# Social Network Analysis -- Recommendations

Instructor: Jen-Wei Huang

Office: 92528 in the EE building
jwhuang@mail.ncku

# Introduction

- Social network services have attracted lots of attention
- Social networking websites provide users to establish their own personal communities or social networks based on relationships of friends.

http://www.facebook.com/

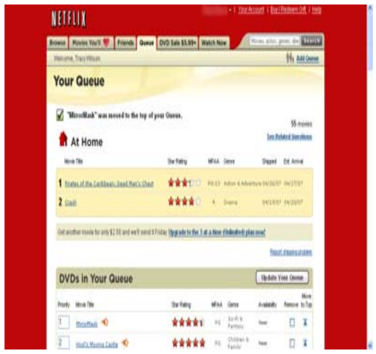
http://twitter.com/

# Introduction

- Recommendation systems have been proposed to integrate with business websites and social networking websites.

http://www.amazon.com/

http://www.ebay.com/



https://www.netflix.com/



# Recommendation Systems

- Recommendation systems usually utilize the following information from the social network.
  - Popularity
  - Similarity
  - Familiarity

# GroupBuyer: A Personalized Group Buying Event Recommender System Using Social Information Filtering

Yun-Hui Hung, Jen-Wei Huang, Ming-Syan Chen

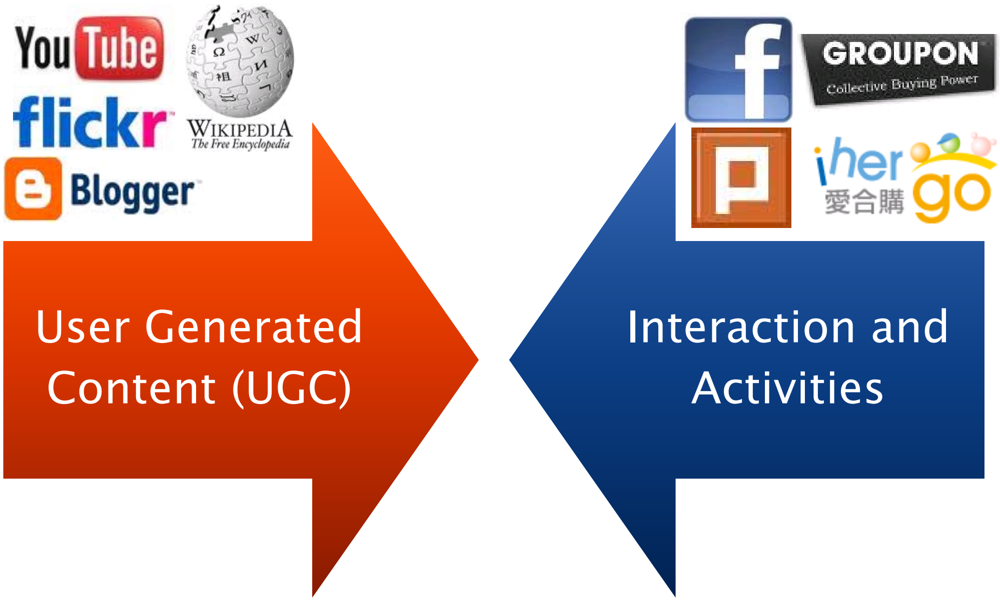
*2011 International Workshop on Behavior Informatics (BI 2011) joint with the 15th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'11)*, May 24, 2011.

# P-SERS: Personalized Social Event Recommender System

Yun-Hui Hung, Jen-Wei Huang, Ming-Syan Chen

*Behavior Computing: Modeling, Analysis, Mining and Decision, Springer, April, 2012.*

# Online Communities





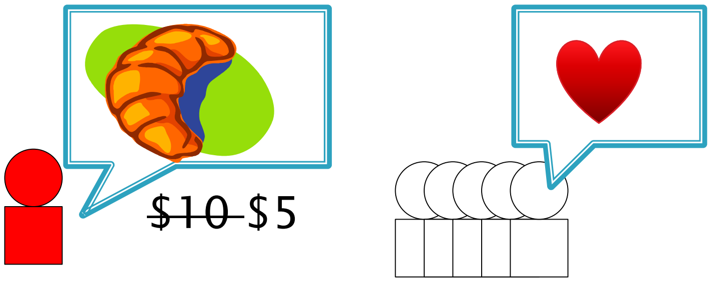

# Social Event Recommender System


| Social Information Filtering | Examples |
|---|---|
| Collaborative Filtering (CF) | •Similar users [2]<br>•Expert [3] |
| Social Filtering (SF) | •Friends Relations [4], [5] |

[2] Resnick, in CSCW, 1994.
[3] Amatriain, in SIGIR, 2009.
[4] Guy, in RecSys, 2009.
[5] Guy, in CHI 2008.

# Online Group Buying Communities

- A group buying event

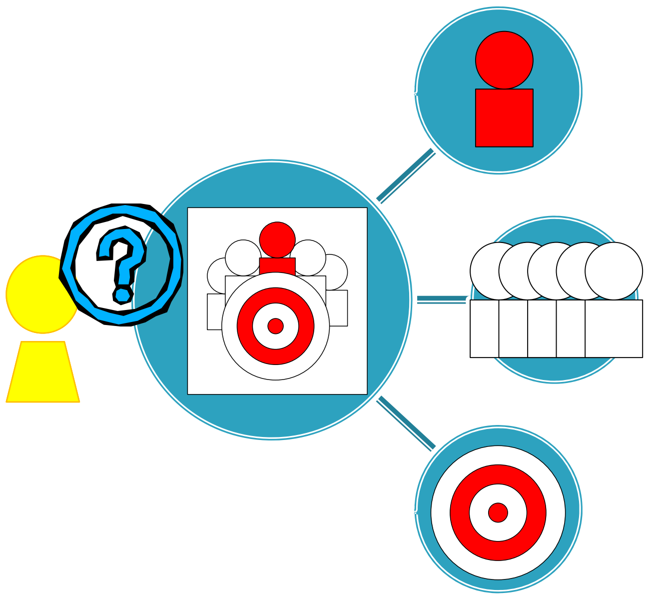



- Social networking functions
  - Knowing friends
  - Joining events



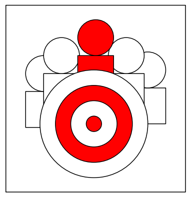
# Major Components

- Initiator $I_{E_i}$
- Participants $P_{E_i}$
- Target $T_{E_i}$



# Contribution

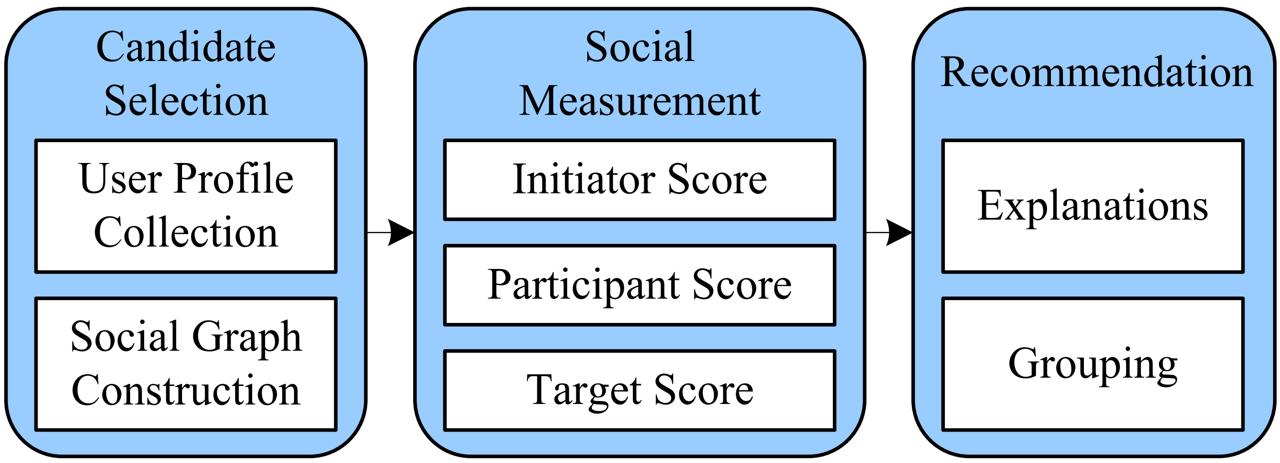
- Propose GroupBuyer, a personalized group buying event recommender system



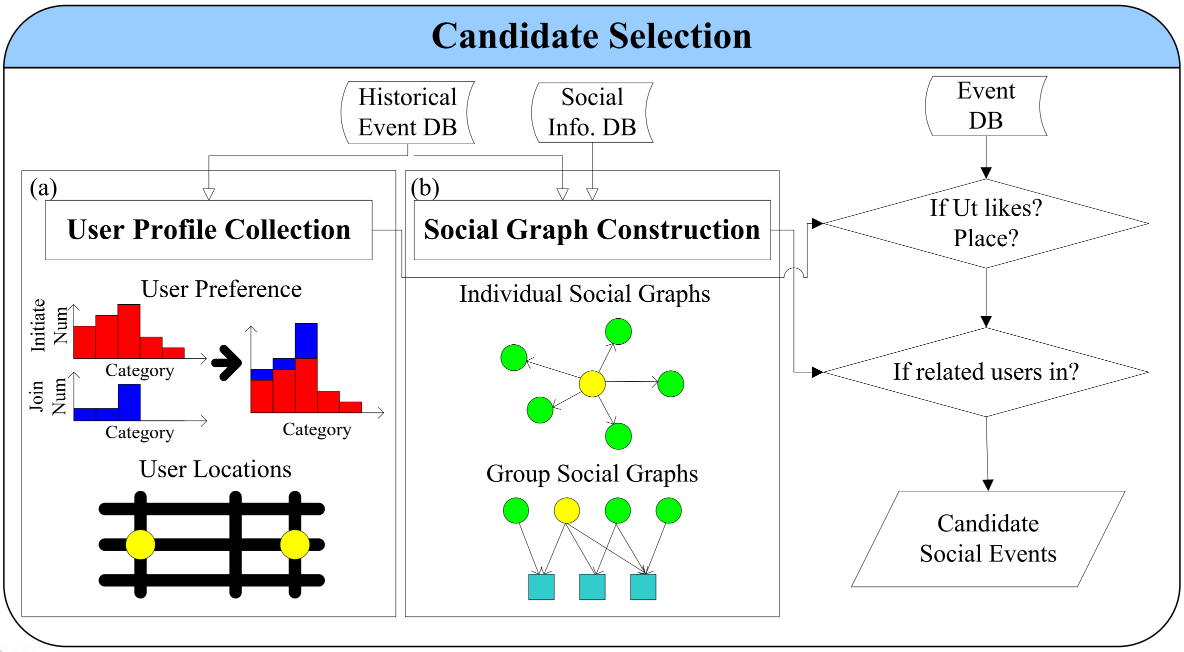
- Use social information
- Model three major components
- Recommend group buying events



# System Model



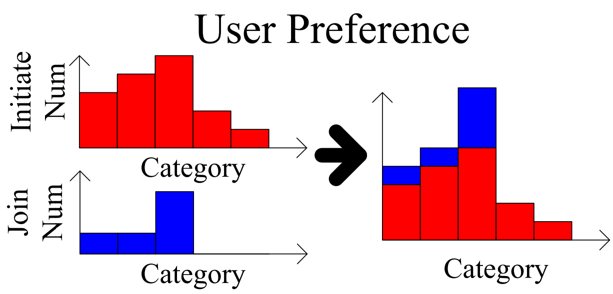
# Candidate Selection



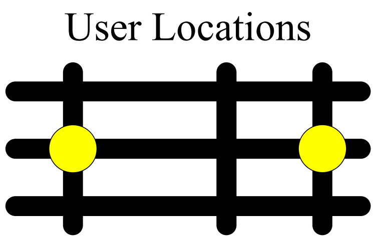



# User Profile Collection

- User preference
  - Historical events

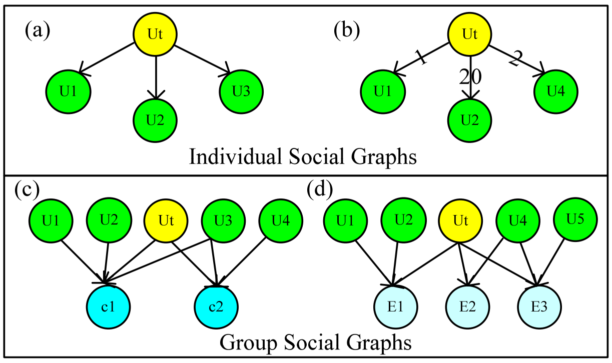



- User locations
  - User settings
  - Historical events



# Social Graph Construction

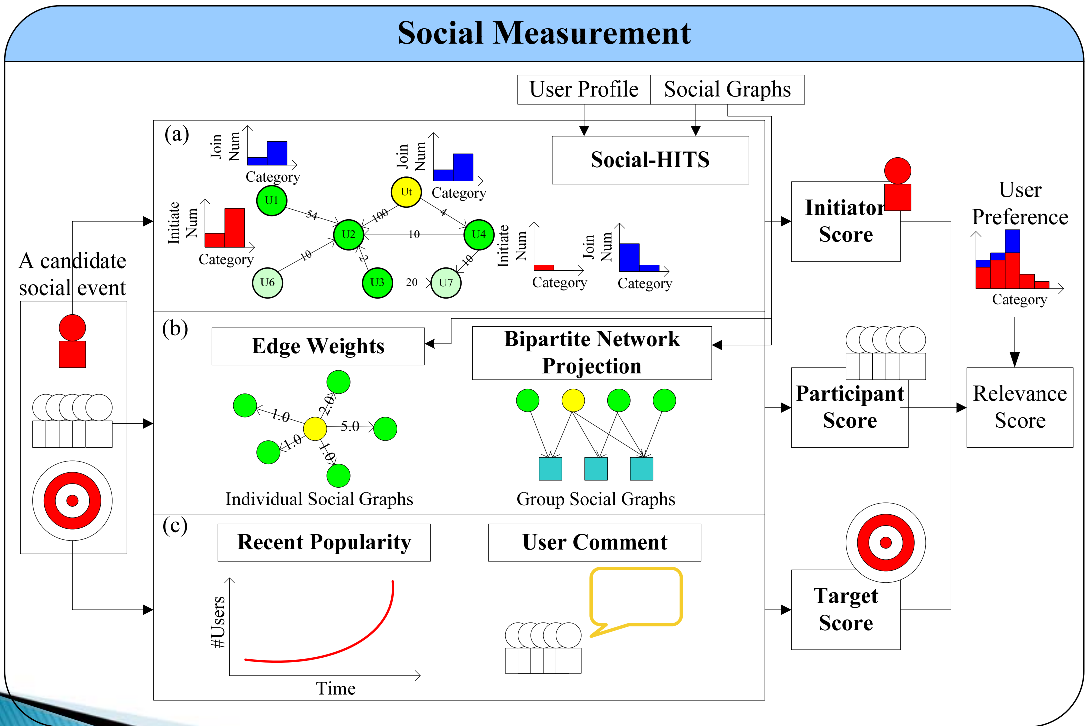
- Social information
  - Friends
  - Co-buying relations




| Graph type | Node type | Level | Social behavior example |
|---|---|---|---|
| Individual social graph | •User | Individual | Add as a friend |
| Group social graph | •User<br>•Group | Group | Co-join a club |

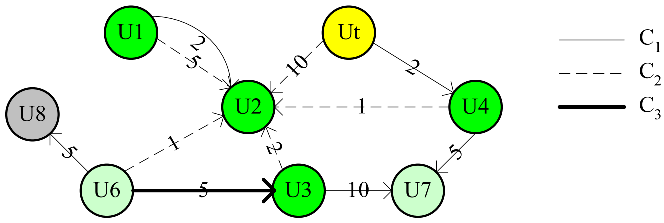


# Social Measurement

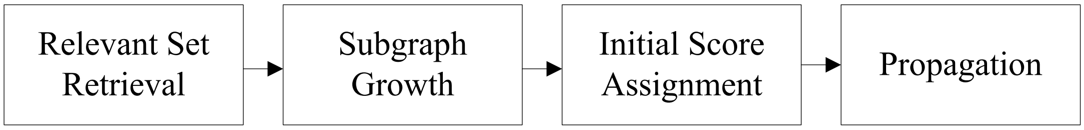


# Initiator Score

- Goal: expertise of initiating
  - Related to the target user
- Expertise graph[6]
  - Joining behavior



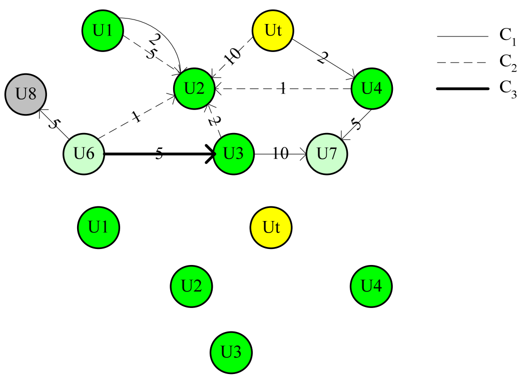

- Social-HITS, similar to HITS[7]

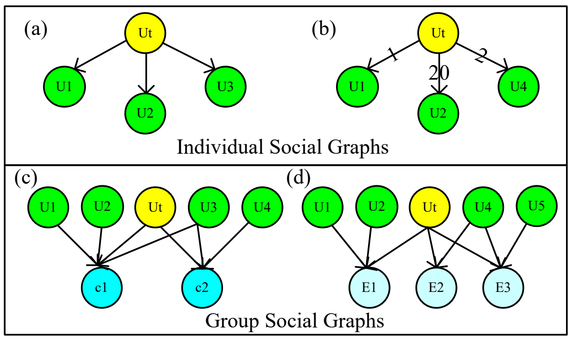


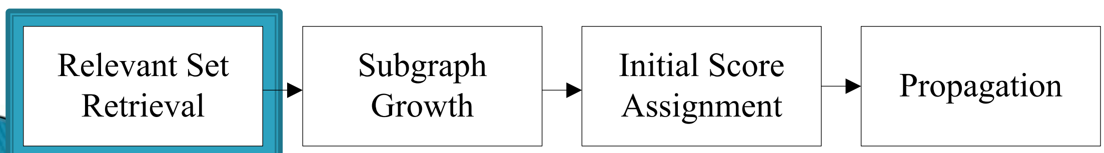



# Relevant Set Retrieval

- Related users



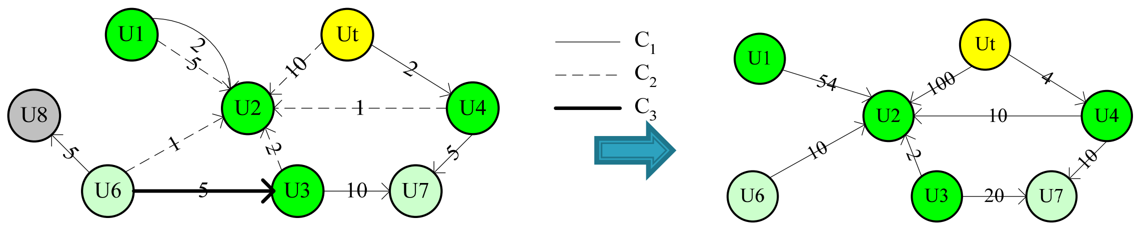



# Subgraph Growth

- Incident edges of relevant set
  - Aggregated by user preference

$$\text{Pref}(U_t, C_1) = 2$$
$$\text{Pref}(U_t, C_2) = 10$$

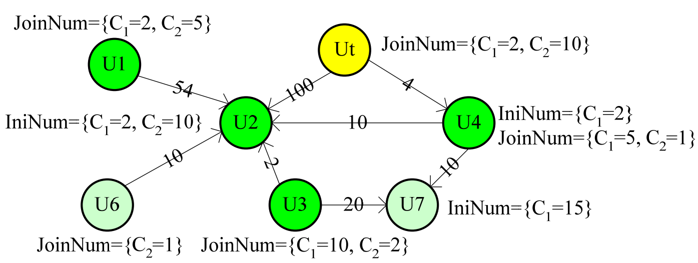



Relevant Set Retrieval → **Subgraph Growth** → Initial Score Assignment → Propagation



# Initial Score Assignment

- Consider user experience
  - Authority score: initiating
  - Hub score: joining
  - Weighted-sum by user preference



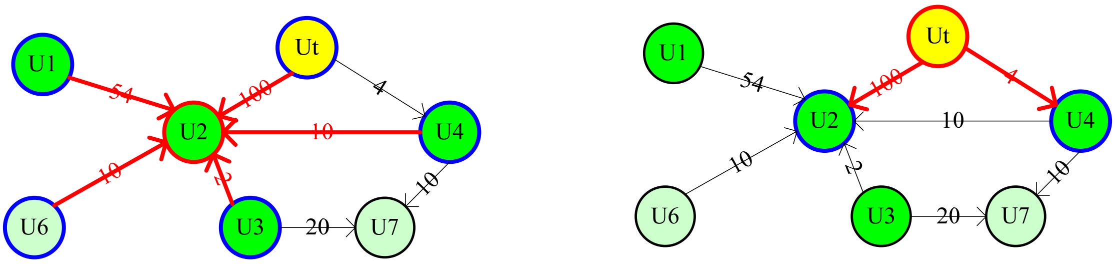

$$InitialAuthority(U_2)$$
$$= (2*2+10*10)/104 = 1.0$$
$$InitialHub(U_1)$$
$$= (2*2+10*5)/104 = 0.52$$

Relevant Set Retrieval → Subgraph Growth → **Initial Score Assignment** → Propagation

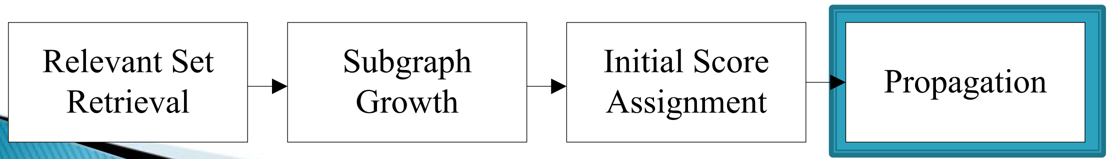
# Propagation

- Automate user recommendation

Hub Propagation                Authority Propagation




- Run iterations
  - Weighted-sum with initial scores and normalize



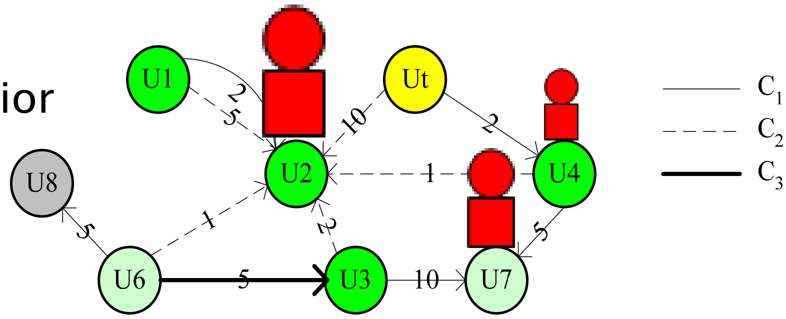



# Social-HITS

- Initiator score

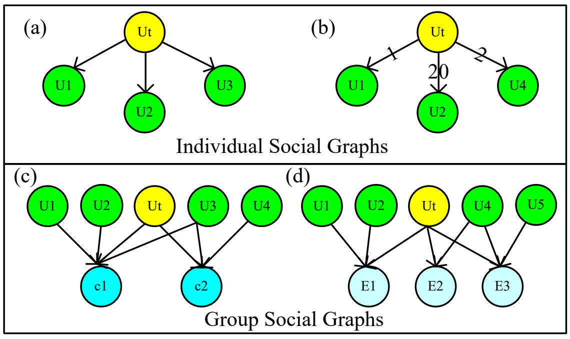
$$IS(I_{E_i}, U_t) = Authority(I_{E_i})$$

- Relative expertise to $U_t$
  - Related users
  - Preference
  - Experience
  - Joining behavior



# Participant Score

- Goal: social influence
- Related users

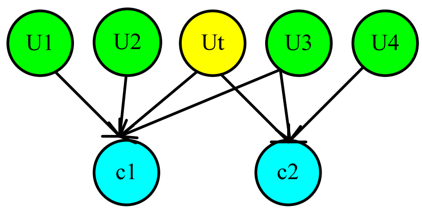



| Graph type | Influential power $p(U_j, U_t)$ |
|---|---|
| Individual social graphs | Weights on direct links |
| Group social graphs | Bipartite network projection [8] |



# Bipartite Network Projection

- $U_t$ is influenced by the users who co-join the group nodes.




$$f'(U_t) = \frac{1}{2} f(c_1) + \frac{1}{2} f(c_2)$$

$$f(c_1) = \frac{1}{4} f(U_1) + \frac{1}{4} f(U_2) + \frac{1}{4} f(U_t) + \frac{1}{4} f(U_3),$$

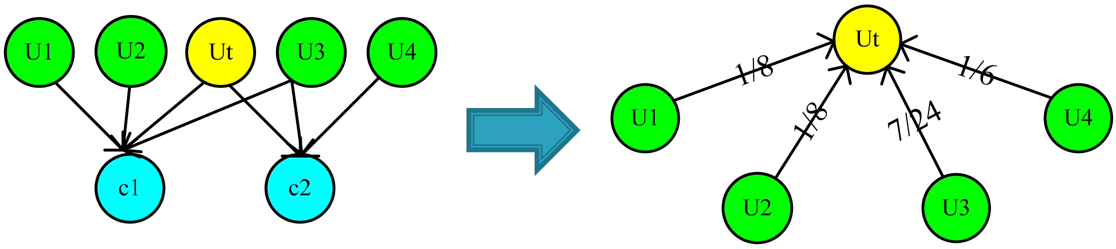
$$f(c_2) = \frac{1}{3} f(U_t) + \frac{1}{3} f(U_3) + \frac{1}{3} f(U_4),$$

$$f'(U_t) = \frac{1}{8} f(U_1) + \frac{1}{8} f(U_2) + \frac{7}{24} f(U_t) + \frac{7}{24} f(U_3) + \frac{1}{6} f(U_4).$$

# Bipartite Network Projection

- Projection graph

$$f'(U_t) = \frac{1}{8} f(U_1) + \frac{1}{8} f(U_2) + \frac{7}{24} f(U_t) + \frac{7}{24} f(U_3) + \frac{1}{6} f(U_4).$$




- Influential power $p(U_j, U_t)$
  - Projection edge weight



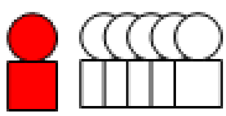
# Participant Score

- Influential power $p(U_j, U_t)$
  - Individual graphs: edge weights
  - Group graphs: projection edge weights

$$p(U_j, U_t) = \hat{p}_{friend}(U_j, U_t) + \hat{p}_{join}(U_j, U_t)$$

$$+ \hat{p}_{co-join}(U_j, U_t) + \hat{p}_{co-club}(U_j, U_t)$$

- Participant score

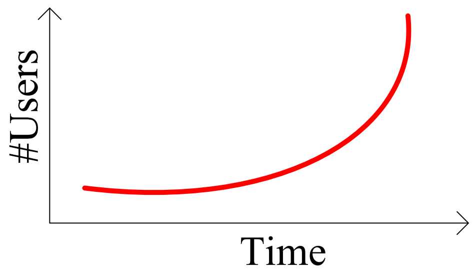


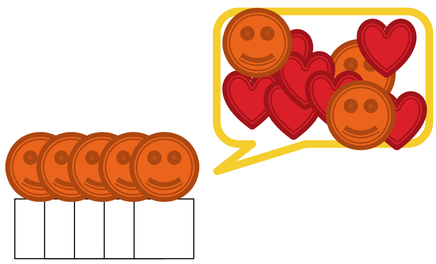
$$PS(P_{E_i} + I_{E_i}, U_t) = p(I_{E_i}, U_t) + \sum_{U_j \in P_{E_i}} p(U_j, U_t)$$

# Target Score

- Goal: global popularity of the target
- Two attributes
  - Recent popularity
  - Good comments

$$TS(T_{E_i}, U_t) = Num * (1 + avgCmt)$$





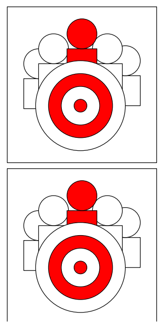
# Relevance Score

- Every candidate event $E_i$

$$RS(E_i, U_t) = \Pr ef(C_{E_i}, U_t) * \left[\alpha * IS(I_{E_i}, U_t) + \beta * PS(I_{E_i} + P_{E_i}, U_t) + \gamma * TS(T_{E_i}, U_t)\right]$$

- Recommendation list
  - Ranking candidates by $RS(E_i, U_t)$

Ranking Candidates



# Dataset and Prior User Study

- Group buying event recommendation on IHERGO[1] (from 2010/1/1~2010/4/30)
  - Historical group buying events
  - 460,000 events, 124,000 users
- Three parameters by 5000 questionnaires
  - Initiator 60%
  - Target 70%
  - Participants 40%

[1] IHERGO, http://www.ihergo.com



# Experiment Settings

- Recommendation satisfaction
  - Comparison with Random+Pl, CF+Pl, SF+Pl
  - 671 users
  - Top-15 lists

* Pl: place constraint

| Method | #Users |
|---|---|
| Random+Pl | 117 |
| CF+Pl | 78 |
| SF+Pl | 121 |
| GroupBuyer | 178 |

# Recommendation Satisfaction

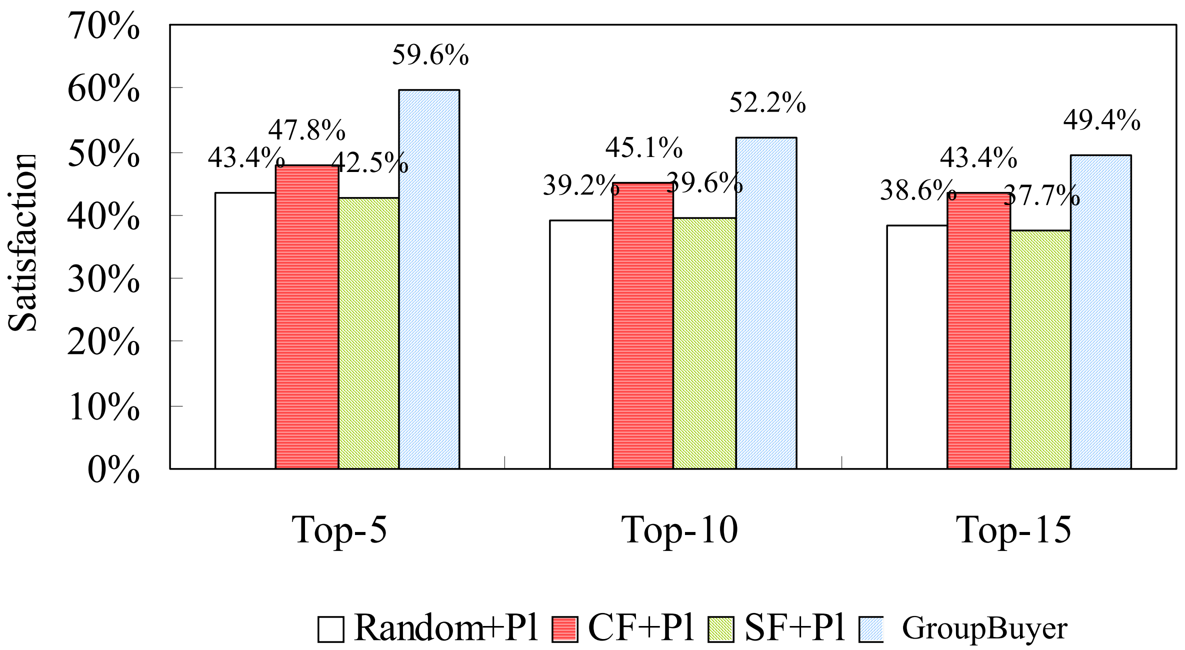
- User satisfaction with top-k list
  - k=1, satisfaction=1; k=2, satisfaction=1/2

$$Satisfaction(k, method) = \frac{\sum_{U_j \in method} \sum_{l=1}^{k} \left(ans_l(U_j) * \frac{1}{l}\right)}{|U_j \in method| \sum_{l=1}^{k} \frac{1}{l}}$$

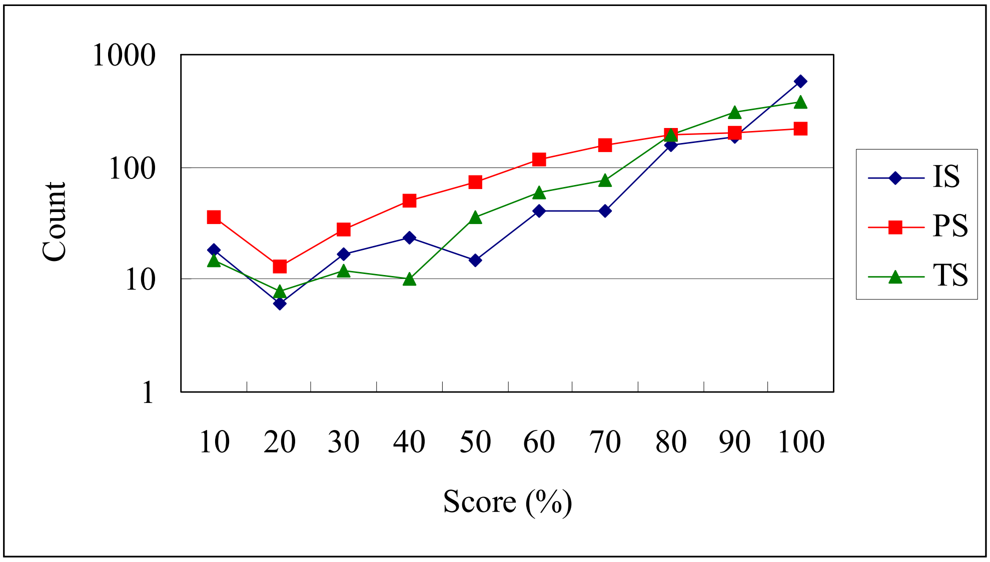

Event: Noodle Group Buying Event
Initiator: Alice
Store: Boss Q Noodles
Place: Taipei Main Station
**Are you interested in this event? YES, NO.**



# Satisfaction Comparison



# Score Trends






# Conclusions

- We proposed GroupBuyer to recommend the most relevant events to users
  - Model three major components
  - Recommend group buying events for the online group buying community, IHERGO

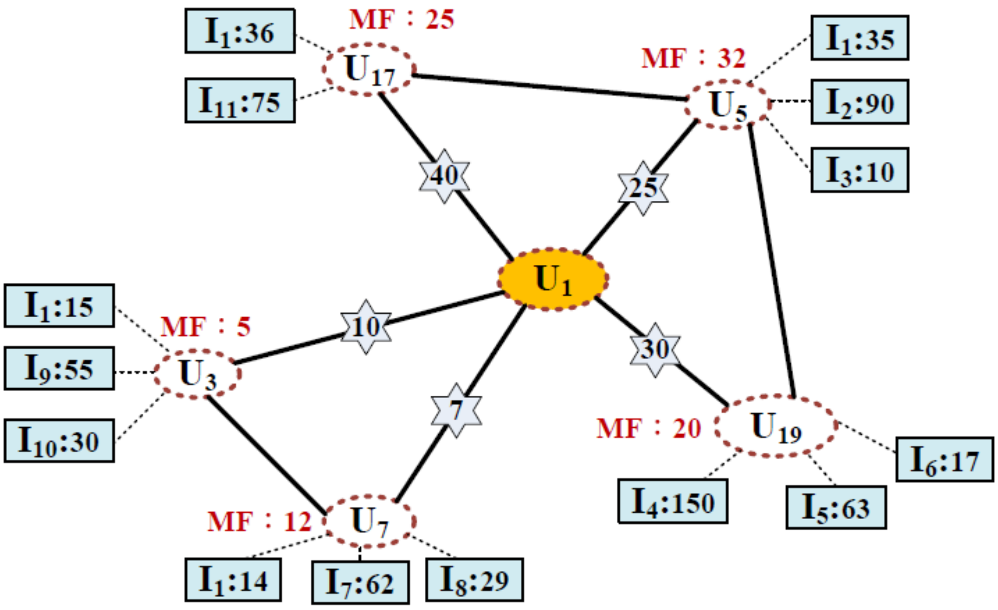
# Discovering Unknown But Interesting Items on Personal Social Network

Juang-Lin Duan, Shashi Prasad, Jen-Wei Huang

*Proc. of the 16th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'12), pp. 145-156, May 29-Jun. 1, 2012.*

# Introduction

- Traditional recommendation systems may have the following problems:
  - Popular items always occupy the recommendation list but they are usually already known by the user.
  - Items recommended by familiar users who frequently communicate with the target user may not be interesting.
  - Items from similar users with lower popularity are ignored.






# Introduction

- We propose UbiMiner, Unknown But Interesting Miner, to discover unknown but interesting items through the target user's social networks.
- UbiMiner considers
  - the popularity of items,
  - the social behavior of the target user, and
  - the similarity between users

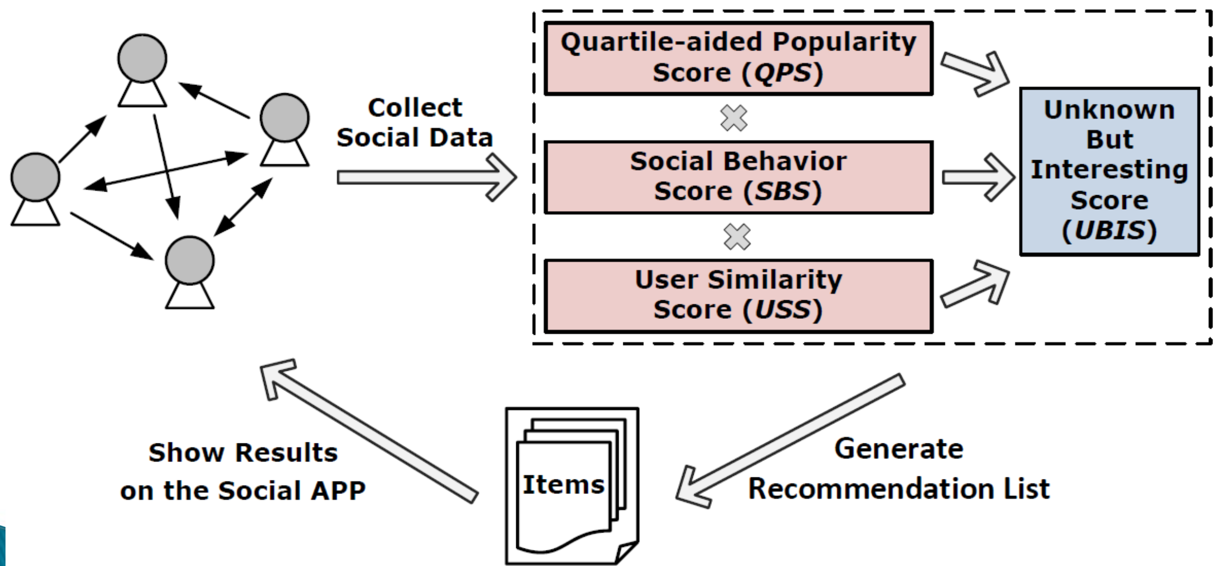
# Preliminary

- There are three major ways to develop recommendation systems.
  - Content-based recommendation
  - Collaborative filtering
  - Hybrid approaches

- Others



# System Architecture



# UBIS

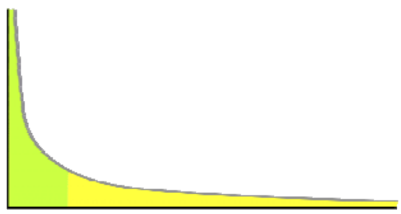
- Unknown But Interesting Score (UBIS) contains:
  - Quartile-aided Popularity Score (QPS) of each item.
  - Social Behavior Score (SBS) by considering social interactions of users.
  - User Similarity Score (USS) of each friend of the target user.



# Quartile-aided Popularity Score

- Popularity Score, $PS_{iv}$ , is defined as the popularity, i.e., number of comments and likes, of a certain item $i$ posted by user $v$.

- We use the concept of quartile and the long tail phenomenon, and sort items by popularity score $PS_{iv}$ ascendingly.



http://en.wikipedia.org/wiki/File:Long_tail.svg

# Quartile-aided Popularity Score

- Concept of quartile, $Q_r = \lfloor r(n + 1) / 4 \rfloor$.
- Upper quartile function, ***Q3(x)***

- Quartile score, $QS_{iv}$

$$QS_{iv} = PS_{iv} - Q_3(PS_{iv})$$



# Quartile-aided Popularity Score

- Quartile-aided Popularity Score, $QPS_{iv}$

$$QPS_{iv} = 1 - \frac{QS_{iv}}{Max(QS_{iv}) + 1}$$

- We can capture the popularity of items adjusted by the upper quartile.

| $iv$ | $I_3U_5$ | $I_1U_7$ | $I_1U_3$ | $I_6U_{19}$ | $I_8U_7$ | $I_{10}U_3$ | $I_1U_5$ | $I_1U_{17}$ | $I_9U_3$ | $I_7U_7$ | $I_5U_{19}$ | $I_{11}U_{17}$ | $I_2U_5$ | $I_4U_{19}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $PS_{iv}$ | 10 | 14 | 15 | 17 | 29 | 30 | 35 | 36 | 55 | 62 | 63 | 75 | 90 | 150 |
| $QS_{iv}$ | 53 | 49 | 48 | 46 | 34 | 33 | 28 | 27 | 8 | 1 | 0 | 12 | 27 | 87 |
| $QPS_{iv}$ | 0.40 | 0.44 | 0.45 | 0.48 | 0.61 | 0.63 | 0.68 | 0.69 | 0.91 | 0.99 | 1 | 0.86 | 0.69 | 0.01 |

# Social Behavior Score

- UbiMiner includes two factors from the social behavior.
  - ***MF**$_{uv}$* : the number of Mutual Friends between user u and user v.
  - ***DC**$_{uv}$* : the number of Direct Communication between user u and user v.



# Social Behavior Score

- Social Behavior Score, ***SBS**$_{uv}$*

$$SBS_{uv} = \left(1 - \frac{MF_{uv}}{\underset{v \in friends\ of\ u}{Max}(MF_{uv}) + 1}\right) \times \left(1 - \frac{DC_{uv}}{\underset{v \in friends\ of\ u}{Max}(DC_{uv}) + 1}\right)$$

| $uv$ | $U_1U_3$ | $U_1U_7$ | $U_1U_{19}$ | $U_1U_5$ | $U_1U_{17}$ |
|---|---|---|---|---|---|
| $MF_{uv}$ | 5 | 12 | 20 | $\boxed{32}$ | 25 |
| $DC_{uv}$ | 10 | 7 | 30 | 25 | $\boxed{40}$ |
| $SBS_{uv}$ | 0.64 | 0.53 | 0.1 | 0.01 | 0.007 |

# User Similarity Score

- The worth value of each action $j$, $WV_{jv}$:

$$WV_{jv} = \frac{\sum_{j \in all\ actions} times\ of\ action\ j}{times\ of\ action\ j}$$

| | $U_1$ | | $U_3$ | | $U_{19}$ | |
|---|---|---|---|---|---|---|
| | Number | $WV_{jv}$ | Number | $WV_{jv}$ | Number | $WV_{jv}$ |
| Article | 100 | 16 | 200 | 9 | 50 | 31 |
| Comment | 500 | 3.2 | 400 | 4.5 | 600 | 2.5 |
| Like | 1000 | 1.6 | 1200 | 1.5 | 900 | 1.7 |

$$IS_{iv} = \sum_{j \in all\ actions} WV_{jv}$$



# User Similarity Score

- User Behavior, $UB_u$:

$$UB_u = \{IS_{I_1}, \cdots, IS_{I_n}\}$$

- User Similarity Score, $USS_{uv}$ :

$$USS_{uv} = \frac{UB_u \cdot UB_v}{||UB_u||\ ||UB_v||}$$

# User Similarity Score

| | $I_1$ | $I_2$ | $I_3$ | $I_4$ | $I_5$ |
|---|---|---|---|---|---|
| $U_1$ | Post + like | Like + Comment | Like | Comment | N/A |
| $U_3$ | Post | Post + Comment | Comment | Post + Like + Comment | Post |
| $U_{19}$ | N/A | Like + Comment | N/A | Like | Like |

| | $IS_{I_1}$ | $IS_{I_2}$ | $IS_{I_3}$ | $IS_{I_4}$ | $IS_{I_5}$ |
|---|---|---|---|---|---|
| $U_1$ | 17.6 | 4.8 | 1.6 | 3.2 | 0 |
| $U_3$ | 9 | 13.5 | 4.5 | 15 | 9 |
| $U_{19}$ | 0 | 4.2 | 0 | 1.7 | 1.7 |

***USS*** between $U_1$ and $U_3$ is 0.62, $U_1$ and $U_{19}$ is 0.28



# Unknown But Interesting Score

- Unknown But Interesting Score, $UBIS_i$

$$UBIS_i = \sum_{v} (QPS_{iv} \times SBS_{uv} \times USS_{uv})$$

- Finally, we sort UBIS and recommend the top-k items to the target user.

| $i$ | $I_9$ | $I_{10}$ | $I_7$ | $I_8$ | $I_5$ | $I_1$ | $I_6$ | $I_2$ | $I_3$ | $I_{11}$ | $I_4$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $UBIS_i$ | 0.29 | 0.20 | 0.15 | 0.09 | 0.07 | 0.04 | 0.03 | 0.006 | 0.004 | 0.0006 | 0.0001 |

# Experimental Setup

- We implement four recommender systems on a popular social networking website, Facebook.
  - Facebook recommendation list is based on latest updates from user's posting (FB).
  - Traditional method is based on popularity of items with similarity among users (PS).
  - TANGENT [Onuma, KDD'09] is based on user's friendship and the frequency of each item.
  - UbiMiner recommends unknown but interesting items based on UBIS.



# TANGENT

- K. Onuma, H. Tong, and C. Faloutsos. Tangent: a novel, 'surprise me', recommendation algorithm. KDD '09

# Questionnaire

**[ The questionnaire of facebook usage ]**

1. How many days in last week did you visited Facebook?

◉ 0 ~ 2 days ○ 3 ~ 5 days ○ Everyday

2. How many hours a day do you spend on Facebook?

◉ Less than 1 hour ○ 1 ~ 3 hours ○ 5 ~ 8 hours ○ All the time

3. What do you spend the most time with on Facebook?

◉ Play games ○ Watch friend's photo / video / message ○ Like good photo / video / message

○ Share photo / video / message ○ Chat ○ Others

submit Reset

**[ Click the button below, Start using the system ]**

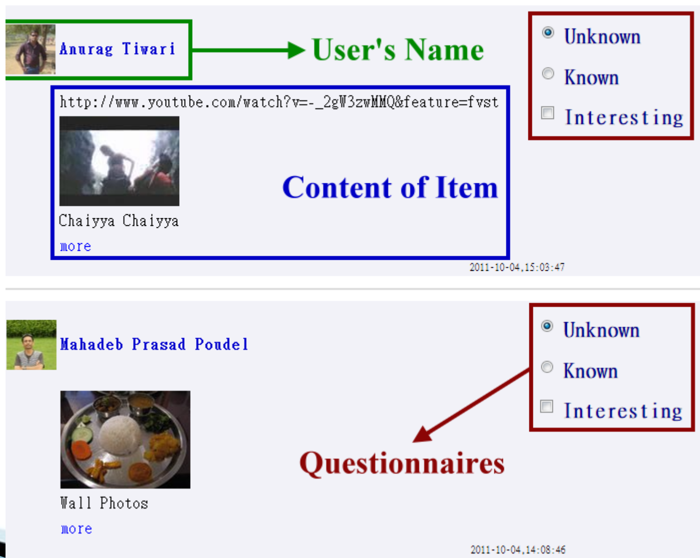
Recommended list 01 | Recommended list 02 | Recommended list 03



# Questionnaire

**Welcom to UBI Recommendation system**

**[Recommended List 01]**



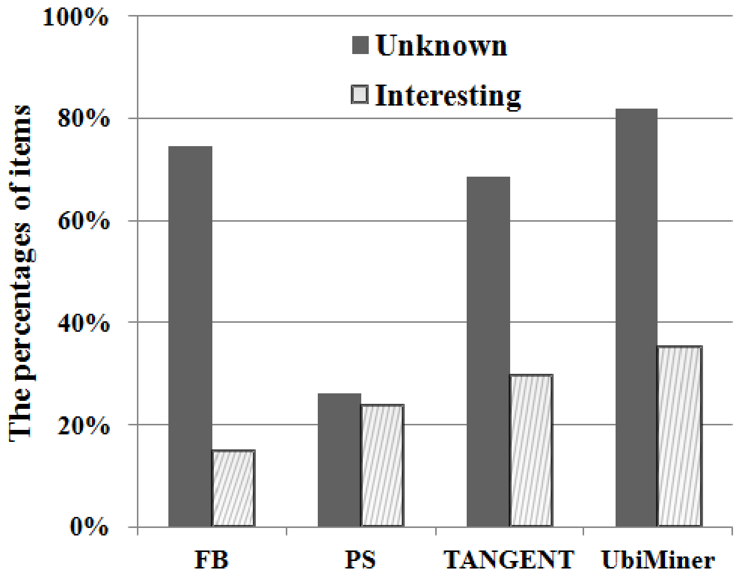
# Experimental Results

- We randomly invited 355 users to participate in our experiment.
- The experiments were conducted from July to September of 2011
- There are in average 185 active users participating per month.



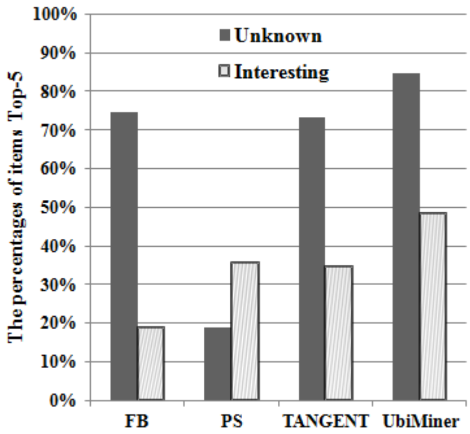
# Satisfaction Percentage

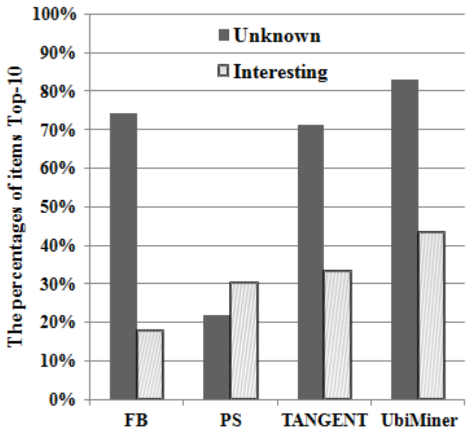
- Overall satisfaction (Top-20 items)



# Satisfaction Percentage

Top-5 list　　Top-10 list

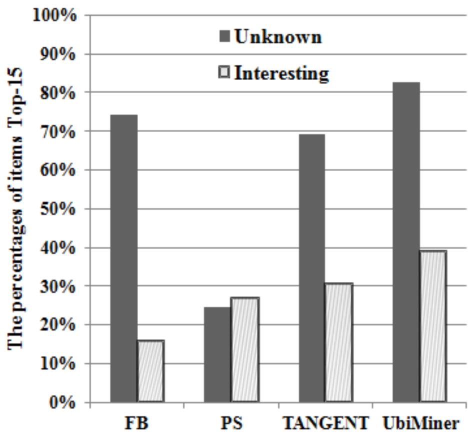


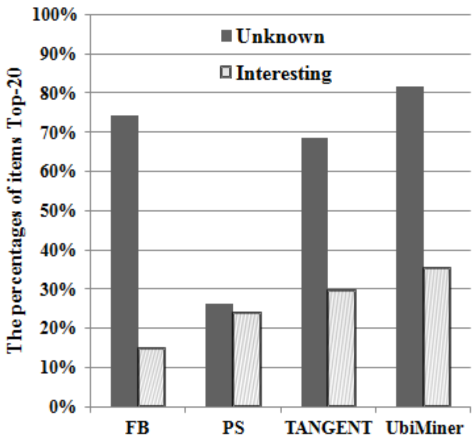






# Satisfaction Percentage

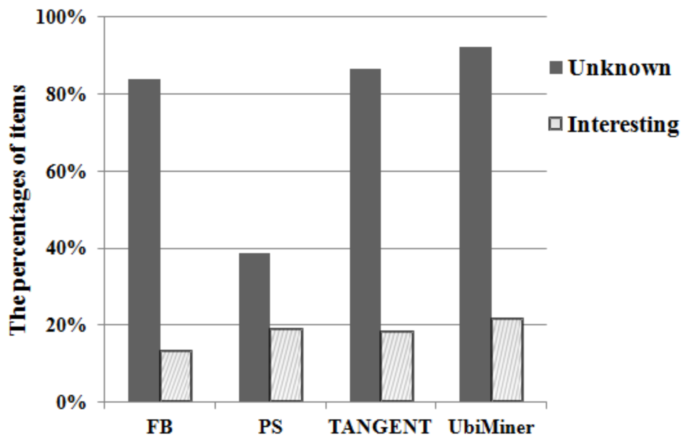
Top-15 list　　Top-20 list



# Behavior Breakdown

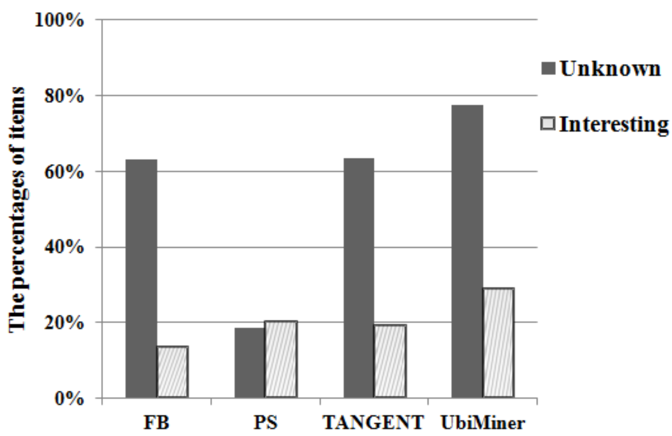
- 119 users visit Facebook every day
- Case 1: Spend time on Facebook
  less than 1 hour.






# Behavior Breakdown

- 119 users visit Facebook every day
- Case 2: Spend time on Facebook
  more than 5 hours.



# Conclusions

- We proposed UbiMiner, which recommends unknown but interesting items by utilizing Quartile-aided Popularity Score, Social Behavior Score, and User Similarity Score.
- Experimental results show that the performance of UbiMiner outperforms that of traditional methods in terms of the percentages of unknown and interesting items in the recommendation lists.



# References

[1] IHERGO http://www.ihergo.com

[2] P. Resnick, N. Iacovou, M. Sushak, P. Bergstrom, and J. Riedl. Grouplens: An open architecture for collaborative filtering of netnews. CSCW, 1994.

[3] X. Amatriain, N. Lathia, J. M. Pujol, H. Kwak, and N. Oliver. A collaborative filtering approach based on expert opinions from the web. SIGIR, 2009.

[4] I. Guy, N. Zwerdling, D. Carmel, I. Ronen, E. Uziel, S. Yogev, and S. Ofek-Koifman. Personalized recommendation of social software items based on social relations. RecSys, 2009.

[5] I. Guy, M. Jacovi, E. Shahar, N. Meshulam, V. Soroka, S. Farrell, Harvesting with SONAR:the value of aggregating social network information, CHI 2008.

[6] J. Zhang, M. S. Ackerman, L. Adamic, Expertise Networks in Online Communities: Structure and Algorithms, WWW 2007.

[7] J. M. Kleniberg. Authoritative sources in a hyperlinked environment. Journal of the ACM, 1999.

[8] T. Zhou, J. Ren, M. c. v. Medo, and Y.-C. Zhang. Bipartite network projection and personal recommendation. *Phys. Rev. E, 76(4):046115, Oct 2007.*

# References

[9] Official facebook blog. People You May Know. http://blog.facebook.com/blog.php?post=15610312130. 2008.

[10]R. Bell, Y. Koren, and C. Volinsky. Modeling relationships at multiple scales to improve accuracy of large recommender systems. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD ’07, pages 95–104. ACM, 2007.

[11]I. Cantador, A. Bellogín, and P. Castells. News@hand: A semantic web approach to recommending news. In W. Nejdl, J. Kay, P. Pu, and E. Herder, editors, *Adaptive Hypermedia and Adaptive Web-Based Systems*, volume 5149 of *Lecture Notes in Computer Science*, pages 279–283. Springer Berlin / Heidelberg, 2008.

[12]J. Chen, W. Geyer, C. Dugan, M. Muller, and I. Guy. Make new friends, but keep the old: recommending people on social networking sites. In *Proceedings of the 27th international conference on Human factors in computing systems*, CHI ’09, pages 201–210. ACM, 2009.



# References

[13]S. Debnath, N. Ganguly, and P. Mitra. Feature weighting in content based recommendation system using social network analysis. In *Proceeding of the 17th international conference on World Wide Web*, WWW ’08, pages 1041–1042. ACM, 2008.

[14]C.-H. Lai and D.-R. Liu. Integrating knowledge flow mining and collaborative filtering to support document recommendation. Journal of Systems and Software, 82:2023–2037, 2009.

[15]K. Onuma, H. Tong, and C. Faloutsos. Tangent: a novel, ’surprise me’, recommendation algorithm. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD ’09, pages 657–666. ACM, 2009.

[16]M. Pazzani and D. Billsus. Content-based recommendation systems. In *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, pages 325–341. Springer Berlin / Heidelberg, 2007.