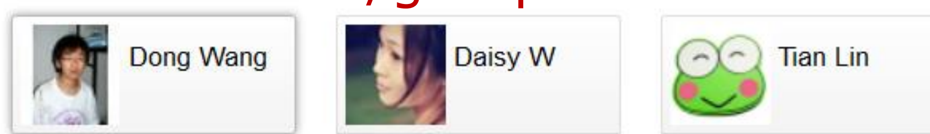
Learning to Infer Social Ties in Large Networks

Wenbin Tang, Honglei Zhuang, Jie Tang
Dept. of Computer Science
Tsinghua University

Real social networks are complex…

- Nobody exists only in one social network.
  - Public network vs. private network
  - Business network vs. family network
- However, existing networks (e.g., Facebook and Twitter) are trying to lump everyone into one big network
  - FB tries to solve this problem via lists/groups
  - However…
- Google+



which circle? Users do not take time to create it.

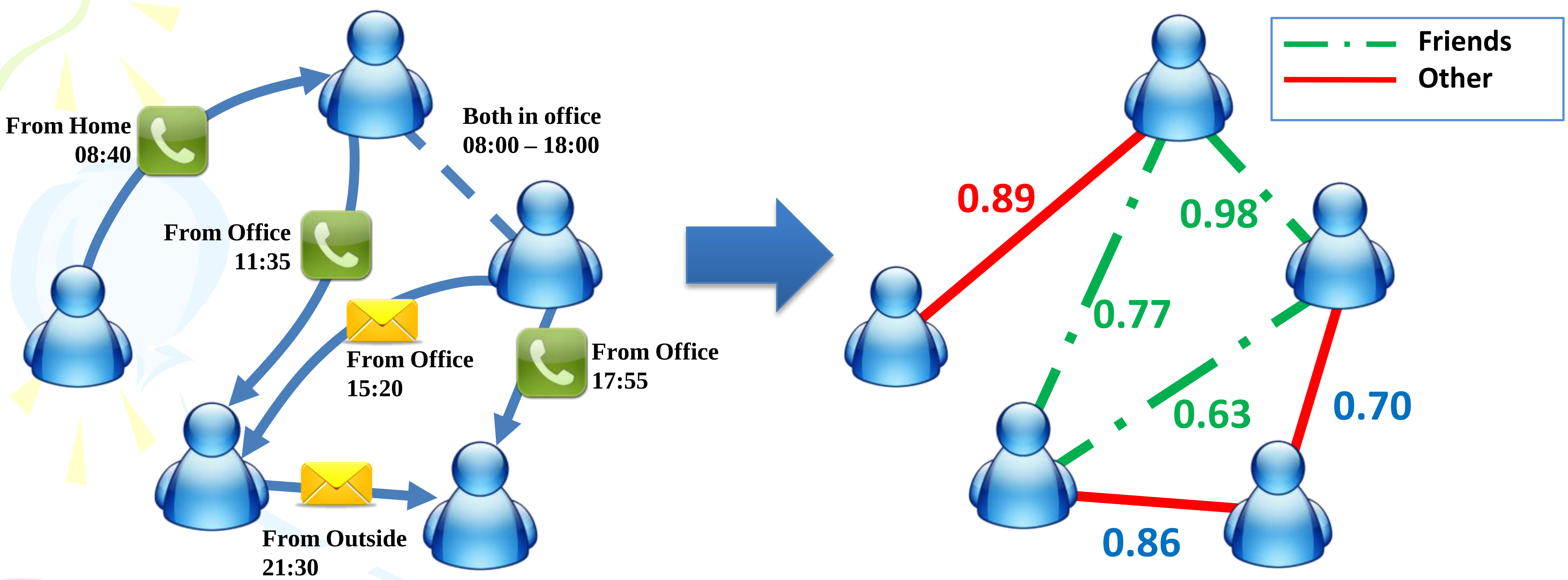
Even complex than we imagined!

- Only 16% of mobile phone users in Europe have created custom contact groups
  - *users do not* take the time to create it
  - *users do not* know how to circle their friends

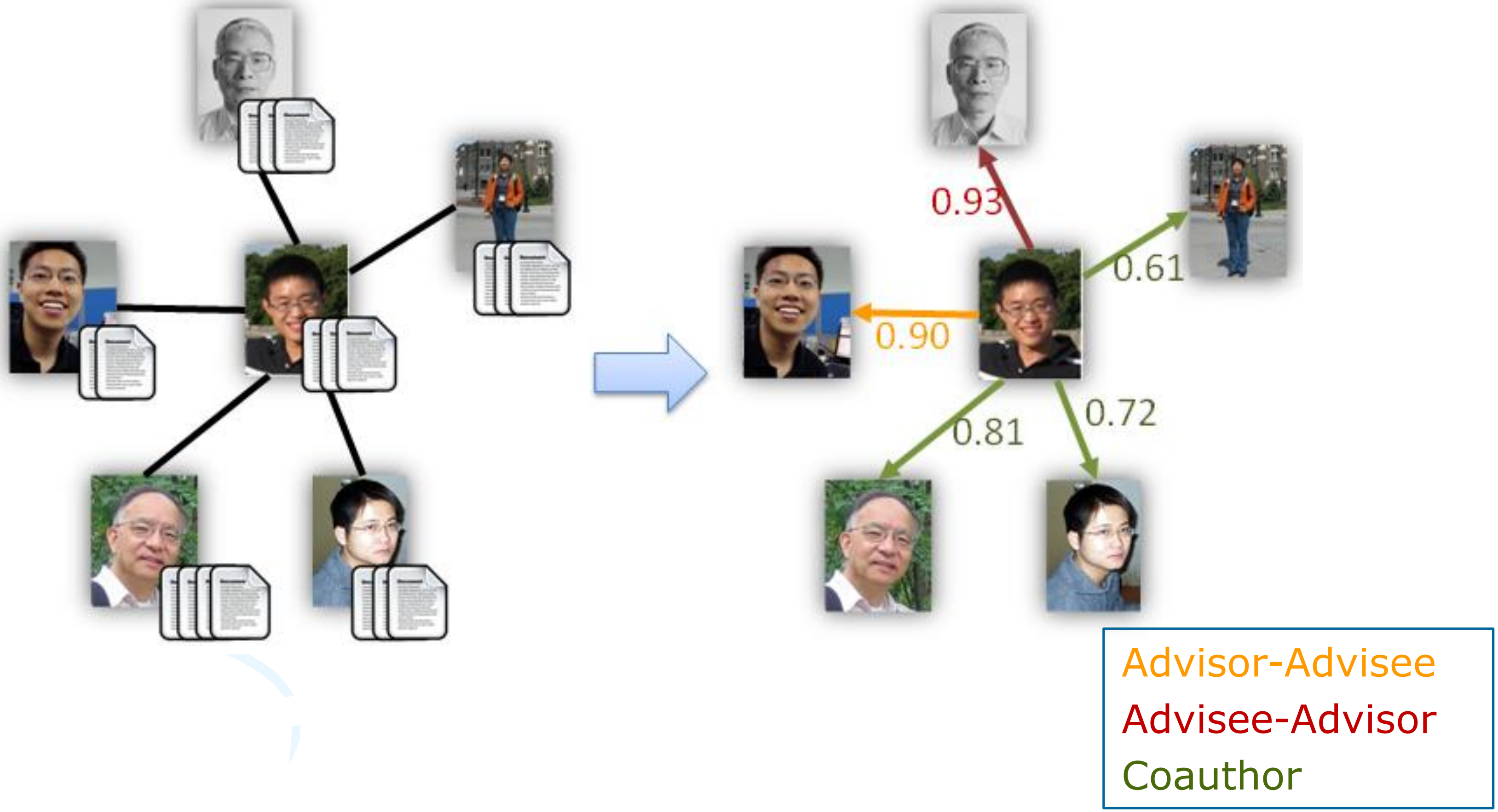
- The fact is that our social network is **black**-white…

How to infer social ties?

Example: Mobile network

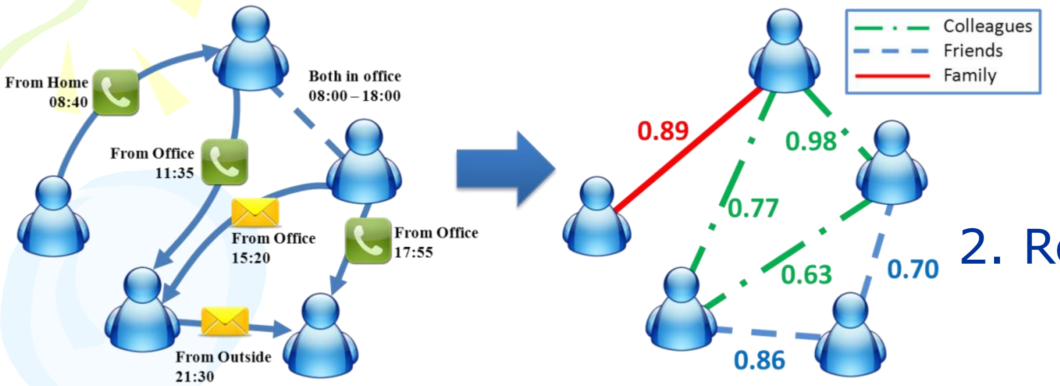


Example: Coauthor networks

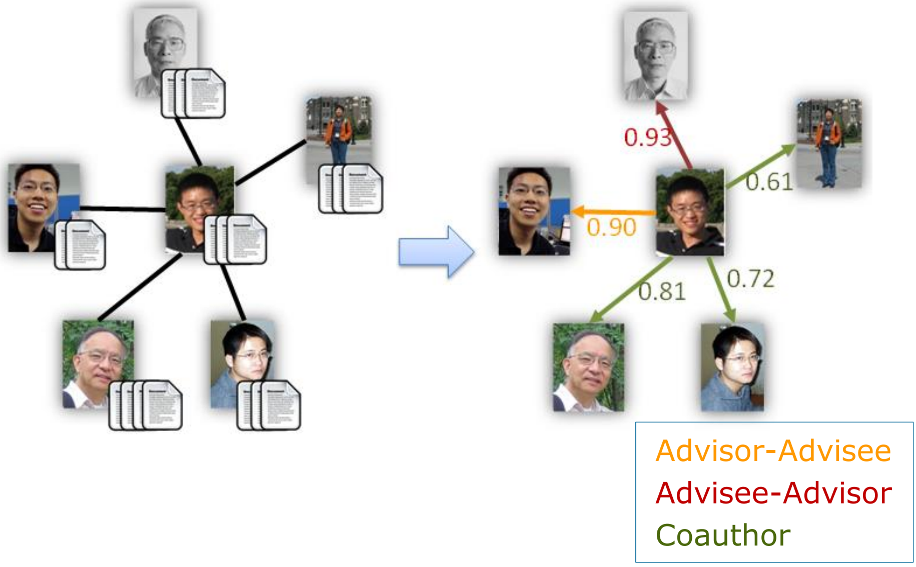


Challenges

1. Relationships in Mobile Network



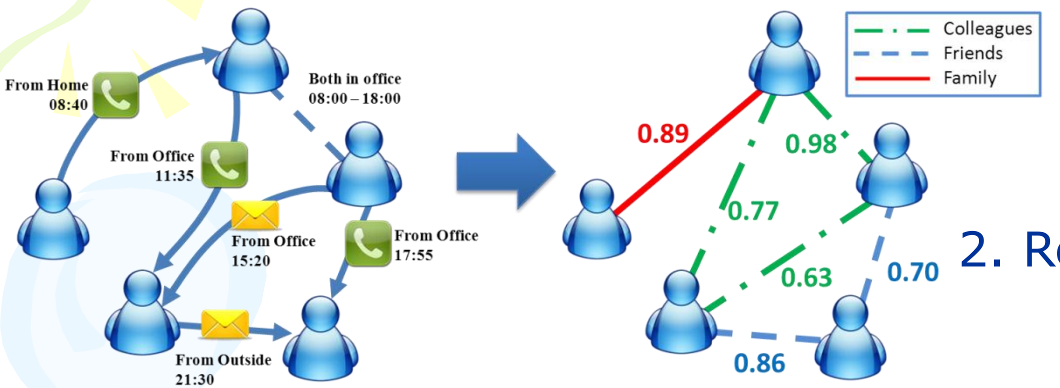
2. Relationships in Publication Network



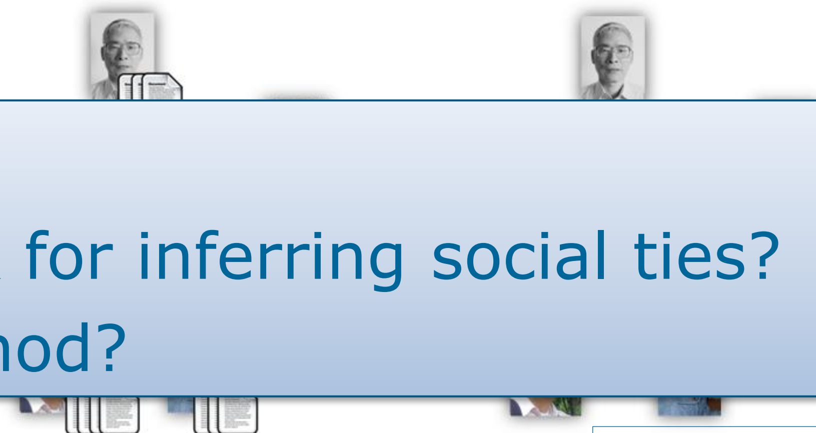
3. Relationships/Roles in Company Email Network

Challenges

1. Relationships in Mobile Network



2. Relationships in Publication Network

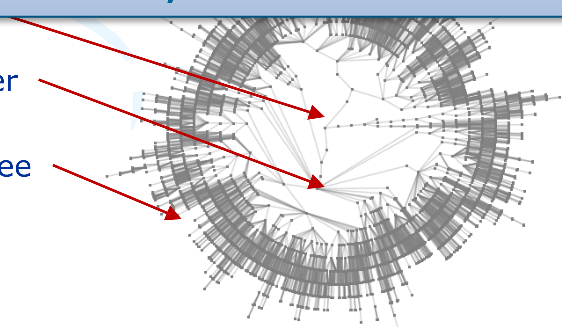


3.
Co

Challenges:

– A generalized framework for inferring social ties?

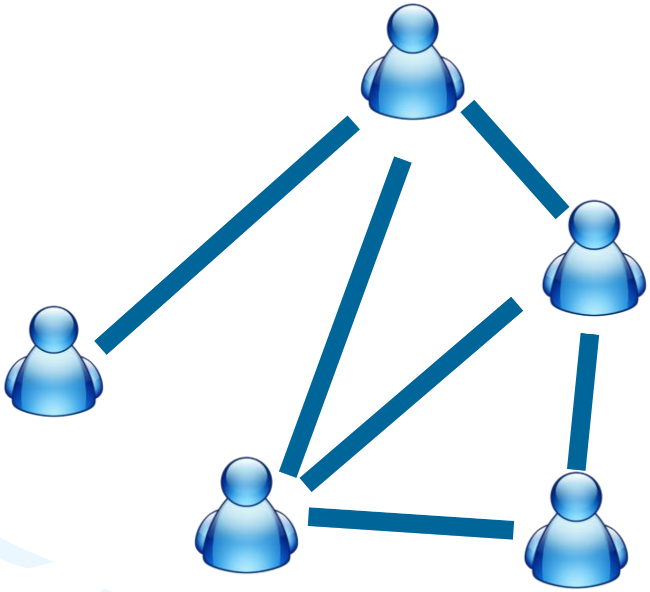
– A scalable, efficient method?



Advisor-Advisee
Advisee-Advisor
Coauthor

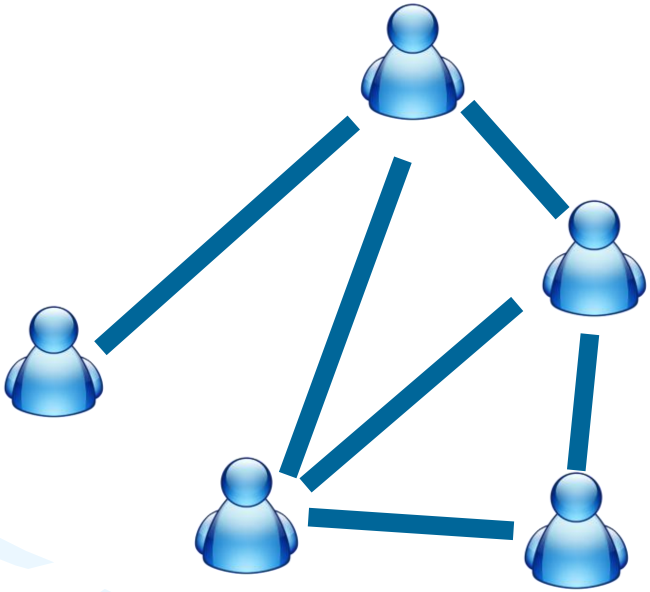
Problem Formulation

Input: $G=(V,E^L,E^U,R^L,W)$



Problem Formulation

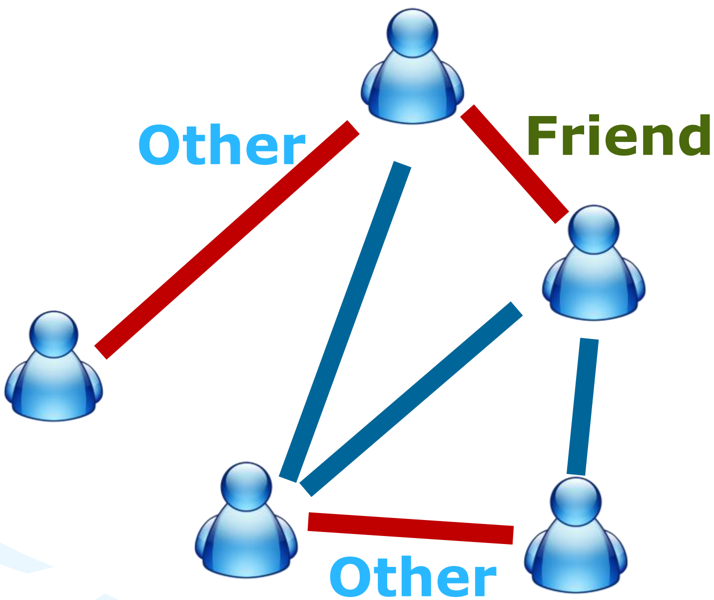
Input: $G=(V,E^L,E^U,R^L,W)$



V: Set of Users

Problem Formulation

Input: $G=(V, E^L, E^U, R^L, W)$

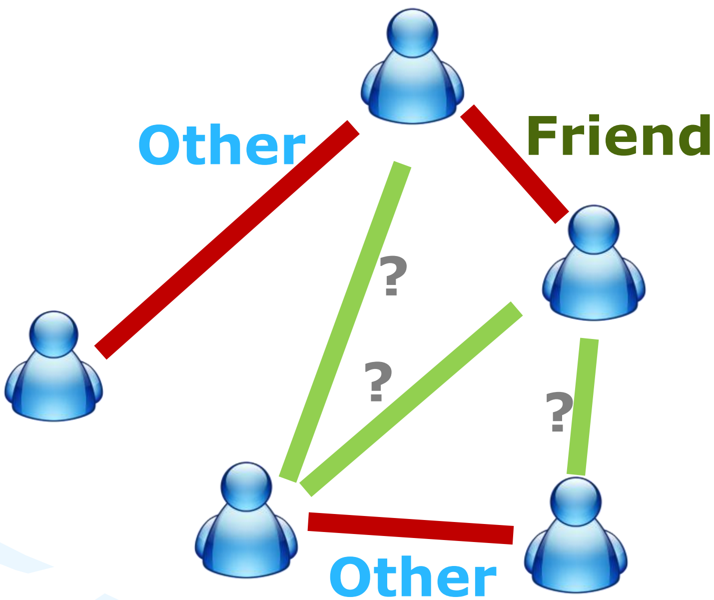


V: Set of Users

E^L, R^L: Labeled relationships

Problem Formulation

Input: $G=(V,E^L,E^U,R^L,W)$



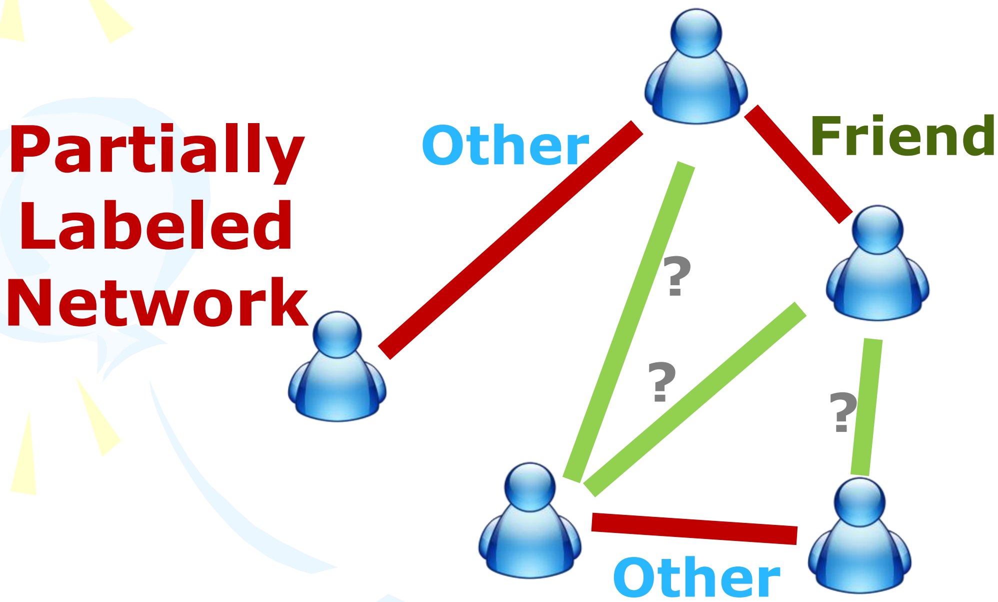
V: Set of Users

E^L,R^L: Labeled relationships

E^U: Unlabeled relationships

Problem Formulation

Input: $G=(V,E^L,E^U,R^L,W)$



V: Set of Users

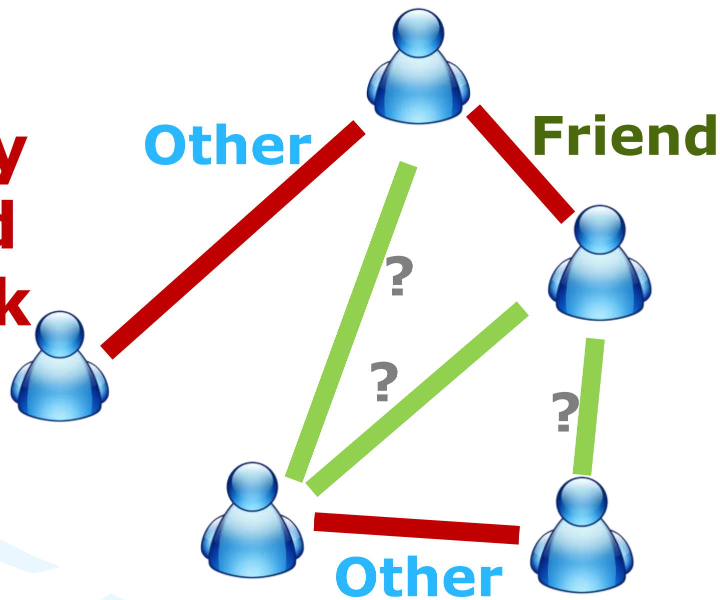
E^L,R^L: Labeled relationships

E^U: Unlabeled relationships

Problem Formulation

Input: $G=(V,E^L,E^U,R^L,W)$

Partially Labeled Network



V: Set of Users

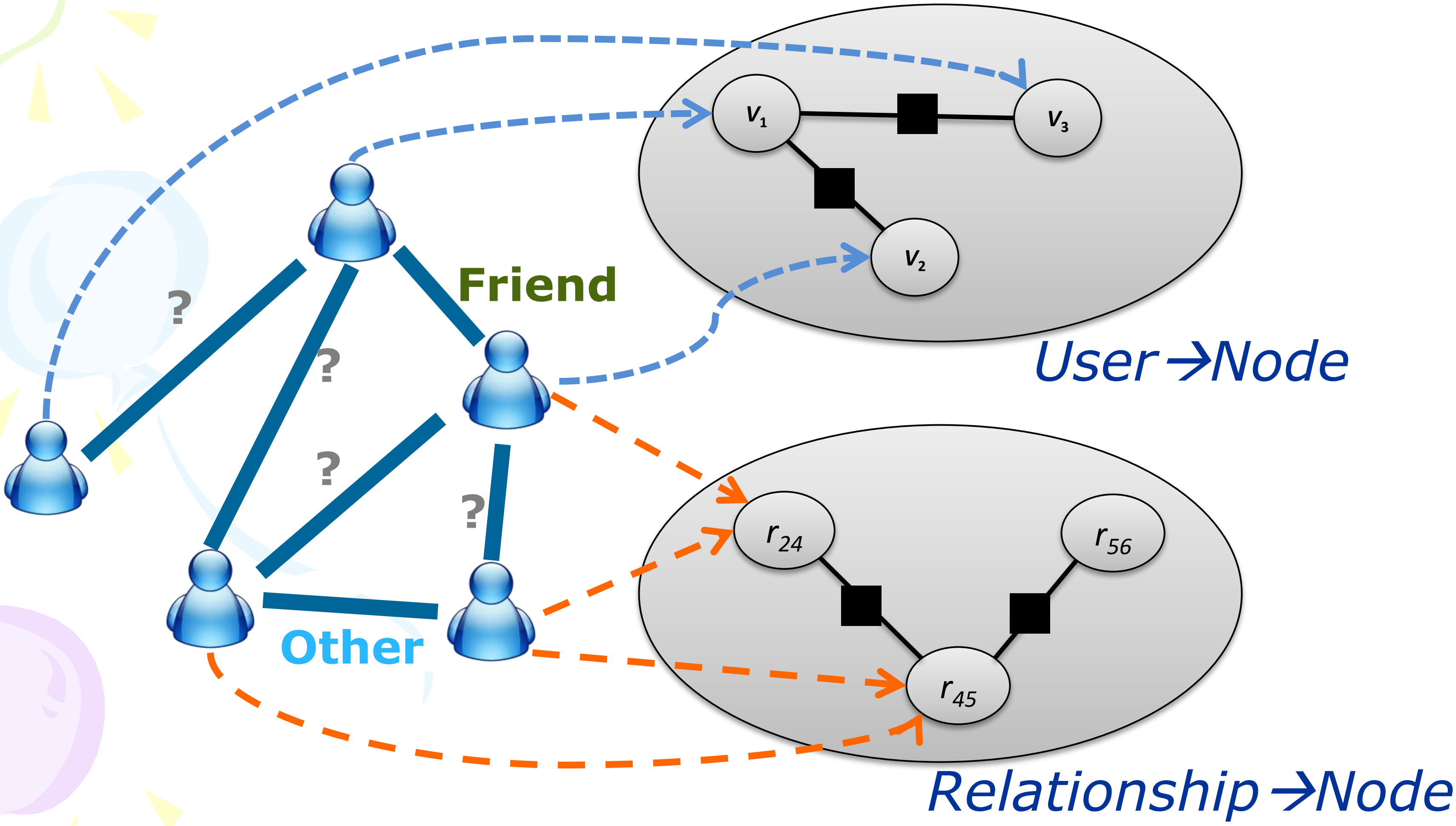
E^L,R^L: Labeled relationships

E^U: Unlabeled relationships

Input:
$G=(V,E^L,E^U,R^L,W)$

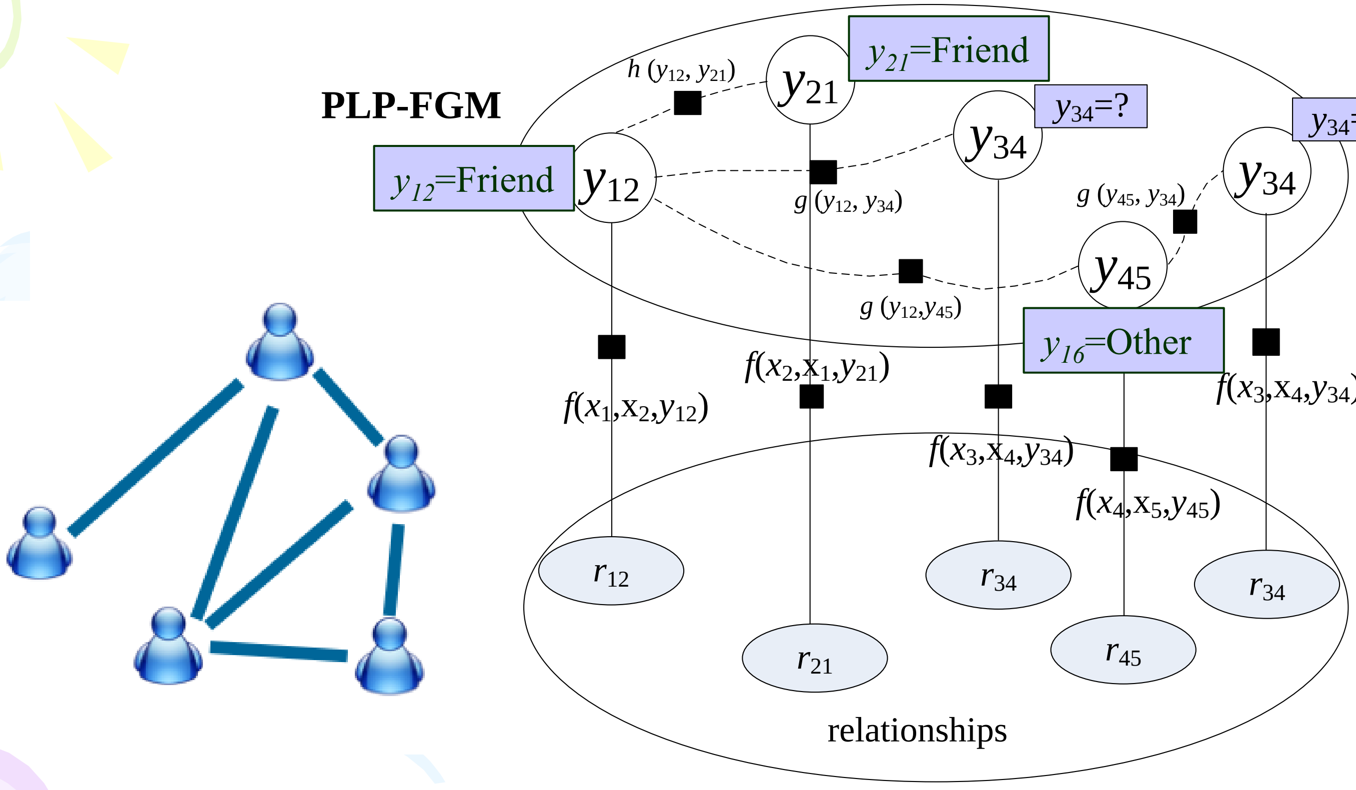
Output:
$f: G \rightarrow R$

Basic Idea

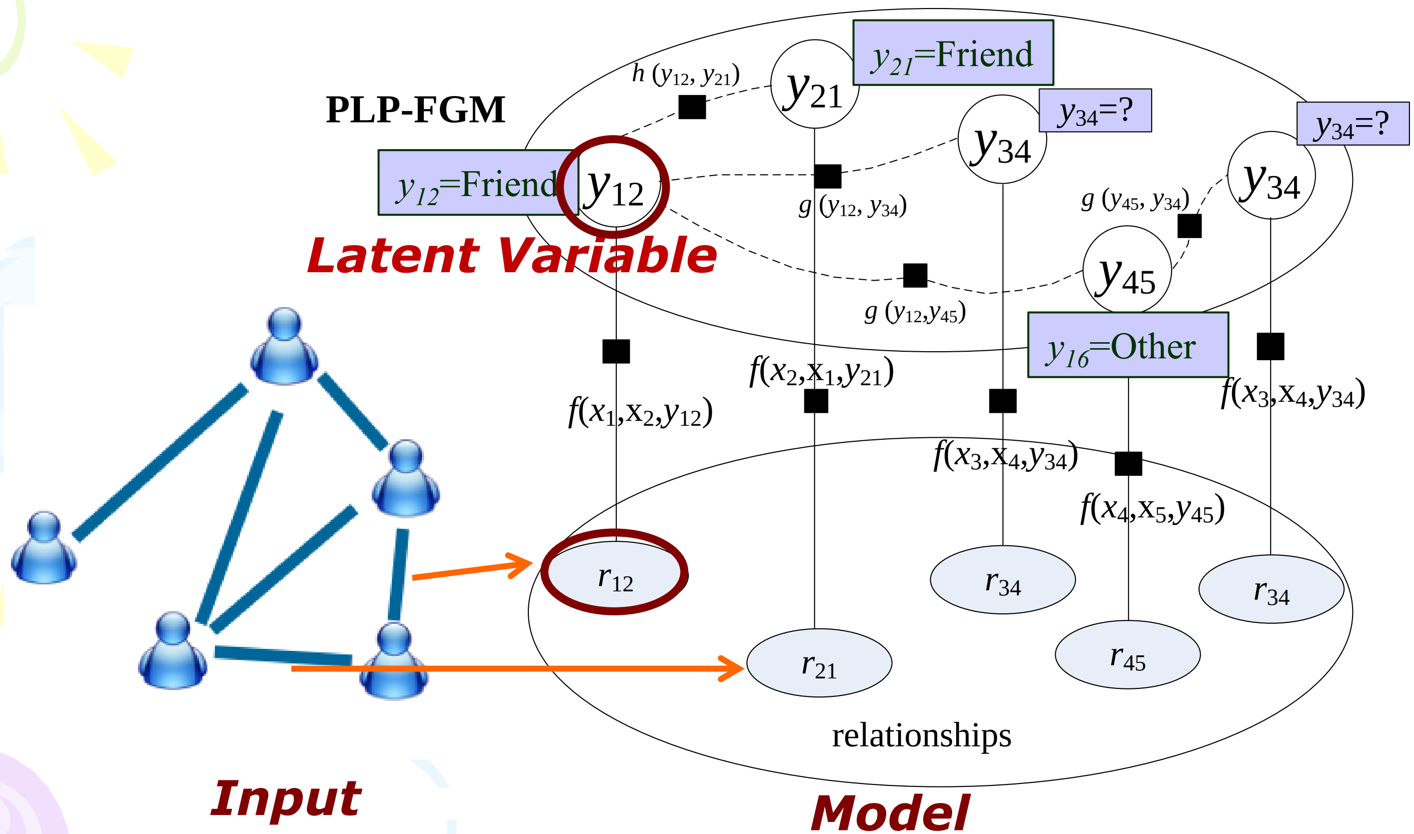


Friend
Other
?
?
?
?
v_1
v_2
v_3
User→Node
r_{24}
r_{56}
r_{45}
Relationship→Node

Partially Labeled Pairwise Factor Graph Model (PLP-FGM)

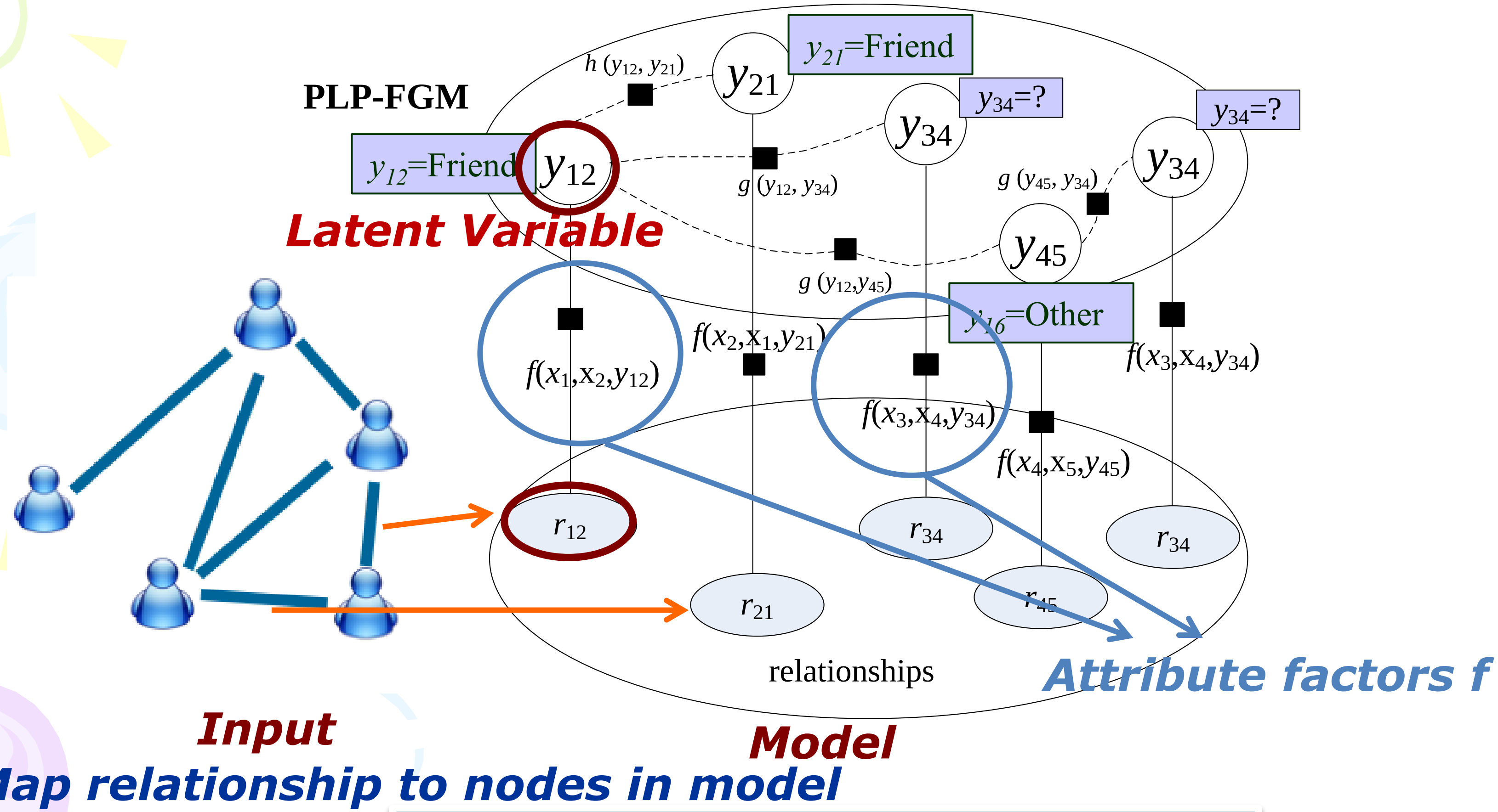


Partially Labeled Pairwise Factor Graph Model (PLP-FGM)



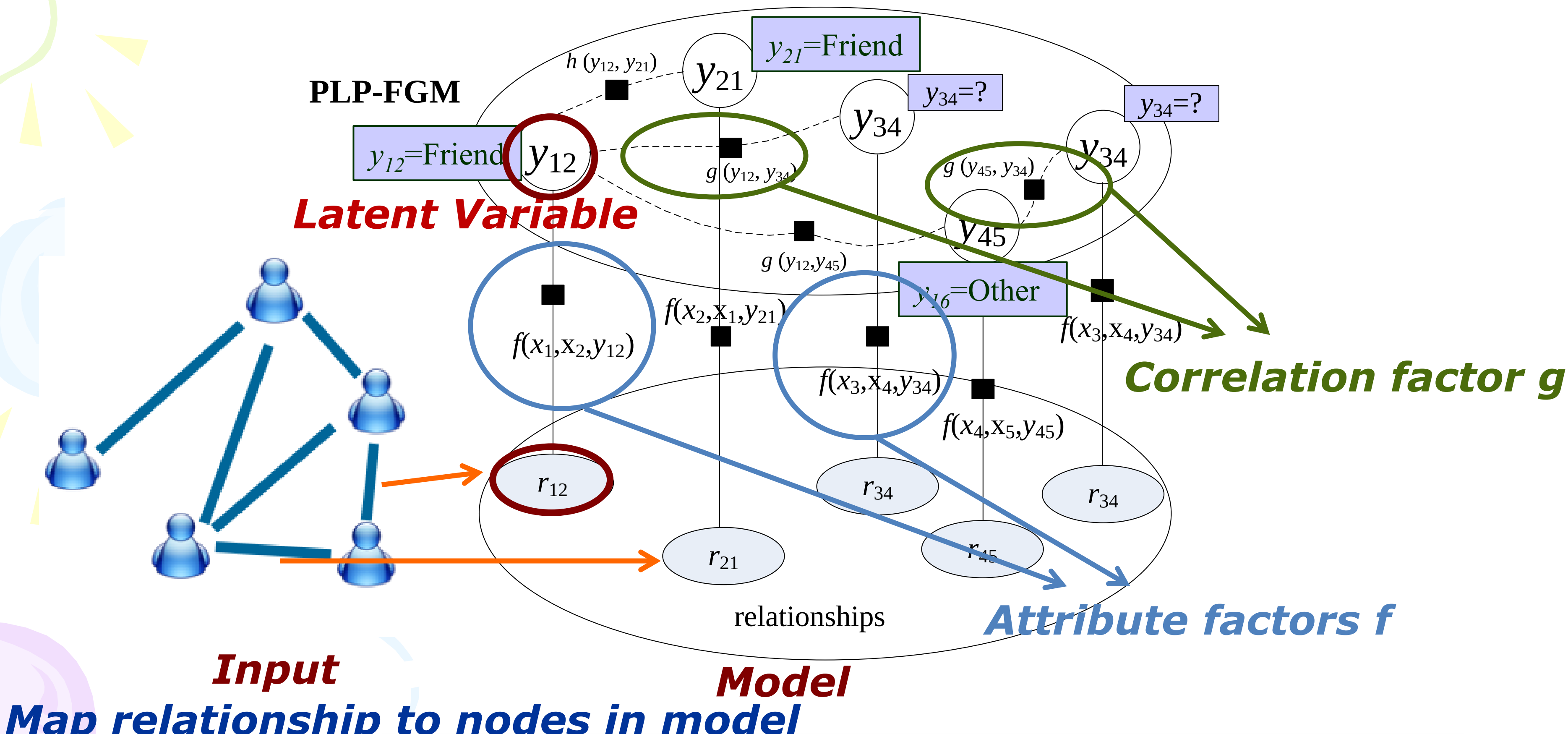
Map relationship to nodes in model

Partially Labeled Pairwise Factor Graph Model (PLP-FGM)



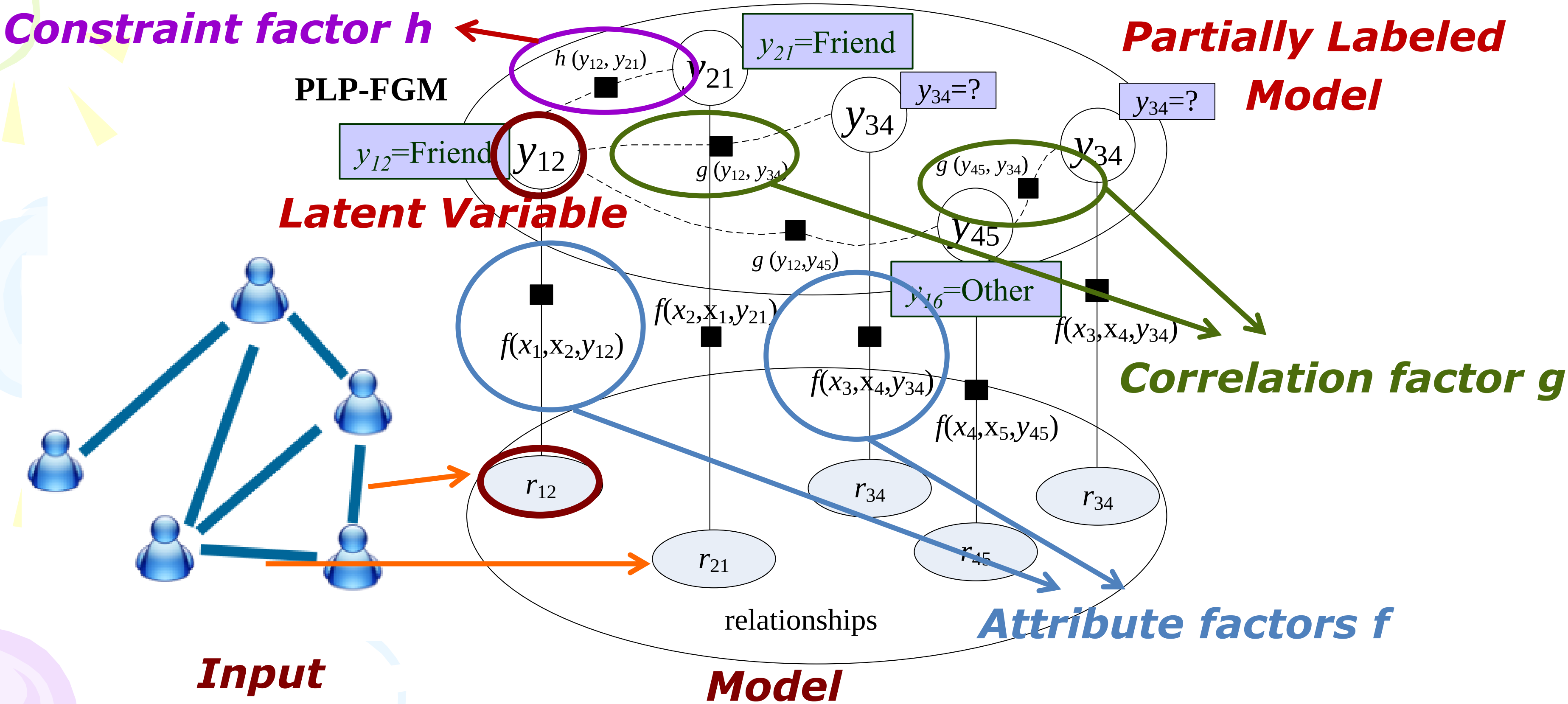
Example:
Call frequency between two users?

Partially Labeled Pairwise Factor Graph Model (PLP-FGM)



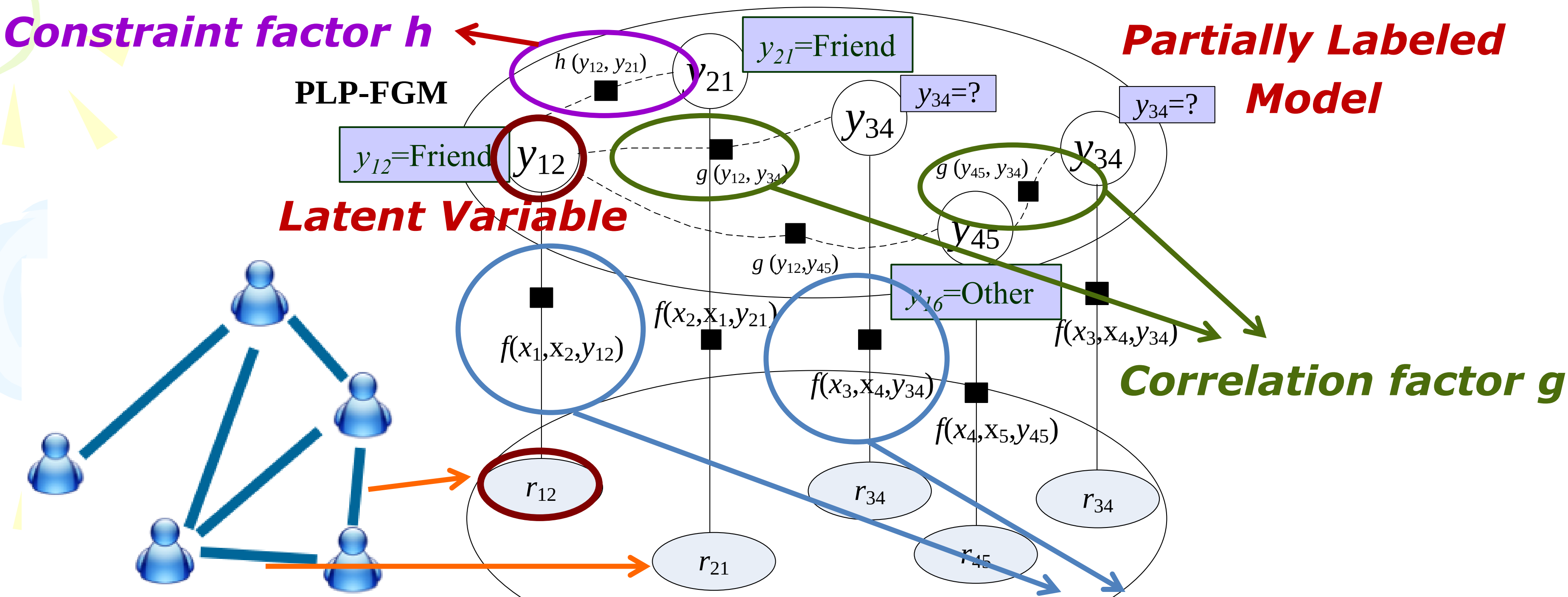
Example:
A makes call to B immediately after the call to C.

Partially Labeled Pairwise Factor Graph Model (PLP-FGM)



Map relationship to nodes in model

Partially Labeled Pairwise Factor Graph Model (PLP-FGM)



Problem:

For each relationship, identify which type has the highest probability?

Solutions$_{\text{(con't)}}$

- Different ways to instantiate factors
  - We use exponential-linear functions
    - Attribute Factor:

$$f(y_i, \mathbf{x}_i) = \frac{1}{Z_\lambda} \exp\{\lambda^T \mathbf{\Phi}(y_i, \mathbf{x}_i)\}$$

    - Correlation / Constraint Factor:

$$g(y_i, G(y_i)) = \frac{1}{Z_\alpha} \exp\{\sum_{y_j \in G(y_i)} \alpha^T \mathbf{g}(y_i, y_j)\}$$

$$h(y_i, H(y_i)) = \frac{1}{Z_\beta} \exp\{\sum_{y_j \in H(y_i)} \beta^T \mathbf{h}(y_i, y_j)\}$$

  - $\theta = [\lambda, \alpha, \beta], s = [\Phi^T, g^T, h^T]^T$
  - Log-Likelihood of labeled data:

$$\mathcal{O}(\theta) = \log \sum_{Y|Y^L} \exp\{\theta^T \mathbf{S}\} - \log \sum_{Y} \exp\{\theta^T \mathbf{S}\}$$

Learning Algorithm

- Maximize the log-likelihood of labeled relationships

Input: learning rate η
Output: learned parameters θ

Initialize θ;
repeat
  Calculate $\mathbb{E}_{p_\theta(Y|Y^L,G)}\mathbf{S}$ using LBP ;
  Calculate $\mathbb{E}_{p_\theta(Y|G)}\mathbf{S}$ using LBP ;
  Calculate the gradient of θ according to Eq. 7:

$$\nabla_\theta = \mathbb{E}_{p_\theta(Y|Y^L,G)}\mathbf{S} - \mathbb{E}_{p_\theta(Y|G)}\mathbf{S}$$

  Update parameter θ with the learning rate η:

$$\theta_{\text{new}} = \theta_{\text{old}} - \eta \cdot \nabla_\theta$$

until *Convergence*;

Algorithm 1: Learning PLP-FGM.

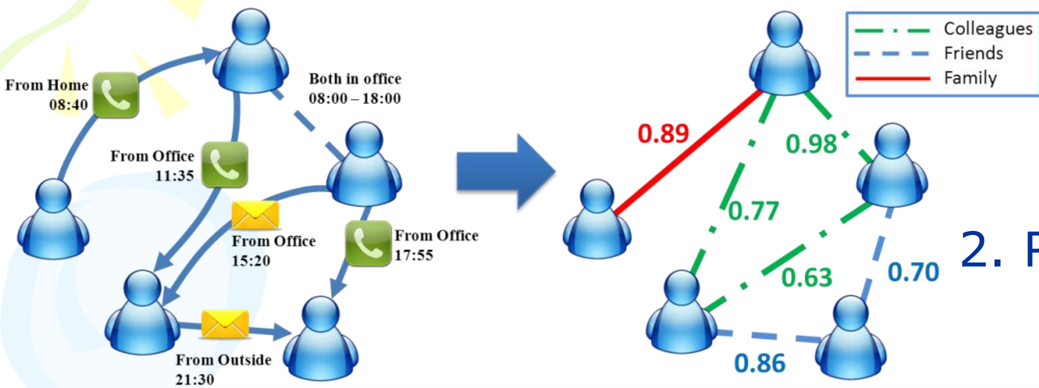
Expectation Computing

Loopy Belief Propagation

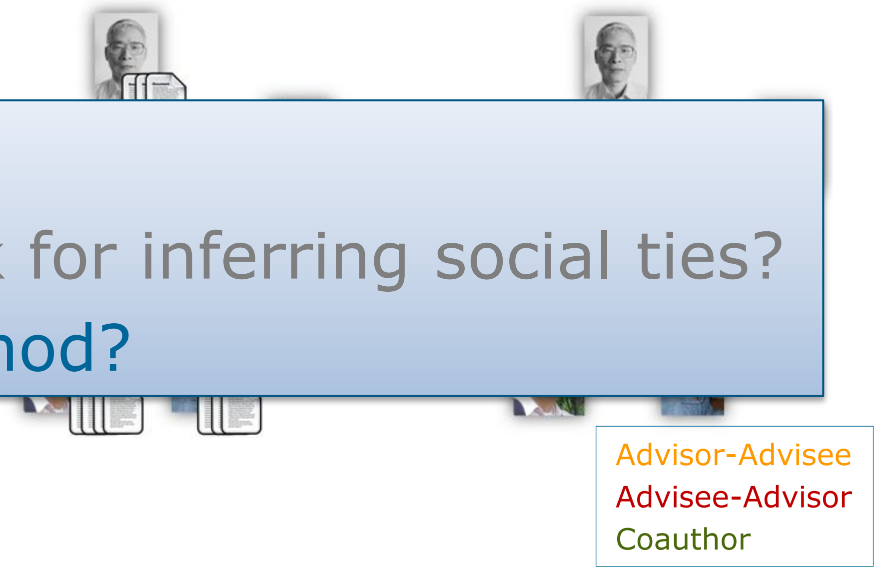
Gradient Decent Method

Challenges

1. Relationships in Mobile Network



2. Relationships in Publication Network

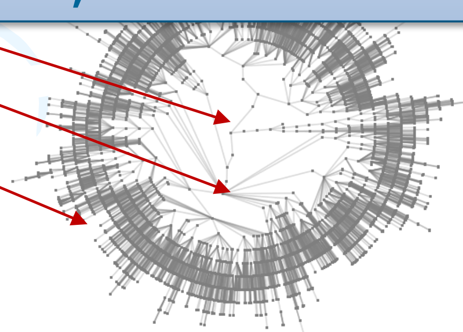


Challenges:

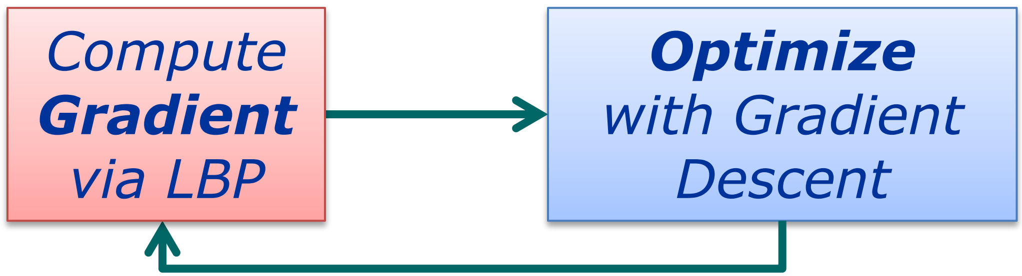
- A generalized framework for inferring social ties?
- A scalable, efficient method?

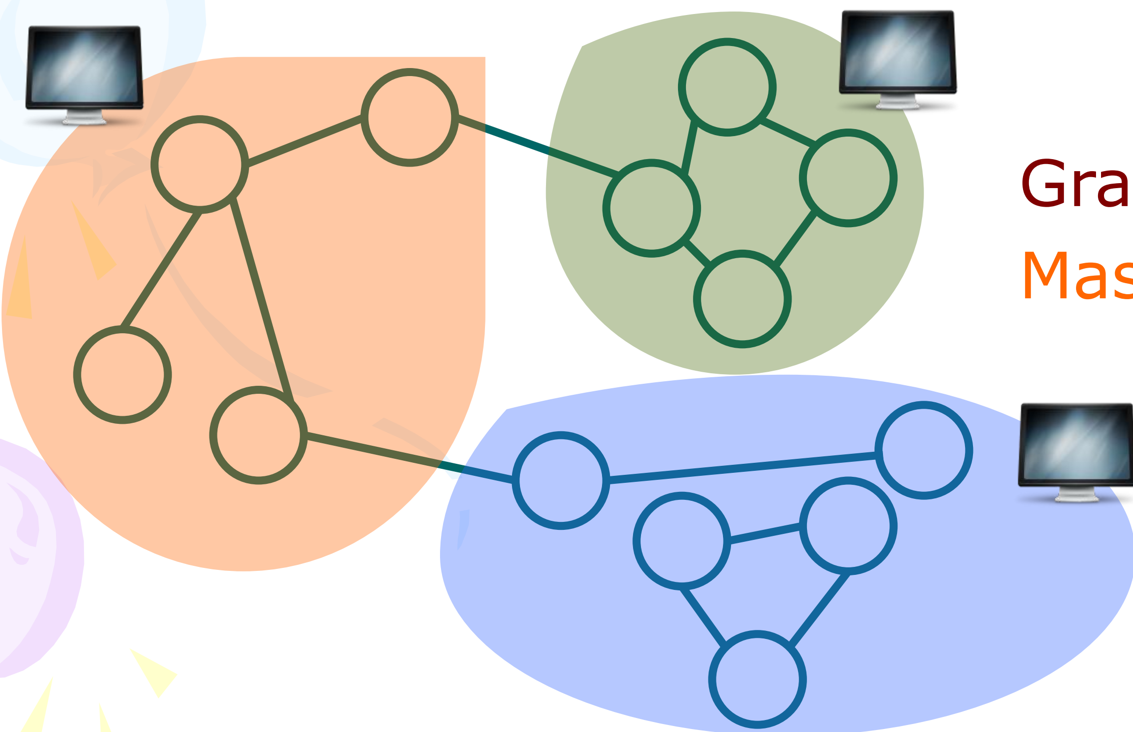
3.
Co

How to infer
Manager
Employee



Distributed Learning





Graph Partition

Master-Slave Computing

Data Sets

- Coauthor Network (Publication)
  - To infer Advisor-Advisee relationship
  - Papers from DBLP
- Email Network (Email)
  - To infer Manger-Subordinate relationship
  - Using Enron Email Dataset
- Mobile Network (Mobile)
  - To infer Friendship
  - 107 users (ten-month). Published by MIT

Data Set	Users	Unlabeled Relationships	Labeled Relationships
Publication	1,036,990	1,984,164	6,096
Email	151	3,424	148
Mobile	107	5,122	314

Baselines

- Baselines:
  - SVM:
    - Use the same features defined in our model to train a classification model
  - TPFG:
    - An unsupervised method to identify advisor-advisee relationships
  - **PLP-FGM-S**
    - Do not use partially-labeled property
    - Train parameters on **the labeled sub-graph**

Performance Analysis

Data Set	Method	Precision	Recall	F_1-score
Publication	SVM	72.5	54.9	62.1
	TPFG	82.8	**89.4**	86.0
	PLP-FGM-S	77.1	78.4	77.7
	PLP-FGM	**91.4**	87.7	**89.5**
Email	SVM	79.1	**88.6**	83.6
	PLP-FGM-S	85.8	85.6	85.7
	PLP-FGM	**88.6**	87.2	**87.9**
Mobile	SVM	**92.7**	64.9	76.4
	PLP-FGM-S	88.1	71.3	78.8
	PLP-FGM	89.4	**75.2**	**81.6**

SVM: Use the same feature to train a classification model

TPFG: An unsupervised method to identify advisor-advisee relationships

PLP-FGM-S:Train PLP-FGM model on the labeled sub-graph

Performance Analysis

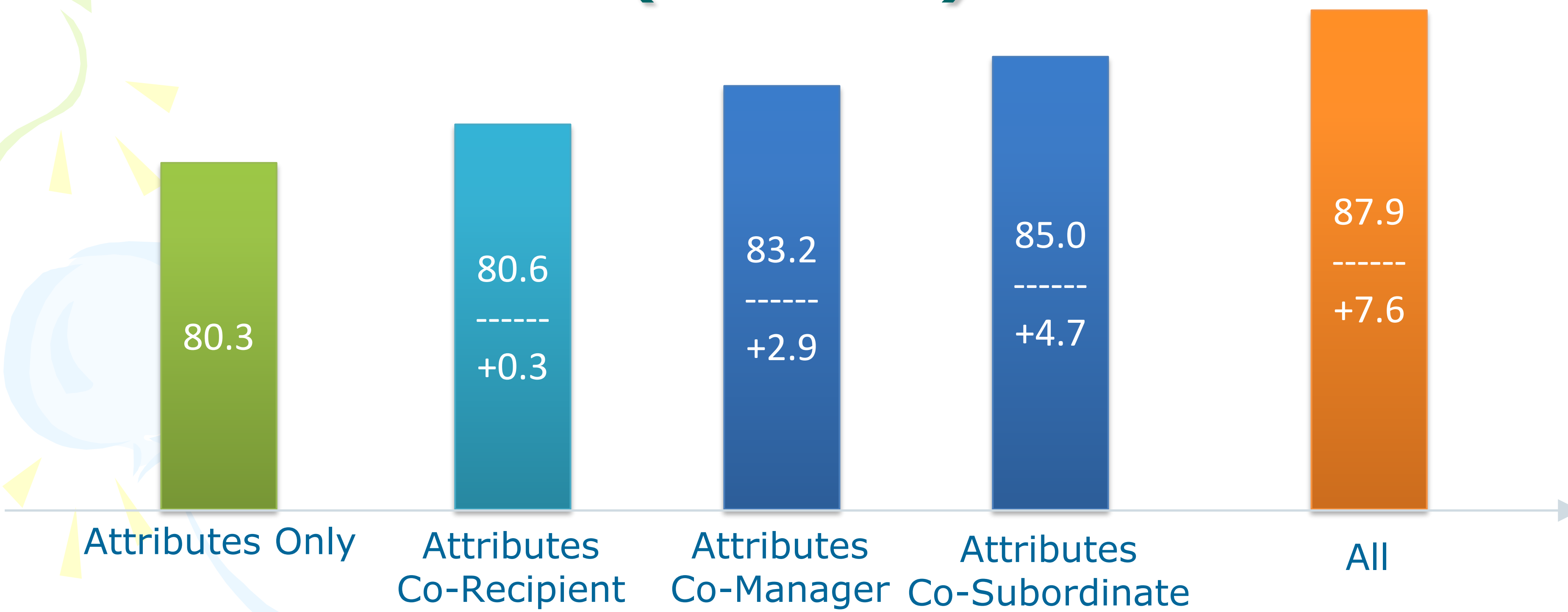
Data Set	Method	Precision	Recall	F_1-score
Publication	SVM	72.5	54.9	62.1
	TPFG	82.8	**89.4**	86.0
	PLP-FGM-S	77.1	78.4	77.7
	PLP-FGM	**91.4**	87.7	**89.5**
Email	SVM	79.1	**88.6**	83.6
	PLP-FGM-S	85.8	85.6	85.7
	PLP-FGM	**88.6**	87.2	**87.9**
Mobile	SVM	**92.7**	64.9	76.4
	PLP-FGM-S	88.1	71.3	78.8
	PLP-FGM	89.4	**75.2**	**81.6**

SVM: Use the same feature to train a classification model

TPFG: An unsupervised method to identify advisor-advisee relationships

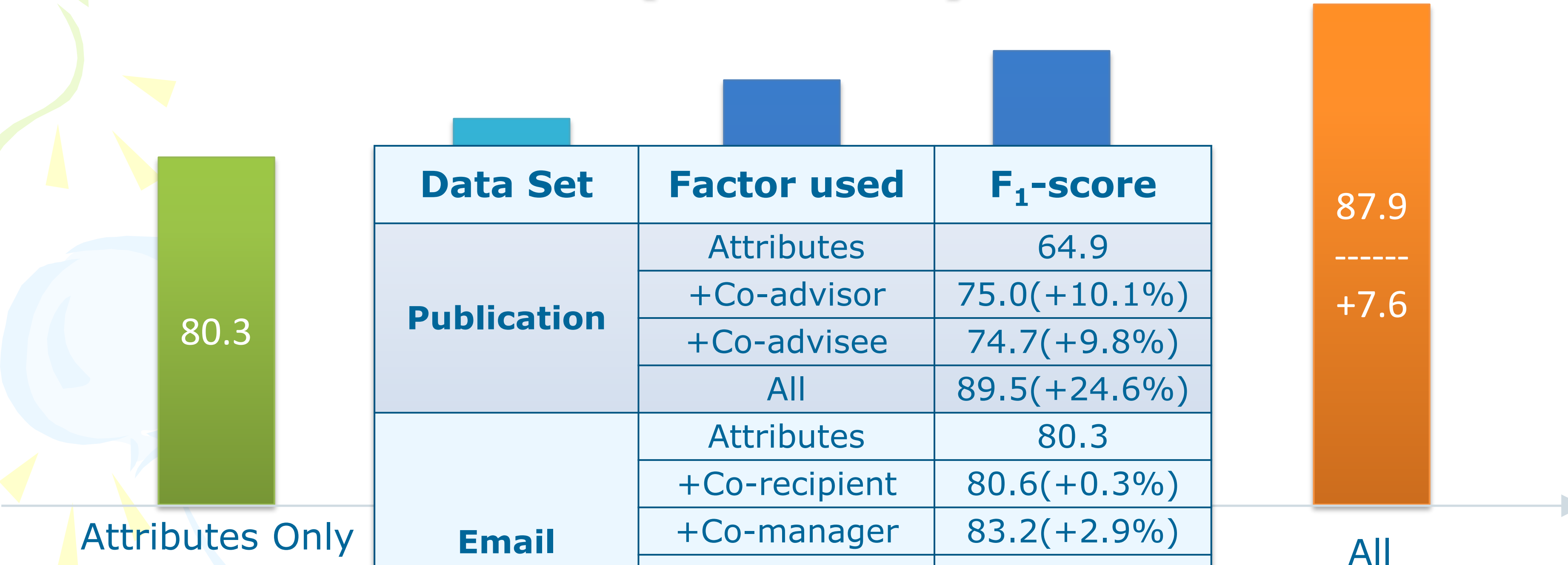
PLP-FGM-S:Train PLP-FGM model on the labeled sub-graph

Factor Contribution Analysis (Email)



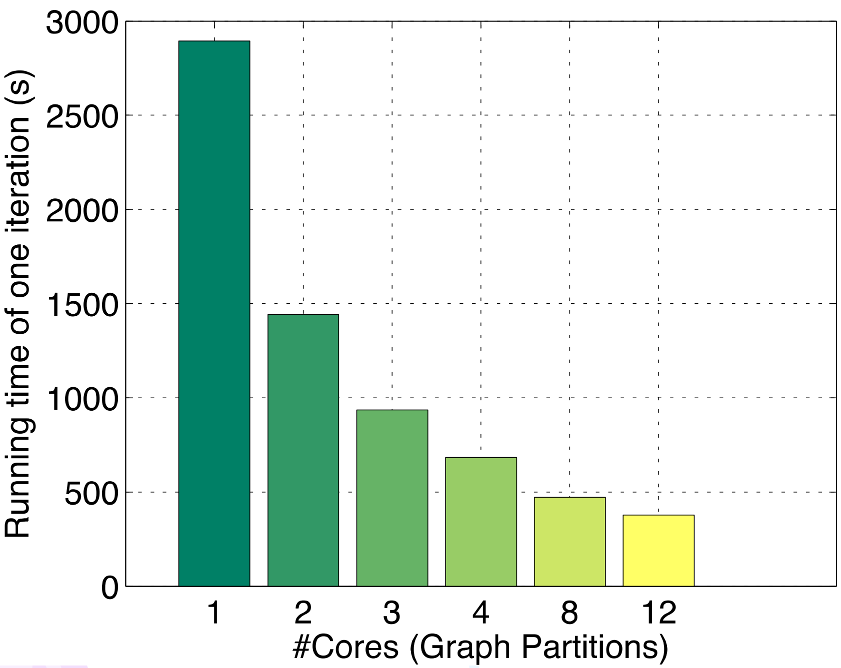
- Co-Recipient : a→(b, c) more than 10 mails
- Co-Manager : (a, b) – (a, c)
- Co-Subordinate: (b, a) – (c, a)

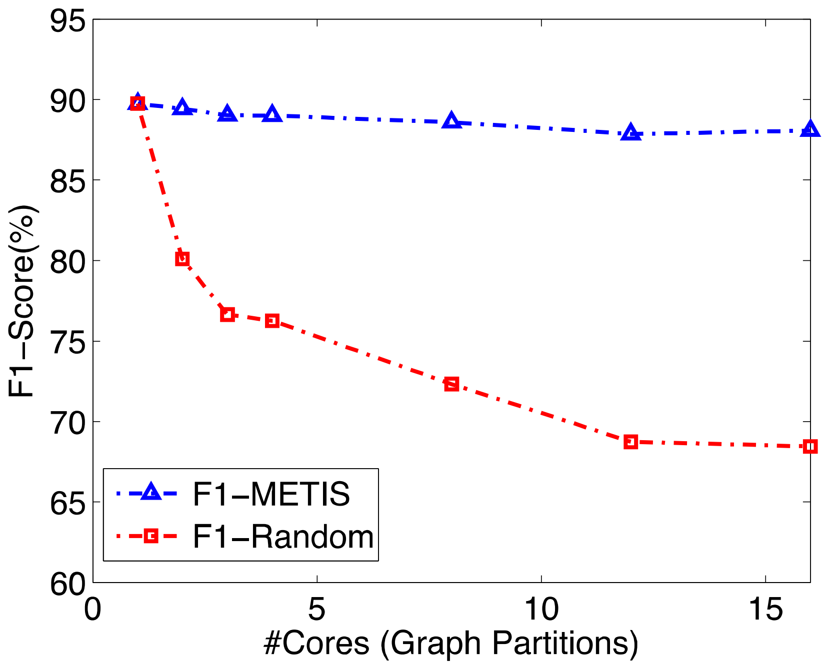
Factor Contribution Analysis (Email)



Data Set	Factor used	F_1-score
Publication	Attributes	64.9
	+Co-advisor	75.0(+10.1%)
	+Co-advisee	74.7(+9.8%)
	All	89.5(+24.6%)
Email	Attributes	80.3
	+Co-recipient	80.6(+0.3%)
	+Co-manager	83.2(+2.9%)
	+Co-subordinate	85.0(+4.7%)
	All	87.9(+7.6%)
Mobile	Attributes	80.2
	+Co-location	80.4(+0.2%)
	+Related-call	80.2(+0.0%)
	All	81.6(+1.4%)

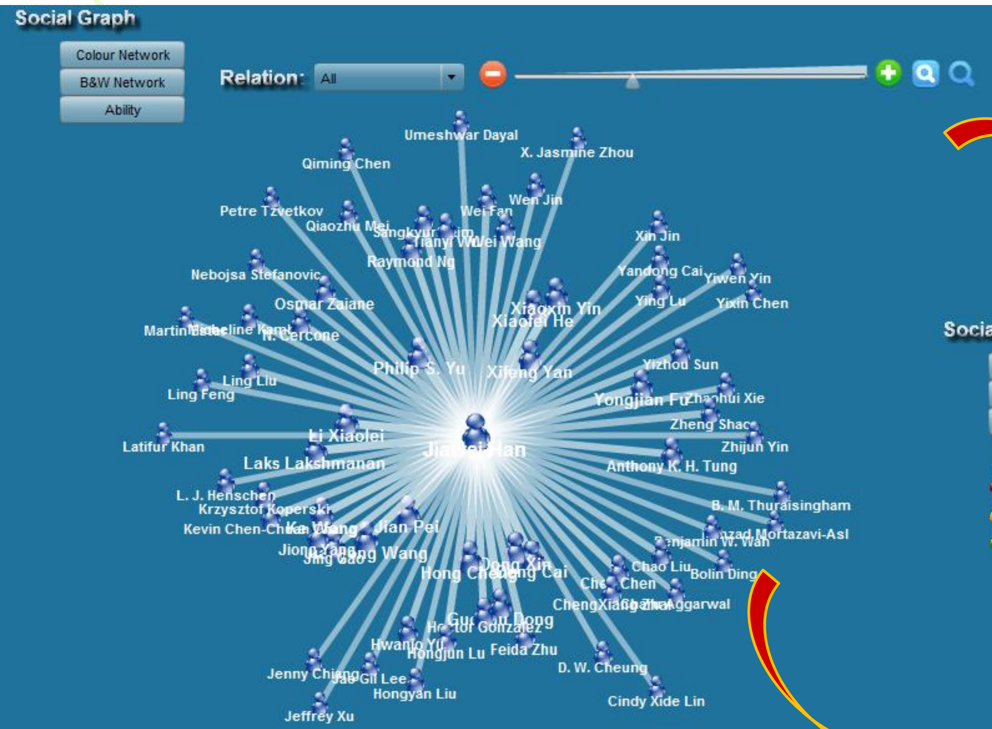
Distributed Learning Performance

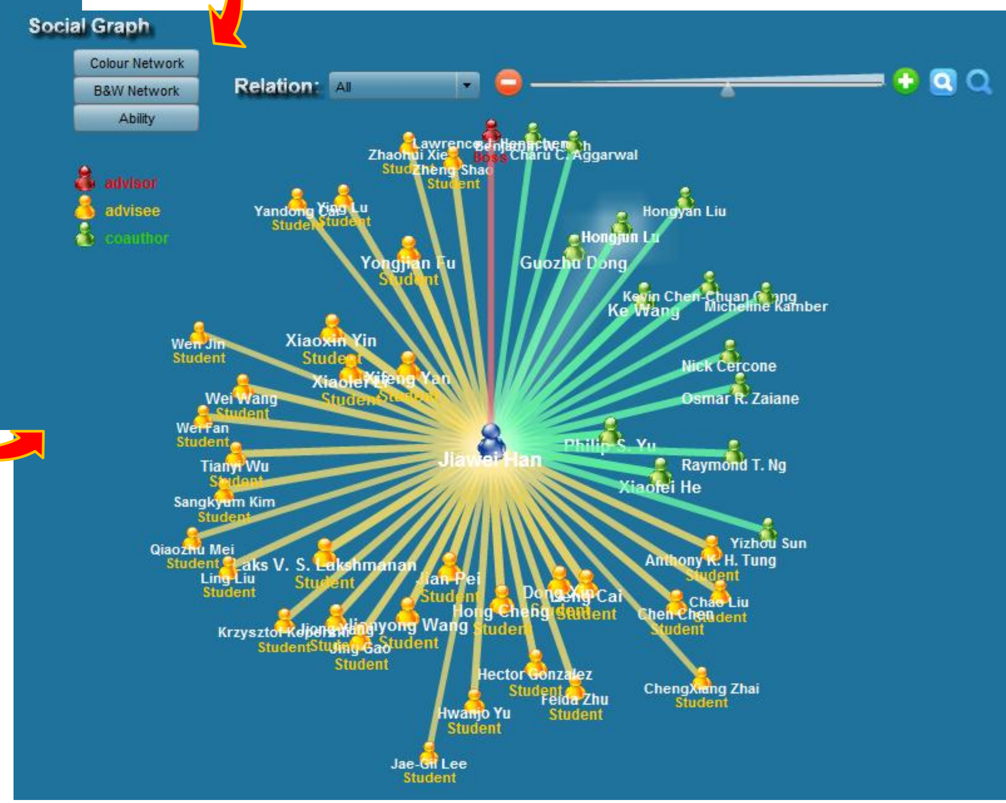




System of Arnetminer

http://arnetminer.org





Conclusion

- Formulate the problem of inferring the types of social ties
- Propose the PLP-FGM model to solve this problem, and present a distributed learning algorithm
- Validate the approach in different real data sets

Future work

- Make online social networks ***colorful***
- How to involve user into learning process?
- Connect with social theories?

Thank you!

Any Questions?

Correlation Definition

- Mobile Dataset:
  - Co-location
    - 3 users in the same location.
  - Related-call
    - A Make a call to B&C at the same place/time

- For more information, please refer to the paper☺

Feature Definition

<table>
<tr><th>Data set</th><th>Factor</th><th colspan="2">Description</th></tr>
<tr><td rowspan="5">Publication</td><td>Paper count</td><td colspan="2">$|P_i|$, $|P_j|$</td></tr>
<tr><td>Paper ratio</td><td colspan="2">$|P_i|/|P_j|$</td></tr>
<tr><td>Coauthor ratio</td><td colspan="2">$|P_i \cap P_j|/|P_i|$, $|P_i \cap P_j|/|P_j|$</td></tr>
<tr><td>Conference coverage</td><td colspan="2">The proportion of the conferences which both v_i and v_j attended among conferences v_j attended.</td></tr>
<tr><td>First-paper-year-diff</td><td colspan="2">The difference in year of the earliest publication of v_i and v_j.</td></tr>
<tr><td rowspan="8">Email</td><td rowspan="8">Traffics</td><td>Sender</td><td>Recipients Include</td></tr>
<tr><td>v_i</td><td>v_j</td></tr>
<tr><td>v_j</td><td>v_i</td></tr>
<tr><td>v_i</td><td>v_k and not v_j</td></tr>
<tr><td>v_j</td><td>v_k and not v_i</td></tr>
<tr><td>v_k</td><td>v_i and not v_j</td></tr>
<tr><td>v_k</td><td>v_j and not v_i</td></tr>
<tr><td>v_k</td><td>v_i and v_j</td></tr>
<tr><td rowspan="6">Mobile</td><td>#voice calls</td><td colspan="2">The total number of voice call logs between two users.</td></tr>
<tr><td>#messages</td><td colspan="2">Number of messages between two users.</td></tr>
<tr><td>Night-call ratio</td><td colspan="2">The proportion of calls at night (8pm to 8am).</td></tr>
<tr><td>Call duration</td><td colspan="2">The total duration time of calls between two users.</td></tr>
<tr><td>#proximity</td><td colspan="2">The total number of proximity logs between two users.</td></tr>
<tr><td>In-role proximity ratio</td><td colspan="2">The proportion of proximity logs in “working place” and in working hours (8am to 8pm).</td></tr>
</table>

Existing Methods…

- [Diehl:07] try to identify the relationships by learning a ranking function in Email network.
- Wang et al. [Wang:10] propose an unsupervised algorithm for mining the advisor-advisee relationships from the Publication network.

- Both algorithms focus on a specific domain
  - not easy to extend to other problems.