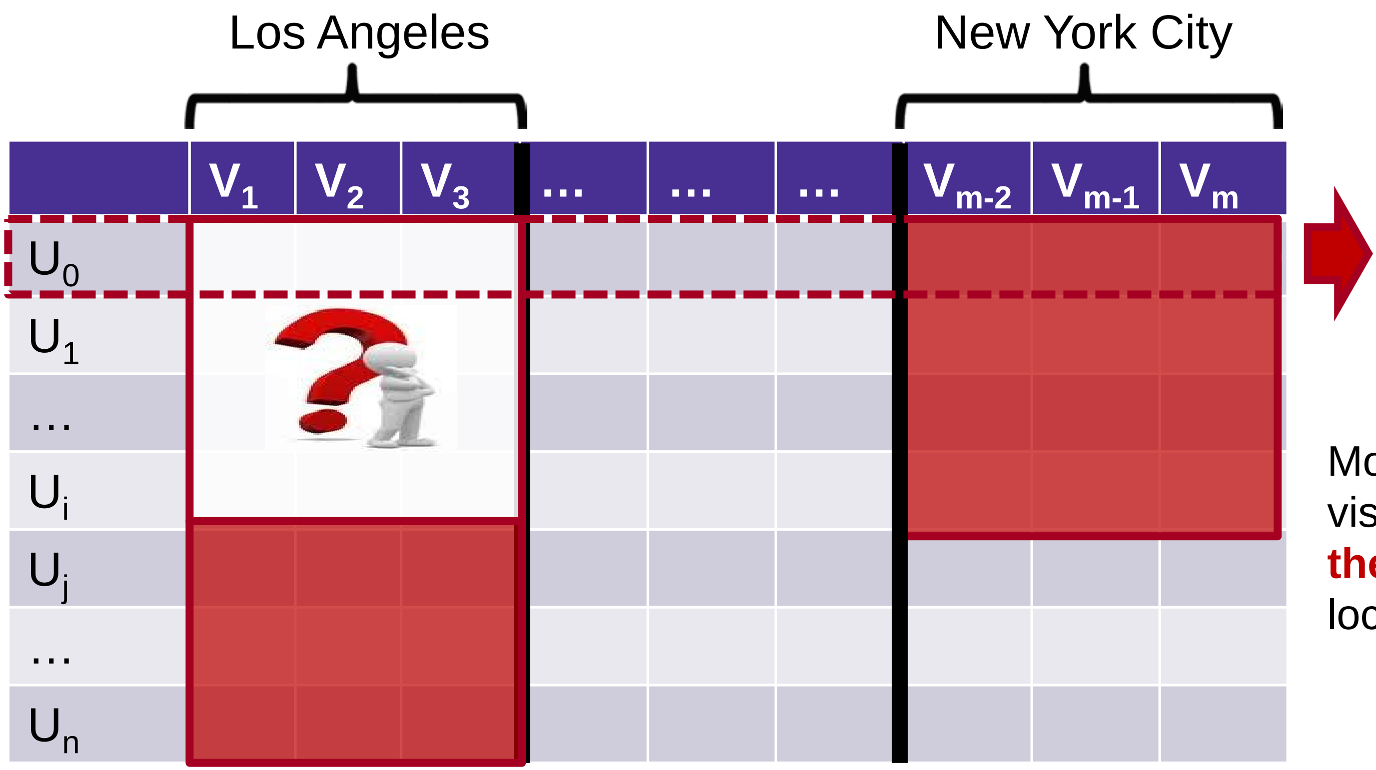
References

- Hongzhi Yin, Bin Cui, Xiaofang Zhou, Weiqing Wang, Zi Huang, Shazia Sadiq. "Joint Modeling of User Check-in Behaviors for Real-time Point-of-Interest Recommendation". TOIS 2016.
- Saeid Hosseini, Hongzhi Yin, Meihui Zhang, Xiaofang Zhou and Shazia Sadiq. Jointly Modelling Heterogeneous Temporal Properties in Location Recommendation. (DASFAA'17)
- Min Xie, Hongzhi Yin, Fanjiang Xu, Hao Wang, Weitong Chen and Sen Wang. "Learning Graph-based POI Embedding for Location-based Recommendation". CIKM 2016.
- Hongzhi Yin, Xiaofang Zhou, Yingxia Shao, Hao Wang, Shazia Sadiq. "Joint Modeling of User Check-in Behaviors for Point-of-Interest Recommendation". CIKM 2015
- Hongzhi Yin, Bin Cui, Xiaofang Zhou. "Spatio-Temporal Recommendation in Geo-Social Networks" in Reda Alhajj and Jon Rokne (eds), "Encyclopedia of Social Network Analysis and Mining". Springer, 2017

Outline

- Introduction
- Challenges of ST-Recommendation
- Effective Recommender Models
  - To Address the Challenges with Spatial Factors
  - To Address the Challenges with Temporal Factors

Data Sparsity and Travel Locality



Millions of POIs around the world. *A* user checks-in less than **100 POIs.**

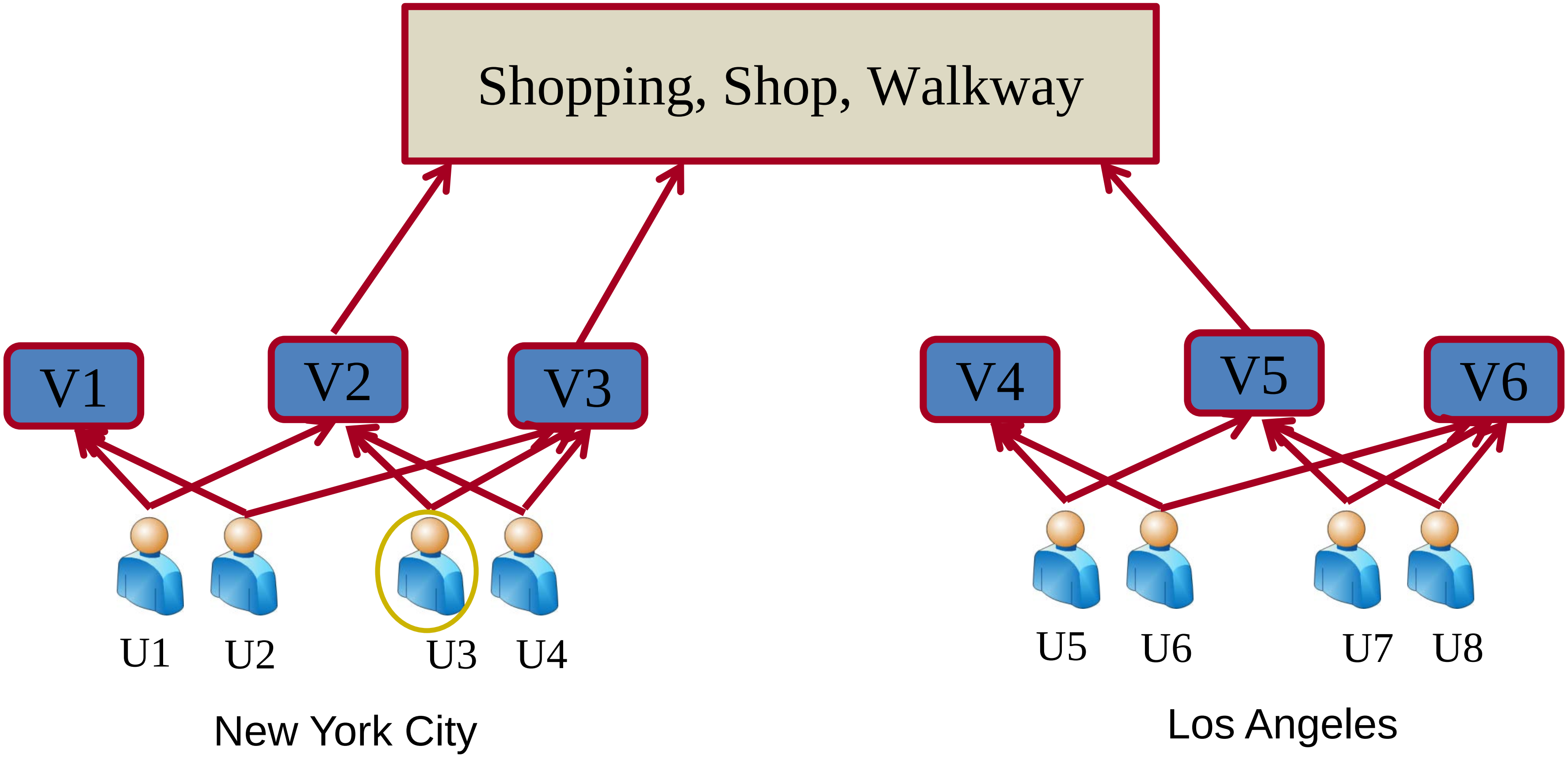
Most of POIs visited by users are located in **their hometowns** due to the locality of user travel.[1]

Problem: When users travel to an unfamiliar region, how to make recommendations to them?

[1] Levandoski *et al.* Lars: A Location-Aware Recommender System. In ICDE, 2012

Identifying and Transferring User Interests

As only the user-item interaction matrix is not enough to identify and transfer user interests, we leverage the content information of spatial items as medium.



The users in one side and the items in the other side can be linked together by the item contents.

Leverage the Wisdom of Crowds

- Leveraging the wisdom of crowds to deal with issue of user interest drift
  - By analyzing the word-of-mouth opinions from people who have visited *l* before, i.e., when people travel in city *l*, what do most of them do? Which POIs have they visited? Which events attended? Exploiting the crowd's behaviors to overcome the data sparsity of individual users in the unfamiliar regions.


Intuitive Ideas

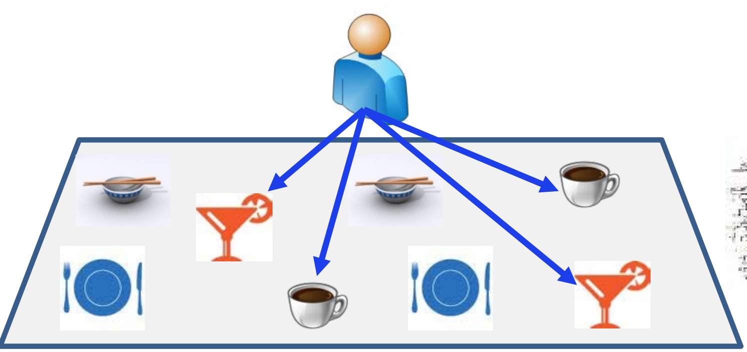
User Personal Interests/Preferences

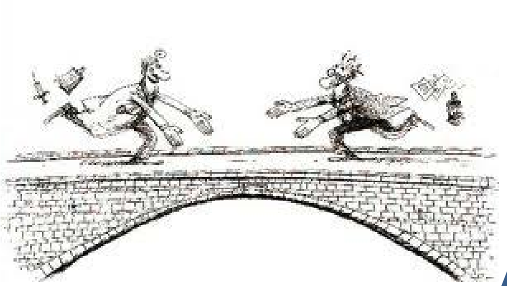
The crowd's preference in each region

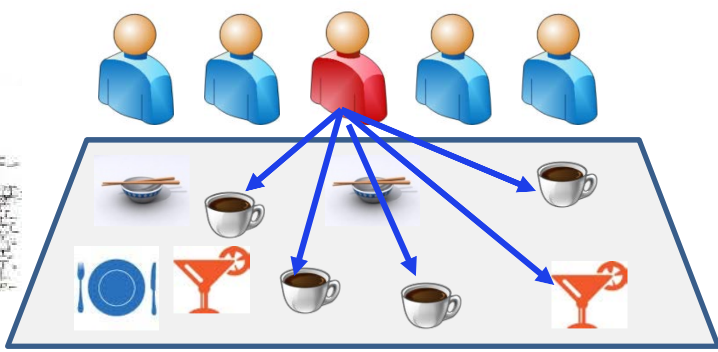
Main idea #1:
Identify **user interest** **according to contents of their visited spatial items.**

Main idea #3:
Combine **personal interest** & **region-aware crowd's preferences**

Main idea #2: Discover the crowd's preferences w.r.t each region







[1] Yin *et al.* LCARS: a location-content-aware recommender system. In KDD, 2013

Implementation of Intuitive Ideas

- How to represent a user's personal interests?
- How to represent the crowd's preferences with respect to a specific region?
- How to combine the two factors in a principal way to make recommendations?

Location-Content-Aware LDA Model

- How to represent a user's personal interests?
  - A multinomial distribution over a set of topics
- How to represent the crowd's preferences with respect to a specific region?
  - A multinomial distribution over a set of topics
- How to combine the two factors in a principal way?
  - By introducing a "switch" variable to indicate which factor will be used to generate the user's current check-in behavior
- Using topics to characterize both user interests and crowd preferences.

[1] Yin *et al.* LCARS: a location-content-aware recommender system. In KDD, 2013

How to represent a topic

- Topic representation in topic models (LDA, PLSA): a multinomial distribution over a set words

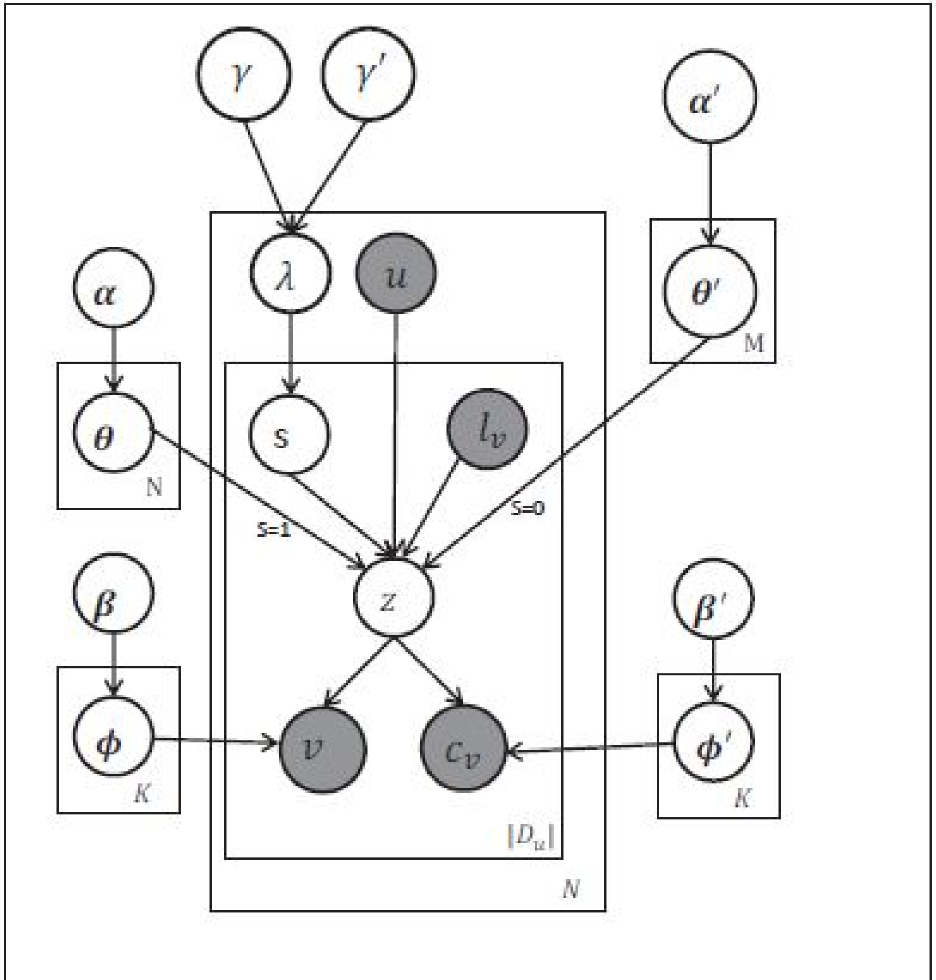
Topics discovered by LDA from DBLP

Topic 1	Topic 2	Topic 3	Topic 4
retrieval 0.13	**mining 0.11**	**neural 0.06**	**web 0.05**
information 0.05	**data 0.06**	**learning 0.02**	**services 0.03**
document 0.03	**discovery 0.03**	**networks 0.02**	**semantic 0.03**
query 0.03	**databases 0.02**	**deep 0.02**	**services 0.03**
text 0.03	**rules 0.02**	**analog 0.01**	**peer 0.02**
search 0.03	**association 0.02**	**vlsi 0.01**	**ontologies 0.02**
evaluation 0.02	**patterns 0.02**	**neurons 0.01**	**rdf 0.02**
user 0.02	**frequent 0.01**	**gaussian 0.01**	**management 0.01**
relevance 0.02	**streams 0.01**	**network 0.01**	**ontology 0.01**

LCA-LDA Model

- **Topic**: A topic z in LCA-LDA correspond to two distributions ϕ_z and $\phi'_{z.}$ The former is a multinomial distribution over items (item ID) and the latter is a distribution over content words.
  - Enabling clustering of both content-similar and co-visited spatial items into the same topics with high probability
  - Integrating both CF information and content information
- **User Interests**: The intrinsic interests of user u are represented by θ_u, a multinomial distribution over topics.
- **Crowd Preferences:** The crowd preferences w.r.t a region l are represented by θ_l, a multinomial distribution over topics.
- **"Switch" Variable**: A switch variable s is introduced to indicate which factor will be responsible for generating the check-in.

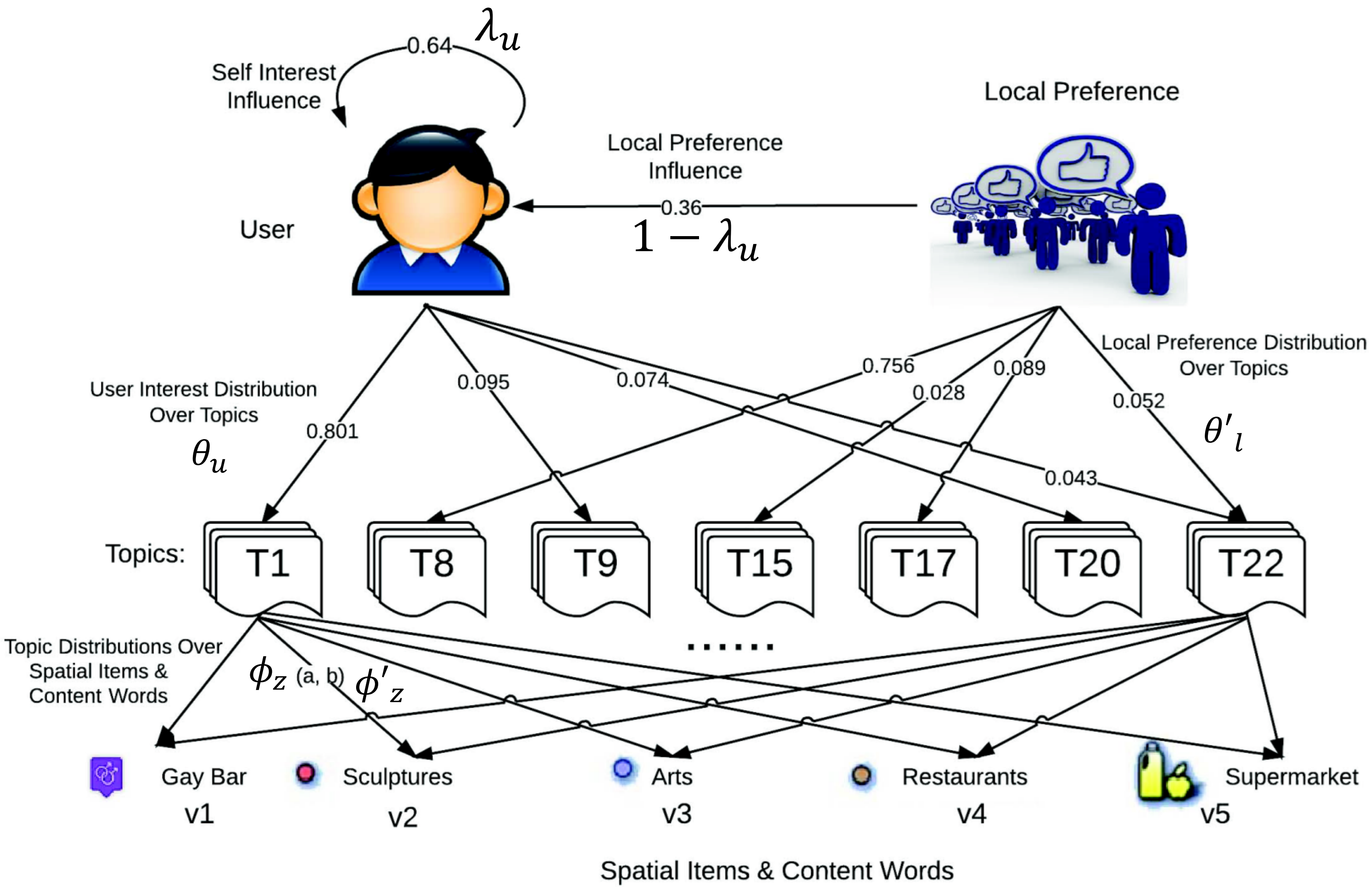
The Generative Process of LCA-LDA



Algorithm 1: Probabilistic generative process in LCA-LDA

for *each topic* z **do**
  Draw $\phi_z \sim Dirichlet(\cdot|\beta)$;
  Draw $\phi'_z \sim Dirichlet(\cdot|\beta')$;
end
for *each* D_u *in* D **do**
  for *each record* $(u,\ v_{ui},\ l_{ui},\ c_{ui}) \in D_u$ **do**
    Toss a coin s_{ui} according to $bernoulli(s_{ui}) \sim beta(\gamma, \gamma')$;
    if $s_{ui} = 1$ **then**
      Draw $\theta_u \sim Dirichlet(\cdot|\alpha)$;
      Draw a topic $z_{ui} \sim multi(\theta_u)$ according to the interest of user u;
    end
    if $s_{ui} = 0$ **then**
      Draw $\theta'_{l_{ui}} \sim Dirichlet(\cdot|\alpha')$;
      Draw a topic $z_{ui} \sim multi(\theta'_{l_{ui}})$ according to the local preference of l_{ui};
    end
    Draw an item $v_{ui} \sim multi(\phi_{z_{ui}})$ from z_{ui}-specific spatial item distribution;
    Draw a content word $c_{ui} \sim multi(\phi'_{z_{ui}})$ from z_{ui}-specific content word distribution;
  end
end

Structure of LCA-LDA



$$P(v|\theta_u^{user}, \theta_l^{crowd}) = \lambda_u P(v|\theta_u^{user}) + (1-\lambda_u) P(v|\theta_l^{crowd})$$

Online Recommendation

- The model parameters in LCA-LDA are estimated by Gibbs sampling.
- Given a query $q=(u,l)$, the ranking score of each item v is computed as the inner product of the two vectors:

$$S(q,v) = \sum_z F(v,z)W(q,z)$$

$$W(q,z) = \hat{\lambda}_u \hat{\theta}_{uz} + (1-\hat{\lambda}_u)\hat{\theta}'_{l_u z}$$

$$F(v,z) = \begin{cases} \hat{\phi}_{zv} \sum_{c_v \in \mathcal{C}_v} \hat{\phi}'_{zc_v} & v \in \mathcal{V}_{l_u} \\ 0 & v \notin \mathcal{V}_{l_u} \end{cases}$$

Example: q=(u, l) Parameters ={ $\lambda_u = 0.4$, $\theta_{uz}, \theta'_{lz}, F_{zv}$ }

λ_u θ_u

Arts	0.25
Food	0.2
Shop	0.35
Night Life	0.2

\+ $(1-\lambda_u)$ θ'_l

Arts	0.1
Food	0.5
Shop	0.3
Night Life	0.1

= W_q

Arts	0.16
Food	0.38
Shop	0.32
Night Life	0.14

S(q,v) = W_{qz}

Arts	0.16
Food	0.38
Shop	0.32
Night Life	0.14

× F_{zv}

Arts	0.0
Food	0.2
Shop	0.0
Night Life	0.0

= 0.076

Limitations of LCA-LDA Model

- Inference complexity
- LCA-LDA considers multiple factors by introducing the additional "switch" variable s to decide whether a topic is drawn from the user's interests or the crowd's preferences.
  - We need to sample both switch and topic for each check-in record
- From the perspective of mixture models, it needs to estimate a mixture weight λ_u for each user by the switch variable s.
- It is not only computationally expensive to learn personalized mixture weights for individual users but also difficult to learn these mixture weights accurately given sparse datasets.

Limitations of LCA-LDA Model

- LCA-LDA ignores the roles of users.
  - Users with different roles tend to have different preferences regarding a region, such as local people vs. tourists
- The location l in LCA-LDA is fixed granularity, such as city. When the location granularity changes, LCA-LDA model needs to be retrained from scratch.
  - That is very time-consuming and infeasible.

Geo-Sparse Additive Generative Model

User Personal Interests/Preferences

How to combine?

Crowd's preferences in a querying region

- **LCA-LDA**

$$P(z|\theta_u^{user}, \theta_l^{crowd}) = \lambda_u P(z|\theta_u^{user}) + (1-\lambda_u) P(z|\theta_l^{crowd})$$

When the data is sparse, infer the mixture weight by the switch variable for each user is both expensive and inaccurate.

- **Geo-SAGE**

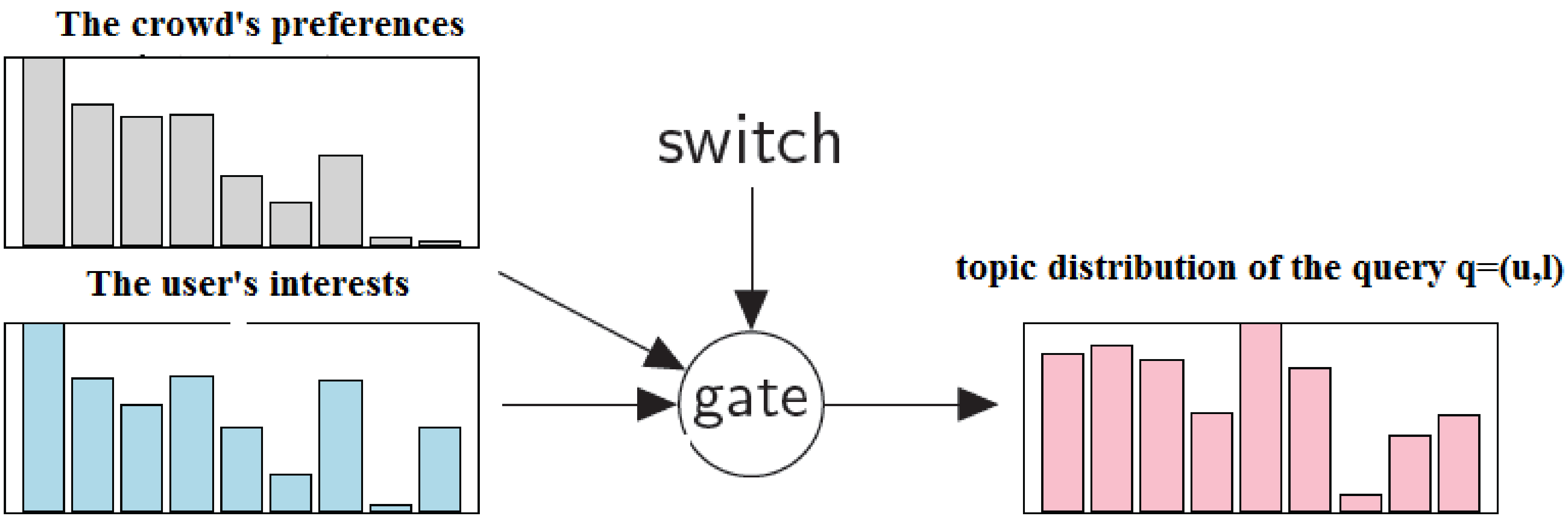
$$P(z|\theta_u^{user}, \theta_l^{crowd}) = \frac{exp(\theta_{u,z}^{user} + \theta_{l,z}^{crowd})}{\sum_{z'} exp(\theta_{u,z'}^{user} + \theta_{l,z'}^{crowd})}$$

We can combine generative facets through simple addition in log space, avoiding the need for latent switching variables

Eisenstein et al. Sparse Additive Generative Models of Text (ICML 11)
Wang et al. Geo-SAGEA Geographical Sparse Additive Generative Model for Spatial Item Recommendation (KDD'15)

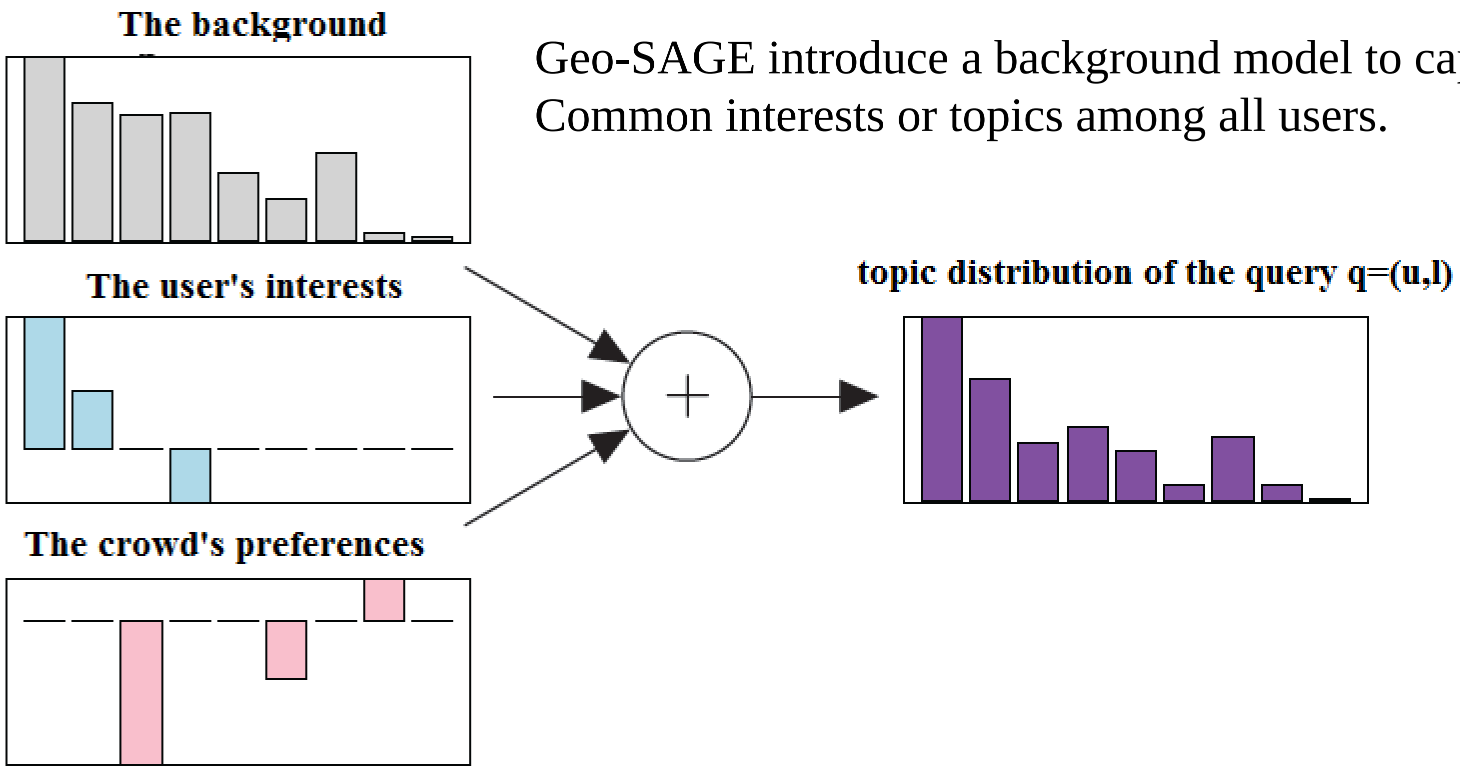
Illustration of LCA-LDA

LCA-LDA model using Dirichlet-multinomials



Both the user’s interests and the crowd’s preferences are represented by a probabilistic distributions and the mixture occurs in the distribution.

Illustration of Geo-SAGE



Geo-SAGE introduce a background model to capture the Common interests or topics among all users.

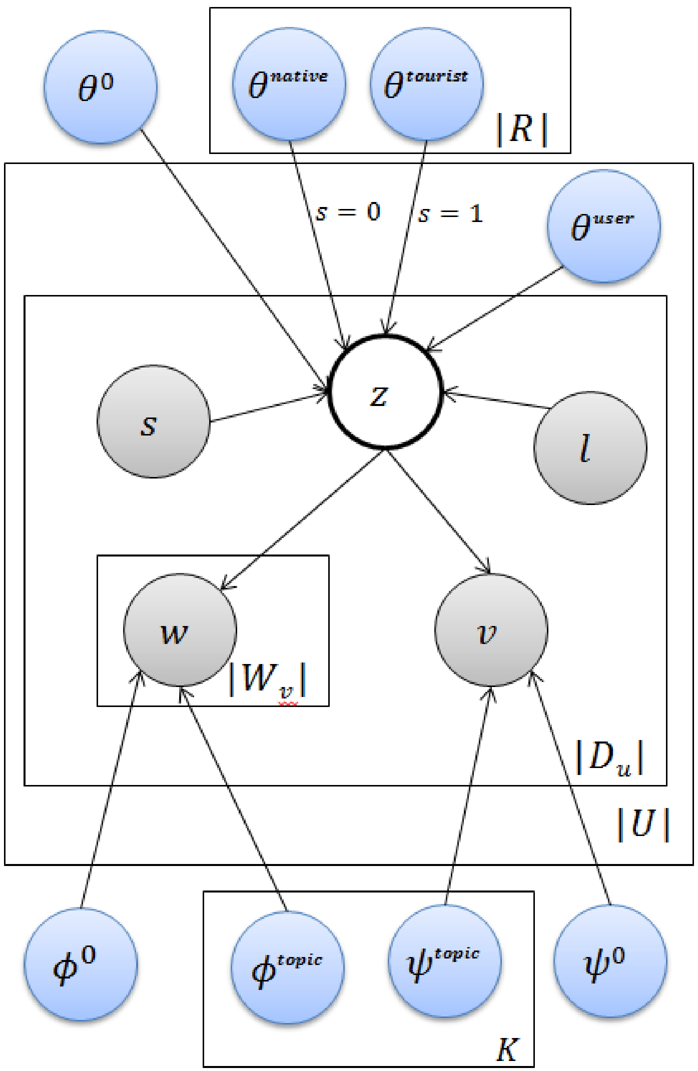
Both the user's interests and the crowd's preferences are represented by a vector with zero-mean variation .

The key difference between LCA-LDA and Geo-SAGE is that the mixture occurs in terms of natural parameters of the exponential family rather than distribution.

Role-Aware Crowd's Preferences

- Generally, the crowds with different roles tend to have different preferences.
- In Geo-SAGE, we refine the crowd's preferences
  - Native preferences: the common preferences of local people
  - Tourist preferences: the common preferences of tourists

Generative Process of Geo-SAGE



For each user activity record $(u,\ v,\ l_v,\ W_v,\ s)$

1. Draw a topic index *z* according to u's interests and the role-aware crowd's preferences.

$$P(z_{u,i}|s_{u,i},l_{u,i},\boldsymbol{\theta}^0,\boldsymbol{\theta}^{user},\boldsymbol{\theta}^{native},\boldsymbol{\theta}^{tourist})$$
$$=P(z_{u,i}|\boldsymbol{\theta}^0+\boldsymbol{\theta}_u^{user}+(1-s_{u,i})\times\boldsymbol{\theta}_{l_{u,i}}^{native}+s_{u,i}\times\boldsymbol{\theta}_{l_{u,i}}^{tourist})$$

2. Draw content words according to *z*'s distributions over the words

$$P(w_{u,i,n}|\boldsymbol{\phi}^0,z_{u,i},\boldsymbol{\phi}^{topic})=P(w_{u,i,n}|\boldsymbol{\phi}^0+\boldsymbol{\phi}_{z_{u,i}}^{topic})$$

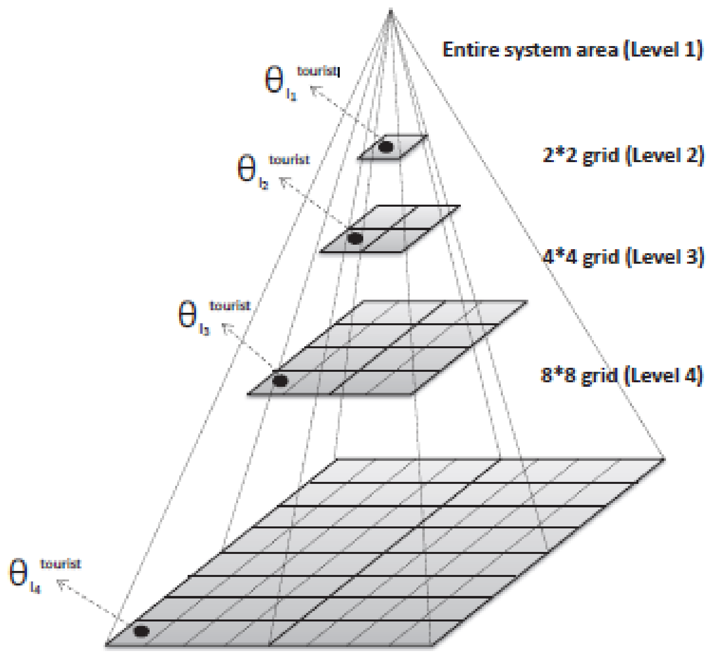
3. Draw a spatial item according to *z*'s distributions over the spatial items

$$P(v_{u,i}|\boldsymbol{\psi}^0,z_{u,i},\boldsymbol{\psi}^{topic})=P(v_{u,i}|\boldsymbol{\psi}^0+\boldsymbol{\psi}_{z_{u,i}}^{topic})$$

Leveraging Spatial Correlation for Data Sparsity

When there are few check-ins in a region, the inference of the role-aware crowd's preferences may be inaccurate for this region.

As close regions have similar urban compositions and cultures, crowd's preferences at these regions should be similar. Thus, we can "borrow" the check-in records from nearby regions to smooth crowd preferences at region r.



Spatial pyramid (a tree structure)

[2] Gale *et al.* Philosophy in Geography. 1979

1. Spatial pyramid: partition the whole area into grids of varying sizes at different levels
2. Representing each region by a path from root to its corresponding cell.
3. Based on the path-based region representation, a hierarchically additive framework to represent crowd's preferences

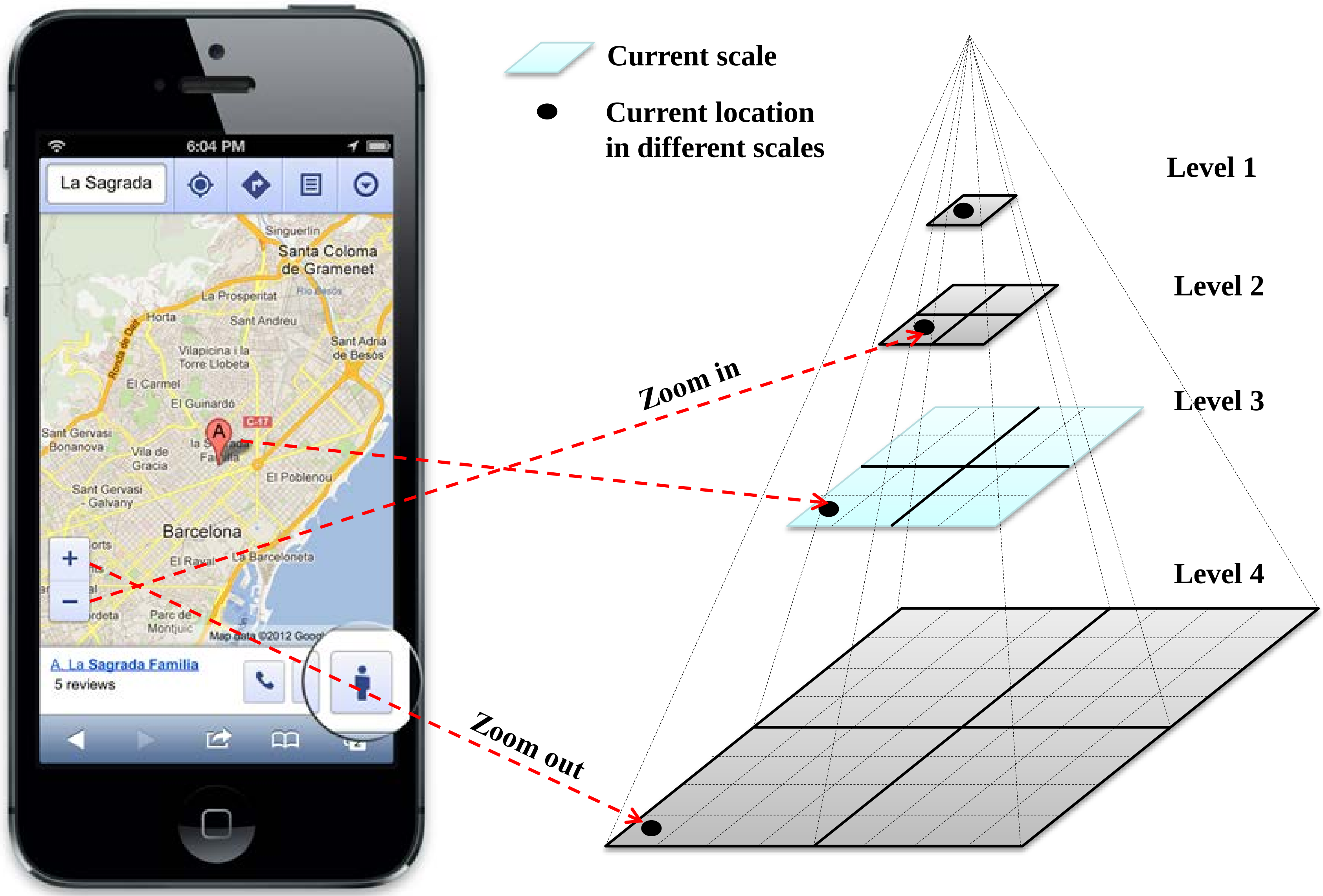
$$\theta_l^{native} = \sum_{h=1}^{H} \theta_{l_h}^{native}$$

$$\theta_l^{tourist} = \sum_{h=1}^{H} \theta_{l_h}^{tourist}$$

Advantages of hierarchically additive representations

- Alleviating Data Sparsity.
  - if there are few or no check-in records at region l, we can still infer crowd preferences based on the check-in data generated at l's ancestor regions.
- Can be seamlessly integrated into Geo-SAGE model.
  - Sharing the same additive feature
- When the granularity of regions changes, we do not need to retrain the model.

Automatic Adaption to the changing region size



Data sets

- Foursquare
  Publicly available
  Contain 483,813 check-in records of 4163 users who live in the California, USA
- Twitter
  Publicly available
  Contain 1,434,668 check-ins of 114,058 users who live across whole USA
- Distributions

Foursquare

Twitter

Comparative Approaches

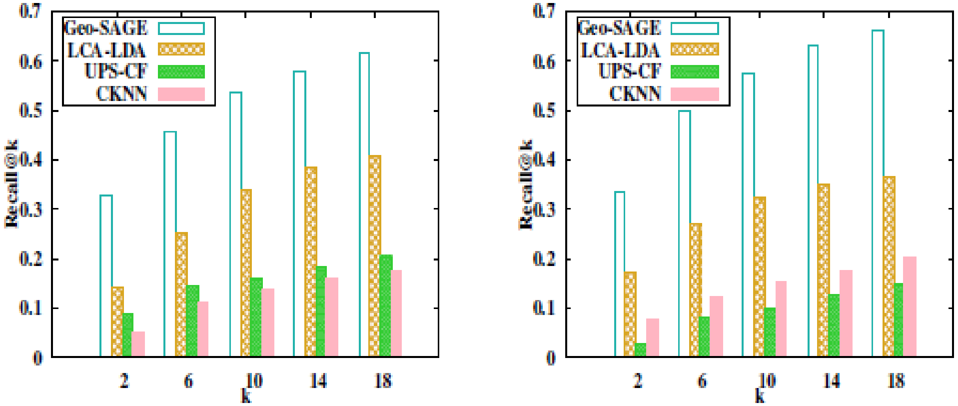
- State-of-Art Spatial Items Recommendation
  - LCA-LDA
    - Latent factor model, Does not distinguish locals from tourists, without SAGE
  - CKNN
    - Local experts based method
  - UPS-CF
    - Friend-based collaborative filtering method
- Variant versions of Geo-SAGE
  - Geo-SAGE-S1
    - Without crowd's preferences
  - Geo-SAGE-S2
    - Ignore the crowd's roles
  - Geo-SAGE-S3
    - Without spatial pyramid

[1] Yin *et al*. Lcars: A location-content-aware recommender system. In KDD, 2013

[2] Bao *et al*. Location-based and preference-aware recommendation using sparse geo-social networking data. In GIS, 2012

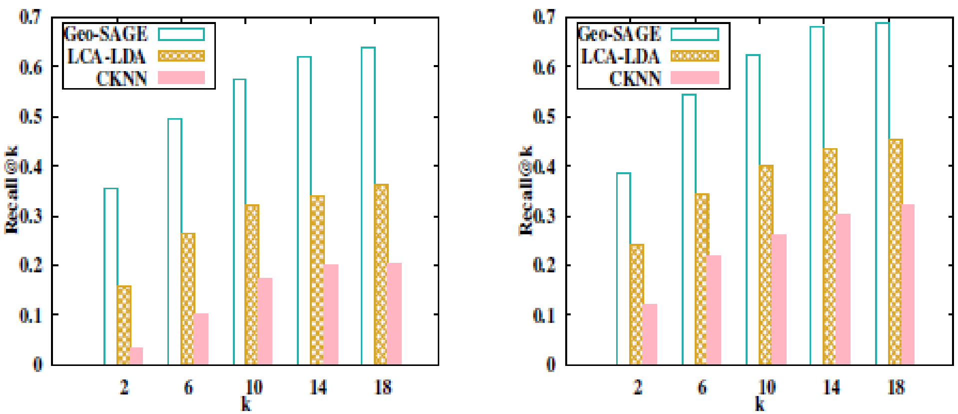
[3] Ference *et al*. Location recommendation for out-of-town users in location-based social networks, In CIKM, 2013

Recommendation Effectiveness



(a) Out-of-Town (b) Home-Town

Figure 4: Performance on Foursquare Dataset



(a) Out-of-Town (b) Home-Town

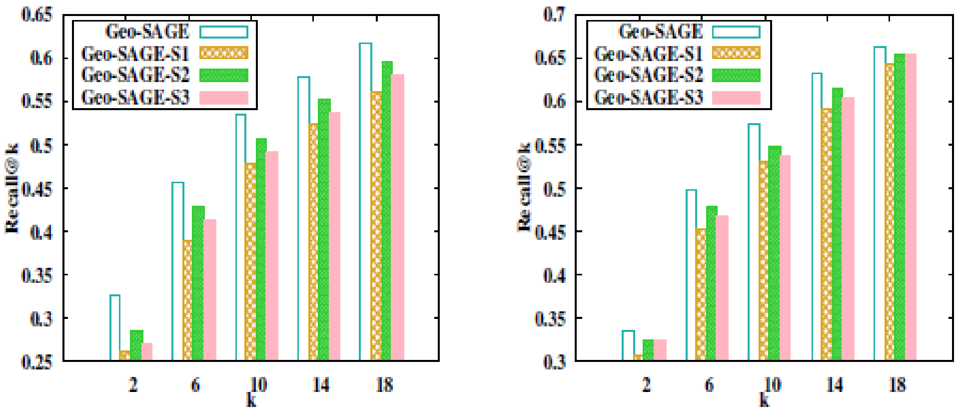
Figure 5: Performance on Twitter Dataset

1. **Geo-SAGE and LCA-LDA perform much better than CKNN:**
   leverage the crowd's preferences to address the challenges of user interest drift.

2. **Geo-SAGE and LCA-LDA perform much better than UPS-CF** :
   exploit the content information to identify and transfer user interests to overcome the changes of data sparsity and travel locality.

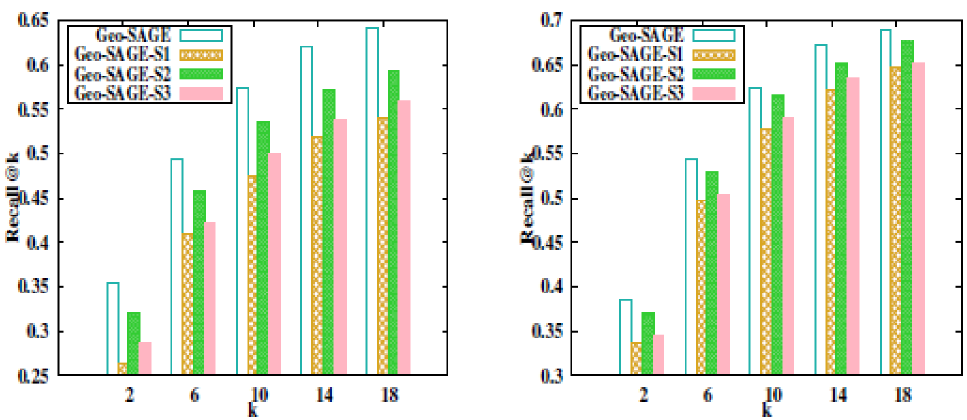
3. **Geo-SAGE performs much better than LCA-LDA:**
   apply the sparse additive model;
   role-aware crowd's preferences.

Impact of Different Factors



(a) Out-of-Town (b) Home-Town

Figure 6: Impact of Different Factors on Foursquare Dataset



(a) Out-of-Town (b) Home-Town

Figure 7: Impact of Different Factors on Twitter Dataset

1. **Geo-SAGE consistently performs better than the three variant versions:**
   indicate the benefits brought by each factor .
   Geo-SAGE-S1: without crowd's preferences
   Geo-SAGE-S2: ignore users' role
   Geo-SAGE-S3: without spatial pyramid

2. **Geo-SAGE-S2 and Geo-SAGE-S3 always perform better than Geo-SAGE-S1** :
   show the advantage of integrating the crowd's preferences.

3. **The performance gap in home-town recommendation is smaller than out-of-town recommendation:**
   performance improvement become less obvious when people travel in home town.

Short Summary

- Challenges Related with Spatial Factor
  - Travel Locality
    - Exploiting the content information of spatial items to identity and transfer users' intrinsic interests
  - Spatial Dynamics of User Interests
    - Exploiting role-aware crowd's preferences with respect to each region
  - Data Sparsity
    - Leveraging the spatial auto-correlation to borrow check-in data from other close regions

Out-of-town Recommendation

References

- Hongzhi Yin, Yizhou Sun, Bin Cui, Zhiting Hu, Ling Chen. “LCARS: A Location-Content-Aware Recommender System”. KDD 2013.
- Hongzhi Yin, Bin Cui, Yizhou Sun, Zhiting Hu, Ling Chen. “LCARS: A Spatial Item Recommender System”. TOIS 2014.
- Weiqing Wang, Hongzhi Yin, Ling Chen, Yizhou Sun, Shazia Sadiq, Xiaofang Zhou. “Geo-SAGE: A Geographical Sparse Additive Generative Model for Spatial Item Recommendation” . KDD 2015
- Hongzhi Yin, Xiaofang Zhou, Bin Cui, Hao Wang, Kai Zheng, Quoc Viet Hung Nguyen. “Adapting to User Interest Drift for POI Recommendation”. TKDE 2016.
- Weiqing Wang, Hongzhi Yin, Ling Chen,Yizhou Sun and Xiaofang Zhou. ST-SAGE: A Spatial-Temporal Sparse Additive Generative Model for Spatial Item Recommendation (ACM TIST’17)
- Eisenstein et al. Sparse Additive Generative Models of Text (ICML 11)

Outline

- Introduction
- Challenges of ST-Recommendation
- Effective Recommender Models
  - To Address the Challenges with Spatial Factors
  - To Address the Challenges with Temporal Factors

Sequential Effect

- **Human movement exhibits sequential patterns** [1, 2]
- **Result from many factors**
  - Temporal Effect, such as time in one day [3]
    - People tend to go to restaurants at dinner time and then relax in cinemas or bars at night.
  - Geographical Influence, geographical proximity [4]
    - Tourists often sequentially visit London Eye, Big Ben and Downing Street.
  - Other life-style related factors [5]
    - People usually check in at a Gym before a restaurant instead of the reverse way because it is not healthy to exercise right after a meal.

[1] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: User movement in location-based social networks," KDD, 2011

[2] C. Song, Z. Qu, N. Blumm, and A.-L. Barabasi, "Limits of predictability in human mobility," Science, 2010.

[3] J.-D. Zhang and C.-Y. Chow, "Spatiotemporal sequential influence modeling for location recommendations: A gravity-based approach," TIST, 2015

[4] Z. Yin, L. Gao, J. Han, J. Luo and T. S. Huang, "Diversified trajectory pattern ranking in geo-tagged social media", SIAM, 2011

[5] H. P. Hsieh, C, T. Li and S. D. Lin, "Measuring and recommending time-sensitive routes from location-based data", TIST 2014

Challenges-Modeling Sequential Influence

- The widely adopted **Markov chain-based** methods encounter the challenge of **huge state prediction space**
- Classical nth-order Markov chain
  - Predict the next possibly visiting spatial items based on all historical visited ones (Zhang et al., IEEE MDM, 2014).
  - Disadvantage: The prediction state space is $O(|V|^{n+1})$. When the number of items $|V|$is slightly large, this method does not work.
- First-order Markov chain
  - Predict the next possibly visiting spatial item based on only the latest visited one (Chen et al., IEEE ICDE, 2011; Cheng et al., ACM MM, 2011; Cheng et al., IJCAI, 2013; Kurashima, ACM CIKM, 2010; Zheng et al., ACM TIST, 2012).
  - Disadvantage: Ignore the effect of other recent visited spatial items. Even in the first-order Markov chain model, the prediction state space is also very huge when there are millions of spatial items.

Challenges-Unifying Personalization and Sequential Effect

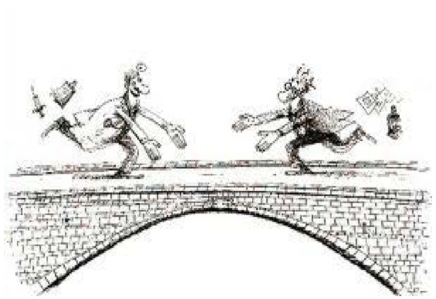
- Unifying personalization and sequential effect
  - Traditional recommender system
    - Focusing on personalization
    - Neglect the sequential effect
  - Existing sequential recommender system (i.e. Markov Chain)
    - Assume the same transition probabilities between items for all users
    - Ignore personalization

SAGE-based Solution: SPORE

Main idea #1: Identify **the personal interests of users** **based on** **topics (e.g., categories)**.

Main idea #3: Combine **personal interests** & **sequential influence** **in** **the additive framework-SAGE**

Main idea #2: Extracting **sequential influence** of all her recently visited spatial items in topic level instead of item level



Wang et al. SPORE: A sequential personalized spatial item recommender system. (ICDE'16)

SPORE

- We model personal interests and sequential influence based on the **latent variable topic-region** in SPORE.
  - A topic-region *z* jointly corresponds to a semantic topic (i.e., a soft cluster of words describing spatial items, referring to categories) and a geographical region (i.e., a soft cluster of locations of spatial items)
  - By introducing the topic-region, we decompose the spatial item prediction problem into two sub-problems:
    - predicting the topic-region z of the user's next activity based on her personal interests and her recently visited spatial items
    - then, predicting the next spatial items given the predicted topic-region z

Advantages by introducing latent Topic-Region

- Overcoming the data sparsity and low sampling rate
  - by focusing on the high-level topic-region rather than the fine granularity - spatial items
- Significantly reducing the prediction space
  - For each item v, we learn a distribution θ_v^{seq} over a set of topic-regions, and $\theta_{v,z}^{seq}$ represents the probability of visiting topic-region z after visiting item v.
  - The number of topic-region is much smaller than the number of items, e.g., less than 100. Thus, the state space is reduced to $|V| \times K$ from $|V|^2$ compared with 1-order Markov chain model.

Additive Sequential Influence

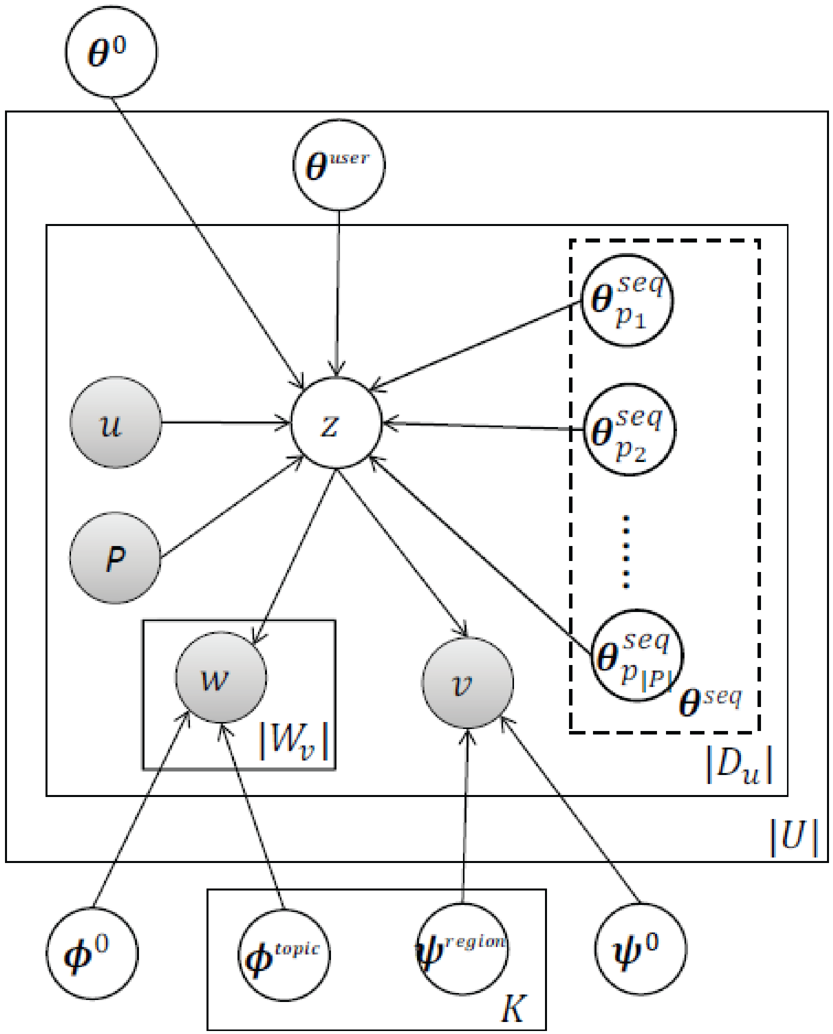
- P_u is the set of spatial items recently visited by u. Actually it is equivalent to a session in the online shopping scenario.
- How to combine the sequential influences from all items in P_u
  - Similar to the fusion of personal interests and sequential influence in SAGE model, we combine the sequential influence in the same way.

$$P(z|\boldsymbol{\theta}_{P_u}^{seq}) = \boldsymbol{\theta}_{P_u,z}^{seq} = \frac{exp(\sum_{v \in P_u} \boldsymbol{\theta}_{v,z}^{seq})}{\sum_{z'} exp(\sum_{v \in P_u} \boldsymbol{\theta}_{v,z'}^{seq})}$$

  - Compared with classical n-order Markov chain model
    - We reduce the state parameter space from $|V|^{n+1}$ to $|V| \times K$
  - Compared with n-order additive Markov chain model (Lore) that works in a traditional mixture way
    - We reduce the state parameter space from $|V|^2$ to $|V| \times K$. Besides, we avoid to compute the mixture weights.

Lore: Exploiting sequential influence for location recommendations," in *SIGSPATIAL*, 2014

Generative Process of SPORE



- Draw a topic-region index $z_{u,i}$
  $z_{u,i} \sim P(z_{u,i}|P_{u,t_{u,i}}, \boldsymbol{\theta}^0, \boldsymbol{\theta}^{user}, \boldsymbol{\theta}^{seq})$
- For each content word $w_{v_{u,i},n}$ in $W_{v_{u,i}}$, draw
  $w_{v_{u,i},n} \sim P(w_{v_{u,i},n}|\boldsymbol{\phi}^0, z_{u,i}, \boldsymbol{\phi}^{topic})$
- Draw a spatial item $v_{u,i}$
  $v_{u,i} \sim P(v_{u,i}|\boldsymbol{\psi}^0, z_{u,i}, \boldsymbol{\psi}^{region})$

$$P(z_{u,i}|P_{u,t_{u,i}}, \boldsymbol{\theta}^0, \boldsymbol{\theta}^{user}, \boldsymbol{\theta}^{seq}) = P(z_{u,i}|\boldsymbol{\theta}^0 + \boldsymbol{\theta}^{user}_u + \boldsymbol{\theta}^{seq}_{P_{u,t_{u,i}}})$$

$$P(v_{u,i}|\boldsymbol{\psi}^0, z_{u,i}, \boldsymbol{\psi}^{region}) = P(v_{u,i}|\boldsymbol{\psi}^0 + \boldsymbol{\psi}^{region}_{z_{u,i}})$$

$$P(w_{v_{u,i},n}|\boldsymbol{\phi}^0, z_{u,i}, \boldsymbol{\phi}^{topic}) = P(w_{v_{u,i},n}|\boldsymbol{\phi}^0 + \boldsymbol{\phi}^{topic}_{z_{u,i}})$$

Limitation of SPORE

- SPORE ignores the effect of time on user mobility behaviors.
- Spatial item recommendation is a time-subtle recommendation task since at different time, users would prefer different successive POIs.
    - A user may go to a restaurant after leaving from office at noon, while he/she may be more likely to go to a gym when he/she leaves office at night.

Zhao et al. STELLAR: Spatial-Temporal Latent Ranking for Successive Point-of-Interest Recommendation. (AAAI’16)

Multi-Granularity Temporal Modeling

- There are multiple time granularities and various temporal cyclic patterns
  - Daily effect
  - Weekly effect
  - Weekday-Weekend pattern
  - Monthly effect
  - Seasonal effect
  - …
- How to integrate various temporal cyclic patterns?
- How to implement a multi-granularity temporal model to automatically adapt to different datasets?

SAGE-Based Solution

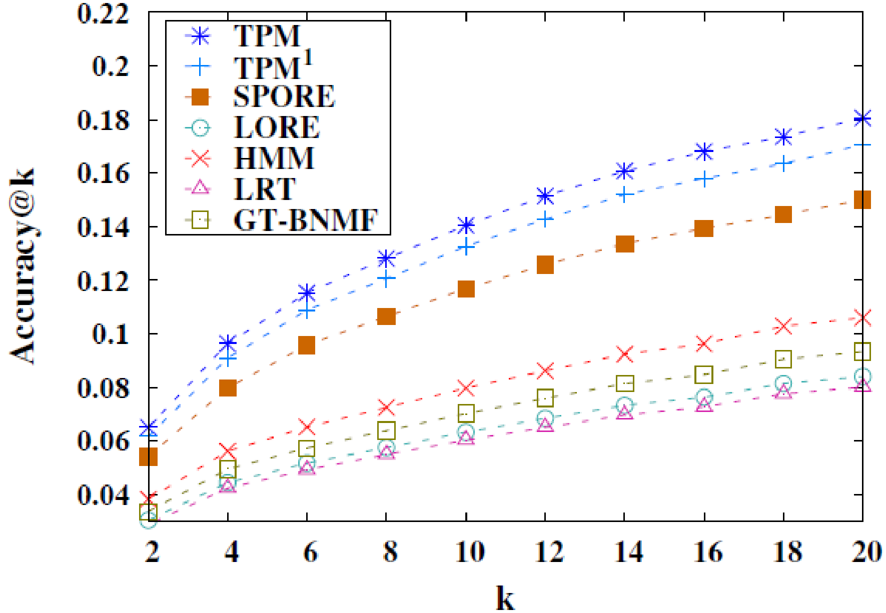
- This method divides the time stamps according to the different granularities separately.
- In this way, each time stamp has multiple time ids with respect to the multiple granularities respectively.
- Then, combine the three temporal patterns in the additive SAGE framework.
  - add the effect of personal interests, multi-granularity temporal cyclic effect and sequential influence in the latent space

$$P(z_{u,i}|P_{u,t_{u,i}}, \boldsymbol{\theta}^{time}, \boldsymbol{\theta}^{user}, \boldsymbol{\theta}^{seq}) = P(z_{u,i}|\boldsymbol{\theta}^{time}_{t_{u,i}} + \boldsymbol{\theta}^{user}_{u} + \boldsymbol{\theta}^{seq}_{P_{u,t_{u,i}}})$$

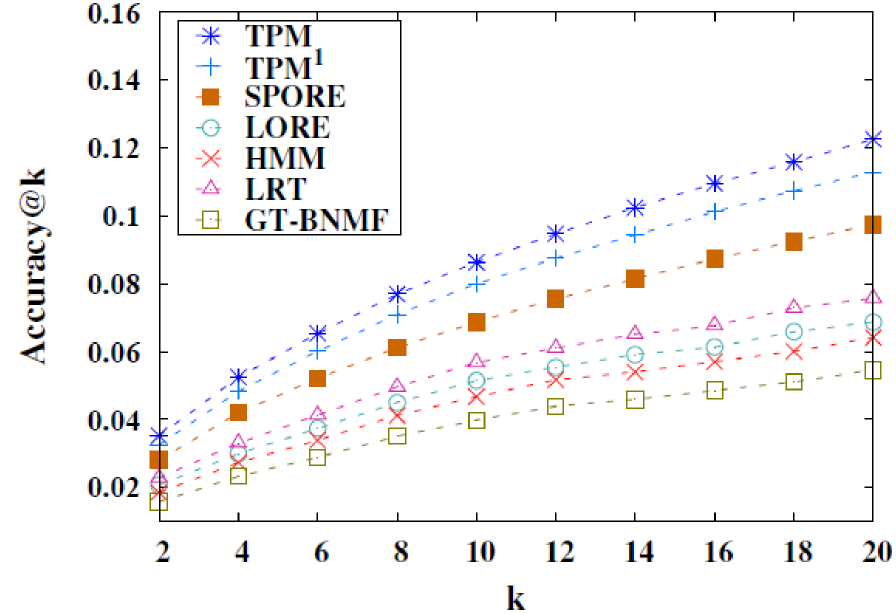
$$\boldsymbol{\theta}^{time}_{t_{u,i}} = \boldsymbol{\theta}^{year}_{t_{month}} + \boldsymbol{\theta}^{week}_{t_{weekday}} + \boldsymbol{\theta}^{day}_{t_{hour}}$$

Wang. et al. TPM: A Temporal Personalized Model for Spatial Item Recommendation (arXiv 2017).

Experimental Results



(a) Foursquare



(b) Twitter

TPM: Temporal-Sequential Model
TPM=SPORE + Temporal Influence
TPM^1 uses the hierarchical three-slice time indexing scheme (Zhao et al. AAAI16)
TPM uses our proposed additive time indexing scheme.

Short Summary

- Challenges Related with Temporal Factor
  - Sequential Effect
    - Low sampling in both time and space
    - Huge state prediction space
  - Temporal Dynamics
    - Multiple-granularity temporal cyclic patterns
    - How to integrate various temporal cyclic patterns to automatically adapt to different datasets
  - How to unify Personal Interests, Sequential Effect and Multi-granularity temporal cyclic patterns

References

- Weiqing Wang, Hongzhi Yin, Shazia Sadiq, Ling Chen, Min Xie, Xiaofang Zhou. "SPORE: A Sequential Personalized Spatial Item Recommender System". ICDE 2016.
- Weiqing Wang, Hongzhi Yin, Shazia Sadiq, Ling Chen, Xiaofang Zhou. TPM: A Temporal Personalized Model for Spatial Item Recommendation (arXiv 2017).
- Hongzhi Yin, Bin Cui, Xiaofang Zhou. "Spatio-Temporal Recommendation in Geo-Social Networks" in Reda Alhajj and Jon Rokne (eds), "Encyclopedia of Social Network Analysis and Mining". Springer, 2017
- Zhao et al. STELLAR: Spatial-Temporal Latent Ranking for Successive Point-of-Interest Recommendation. (AAAI'16)
- Cheng et al. "What's your next move: User activity prediction in location-based social networks." in SDM, 2013
- C. Cheng, H. Yang, M. R. Lyu, and I. King, "Where you like to go next: Successive point-of-interest recommendation," in IJCAI, 2013.
- J.-D. Zhang, C.-Y. Chow, and Y. Li, "Lore: Exploiting sequential influence for location recommendations," in SIGSPATIAL, 2014

Outline

- Part I: Introduction and Preliminaries
- Part II: Recommendation in Heterogeneous Information Networks
- Part III: Recommendation in a Text-Rich Setting
- Part IV: Recommendation with Spatio-Temporal Information
- Part V: Research Frontiers and Summary

Online Recommendation Efficiency

- Real-time Response
  - Given a mobile user, the naive approach to produce online top-k recommendations is to
    - first compute a score for each item
    - and then select k ones with highest scores.
    - However, when the number of available items becomes large, to produce a top-k ranked list using this brute-force method is very time-consuming and slow.
  - To support real-time recommendation in mobile scenario
    - Efficient smart online retrieval algorithms and effective indexing structures are required.

Recommended Reading

Online Recommendation Efficiency Issue:

Chapter 4: Fast Online Recommendation

To support real-time recommendation response,
Smart retrieval algorithms + effective indexing structure

Threshold based algorithm (TA)
Attribute pruning-based algorithm (AP)
Metric-tree-based search algorithm (MP)
Locality-sensitive hashing (LSH)
Asymmetric Locality-sensitive hashing (ALSH)
Learning to hash techniques (L2H)

Cross-domain Recommendation

- Traditional recommender systems suggest items belonging to a single domain
  - movies in Netflix
  - songs in Last.fm
  - POIs in Foursquare
- In reality, users provide feedback for items of different types, and express their opinions on different social media and different providers
  - e.g., Facebook, Twitter, Amazon, Netflix
- Even items (or entities) from different domains and platforms are not independent or isolated
  - Sharing some attributes, semantics or hidden factors
  - Implicit links exist between these heterogeneous items

- Can we leverage all the available personal data provided in distinct domains to generate better recommendations
  - Linking Users across domains or platforms
  - Multi-view user modeling
- Can we exploit and leverage the common attributes, semantics and other hidden knowledge across distinct domains generate better recommendations?
  - Very helpful to overcome the issue of the cold-start items

Cold-Start User in New Systems or Startup

- How to recommend items to new users?
- How to get user interest quickly?
- When new user comes, his feedback on what items can help us better understand his interest?
  - How to choose *k* items to get most of the user's interests?
    - Not very popular
    - Can represent a group of items
    - Users who like this item have different preference with users who dislike this item
    - The items that can reduce the entropy of the user's interests to the maximum extent.

Summary

- Information Network Approach to model context-rich environment
- Recommendation Techniques in Heterogeneous Information Networks
- Recommendation Techniques in Text-Rich Setting
- Recommendation Techniques with Spatio-Temporal Information

Q & A

THANK YOU!