Affiliation Recommendation using Auxiliary Social Networks

August 19, 2010

Outline

What to look out for?

What to look out for?

- Affiliation recommendation problem.

What to look out for?

- Affiliation recommendation problem.
- Affiliation recommendation algorithms.

Outline

Outline

Social networks

[1] TODO: Pictures of orkut, facebook, yeast network.

Social networks

- **[TODO]]** [1]

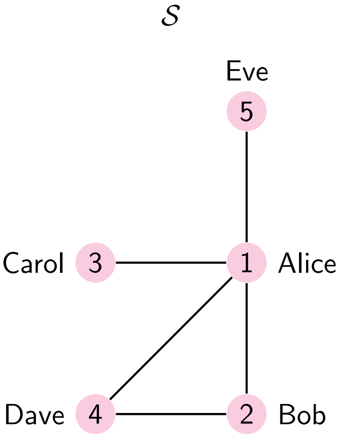
[1] TODO: Pictures of orkut, facebook, yeast network.

Social networks

- **[TODO]**] [1]
- Not necessarily among people - Yeast gene network.

[1] TODO: Pictures of orkut, facebook, yeast network.

Social network S: An undirected graph.



Affiliation networks

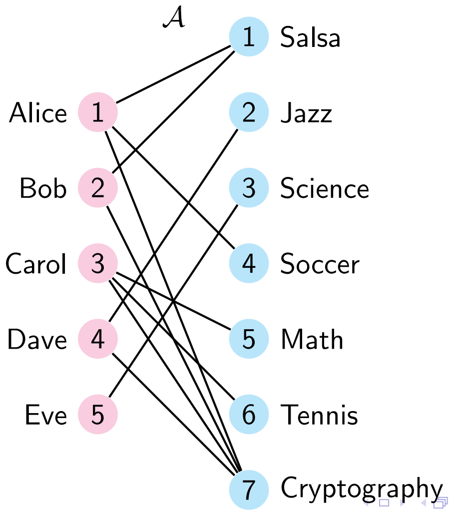
Affiliation networks

- Pictures of orkut and facebook communities.

Affiliation networks

- Pictures of orkut and facebook communities.
- Not necessarily among people - gene-disease network.

Affiliation network A: A bipartite graph.



Outline

Problems in social network analysis.

Problems in social network analysis.

- Modelling network evolution.

Problems in social network analysis.

- Modelling network evolution.
- Link prediction.

Problems in social network analysis.

- Modelling network evolution.
- Link prediction.
- Community identification.

Affiliation Recommendation.

Affiliation Recommendation.

- Exploiting social network in making affiliation recommendations.

Affiliation Recommendation.

- Exploiting social network in making affiliation recommendations.
- Generalizable to the item recommendation problem.

Outline

Outline

The combined network

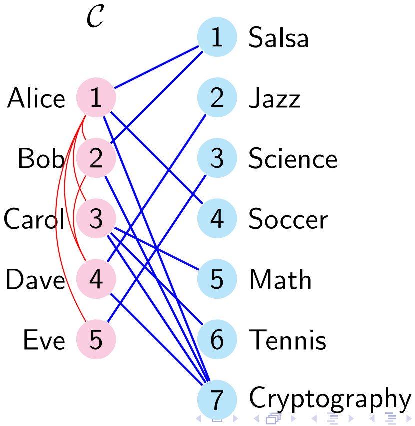
The combined network

- λ: relative weight associated with information in S.

The combined network

- λ: relative weight associated with information in S.
- $\mathbf{D}$: unobserved.

$$\mathbf{C}(\lambda, \mathbf{D}) = \begin{bmatrix} \lambda S & A \\ A^T & \mathbf{D} \end{bmatrix}$$



Outline

Modelling A alone

Modelling A alone

- User-group affinity as a product of low dimensional vectors: $\mathbf{A}_{i,j} \approx \langle \mathbf{U}(i,:), \mathbf{G}(i,:) \rangle$.

Modelling A alone

- User-group affinity as a product of low dimensional vectors: $\mathbf{A}_{i,j} \approx \langle \mathbf{U}(i,:), \mathbf{G}(i,:) \rangle$.
- $\mathbf{A} \approx \mathbf{UG}$, $rank(\mathbf{U}) \leq k$, $rank(\mathbf{G}) \leq k$. (U as user preferences, G as group properties.)

Modelling A alone

- User-group affinity as a product of low dimensional vectors: $\mathbf{A}_{i,j} \approx \langle \mathbf{U}(i,:), \mathbf{G}(i,:) \rangle$.
- $\mathbf{A} \approx \mathbf{UG}, rank(\mathbf{U}) \leq k, rank(\mathbf{G}) \leq k$. (U as user preferences, G as group properties.)
- For user j, recommend affiliations with high affinity.

Modelling C

Modelling C

- A **good model** will account for edges in **S** too.
$\mathbf{C}(\lambda, \mathbf{D}) \approx \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \end{bmatrix} \mathbf{L} \left[\mathbf{V}_1^T \mathbf{V}_2^T \right]$, rank of $\mathbf{V}_i$ and $\mathbf{L} \leq k$.

Modelling C

- A **good model** will account for edges in $\mathbf{S}$ too.
  $\mathbf{C}(\lambda, \mathbf{D}) \approx \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \end{bmatrix} \mathbf{L} \left[\mathbf{V}_1^T \mathbf{V}_2^T \right]$, rank of $\mathbf{V}_i$ and $\mathbf{L} \leq k$.
- So $\mathbf{A} \approx \mathbf{V}_1 \mathbf{L} \mathbf{V}_2^T$.

Modelling C

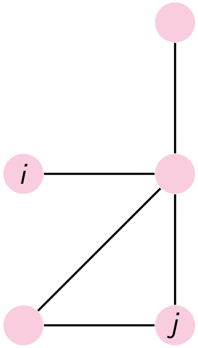
- A **good model** will account for edges in $\mathbf{S}$ too. $\mathbf{C}(\lambda, \mathbf{D}) \approx \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \end{bmatrix} \mathbf{L} \left[\mathbf{V}_1^T \mathbf{V}_2^T \right]$, rank of $\mathbf{V}_i$ and $\mathbf{L} \leq k$.
- So $\mathbf{A} \approx \mathbf{V}_1 \mathbf{L} \mathbf{V}_2^T$.
- $\mathbf{V}_1 \mathbf{L} \mathbf{V}_2^T$ is a similarity score matrix for ranking potential affiliations.

Outline

Proximity between users in a social network

$\mathbf{C}^2_{i,j}$: Number of paths of length 2 between i and j.

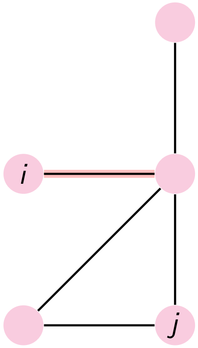
$$\mathcal{C} \rightarrow \mathcal{C}^2$$



Proximity between users in a social network

$\mathbf{C}^2_{i,j}$: Number of paths of length 2 between i and j.

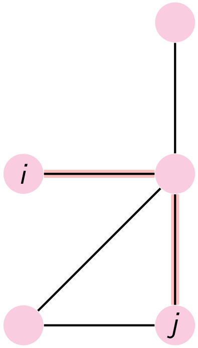
$$\mathcal{C} \rightarrow \mathcal{C}^2$$



Proximity between users in a social network

$\mathbf{C}^2_{i,j}$: Number of paths of length 2 between i and j.

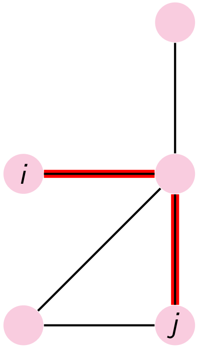
$$\mathcal{C} \rightarrow \mathcal{C}^2$$



Proximity between users in a social network

$\mathbf{C}^2_{i,j}$: Number of paths of length 2 between i and j.

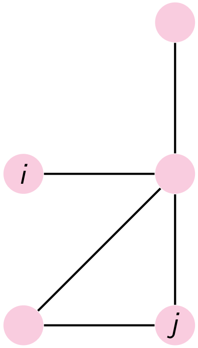
$$\mathcal{C} \rightarrow \mathcal{C}^2$$



Proximity between users in a social network

$\mathbf{C}^3_{i,j}$: Number of paths of length 3 between i and j.

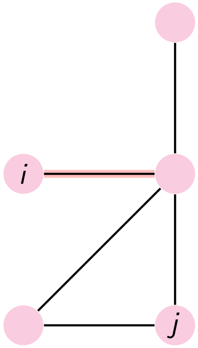
$$\mathcal{C} \rightarrow \mathcal{C}^3$$



Proximity between users in a social network

$\mathbf{C}^3_{i,j}$: Number of paths of length 3 between i and j.

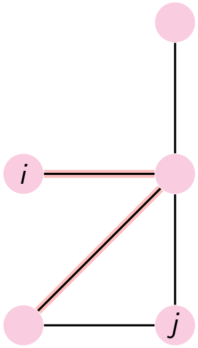
$$\mathcal{C} \rightarrow \mathcal{C}^3$$



Proximity between users in a social network

$\mathbf{C}^3_{i,j}$: Number of paths of length 3 between i and j.

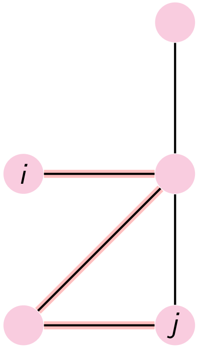
$$\mathcal{C} \rightarrow \mathcal{C}^3$$



Proximity between users in a social network

$\mathbf{C}^3_{i,j}$: Number of paths of length 3 between i and j.

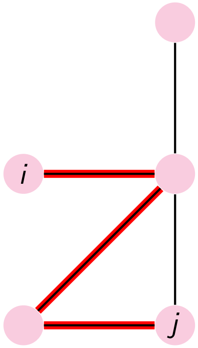
$$\mathcal{C} \rightarrow \mathcal{C}^3$$



Proximity between users in a social network

$\mathbf{C}^3_{i,j}$: Number of paths of length 3 between i and j.

$$\mathcal{C} \rightarrow \mathcal{C}^3$$



Graph proximity Model

Graph proximity Model

- $\text{Proximity}(i,j) = \beta^2 \mathbf{C}_{i,j}^2 + \beta^3 \mathbf{C}_{i,j}^3 + \ldots$

Graph proximity Model

- $\text{Proximity}(i,j) = \beta^2 \mathbf{C}^2_{i,j} + \beta^3 \mathbf{C}^3_{i,j} + \ldots$
- Known as Katz measure, when the series is convergent ($\|\beta C\|_2 < 1$).

Graph proximity Model

- Proximity$(i, j) = \beta^2 \mathbf{C}^2_{i,j} + \beta^3 \mathbf{C}^3_{i,j} + \dots$
- Known as Katz measure, when the series is convergent ($\|\beta C\|_2 < 1$).
- To save time, compute the truncated Katz measure, tKatz$(\mathbf{C}, \beta, k) = \sum_{i=1}^{k} \beta^i \mathbf{C}^i$.

Graph proximity Model

- Proximity$(i, j) = \beta^2 \mathbf{C}^2_{i,j} + \beta^3 \mathbf{C}^3_{i,j} + \dots$
- Known as Katz measure, when the series is convergent ($\|\beta C\|_2 < 1$).
- To save time, compute the truncated Katz measure, tKatz$(\mathbf{C}, \beta, k) = \sum_{i=1}^{k} \beta^i \mathbf{C}^i$.
- Recommend user-group affiliations based on proximity in $\mathbf{C}$.

User-group proximity: Paths considered

User-group proximity: Paths considered

- user $i \xrightarrow{\mathbf{S}}$ user $j \xrightarrow{\mathbf{A}}$ group n (in $\mathbf{C}^2$)

User-group proximity: Paths considered

- user $i \xrightarrow{\mathbf{S}}$ user $j \xrightarrow{\mathbf{A}}$ group n (in $\mathbf{C}^2$)
- user $i \xrightarrow{\mathbf{S}}$ user $j \xrightarrow{\mathbf{AA}^T} k \xrightarrow{\mathbf{A}}$ group n (in $\mathbf{C}^4$)

Outline

Scalable recommenders for the graph proximity model

Scalable recommenders for the graph proximity model

- $\text{tKatz}(\mathbf{C}, \beta, k) = \sum_{i=1}^{k} \beta^i \mathbf{C}^i$.

Scalable recommenders for the graph proximity model

- $\text{tKatz}(\mathbf{C}, \beta, k) = \sum_{i=1}^{k} \beta^i \mathbf{C}^i$.
- $\mathbf{C}^i$ gets denser, so hard to compute.

Scalable recommenders for the graph proximity model

- $\mathrm{tKatz}(\mathbf{C}, \beta, k) = \sum_{i=1}^{k} \beta^i \mathbf{C}^i$.
- $\mathbf{C}^i$ gets denser, so hard to compute.
- Use Low rank approximations of $\mathbf{C} \approx \mathbf{V}\Lambda^i\mathbf{V}^T$. Use $\mathbf{C}^i \approx \mathbf{V}\Lambda^i\mathbf{V}^T$. [3]

Common Subspace Approximation

Common Subspace Approximation

- $\text{tKatz}(\mathbf{C}; \beta, 3)_{12} = \beta\mathbf{A} + \beta^2\lambda\mathbf{SA} + \beta^3(\lambda^2\mathbf{S}^2\mathbf{A} + \mathbf{AA}^T\mathbf{A})$.

Common Subspace Approximation

- $\text{tKatz}(\mathbf{C}; \beta, 3)_{12} = \beta\mathbf{A} + \beta^2\lambda\mathbf{S}\mathbf{A} + \beta^3(\lambda^2\mathbf{S}^2\mathbf{A} + \mathbf{A}\mathbf{A}^T\mathbf{A})$.
- $(\mathbf{A}\mathbf{A}^T)^i, \mathbf{S}^i, (\mathbf{A}\mathbf{A}^T)^j\mathbf{S}^i$ gets denser, so hard to compute.

Common Subspace Approximation

- $\mathrm{tKatz}(\mathbf{C}; \beta, 3)_{12} = \beta\mathbf{A} + \beta^2\lambda\mathbf{S}\mathbf{A} + \beta^3(\lambda^2\mathbf{S}^2\mathbf{A} + \mathbf{A}\mathbf{A}^T\mathbf{A})$.
- $(\mathbf{A}\mathbf{A}^T)^i, \mathbf{S}^i, (\mathbf{A}\mathbf{A}^T)^j\mathbf{S}^i$ gets denser, so hard to compute.
- Use $\mathbf{S} \approx \mathbf{Q}(\mathbf{Q}^T\mathbf{S}\mathbf{Q})\mathbf{Q}^T \equiv \mathbf{Q}\mathbf{D}_\mathbf{S}\mathbf{Q}^T, \mathbf{A} \approx \mathbf{Q}(\mathbf{Q}^T\mathbf{A}\mathbf{V})\mathbf{V}^T \equiv \mathbf{Q}\mathbf{D}_\mathbf{A}\mathbf{V}^T$.

Common Subspace Approximation

- $\text{tKatz}(\mathbf{C}; \beta, 3)_{12} = \beta\mathbf{A} + \beta^2\lambda\mathbf{S}\mathbf{A} + \beta^3(\lambda^2\mathbf{S}^2\mathbf{A} + \mathbf{A}\mathbf{A}^T\mathbf{A})$.
- $(\mathbf{A}\mathbf{A}^T)^i, \mathbf{S}^i, (\mathbf{A}\mathbf{A}^T)^j\mathbf{S}^i$ gets denser, so hard to compute.
- Use $\mathbf{S} \approx \mathbf{Q}(\mathbf{Q}^T\mathbf{S}\mathbf{Q})\mathbf{Q}^T \equiv \mathbf{Q}\mathbf{D}_\mathbf{S}\mathbf{Q}^T, \mathbf{A} \approx \mathbf{Q}(\mathbf{Q}^T\mathbf{A}\mathbf{V})\mathbf{V}^T \equiv \mathbf{Q}\mathbf{D}_\mathbf{A}\mathbf{V}^T$.
- Now $(\mathbf{A}\mathbf{A}^T)^i \approx \mathbf{Q}(\mathbf{D}_\mathbf{A}\mathbf{D}_\mathbf{A}^T)^i\mathbf{Q}^T$, $(\mathbf{S})^i \approx \mathbf{Q}\mathbf{D}_\mathbf{S}^i\mathbf{Q}^T$, $(\mathbf{S}\mathbf{A}\mathbf{A}^T) \approx \mathbf{Q}\mathbf{D}_\mathbf{S}(\mathbf{D}_\mathbf{A}\mathbf{D}_\mathbf{A}^T)\mathbf{Q}^T$.

Outline

Outline

Data sets

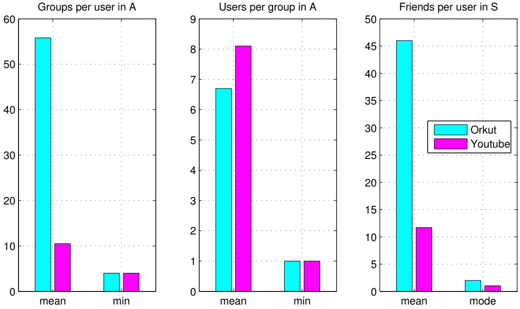


Figure: Statistics of extracted social and affiliation networks: *Orkut* and *Youtube* data sets [1]; Orkut: $N_u = 9123$, $N_g = 75546$. Youtube: $N_u = 16575$, $N_g = 21326$.

Outline

1. Outline
2. Social network analysis
   - Social and affiliation networks
   - Social network analysis: Problems
3. Recommendation algorithms
   - Modelling user-affiliation affinity
   - Latent factors model
   - Graph proximity model
   - Scalability for graph proximity model
4. Experiments
   - Datasets and their statistics
   - Evaluation methods
   - Results and discussion
5. Conclusion

Evaluation methods

Evaluation methods

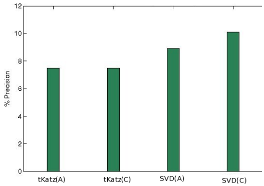
- Choosing appropriate evaluation method — Not obvious!

Evaluation methods

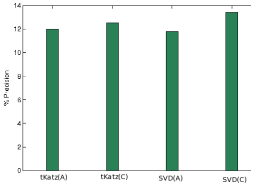
- Choosing appropriate evaluation method — Not obvious!
- "Global" sensitivity vs "Per-user" sensitivity.

Evaluation methods

- Choosing appropriate evaluation method — Not obvious!
- "Global" sensitivity vs "Per-user" sensitivity.



(a) Orkut dataset

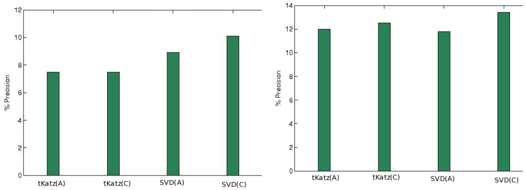


(b) Youtube dataset

Figure: Comparison of recommendation algorithms using "global sensitivity".

Evaluation methods

- Choosing appropriate evaluation method — Not obvious!
- "Global" sensitivity vs "Per-user" sensitivity.



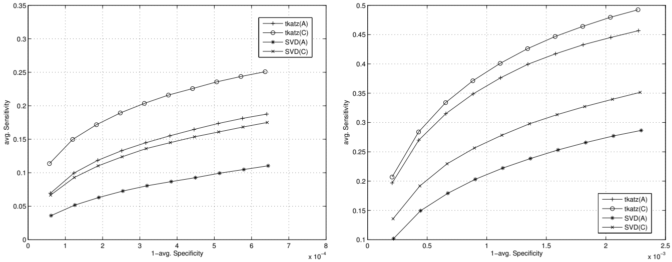
(a) Orkut dataset (b) Youtube dataset

Figure: Comparison of recommendation algorithms using "global sensitivity".

- Consider the top 50 recommendations made for a user.

Outline

Results



(a) Orkut dataset (b) Youtube dataset

Figure: Comparison of recommendation algorithms using "Per-user" sensitivity.

Results

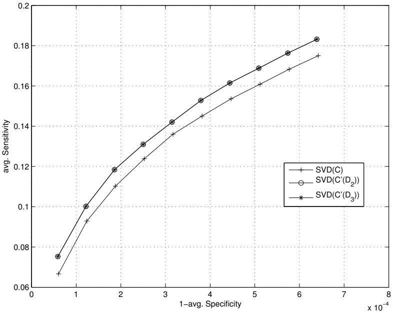


Figure: Comparison of latent factors based algorithms for various choices of $\mathbf{D}$, for the Orkut dataset: $\mathbf{D}_2 = \mathbf{A}^T\mathbf{A}, \mathbf{D}_3 = \lambda\mathbf{A}^T\mathbf{A}$.

Results

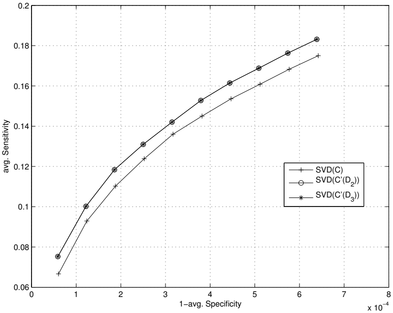
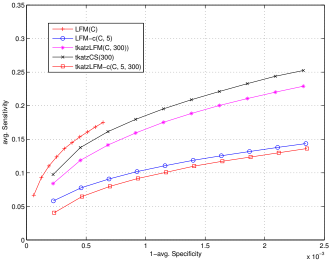


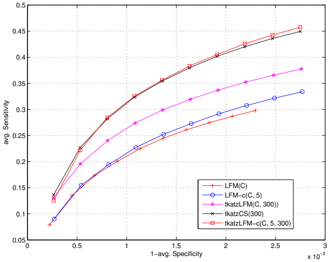
Figure: Comparison of latent factors based algorithms for various choices of $\mathbf{D}$, for the Orkut dataset: $\mathbf{D}_2 = \mathbf{A}^T\mathbf{A}, \mathbf{D}_3 = \lambda\mathbf{A}^T\mathbf{A}$.

- Choice of **D** is immaterial! (Consistent on Youtube).

Results



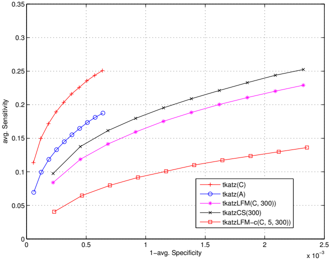
(a) Orkut-1c data set



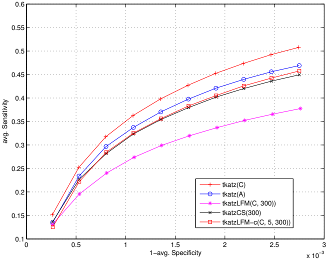
(b) Youtube-1c data set

Figure: Scalable approximations

Results



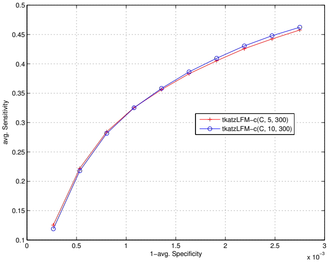
(a) Orkut-1c data set



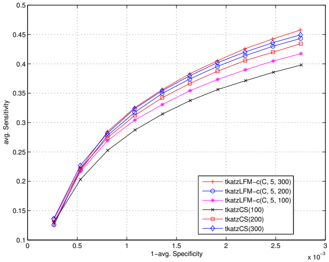
(b) Youtube-1c data set

Figure: Scalable approximations: Clustering

Results



(a) Effect of changing the number of clusters.



(b) Effect of changing the number of factors.

Outline

1. Outline
2. Social network analysis
   - Social and affiliation networks
   - Social network analysis: Problems
3. Recommendation algorithms
   - Modelling user-affiliation affinity
   - Latent factors model
   - Graph proximity model
   - Scalability for graph proximity model
4. Experiments
   - Datasets and their statistics
   - Evaluation methods
   - Results and discussion
5. Conclusion

Conclusion

Conclusion

- Friendship network is indeed useful in recommending affiliations!

Conclusion

- Friendship network is indeed useful in recommending affiliations!
- Community recommendation – link prediction perspective.

Conclusion

- Friendship network is indeed useful in recommending affiliations!
- Community recommendation – link prediction perspective.
- Two ways of modeling the information from auxiliary networks.

Conclusion

- Friendship network is indeed useful in recommending affiliations!
- Community recommendation – link prediction perspective.
- Two ways of modeling the information from auxiliary networks.
- Choice of evaluation strategy is important.

Future work

Future work

- Using affiliation networks for link prediction in friendship networks – Seems harder.

Future work

- Using affiliation networks for link prediction in friendship networks – Seems harder.
- More sources of information – How do you use them all?

Future work

- Using affiliation networks for link prediction in friendship networks – Seems harder.
- More sources of information – How do you use them all?
- Huge networks – Scalability.

The take home message

The take home message

- Graph proximity on Combined user/ item network $\longrightarrow$ Good item recommendations.

The take home message

- Graph proximity on Combined user/ item network $\longrightarrow$ Good item recommendations.
- Can make this scalable.

References

Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee.
Measurement and analysis of online social networks.
In *IMC '07: Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, pages 29–42, New York, NY, USA, 2007. ACM.

Vishvas Vasuki, Nagarajan Natarajan, Zhengdong Lu, and Inderjit.S Dhillon.
Affiliation recommendations using auxiliary friendship networks.
In *RecSys '10: Proceedings of the Third ACM Conference on Recommender Systems*, Barcelona, Spain, 2010.

Vishvas Vasuki, Nagarajan Natarajan, Zhengdong Lu, Berkant Savas, and Inderjit.S Dhillon.

Scalable affiliation recommendations using auxiliary friendship networks.

Submitted for review to ACM Transactions on Intelligent Systems and Technology special issue on Social Recommender Systems, 2010.

pdf, bib.