Recommendation using Extended Paths in Complex Networks

Robin Burke
Fatemeh Vahedian

Center for Web Intelligence
School of Computing
DePaul University

Chicago, Illinois USA

DePaul University
College of Computing and Digital Media

Other contributors

- Bamshad Mobasher
- Jonathan Gemmell
- Thomas Schimoler
- Yong Zheng

Ad for our group's other presentations:
"Incorporating Context Correlation into Context-aware Matrix Factorization" IP workshop, Monday
"Adapting Recommendations to Contextual Changes Using Hierarchical Hidden Markov Models"
Main conference, session SIST3, Friday

Outline

- Historical background
  - Social tagging systems
- Multi-component hybrid using metapaths
- Multi-relational matrix factorization

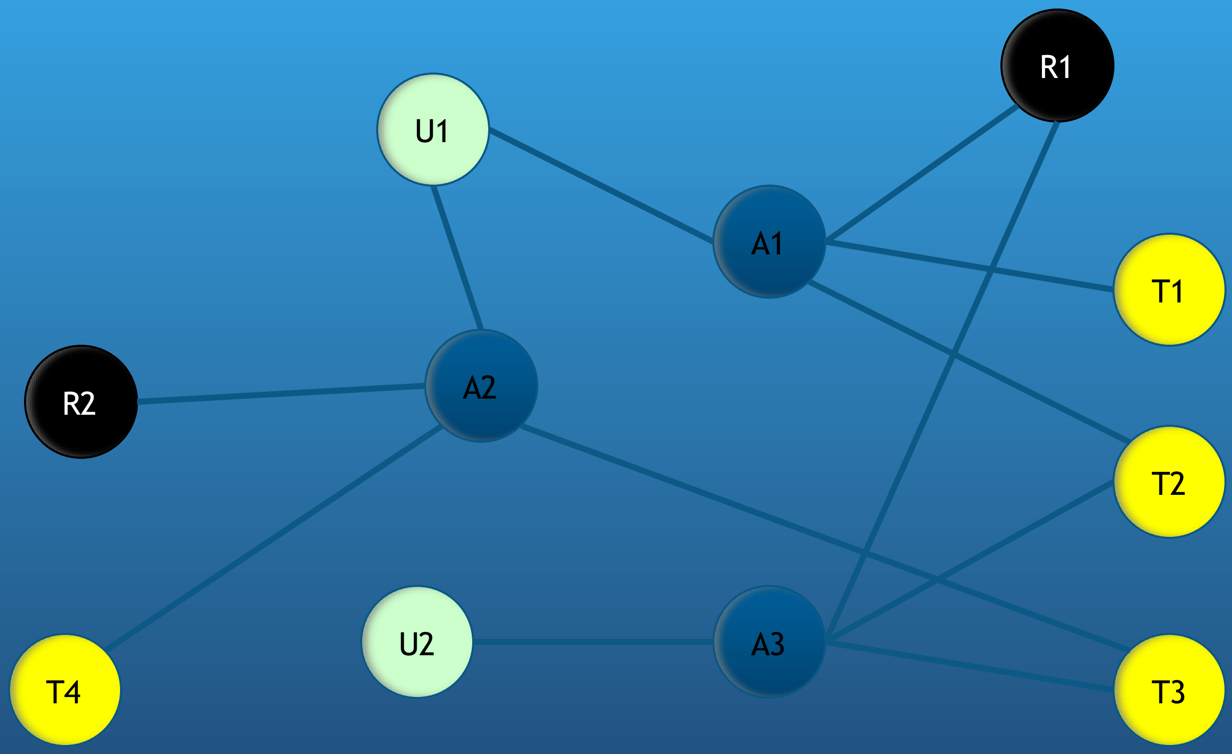
Heterogeneous networks

- No explanation needed for this audience
- A variety of data sets in this work
  - social tagging systems
    - users, resources, tags
  - social media sites
    - users, businesses, locations, categories (Yelp)
  - informal education
    - students, schools, organizations, programs, offerings
  - scientific publications
    - authors, publications, venues, series
  - commercial
    - users, employers, job ads, applications, schools, etc.

Social Tagging Research

- (Gemmell, et al. 2009, 2011)
- Users apply tags to resources
- Examples
  - delicious.com
  - Amazon.com
  - Last.fm

As a network



Recommendation Options

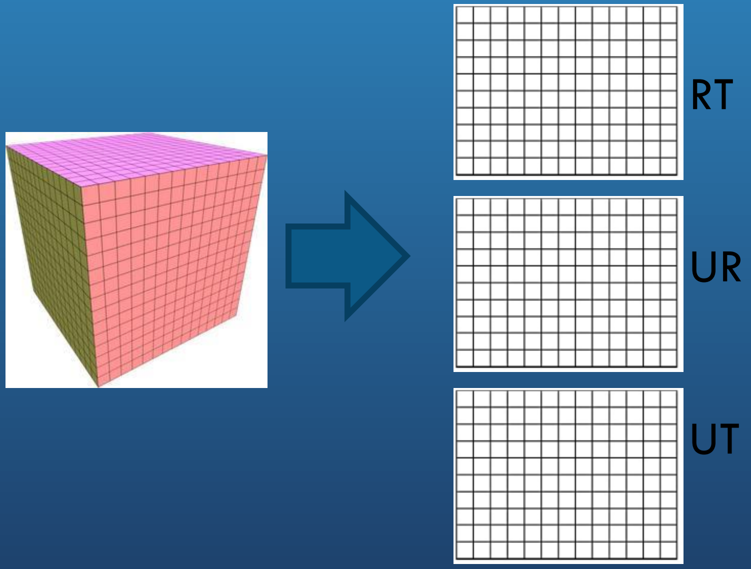
Input	Output
User	Similar users Recommended resources Recommended tags
User, Tag	Recommended resources
User, Resource	*Recommended tags*

Resource Recommendation

- Given a user
  - what resources to recommend
    - most analogous to “normal” recommendation
    - but little studied at the time

Two-Dimensional Projections

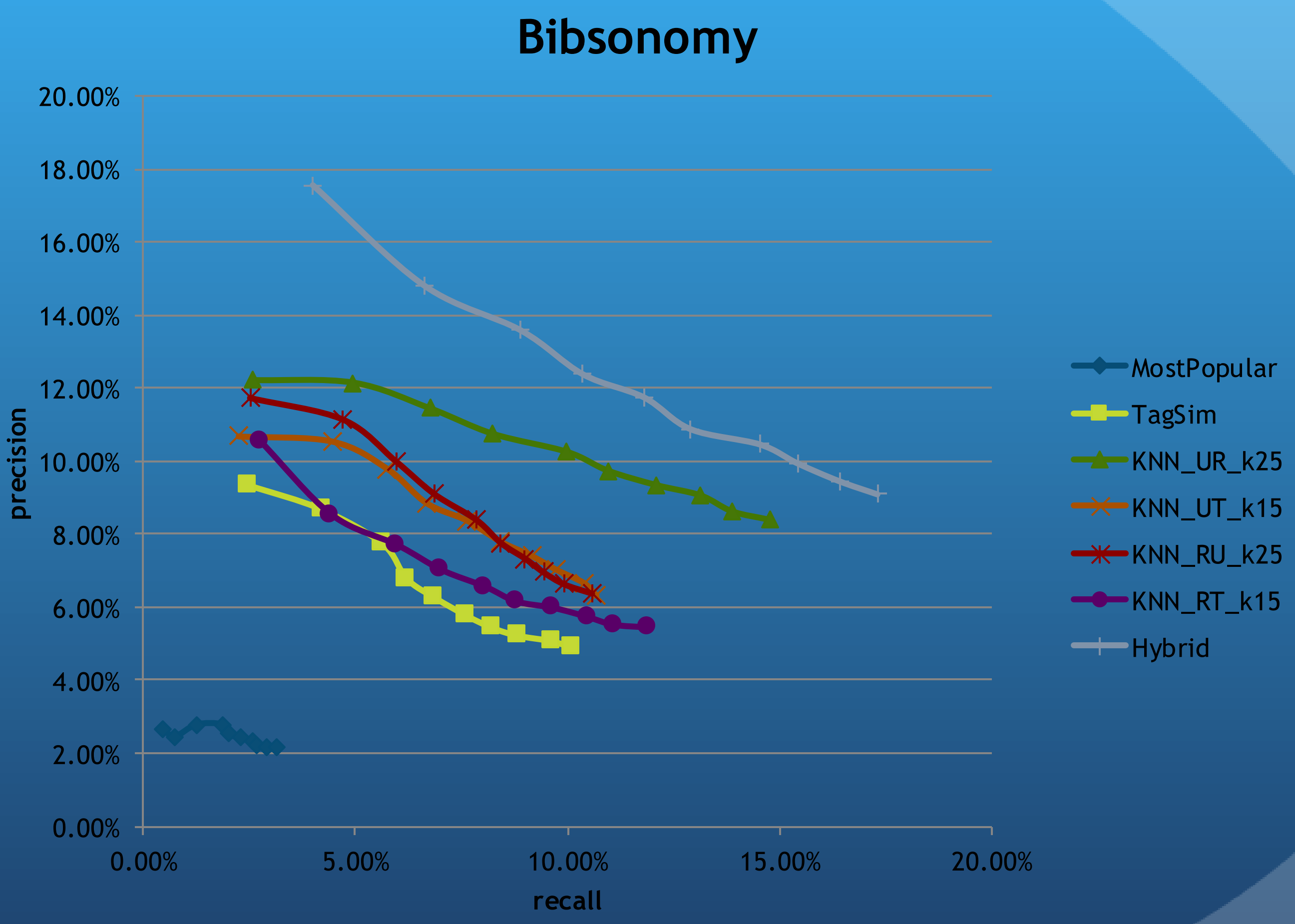
$$RT_{tf}(r,t) = |\{a = \langle u, r, t\rangle \in A : u \in U\}|$$



Approach

- Build weighted hybrids
  - Incorporate all reduced dimension views
- Individual predictions P_i
  - scaled to 0..1 scale
  - weighted by α_i
  - α values sum to 1
  - combined to overall prediction P*
- Learn α values through optimization

Typical results



Learned weights

Dataset	Pop	TagSim	kNN_{ur}	kNN_{ut}	kNN_{ru}	kNN_{rt}
Amazon	0.053	0.254	0.419	0.001	0.131	0.147
Bibsonomy	0.01	0.023	0.431	0.020	0.209	0.307
Delicious	0.004	0.263	0.512	0.069	0.119	0.033
LastFM	0.006	0.153	0.410	0.005	0.425	0.001

kNNur weights similar across datasets

kNNrt is inconsistent

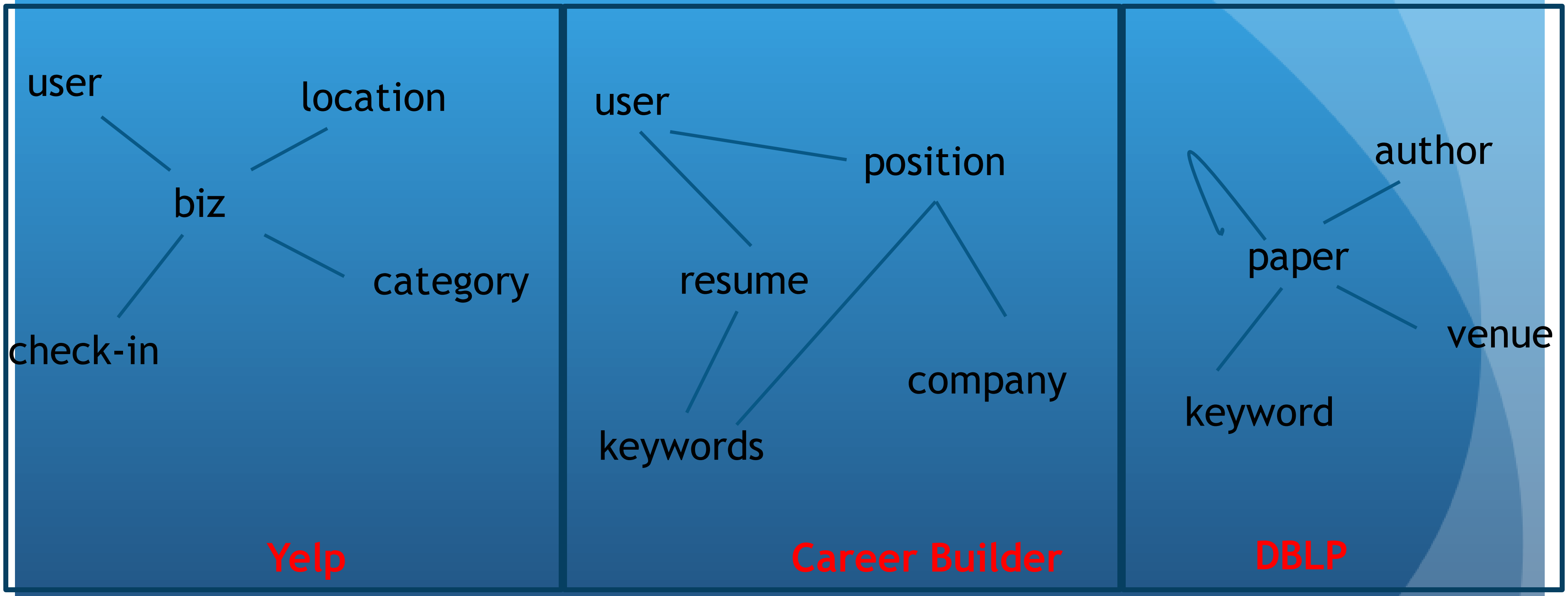
Key findings

- Hybrid always does better than any single component
- kNN_{ur} also does well
  - makes sense since we are using users to predict resources
- kNN_{ru} and kNN_{rt} inconsistent
  - compare Bibsonomy and LastFM
  - tags in LastFM are not good descriptors for resources
- Not shown
  - Hybrid performs better than the well-known PITF algorithm
    - for tag recommendation

Extending to heterogeneous networks

- (Burke and Vahedian, 2013; Burke, et al. 2014)
- More types of nodes
- More types of edges
- Possible edges between nodes of the same type
- Increased complexity but application-defined structure

Examples

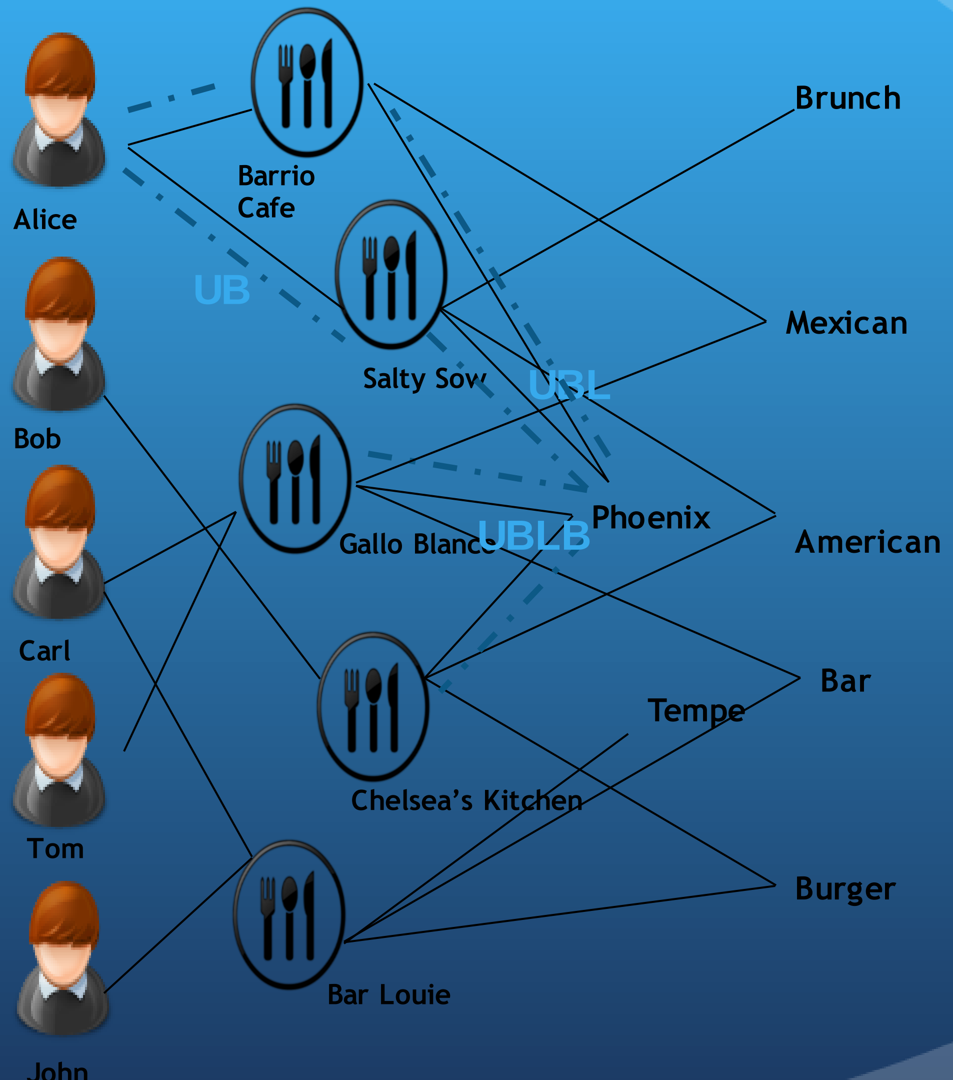


user
location
biz
category
check-in
Yelp
user
position
resume
company
keywords
Career Builder
author
paper
venue
keyword
DBLP

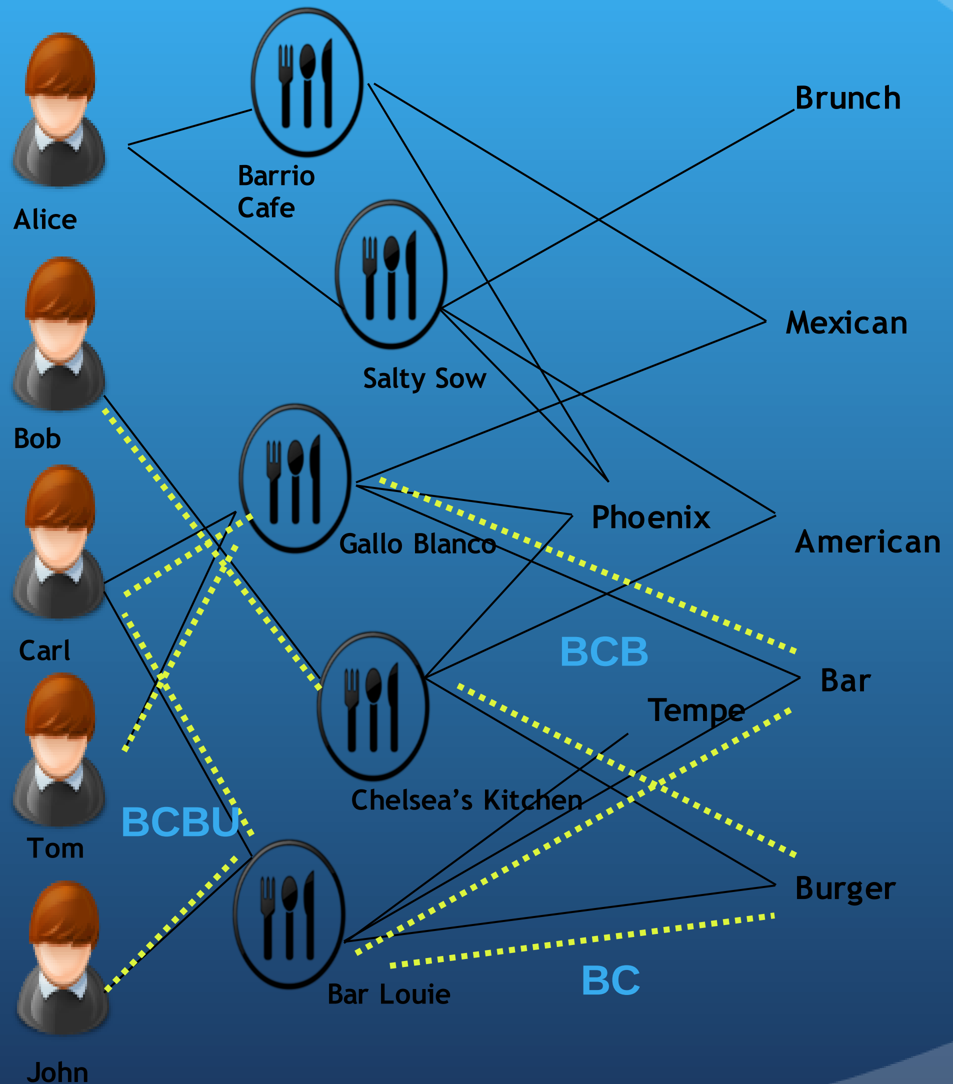
Meta-paths

- In a heterogeneous network
  - many choices for how to represent a user's profile
    - in terms of items preferred
    - in terms of tags given to items
    - in terms of tags supplied by all users for their preferred
    - etc.
- Represent all such options as meta-paths
  - classes of paths through the network
  - each link follows a characteristic typed edge from one node type to another

Meta-path example (UBLB)



Meta-path example (BCBU)



WHyLDR

- Weighted Hybrid of Low-Dimensional Recommenders
  - Take the weighted hybrid approach from tag recommendation
  - Build two-dimensional components using meta-paths,
    - can be multiple steps through the network
    - instead of the one-step relations used in tagging work

Problem: Unbounded

- Component generation is an unbounded process
  - Expensive
  - Not efficient
- Some components make only a minor contribution
- Weight optimization process is slowed by adding components
- Solution: estimate component utility using information gain

Computing Information Gain 1

- Start with probability
  - p(a) = probability of encountering node a (among the other nodes in class A)
  - = probability of a random walk encountering a
  - = (as length of walk -> ∞) degree of a
    - relative to other nodes in A

$$p(a) = \frac{Degree(a)}{\sum_{n \in A} Degree(n)}$$

Computing information gain 2

- Entropy of dimension A

$$H(A) = \sum_{a \in A} -p(a)\log(p(a))$$

$p(b|a)$

- Entropy of dimension A given B

$$H(A|B) = \sum_{a \in A} -p(a|b)\log(p(a|b))$$

$$p(a|b) = \frac{|paths(b \rightarrow a)|}{\sum_{n \in A, m \in B} |paths(m,n)|}$$

Information gain

- $G(A,B) = H(A) - H(A|B)$
- If the gain is small
  - $H(A)$ and $H(A|B)$ are close
- This means that knowing B
  - does not decrease the entropy of A
- Example
  - knowing that a song is tagged “rock”
  - doesn’t decrease its entropy across user profiles in Last.fm
    - because the tag is used so loosely for almost everything

Example: Yelp

- A = users
- B = restaurants

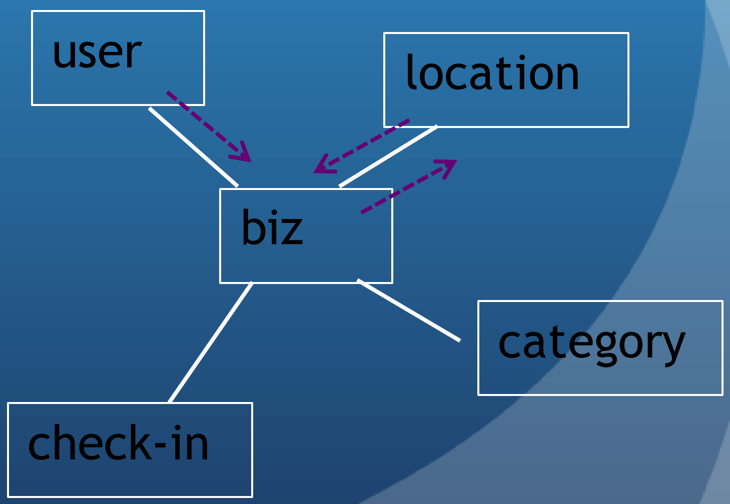
User profile			
	Salty Sow	Barrio Cafe	Bar Louie
Bob	1	2	0
Alice	2	0	1

p(Bob)=1/2

p(Salty Sow|Bob)=1/3

Meta-paths in Yelp

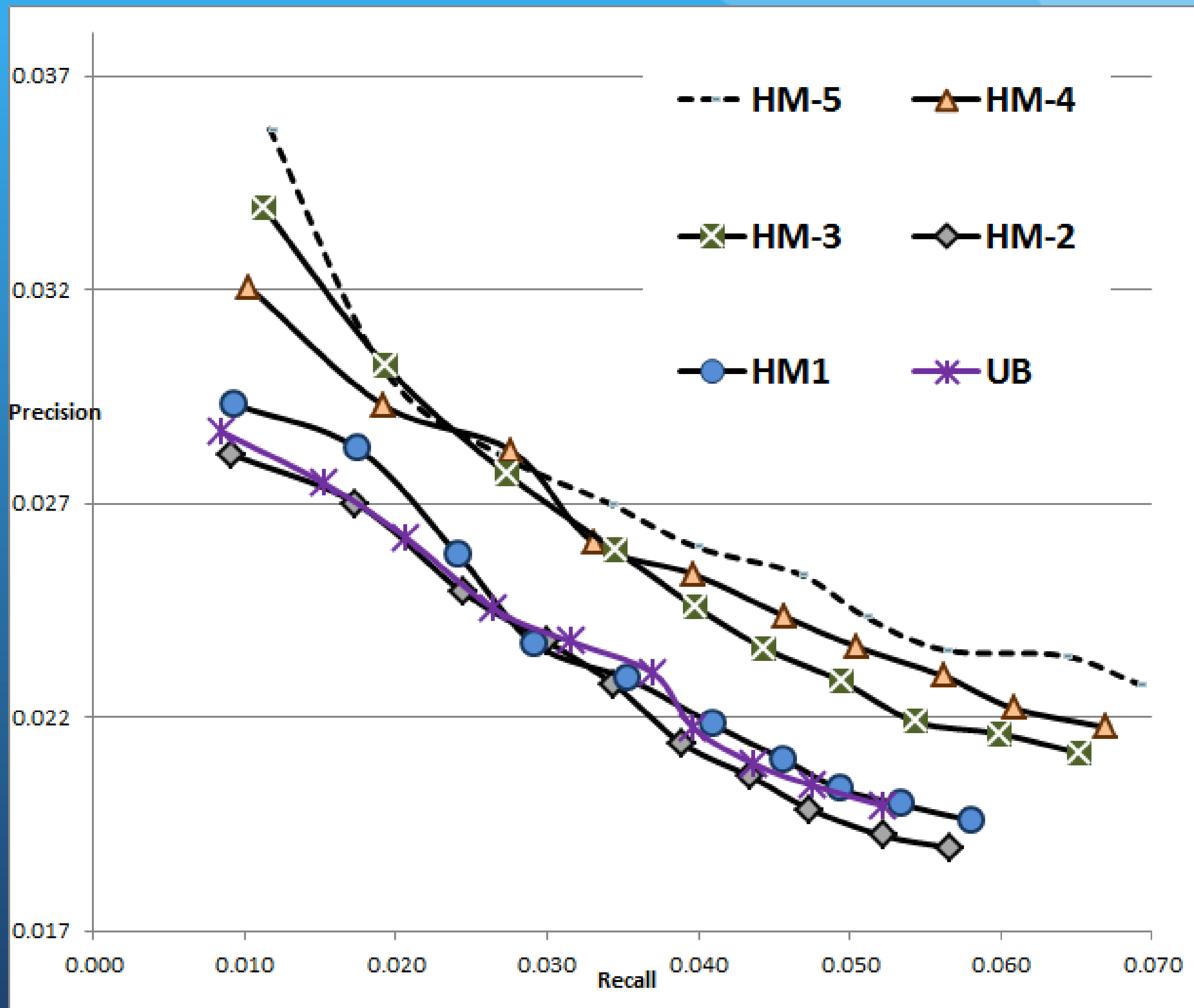
Type	Meta-path
User-based	User-biz User-biz-category User-biz-category-biz User-biz-location User-biz-location-biz
Item-based	Biz-category Biz-user Biz-user-biz-category



Hybrids

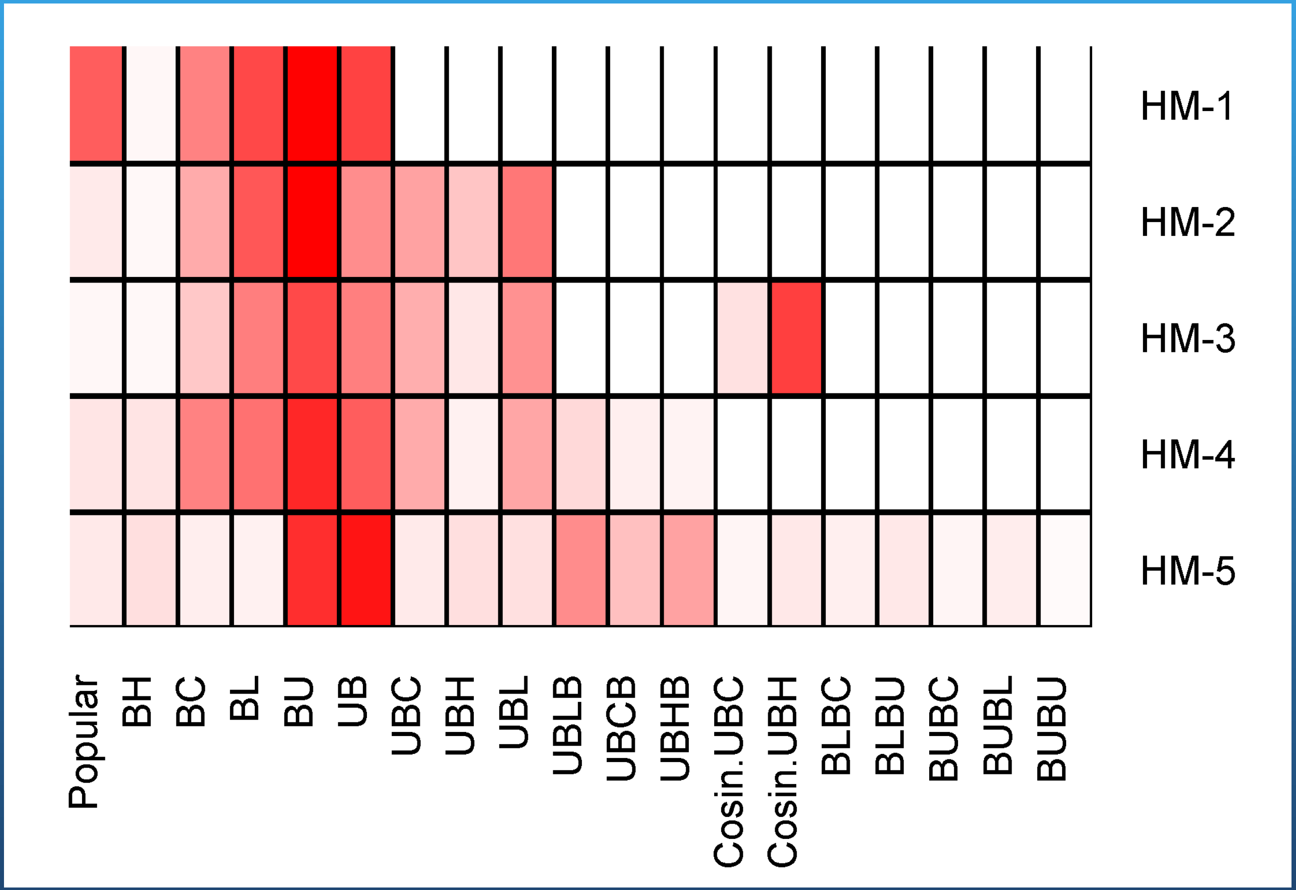
- **HM-1**: User-based and item-based, paths of length 1 plus popularity
- **HM-2**: HM-1 plus user-based and item-based, paths of lengths 2
- **HM-3**: HM-2 plus cosine, paths of length 2
- **HM-4**: HM-3 plus user-based, paths of length 3
- **HM-5**: HM-4 plus item-based, paths of length 3

Results



HM-5
HM-4
HM-3
HM-2
HM1
UB
Precision
0.037
0.032
0.027
0.022
0.017
0.000
0.010
0.020
0.030
0.040
0.050
0.060
0.070
Recall

Component contribution



Correlation

- Information gain vs learned weights

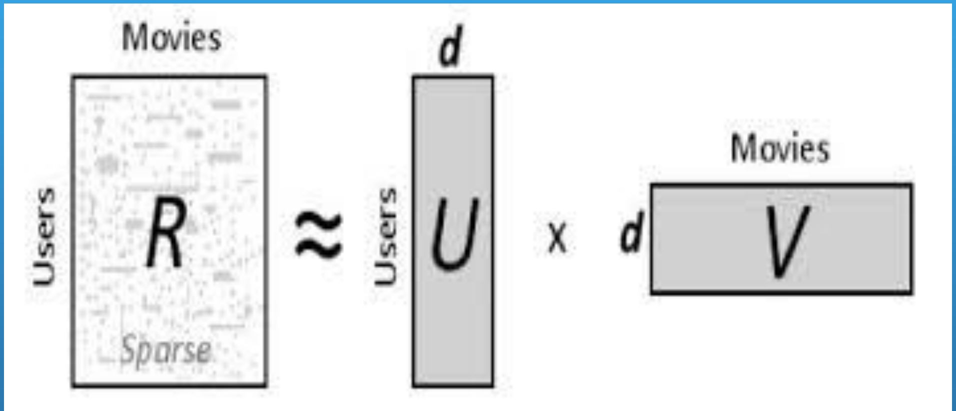
Model	HM-1	HM-2	HM-3	HM-4	HM-5
Correlation	0.788	0.523	0.587	0.90	0.627

- Other work
  - demonstrated that IG could be used to prune the set of components
  - improved learning time
  - without loss of accuracy

Alternative recommendation model

- Multi-Relational Matrix Factorization (Drummond, 2014)
  - assume target relation (i.e. user – business)
  - and auxiliary relations (i.e. business – category)
- Learn the factorization model parameters
  - by optimizing the sum over the loss functions on each relation
  - auxiliary relations act as regularization terms

Single relation
(from Krohn-Grimberghe, et al. 2012)



$$f(\Theta) := L_r(D_r, \hat{y}_r(D_r;\Theta)) + \lambda||\Theta||_2^2$$

Multiple relations

$$(\varphi^*(U), \Phi^*_{UM}) = argmin$$
$$L_{UM}(D_{UM}, \hat{y}_{UM}(.; \varphi_{UM}(U), \Phi_{UM}))+$$
$$\sum_{i=1}^{R} \alpha_{UM,MP_i} L_{MP_i}(D_{MP_i}, \hat{y}_{UM,MP_i}(.; \varphi_{UM}(U), \Phi_{UM,MP_i}))$$

- Note that relations need not be direct associations
- Can be generated by meta-paths
  - as in our weighted hybrid work

DMF / CATSMF (Drummond, et al. 2014)

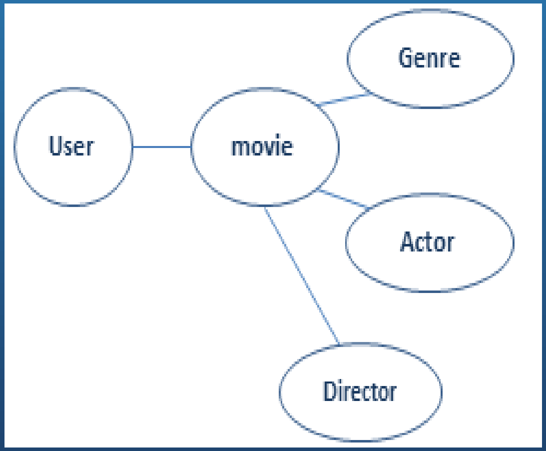
- **DMF** (*Decoupled Target Specific Features Multi-Target Factorization)*
  - different latent feature models are defined for each relation
  - factorization process in such a way that they are optimized for the best performance on each relation individually
- **CATSMF** (*Coupled Auxiliary and Target Specific Features Multi-Target Factorization)*
  - proposed to improve the efficiency of the DMF model when applied to multiple targets
  - better accuracy than DMF in some domains

Methodology

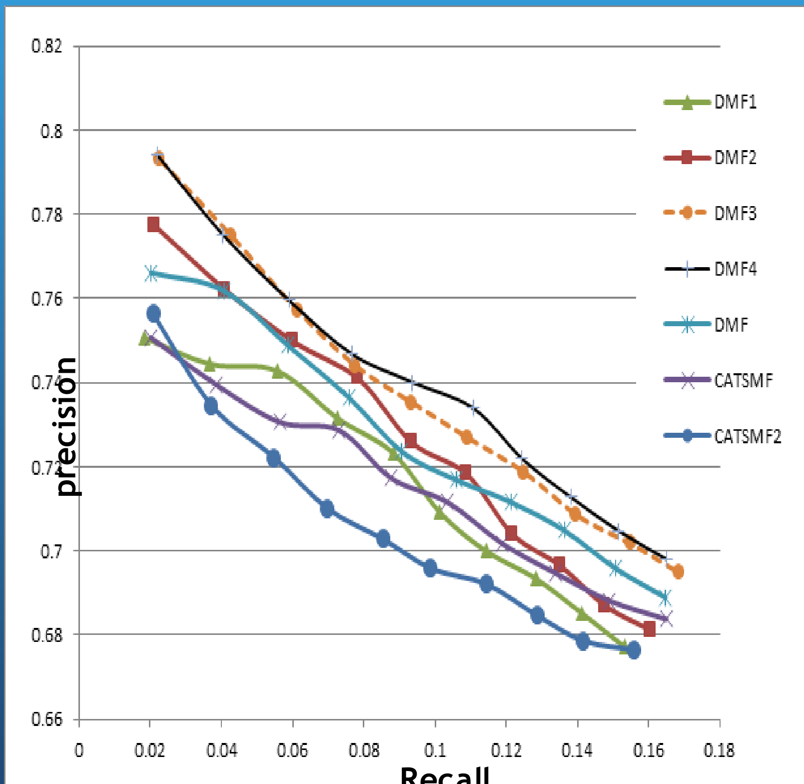
- Use 80% of the data as training set and the rest as test set
  - All meta-paths are generated based on training data
  - 2 step and 3 step versions
- Optimize the factorization model using BPR as loss function
- Generate the list of 10 recommendations
- Measure the recall and precision for top 10 recommendations

MovieLens Dataset Experiments

- Target relation is UM
- The user profile paths: UM, UMA, UMG, UMD, UMGM, UMDM, UMAM
- The item profile paths: MG, MD, MA

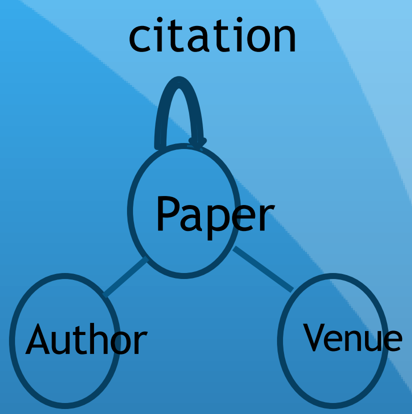


Movie Recommendation Results



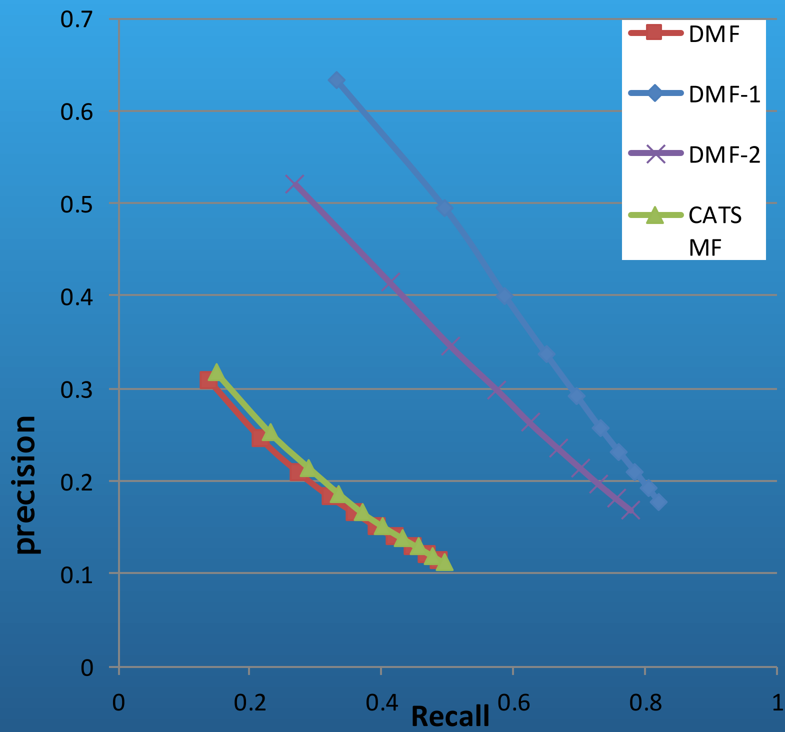
	um	ma	md	mg	uma	umd	umg	umam	umdm	umgm
DMF										
DMF1										
DMF2										
DMF3										
DMF4										
CATSMF										
CATSMF 2										

DBLP dataset Experiment



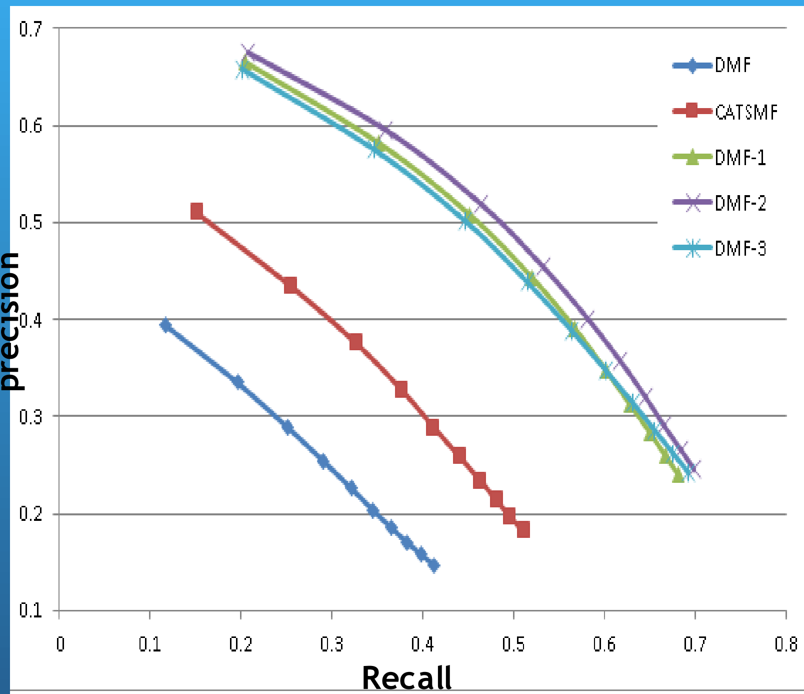
- Venue Recommendation to Author
  - APV is the target relation
  - Direct links: paper-author, paper-citation, paper-venue
  - Meta-paths: Author-paper-Author, Author-paper-citation
- Citation Recommendation
  - Paper-citation is target relation
  - Direct links: paper-author, paper-citation, paper-venue
  - Meta-paths: paper-citation-venue

Venue Recommendation Results



	APV	PA	PV	PC	APC	APA	VPA	VPC	APCA	APCV	APAP	APVP	VPAP	VPCP
DMF-1														
DMF-2														
DMF-3														

Citation-Recommendation Results



	pc	pv	pa	cpc	cpv	cpa	pap	pcp	pca	pcv	papa	papc	papv	pcpv	pcpa
DMF															
DMF-1															
DMF-2															
DMF-3															

Conclusions

- A heterogeneous network approach is valuable for recommendation
  - distant relations through the network can add accuracy
  - (and sometimes diversity)
- Examples
  - weighted hybrid
  - multi-relational factorization
- Information gain
  - correlates with component / relation utility
  - but is probably too simple
    - not sensitive to recommendation task
  - is also computationally intensive

Future work

- Studying information gain-based pruning in multi-relational models
- Better relation / component utility metric
- MRF vs weighted hybrid
  - factorization is not always better
  - when / why

Conclusion and Future work

- Recommendation using multi-relational matrix factorization in networked data can be enhanced through in the inclusion of relations derived from meta-path expansions
- Longer meta-paths are not always good
- Future work
  - Predicting the usefulness of generated meta-paths
  - Weighted meta-paths