Tag-based Social Interest Discovery

Xin Li / Lei Guo / Yihong (Eric) Zhao

Yahoo!Inc – 2008

Presented by: Tuan Anh Le (aletuan@vub.ac.be)

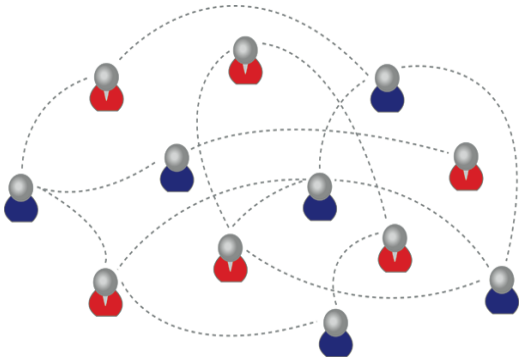
Outline

- Introduction
- Data set collection & Pre-processing
- Architecture for (I) Social Interest Discovery
- Analysis of Tags
- Evaluation results
- Conclusions

Introduction

- Social network systems are becoming more successful popular, and generate challenges

- Discovering social interests shared by group of users is very important
  - Detecting and representing user's interests



- Two types of existing approaches:
  - User-centric: based on social connections among users
  - Object-centric: based on the common objects fetched by users

Introduction

- Paper's approach : discover social interests by utilizing user-generated tags
  - Statistical analyse the real-word traces of tags and web content (*delicious.com*)
    - User-generated tags are consistent with the content they are being attached
  - Develop the Social Interest Discovery system
    - Discovering the common user interests
    - Clustering users and their saved URLs by topic (set of tags)

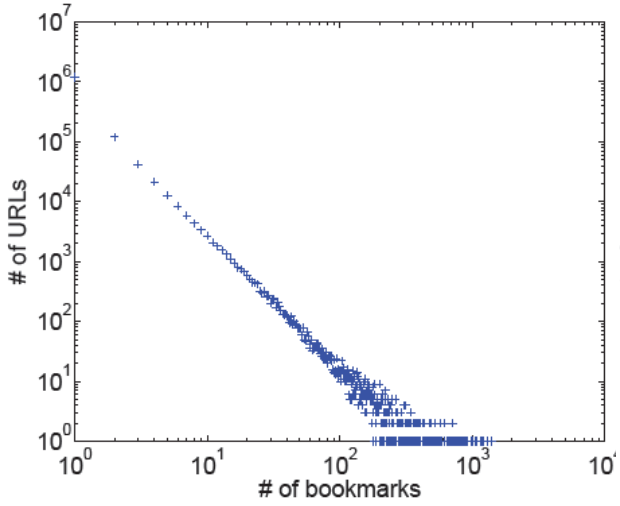
Data set

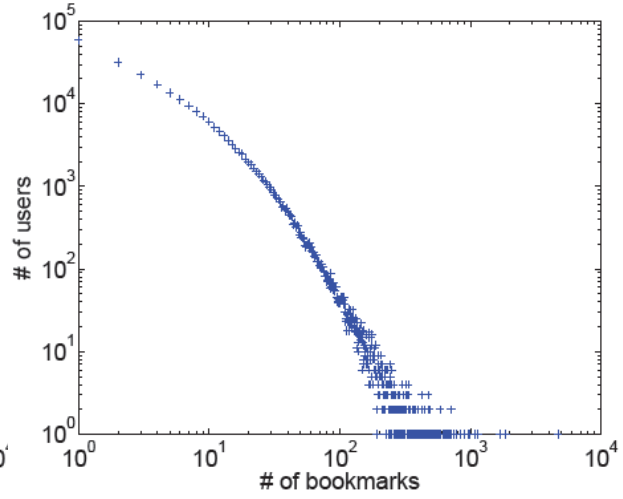
- Data is collect from *delicious.com* database. Each post has form:

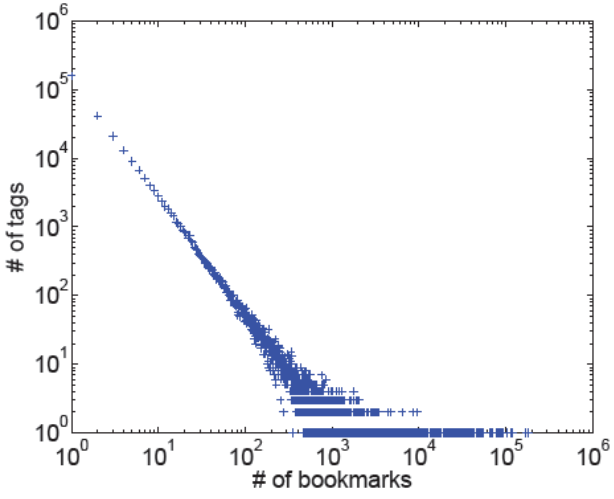
  $p = \{user, URL, tags\}$

- How many data they collected ?
  - *4.3 millions bookmarks, 0.2 millions users, and 1.4m URLs*
  - After pre-processing: *~ 0.3m tags and 4m keywords*
- Data collection & pre-processing
  1. Crawl the URLs and download pages
  2. Discard all non-html object
  3. Coding to UTF 8 & removing non English paper
  4. Stopword List (i.e. "a", "an", "the" etc…)
  5. Porter Stemming algorithm* (i.e. "fishing", "fisher", "fished" → "fish")
  6. Analysis distributions of frequencies (Tags, URLs and User) over the Bookmarks

Data set





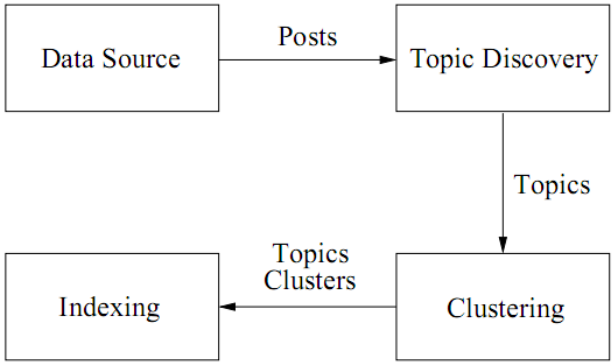


- Statistical view
  - Distributions follow power law* (linear graph in log-log scale)
  - Distributions have long tails ! (~ Pareto principle: 80/20 rule)

- Remarks
  - Most documents are unpopular
  - Most users are inactive
  - Top popular tags connect most of the users

Architecture of SID

- Discovery Social Interest by Tags
  - Idea: Set of tags are frequently used by many users can give a hint that such users may spontaneously from a community of an interest even though *they may not have any physical connection or online connection.*
  - SID is proposed based on "association rules algorithm"
    - Finding frequently co-occurring tags (topics of interest)
    - Building URLs and users clusters for such tag-based topic" (clustering)
    - Importing topics and clusters into indexing system for application queries (indexing)



Architecture of SID

- Data Source
  - Application data repository which store user's post
  - Data source provides SID a series of posts p = (user, URL, tags)
- Topic discovery
  - Using association rule algorithms to discover "*frequent item patterns*" for a set of transactions and then derive the implication relationship among items set for transaction.
  - Remove redundancy from item sets. For example:

100 posts contains tags "food" and "recipes" with support is 30, then {*food, recipes*}, {*food*}, {*recipes*} are "hot" topic.

$\omega(\{recipes, food\}) = \omega(\{food\}) = \omega(\{recipes\})$ → remove {*food*}, {*recipes*}

Architecture of SID

- Clustering
  - For each topic and all the posts contain the tag set, insert URLs and uses into two clusters.
  - Naïve clustering algorithm:

```
1.    for all topic t ∈ T do
2.        t.user ← ∅
3.        t.url ← ∅
4.    end for
5.    for all post p ∈ P do
6.        for all topic t ∈ p do
7.                t.user ← t.user ∪ {p.user}
8.                t.url ← t.url ∪ {P.url}
9.        end for
10.   end for
```

Architecture of SID

- Indexing
  - Clusters types:
    - Url & user clusters are identified by topics.
    - Topic & url clusters are identified by users.
  - Indexing cluster supports some queries:
    - For a given topic, list all URLs contain this topic (have been tagged with all the tags in the topic).
    - For a given topic, list all users who are interest in that topic (have used all the tags in the topic).
    - For a given tag, list all topics contain that tag….

Analysis of Tags

- Statistical model:
  - Use vector space model (VSM) to describe a URL (i.e. book index)
  - Each URL: two vectors
    - One in the space of all tags, one in the space of all document keywords
  - In VSM, matrix with t terms and d documents:
    - Term-document matrix $A = (a_{ij})\ R^{t*d}$
    - Column vector a_j is a set of terms belong to document j
    - a_{ij} : importance of term i in document j (or "weight")

D1 = "I like databases"
D2 = "I hate hate databases"

	I	Like	hate	databases
D1	1	1	0	1
D2	1	0	2	1

Analysis of Tags

- Statistical model
  - Weight (a_{ij}) measurement
    - Tf-based (term frequency based)

$$a_{ij}^{tf} = \frac{f_{ij}}{\sqrt{\sum_{k=1}^{t} f_{kj}^2}}.$$

a_{ij} : importance of term i in document j
f_{ij} : frequency of term i in document j

    - Tf-Idf based (term frequency – inverse document frequency)

$$a_{ij}^{tfidf} = \frac{b_{ij}}{\sqrt{\sum_{k=1}^{t} b_{kj}^2}}$$

b_{ij} : inverse document frequency

$$b_{ij} = f_{ij} \cdot \log(\frac{d}{D_i})$$

D_i : number of documents contains term i
d : total number of documents

Analysis of Tags

- Tags vs. document keywords
  - An intuitive example:

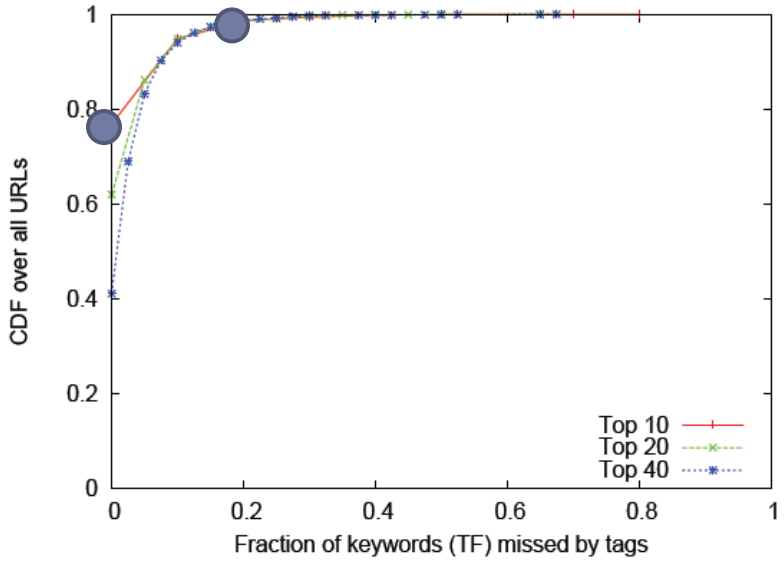
URL	http://kalfsb.home.att.net/resolve.html
Top Tf keywords	**domain**,**name**,**file**,**resolver**,server,conf,network,nameserver,**ip**,**org**,**ampr**
Top Tf idf keyword	**ampr**,**domain**,jnos,nameserver,conf,**kalfsb**,resolver,**ip**,**file**,**name**,server
All tags	linux,howto,network,**sysadmin**,**dns**

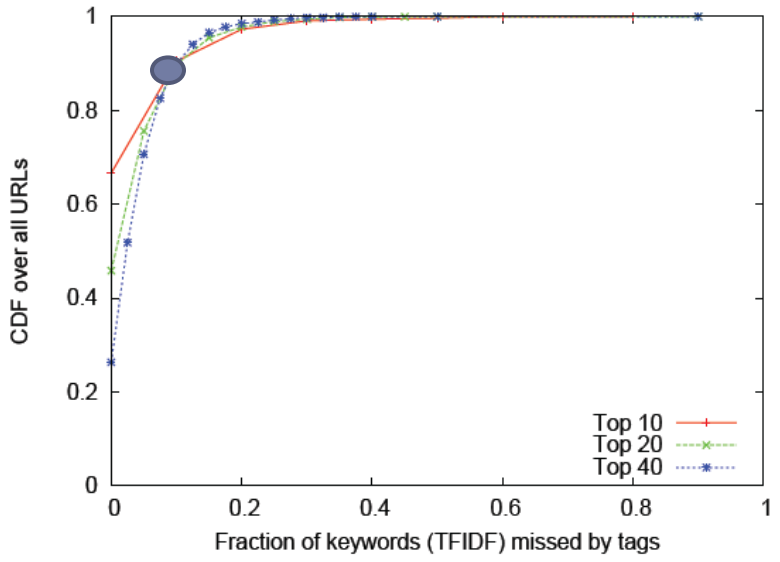
  - Tags & keywords reflect the content, and differ only *literally*
  - Tags are closer to people's understanding of content than keywords
  - Some keywords are unrelated to the content / replaced without changing meaning

Analysis of Tags

- Tags vs. document keywords
  - Vocabulary of tags and keywords:
    - Is vocabulary of important Keywords covered by Tags ? (YES)
    - Statistical method:
      - 7000 randomly English web document
      - Plot cumulative distribution function : $x \mapsto F_X(x) = \mathrm{P}(X \leq x)$

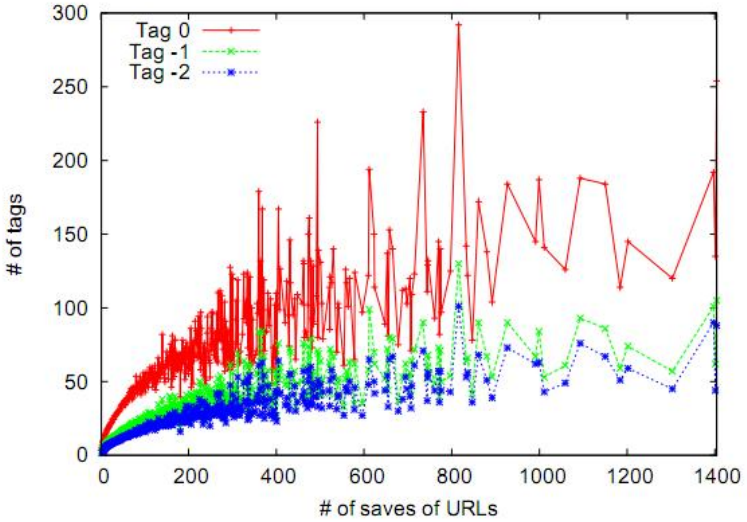
(x is percentages Keywords missed by Tags)





Analysis of Tags

- Convergence of User's Tag Selection
  - Proportions of tags in the bookmarks are quite stable for popular URLs
  - Measure the concentration & convergence of distinct tags used by different user



Y = F(X)

Y: Number of distinct tags
X: Popularity of URLs (#saves of URLs)

Analysis of Tags

- Tags matched by documents
  - How well do user's tags capture the main concepts of documents ?
  - Solutions
    - Human reviews
    - Statistical analysis about correlation between the tags of a URL and the content of its document.

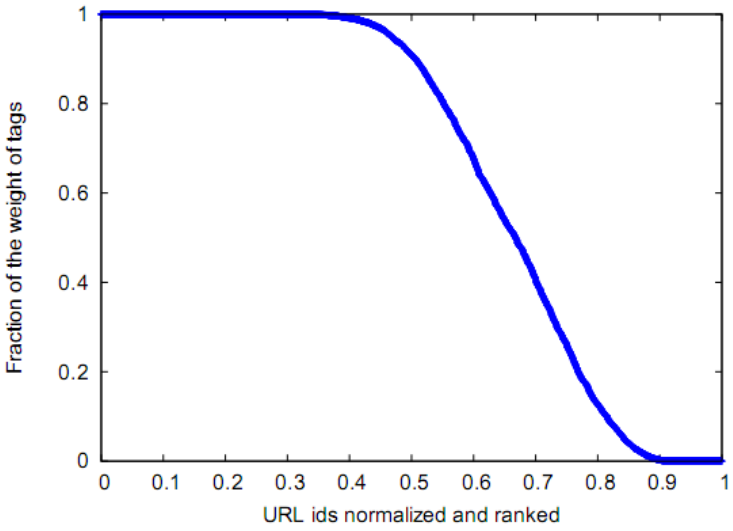
$$e(T,U) = \frac{\sum_{k|t_k \in U} w(t_k)}{\sum_i w(t_i)}$$

T = { t_i } : set of tags attached an URL U

w(t) : weight of tag t (frequency of tag in data)

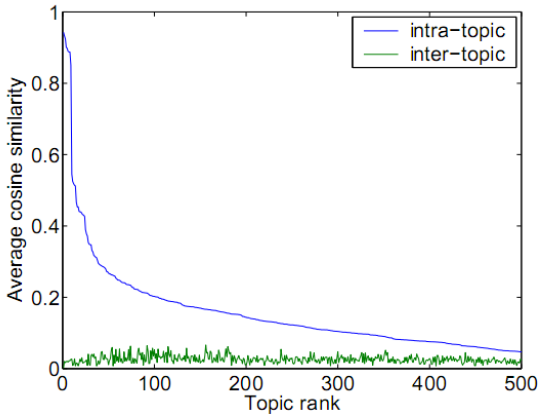
e(T,U) : tag match ratio

Numerator is total weight of tags which also appear in document keyword

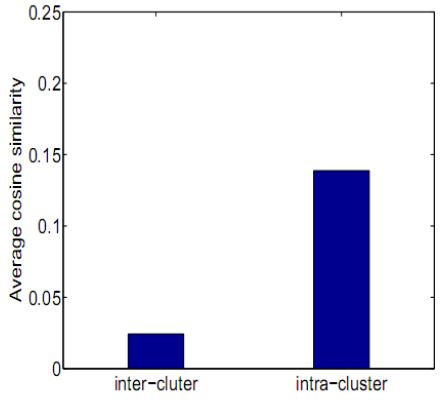


Evaluation results

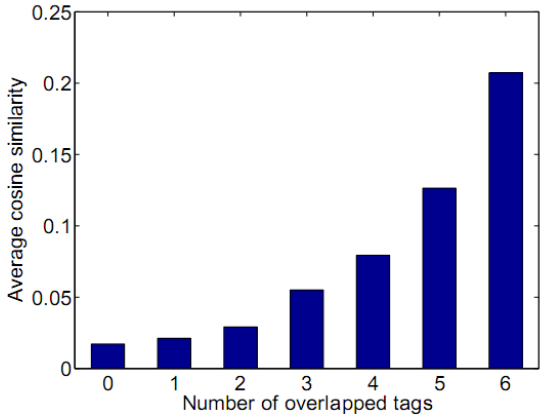
- Effectiveness of SID URL clusters by computing URL similarity within & cross the clusters.
  - Compute the similarity between pair of documents with inner product (cosine similarity) of their **tf * idf keywords vectors.**
  - Select 500 interest topics, each contains > 30 bookmarked URLs that share 5-6 co-occurring user tags.
  - Each topic: compute average cosine similarity of all URL pairs in its cluster (intra-topic)
  - Randomly select 10,000 topic pairs, compute average pairwise document similarity between every two topics (inter-topic)



(a) The comparison of inter-topic and intra-topic cosine similarity



(b) Average inter- and intra-topic cosine similarity

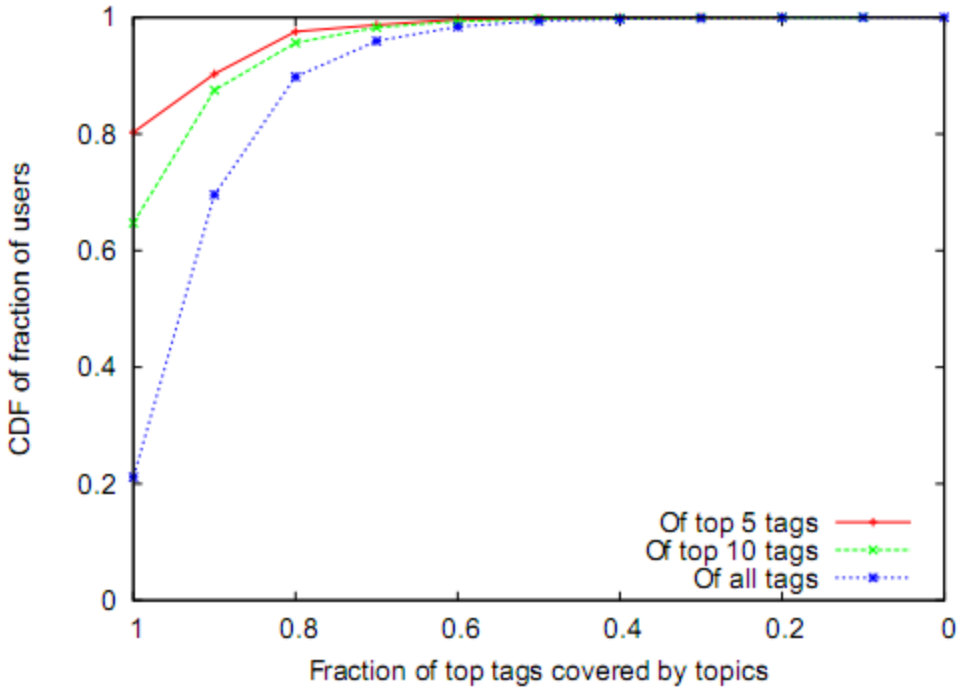


(c) Cosine similarity of topic pairs with different number of co-occurring tags

Figure 10: Tag-based cosine similarity of interest topics (support number = 30)

Evaluation results

- Evaluate how the topic discovered by SID cover the individual interests of users.
  - The more frequently an user uses a tag, the higher interests he has on the corresponding topic represented by the tags.
  - Checking if top used tags of each users are in any topic discover can be discovered by SID

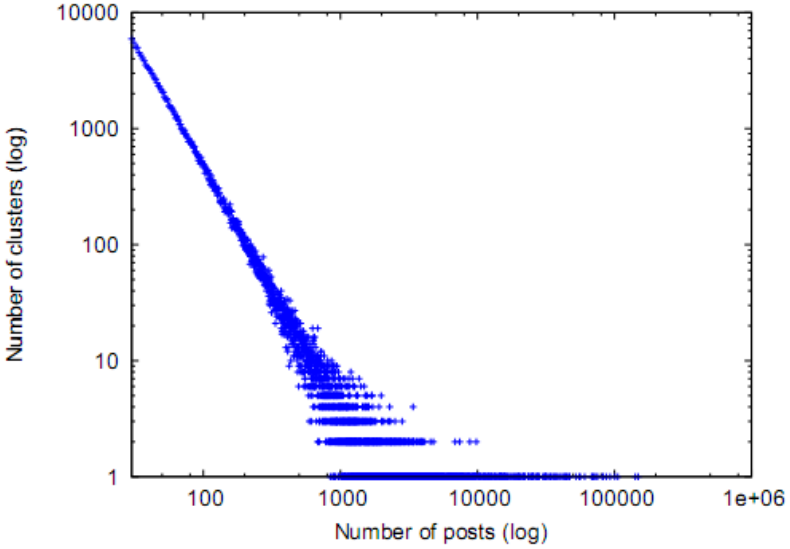


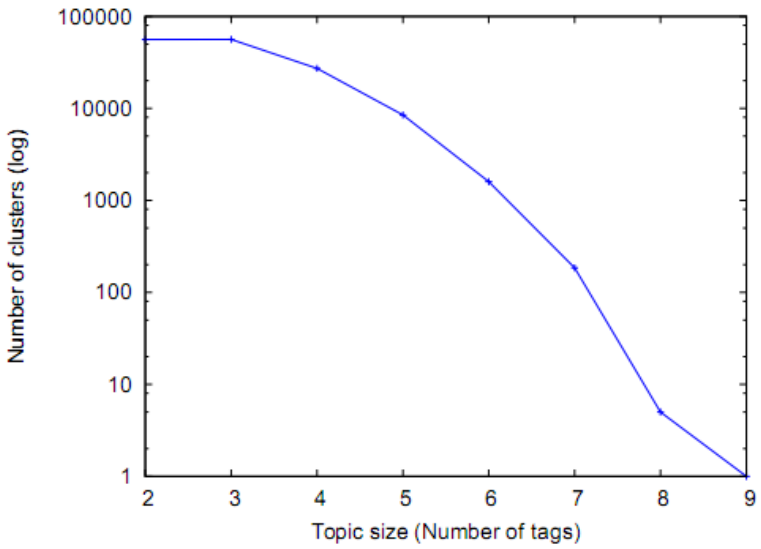
- ✓ 80% users has 100% top 5 tags
- ✓ 10% user has 90% top 5 tags
- ✓ Over 90% user has at least 90% top 5 tags

- ✓ SID has correctly identified & cluster over 90% interest for more than 90% users

Evaluation results

- General properties of topic clusters.
  - The number of clusters with a given cluster size (threshold is 30) follows a power-law distribution.
  - Users tend to use a small number of words to summarize the contents for themselves
  - Distributions of the number of topics as the functions of number of users & number of URLs are also follow power law





Conclusion

- Advance
  - Propose new approach for interest discovery base on user's tags with cost effective*
  - User tags is closer with human understanding, and capture precisely web contents.
  - Applicable system to discover common interest topic in social networks.
  - Don't require online or physical connection between users
- Disadvantage
  - Depend on user / group of user's characteristics. (talk about same thing in # ways)
    - Community culture
    - Users understand about something in different levels
  - Should combine with another approach (user-centric) to give flexibility (users can self-organize their group / or get connection's recommendation)