Social Prediction in Mobile Networks:

Can we infer users' emotions and social ties?

Jie Tang

Tsinghua University, China

Collaborate with

John Hopcroft, Jon Kleinberg (*Cornell*)

Jinghai Rao (Nokia), Jimeng Sun (*IBM TJ Watson*)

Tiancheng Lou, Wenbin Tang, Honglei Zhuang, Yuan Zhang (*Tsinghua*)

Motivation

Social behavior

VS.

Emotion change

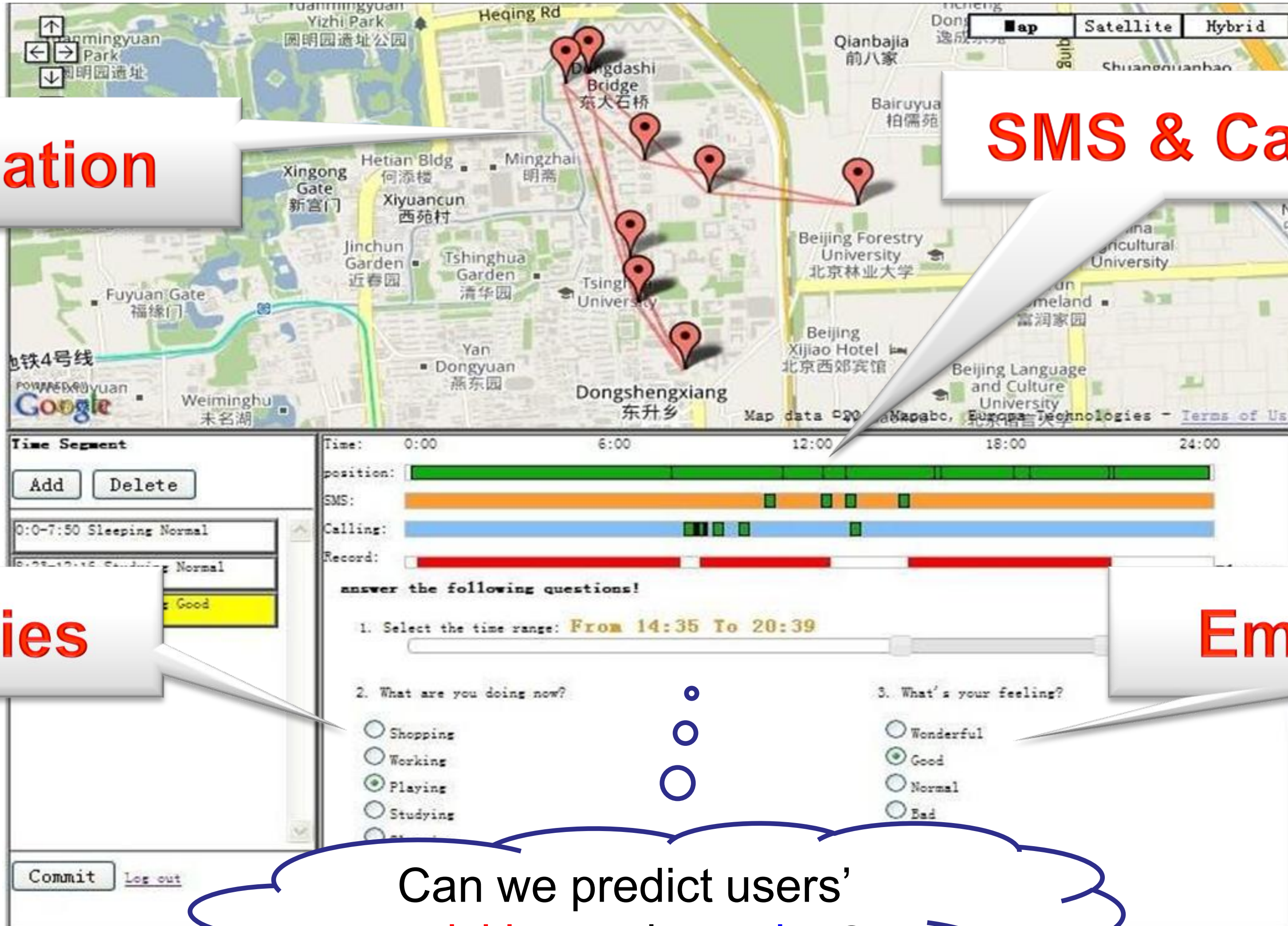
Motivation

Emotion stimulates the mind 3000 times quicker than rational though!!!

It's an emotional world we live in!

Six degree vs. Three degree [Nature; BMJ]

Motivation: A *Happy* System



Location

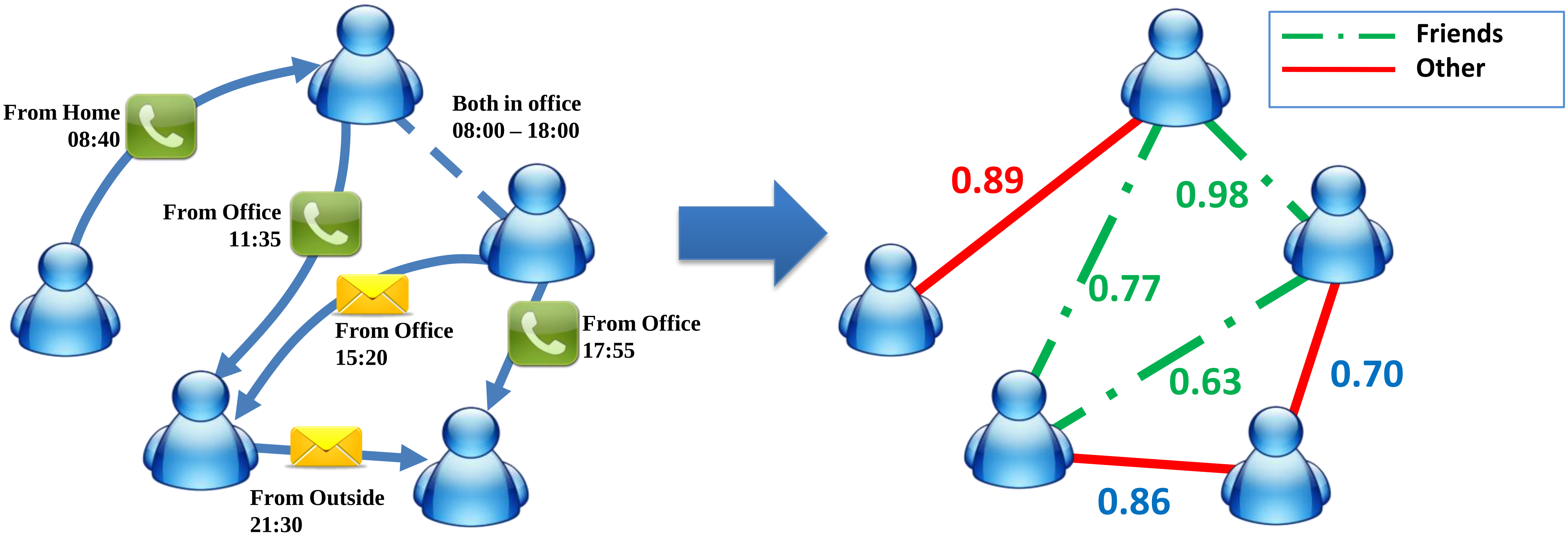
SMS & Calling

Activities

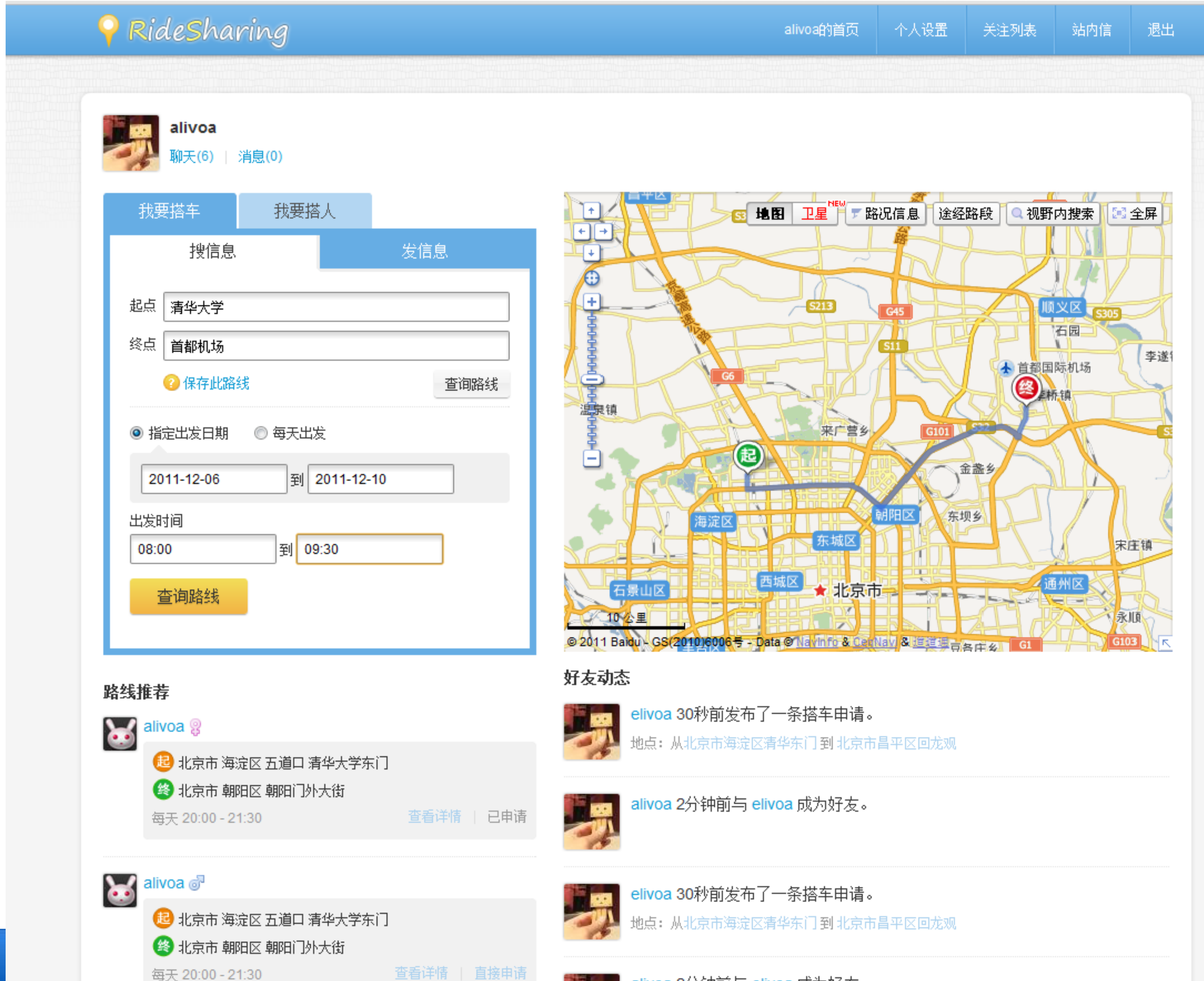
Emotion

Can we predict users' activities and emotion?

Motivation: Inferring Social Ties



Motivation: RideSharing

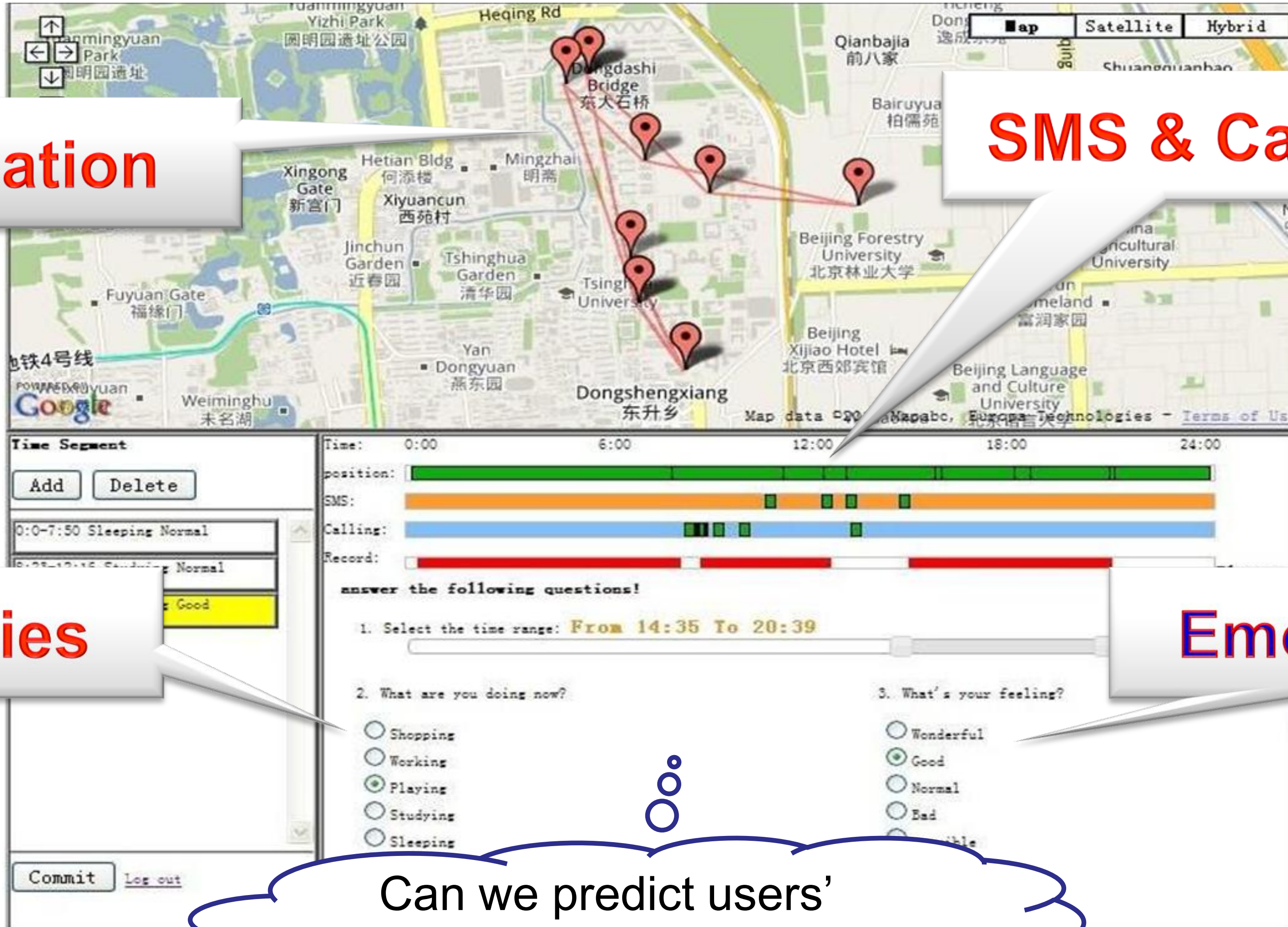


MoodCast: Emotion Prediction via Dynamic Continuous Factor Graph Model

ICDM'10, IEEE Trans. on Affective Computing'11



Happy System



Location

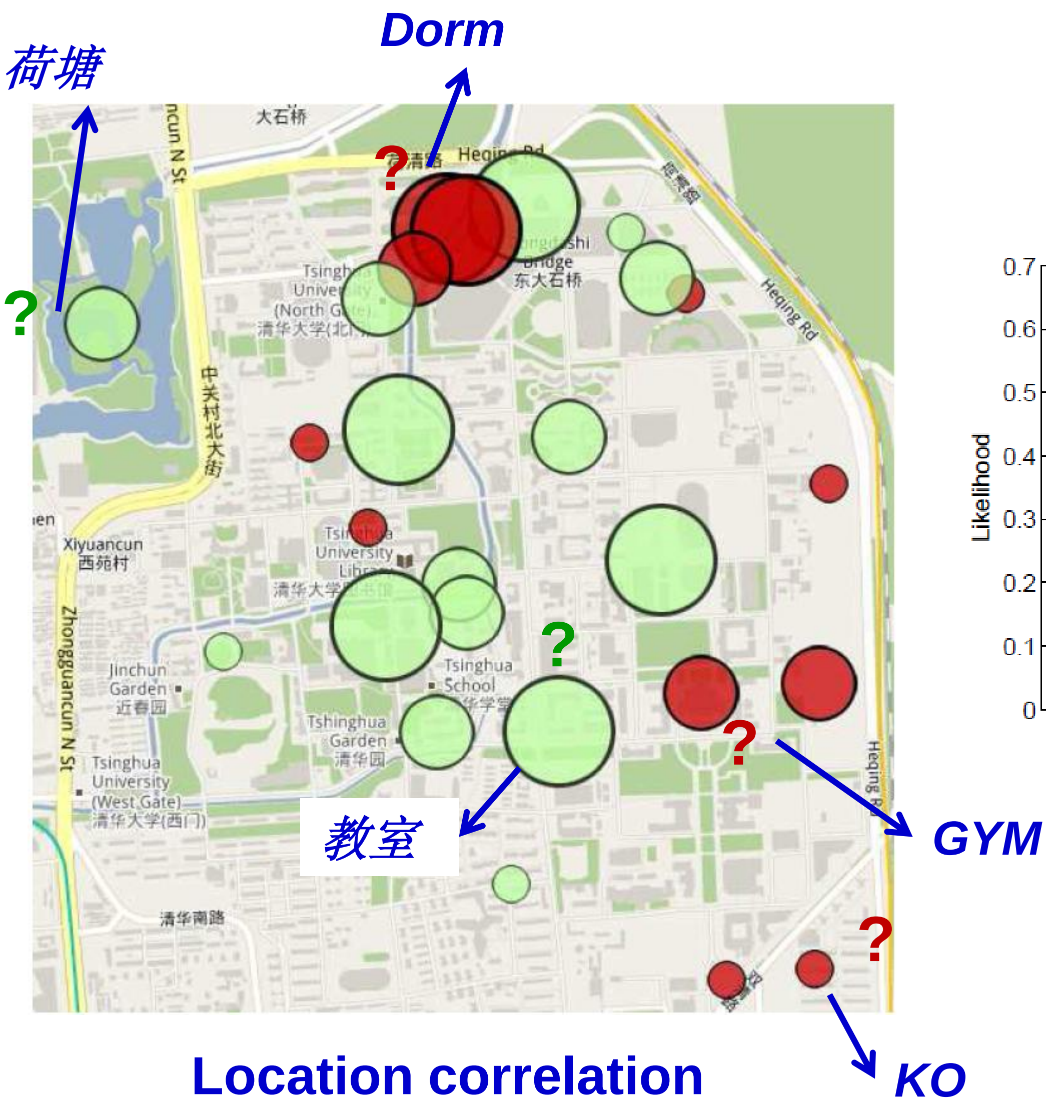
SMS & Calling

Activities

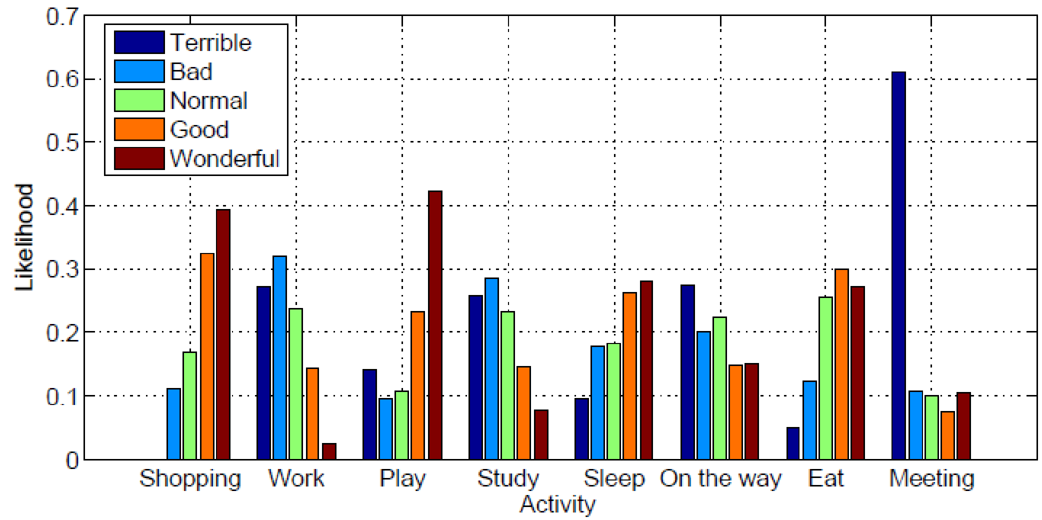
Emotion?

Can we predict users' emotion?

Observations

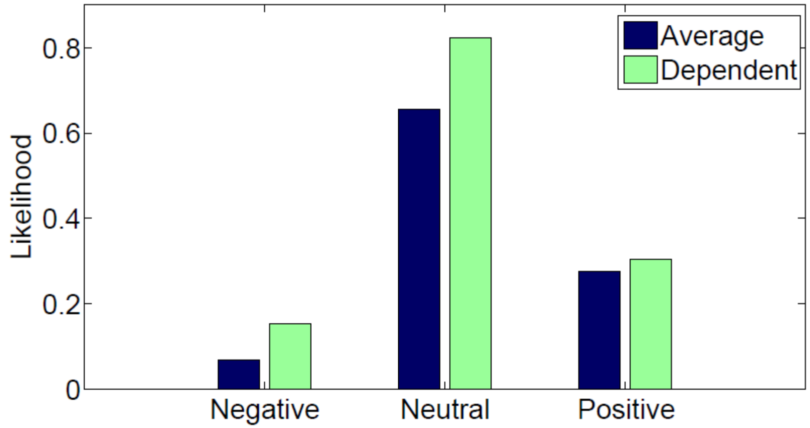


Location correlation
(Red-happy)



Activity correlation

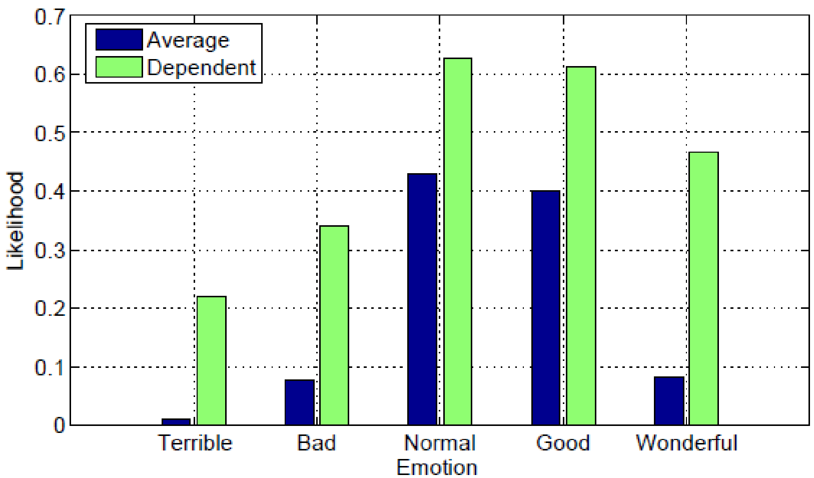
Observations (cont.)



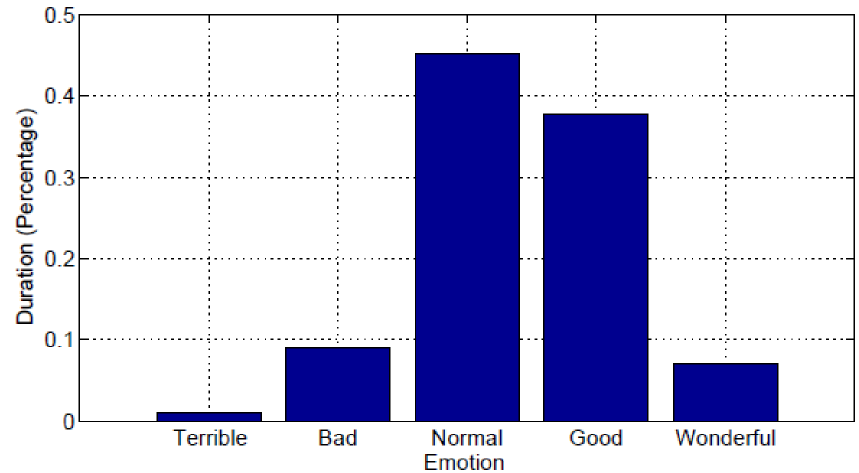
(a) Social correlation

(a) Implicit groups by emotions

Social correlation



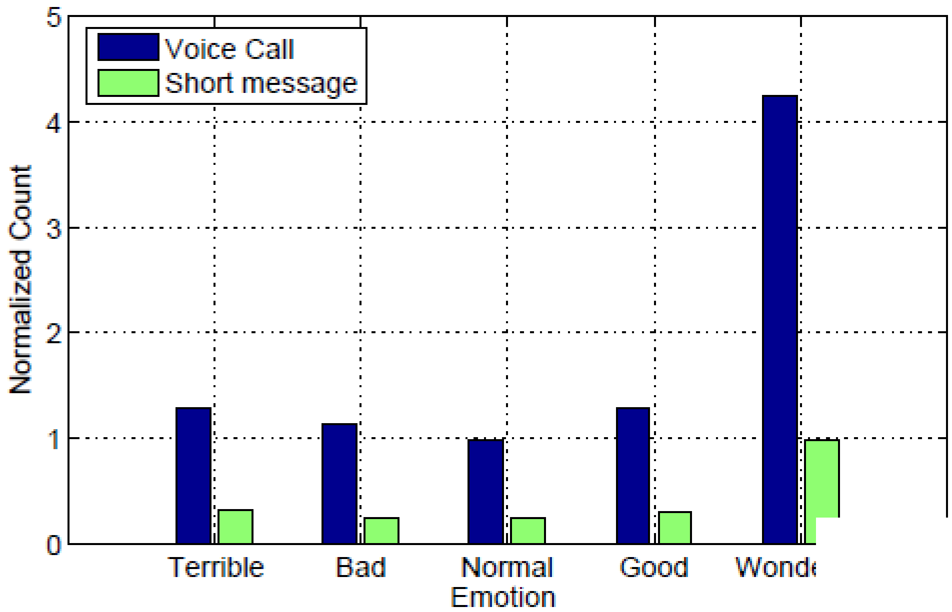
(a) Temporal correlation



(b) Time duration

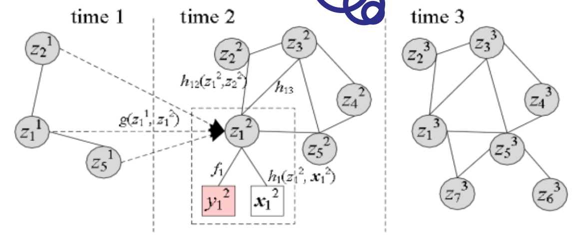
Temporal correlation

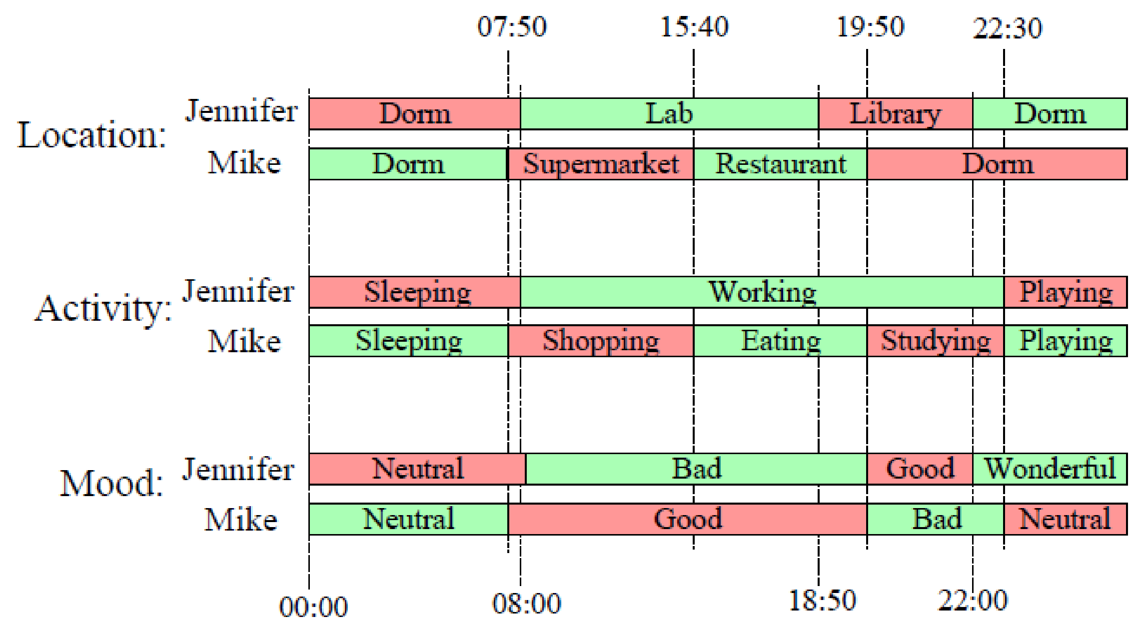
Observations (cont.)



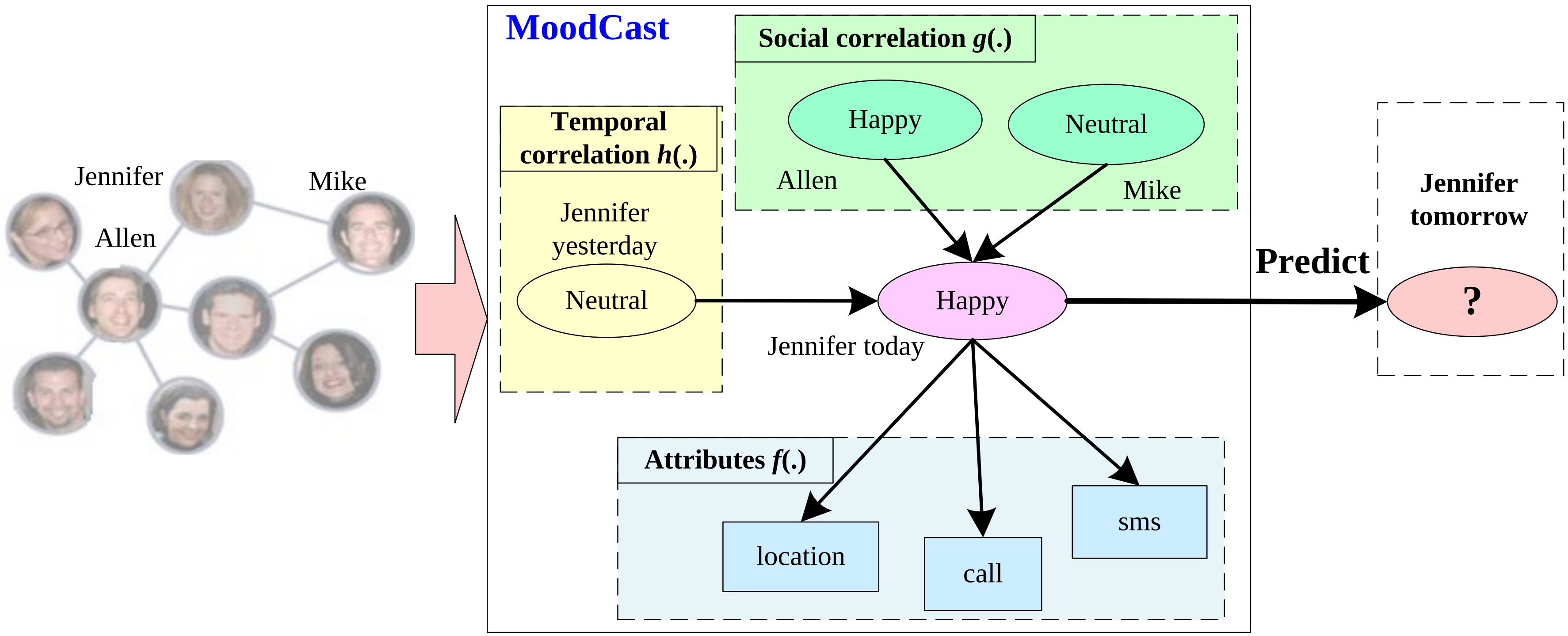
Calling (SMS) correlation

We should not split the data into different time windows …





MoodCast: Dynamic Continuous Factor Graph Model

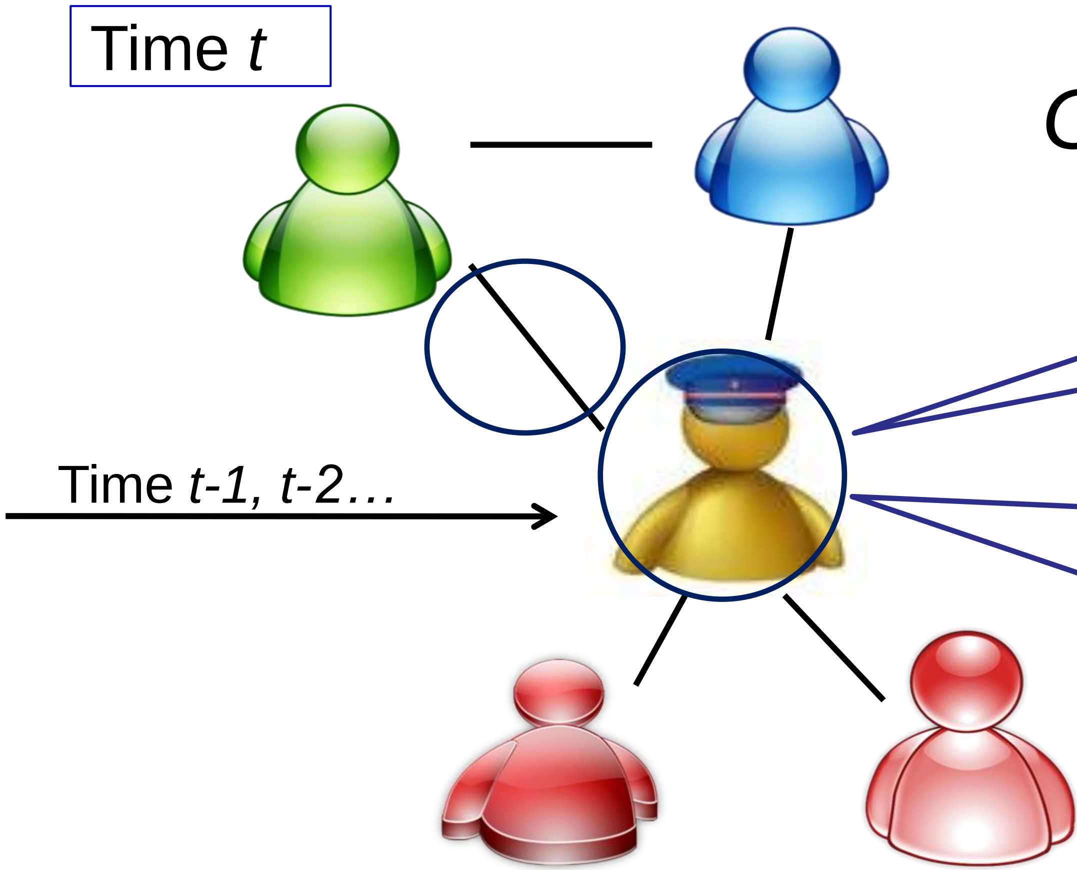


Our solution

1. We directly define continuous feature function;

2. Use Metropolis-Hasting algorithm to learn the factor graph model.

Problem Formulation



$G^t = (V, E^t, X^t, Y^t)$

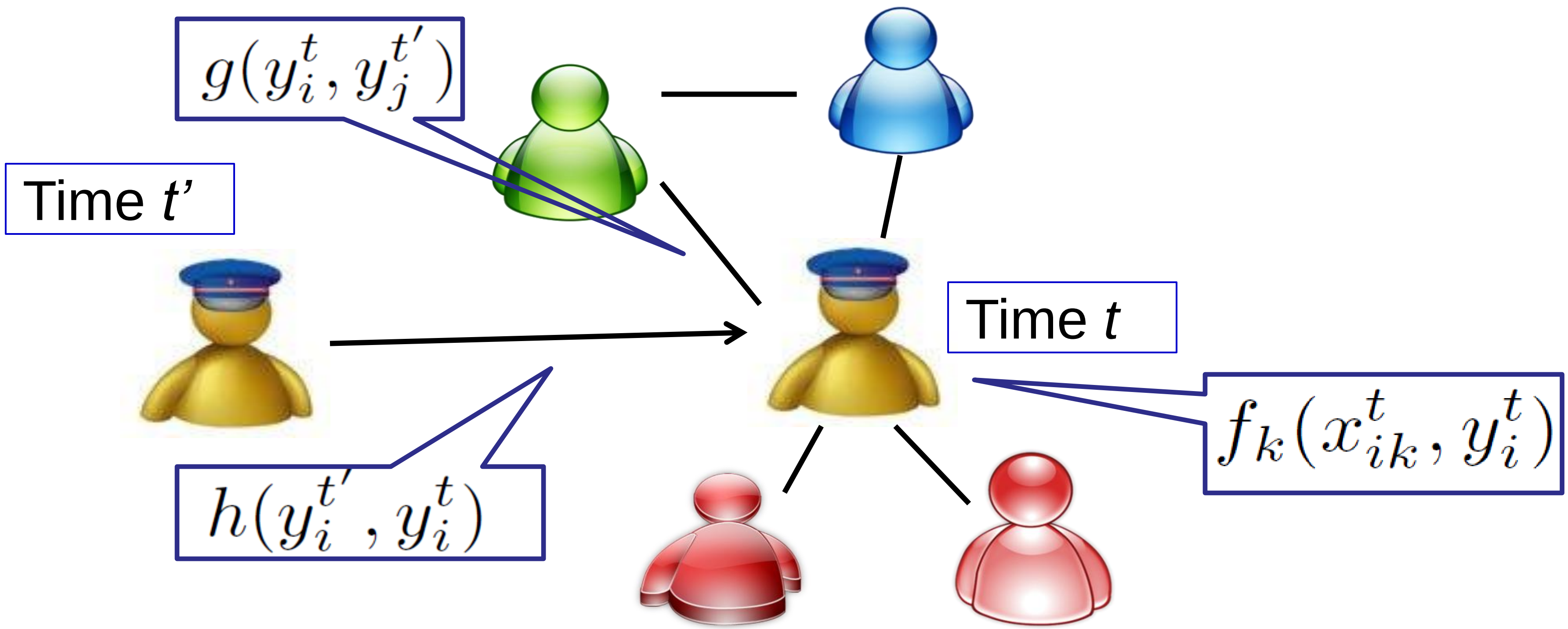
Emotion: Sad

Attributes:
- Location: Lab
- Activity: Working

Learning Task:

$$f(V, E^{(t+1)}, X^{(t+1)} | G^t) \rightarrow Y^{(t+1)}$$

Dynamic Continuous Factor Graph Model

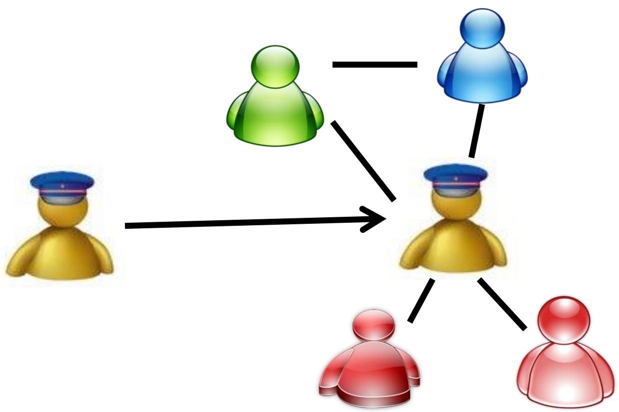


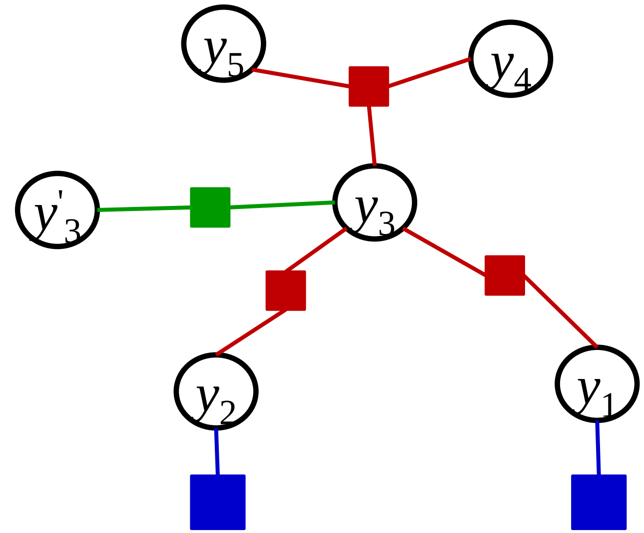
$f_k(x_{ik}^t, y_i^t)$: Binary function

$$g(y_i^t, y_j^{t'}) = \exp\{-\beta_{ji}(t-t')(y_i^t - y_j^{t'})^2\}$$

$$h(y_i^{t'}, y_i^t) = \exp\{-\lambda_i(t-t')(y_i^t - y_i^{t'})^2\}$$

Model Learning





$$p(Y|G^t) = \frac{1}{Z}\exp\{\sum_{v_i \in V}\sum_{x_{ik}^t \in X} \alpha_k f_k(x_{ik}^t, y_i^t)$$ **Attribute**

$$+ \sum_{v_j \in NB(v_i)} \sum_{(y_i^t, y_j^{t'}) \in Y^t} -\beta_{ji}(t-t')(y_i^t - y_j^{t'})^2$$ **Social**

$$+ \sum_{v_i \in V} \sum_{(y_i^t, y_i^{t'}) \in Y^t} -\lambda_i(t-t')(y_i^t - y_i^{t'})^2\}$$ **Temporal**

$$\theta^\star = \arg\max_\theta \; \log\, p(Y = y|x, \theta)$$

MH-based Learning algorithm

Input: number of iterations and learning rate η;
Output: learned parameters $\theta = (\{\alpha_k\}, \{\beta_{ji}\}, \{\lambda_i\})$;

Initialize $\theta = \{\alpha, \beta, \lambda\}$;
repeat
  % sample a new Y' according to $q(Y'|Y)$;
  $Y' \leftarrow q(Y'|Y)$;
  $\tau \sim \min(\frac{p(Y'|G^t,\theta)}{p(Y|G^t,\theta)}, 1)$;
  toss a coin s according to a $Bernoulli(\tau, (1-\tau))$;
  if $(s = 1)$ **then**
    % accept the new configuration Y';
    $Y \leftarrow Y'$;
    if $(Err(Y') < Err(Y)\ \&\ \Delta\theta F < 0)$ **then**
      $\theta^{new} \leftarrow \theta^{old} + \eta(\Delta\theta F)$;
    end
    else if $(Err(Y') > Err(Y)\ \&\ \Delta\theta F \geq 0)$ **then**
      $\theta^{new} \leftarrow \theta^{old} - \eta(\Delta\theta F)$;
    end
  end
until *convergence*;

Random Sampling

Update

Experiment

- Data Set

	#Users	Avg. Links	#Labels	Other
MSN	30	3.2	9,869	>36,000hr
LiveJournal	469,707	49.6	2,665,166	

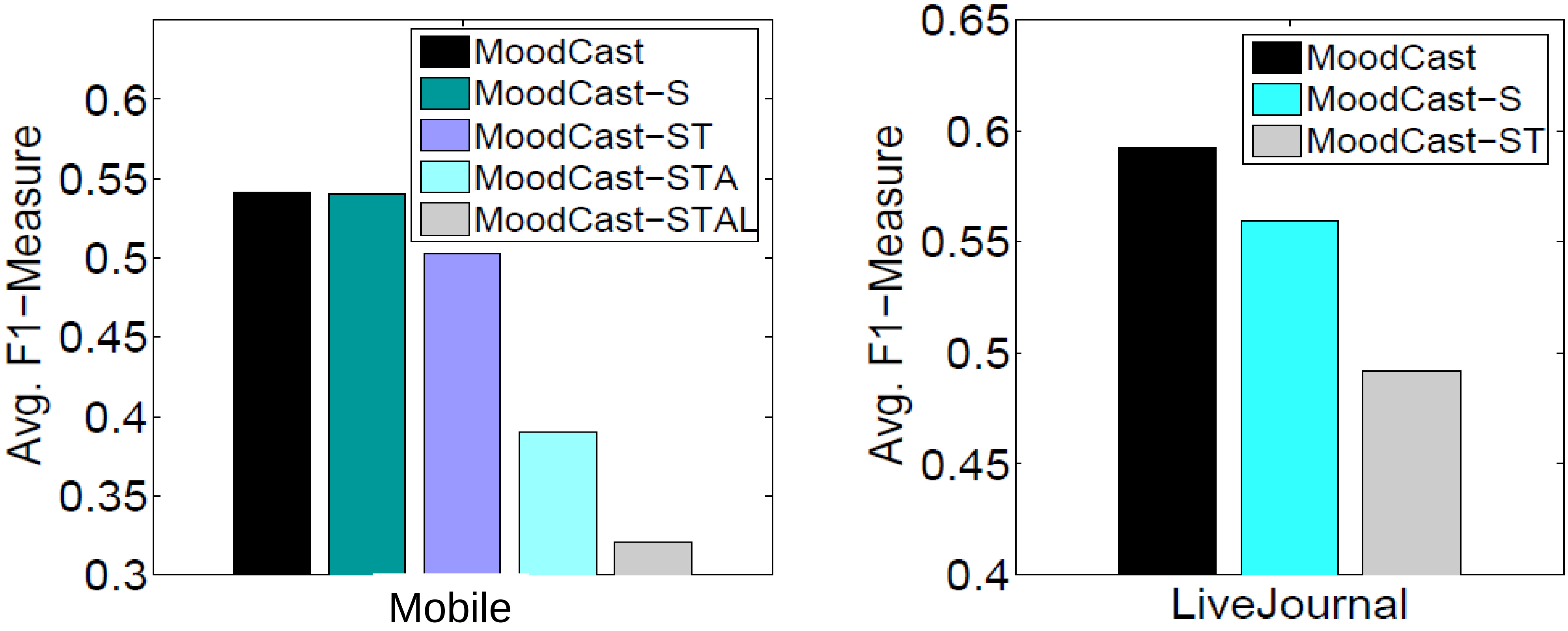
- Baseline
  - SVM
  - SVM with network features
  - Naïve Bayes
  - Naïve Bayes with network features
- Evaluation Measure:

  Precision, Recall, F1-Measure

Performance Result

Classifier	Method	MSN Dataset			LiveJournal Dataset		
		Precision	Recall	F1-score	Precision	Recall	F1-score
Positive	MoodCast	68.42	69.23	**68.82**	52.50	73.68	**61.32**
	SVM-Simple	60.88	71.08	65.58	49.56	48.57	49.06
	SVM-Net	59.12	72.70	65.21	50.72	60.29	55.09
	NB-Simple	67.30	56.21	61.25	57.08	43.34	49.27
	NB-Net	71.89	56.59	63.33	59.1	47.38	52.59
Neutral	MoodCast	67.78	76.57	**71.90**	59.61	84.92	**75.44**
	SVM-Simple	67.39	59.73	63.33	67.58	78.69	72.71
	SVM-Net	68.42	55.11	61.05	71.21	78.13	74.51
	NB-Simple	54.14	68.04	60.30	65.95	54.14	59.46
	NB-Net	51.06	71.62	59.62	61.70	61.53	61.61
Negative	MoodCast	30.77	13.95	19.20	45.45	54.98	**49.77**
	SVM-Simple	5.63	4.54	5.03	71.67	37.39	49.14
	SVM-Net	8.18	16.90	11.02	68.78	37.68	48.68
	NB	14.70	28.16	19.32	54.77	36.61	43.89
	NB-Net	17.88	32.08	**22.96**	51.70	41.18	45.84
Average	MoodCast	55.66	53.25	**53.31**	52.52	71.19	**62.17**
	SVM-Simple	44.63	45.12	44.65	62.94	54.83	56.97
	SVM-Net	45.24	48.23	45.76	63.57	58.70	59.42
	NB-Simple	45.38	50.80	46.95	59.26	44.69	50.87
	NB-Net	46.94	53.43	48.63	57.5	50.03	53.35

Factor Contributions



- All factors are important for predicting user emotions

Inferring Social Ties in Mobile Networks

PKDD 2011 (*Best Paper Runnerup*), WSDM 2012

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.

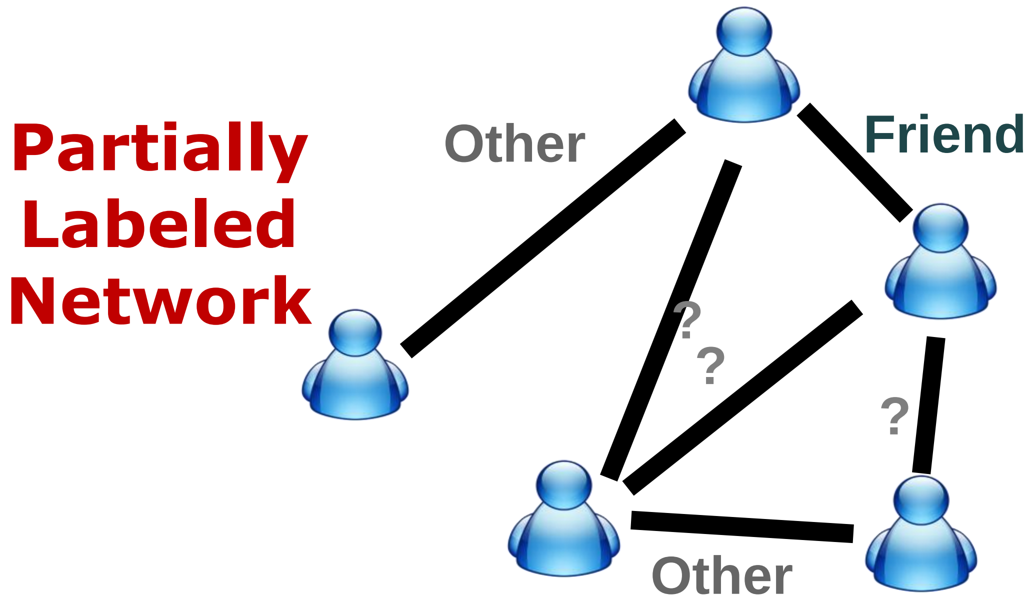
Even complex than we imaged!

- 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…

Problem Formulation

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



V: Set of Users

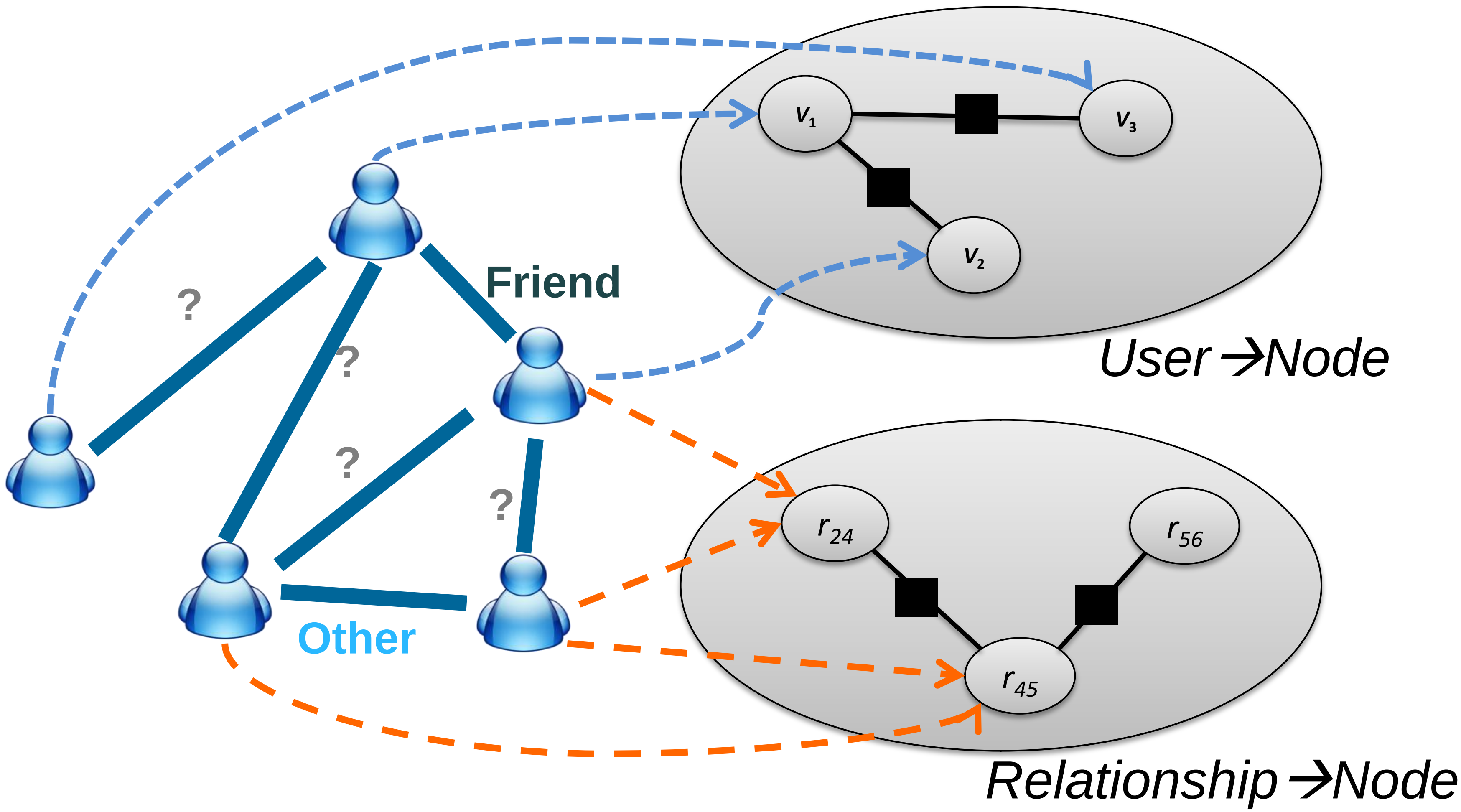
E^L,R^L: Labeled relationships

E^U: Unlabeled relationships

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

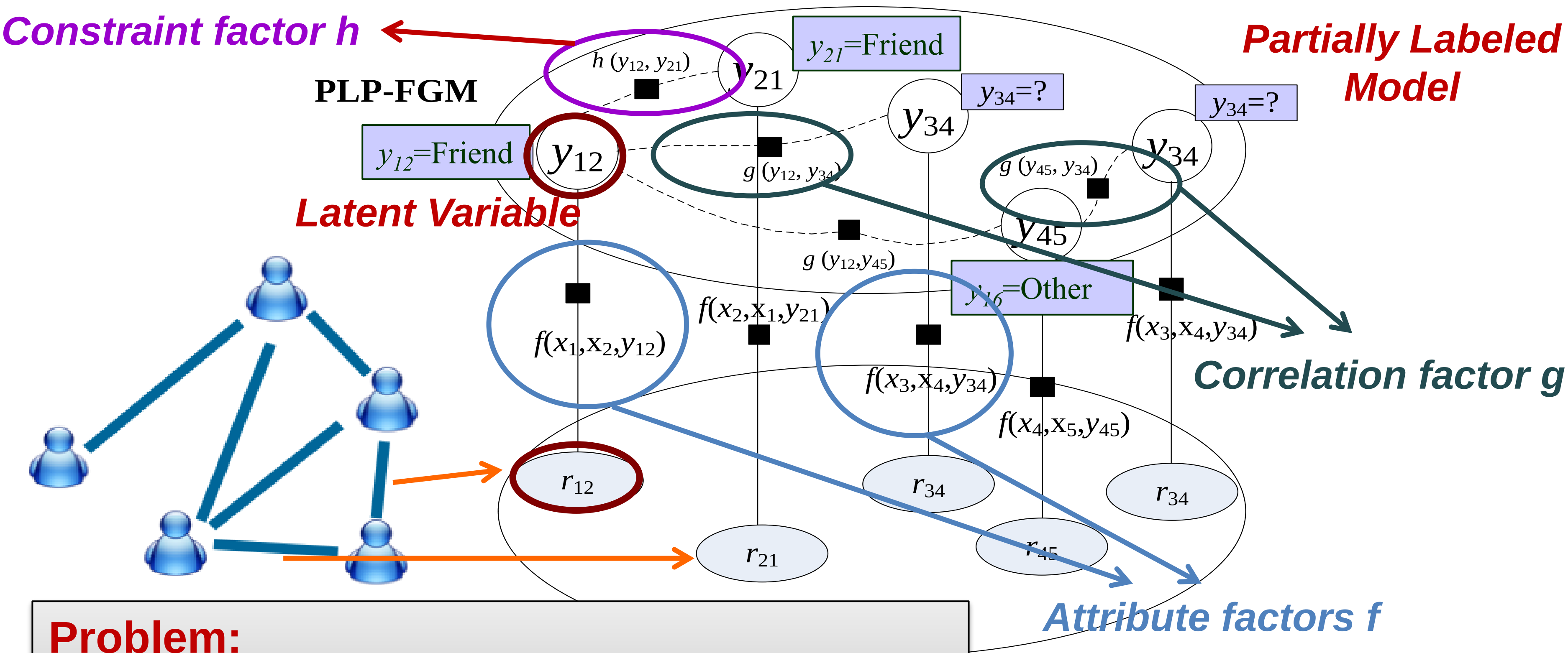
Output:
$f: G \rightarrow R$

Basic Idea



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

Partially Labeled Pairwise Factor Graph Model (PLP-FGM)



Problem:

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

Ma

Example:

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

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

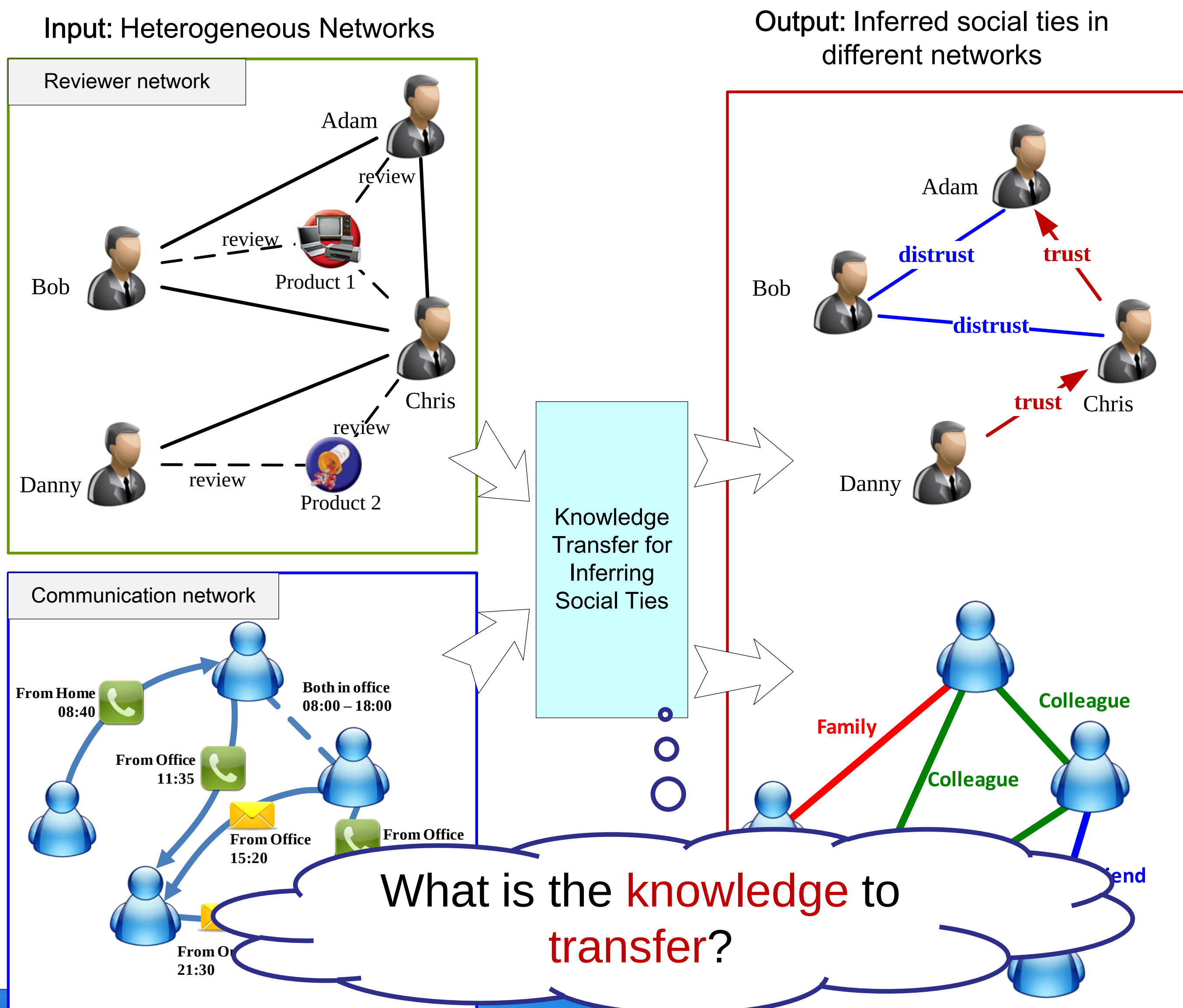
Gradient Ascent Method

Still Challenges?

Questions:

- How to obtain sufficiently training data?
- Can we leverage knowledge from other network?

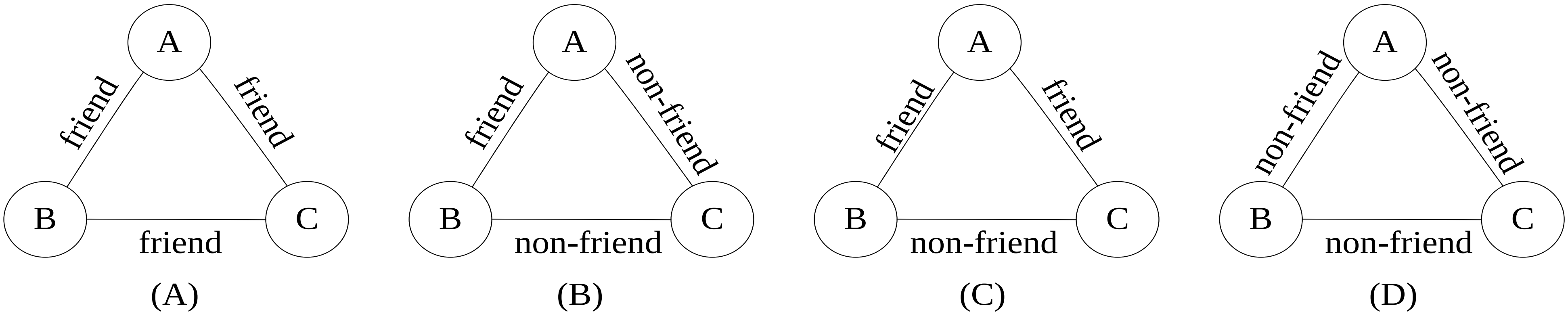
Inferring Social Ties Across Networks



What is the knowledge to transfer?

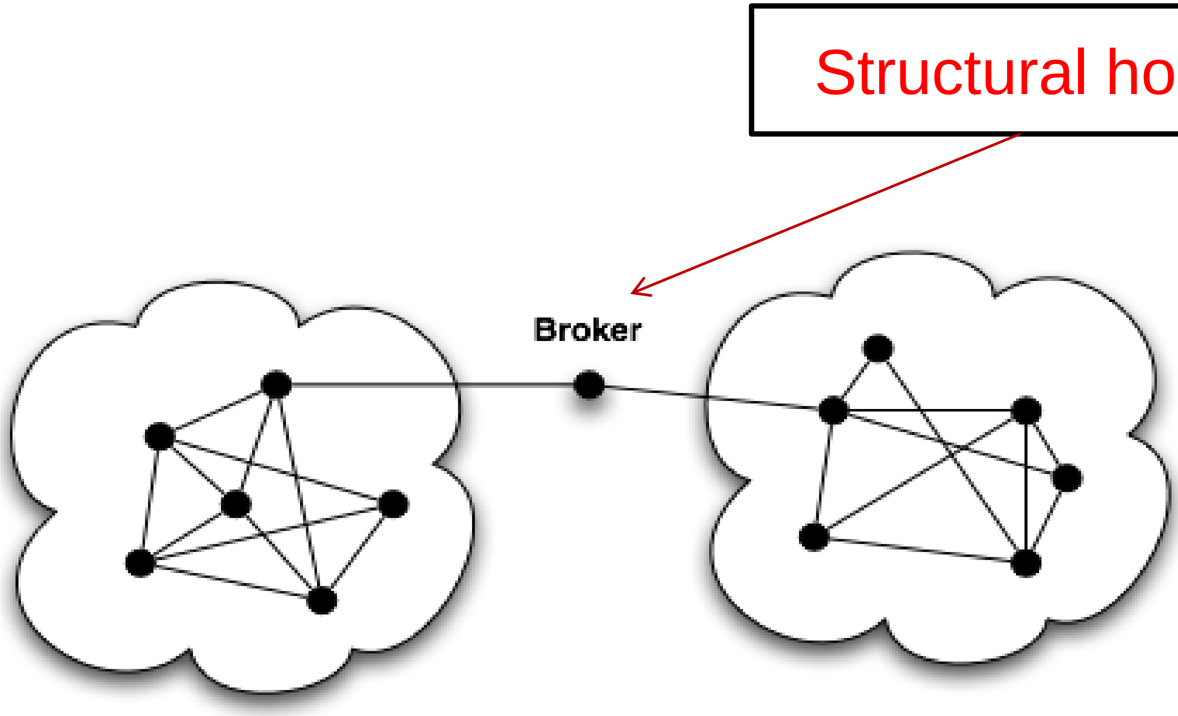
Social Theories

- Social balance theory
- Structural hole theory



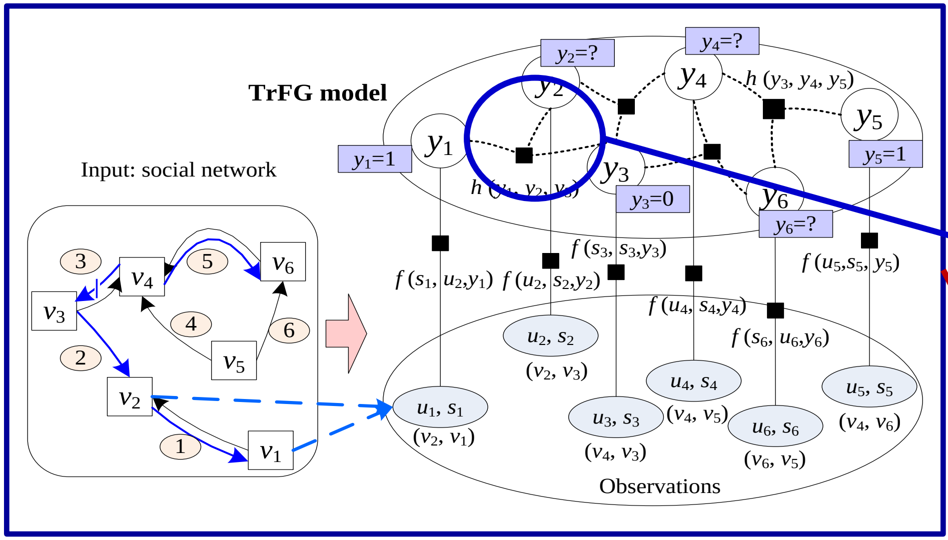
Social Theories—Structural hole

- Social balance theory
- Structural hole theory



Transfer Factor Graph Model

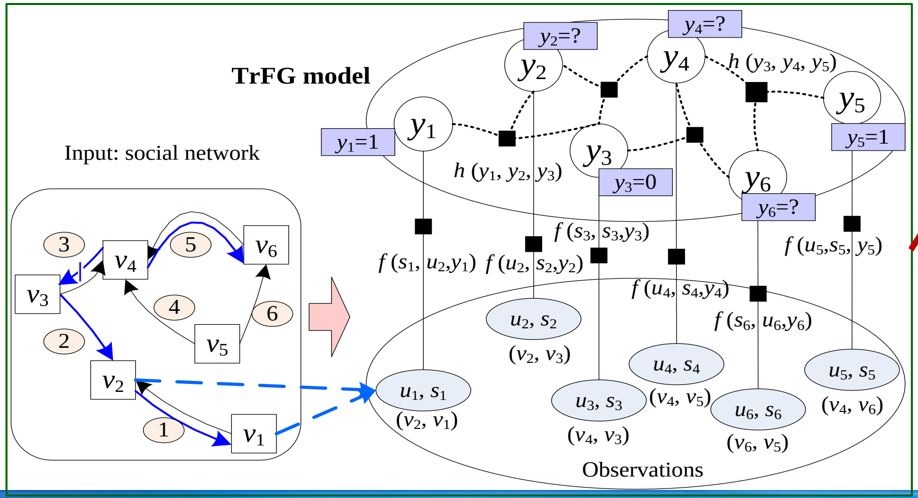
Coauthor network



Triad-based factor

Bridge via social theories

mobile



Mathematical Formulation

Features defined in
different networks

$$\begin{aligned}\mathcal{O}(\alpha,\beta,\mu) &= \mathcal{O}_S(\alpha,\mu)+\mathcal{O}_T(\beta,\mu)\\ &= \sum_{i=1}^{|V_S|}\sum_{j=1}^{d}\boxed{\alpha_j g_j(x_{ij}^S, y_i^S)} + \sum_{i=1}^{|V_T|}\sum_{j=1}^{d'}\boxed{\beta_j g'_j(x_{ij}^T, y_i^T)}\\ &\quad + \sum_k \mu_k\Big(\sum_{c\in G_S}\boxed{h_k(Y_c^S)} + \sum_{c\in G_T}\boxed{h_k(Y_c^T)}\Big)\\ &\quad - \log Z\end{aligned}$$

Triad-based features shared
across networks

Data Sets

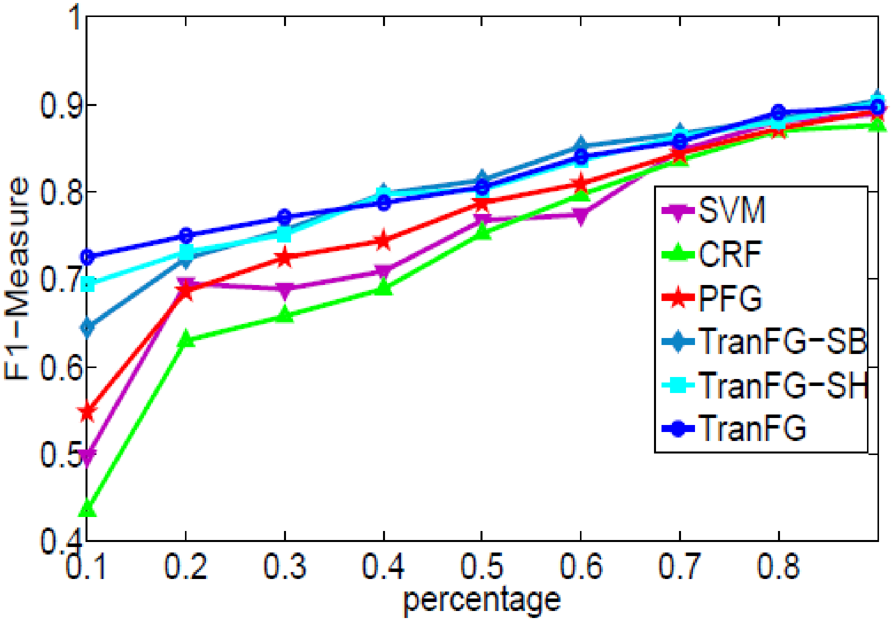
- **Epinions** a network of product reviewers: 131,828 nodes (users) and 841,372 edges
  - trust relationships between users
- **Slashdot**: 82,144 users and 59,202 edges
  - “friend” relationships between users
- **Mobile**: 107 mobile users and 5,436 edges
  - to infer friendships between users

Results

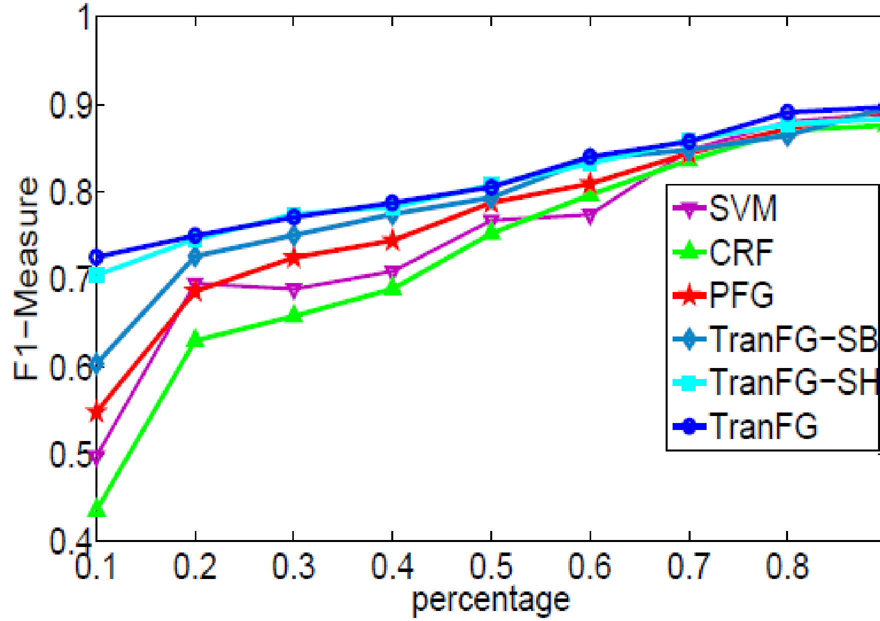
Data Set	Method	Prec.	Rec.	F1
Mobile	SVM	89.83	59.55	71.62
	CRF	94.55	54.17	68.87
	PFG	100.00	59.24	74.40
Epinions to Mobile (40%)	TranFG	82.39	83.44	82.91
Slashdot to Mobile (40%)	TranFG	72.58	85.99	78.72

SVM and CRF are two baseline methods;
PFG is the proposed partially-labeled factor graph model;
TranFG is the proposed transfer–based factor graph model.

Varying the percent of the labeled data

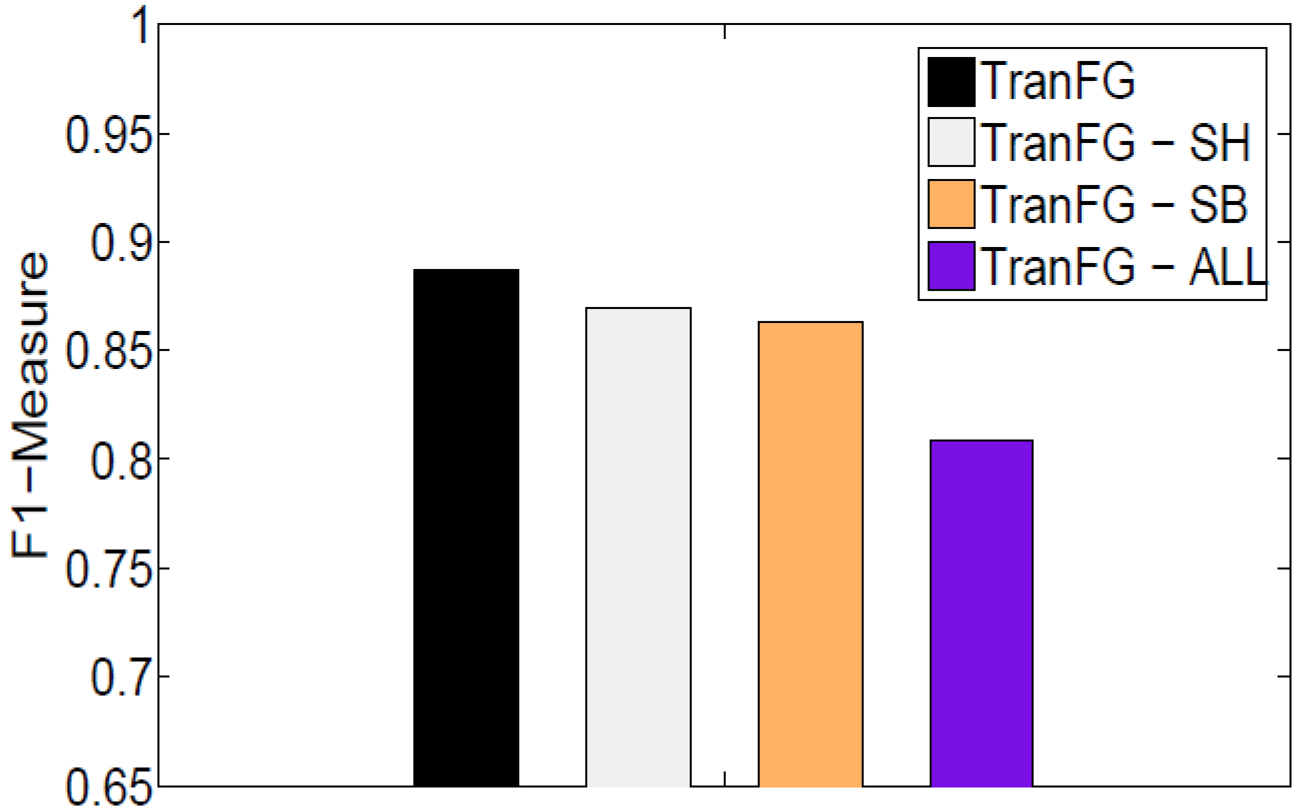


Epinions-to-Mobile



Slashdot-to-Mobile

Factor contribution analysis



SH-Structural hole;
SB-Social balance.

Conclusions

- **Moodcast: emotion prediction**
  - – Emotion stimulates the mind 3000 times quicker than rational though;
  - – We demonstrate that it is possible to accurately predict users' emotions in mobile network.
- **Inferring social ties**
  - – different types of social ties have essentially different influence on people;
  - – By incorporating social theories, our proposed model can significantly improve (+4-14%) the inferring accuracy.

Future Work

- **Emotion:**
  - Emotion diffusion in the mobile network;
  - Predicting activities and emotions simultaneously.

- **Inferring social ties:**
  - Inferring complex relationships between users, e.g., family, colleague, manager-subordinate;
  - Active learning for inferring social ties.

Related Publications

- Jie Tang, Tiancheng Lou, and Jon Kleinberg. Inferring Social Ties across Heterogenous Networks. **WSDM’12**.
- Chi Wang, Jiawei Han, Yuntao Jia, Duo Zhang, Yintao Yu, Jie Tang, Jingyi Guo. Mining Advisor-Advisee Relationships from Research Publication Networks. **KDD’10**.
- Wenbin Tang, Honglei Zhuang, and Jie Tang. Learning to Infer Social Relationships in Large Networks. **PKDD'11.** **(Best Student Paper Runner-up)**
- Jie Tang, Yuan Zhang, Jimeng Sun, Jinghai Rao, Wenjing Yu, Yiran Chen, and ACM Fong. Quantitative Study of Individual Emotional States in Social Networks. **IEEE Transactions on Affective Computing**. 2011
- Yuan Zhang, Jie Tang, Jimeng Sun, Yiran Chen, and Jinghai Rao. MoodCast: Emotion Prediction via Dynamic Continuous Factor Graph Model. **ICDM'10**.
- Chenhao Tan, Jie Tang, Jimeng Sun, Quan Lin, and Fengjiao Wang. Social Action Tracking via Noise Tolerant Time-varying Factor Graphs. **KDD’10**.
- Jie Tang, Jimeng Sun, Chi Wang, and Zi Yang. Social Influence Analysis in Large-scale Networks. **KDD'09**.

Thanks!

HP: http://keg.cs.tsinghua.edu.cn/jietang/

System: http://arnetminer.org