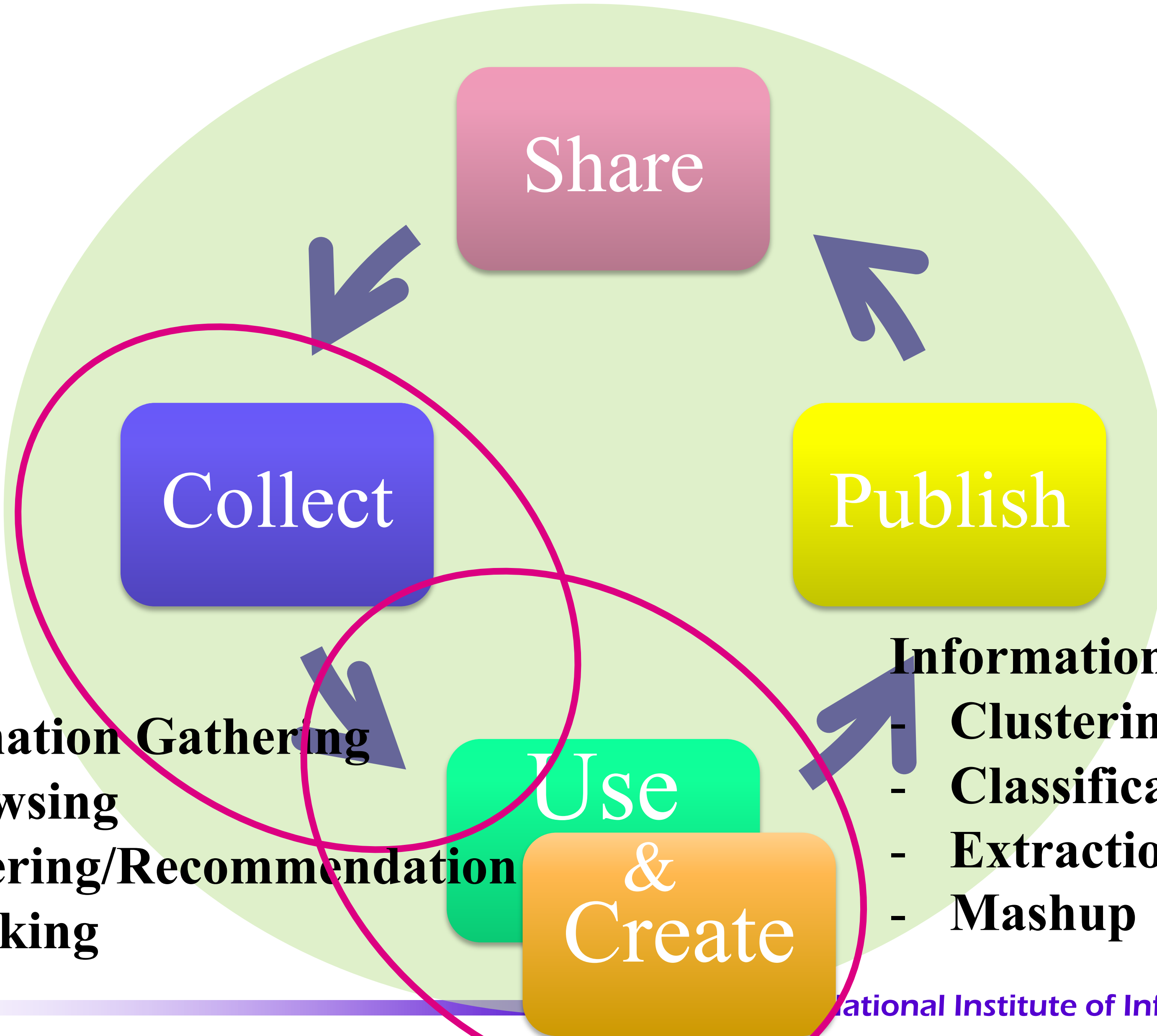
Information Cycle
1. Information Level

Hideaki Takeda
National Institute of Informatics
takeda@nii.ac.jp
http://www-kasm.nii.ac.jp/~takeda/

Information Cycle on Information Level



Information Integratio
- **Clustering**
- **Classification**
- **Extraction**
- **Mashup**

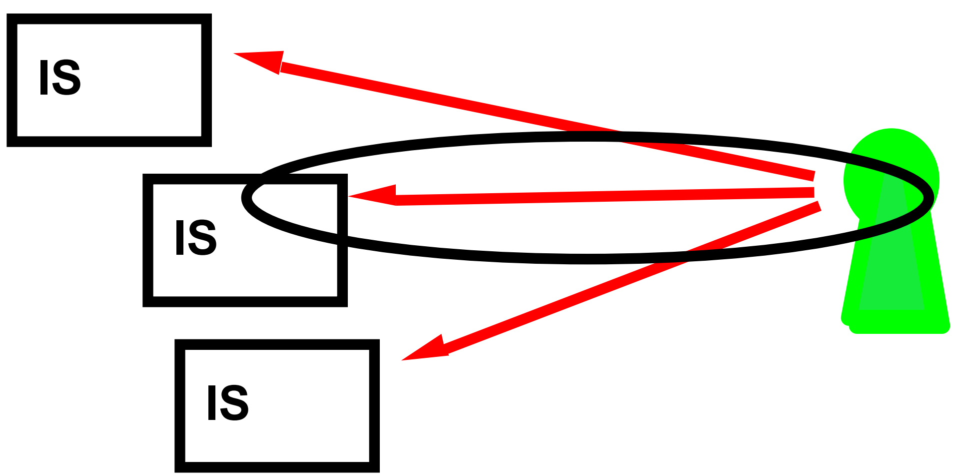
Information Gathering
- **Browsing**
- **Filtering/Recommendation**
- **Ranking**

Information Cycle on Information Level

- Information Gathering
  - Browsing
  - Filtering/Recommendation
  - Ranking
- Information Integration
  - Clustering
  - Classification
  - Extraction
  - Mashup

Information Gathering

- Model: Relationship between agents and information sources
- Main task: how to provide access from the agent to information sources



Methods for Information Gathering

- Information Retrieval
  - Explicit specification of users needs
- Browsing
  - Implicit specification of users needs
  - Finding it by oneself
- Information Filtering/Recommendation
  - Implicit specification of users needs
  - Guessing users preference
- Ranking-based search
  - Explicit specification of users needs + general preference

Browsing

- Characteristics as Information Gathering
  - Pros:
    - Users initiative
    - Applicable even with vague purpose
  - Cons: No warranty to reach the goal
    - Human habit: up to down, side trip
    - Human limitation: Sequential access
- Problems
  - How to support users with keeping users initiative
  - How to obtain users preference

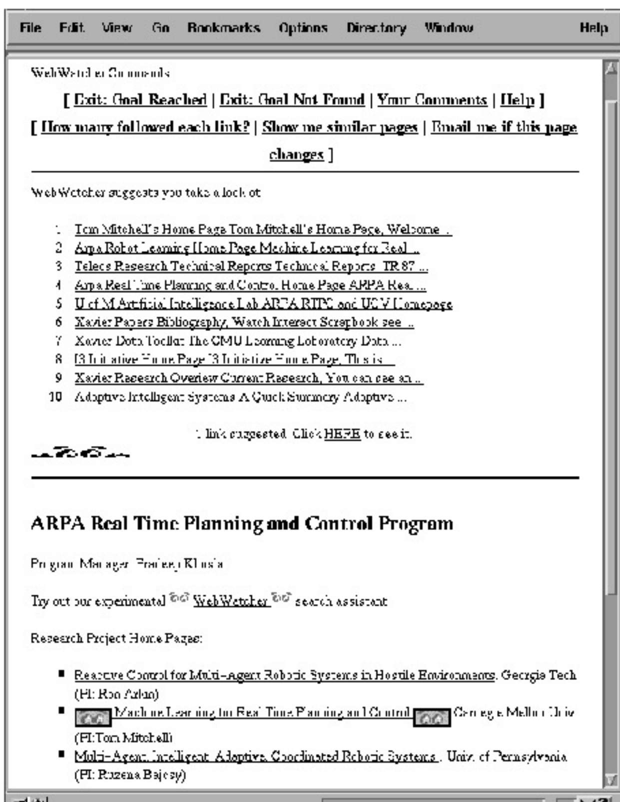
Browsing

- Problems
  - How to support users with keeping users initiative
  - How to obtain users preference

↓

  - How to obtain users preference
    - Web Watcher
    - Letizia
    - Syskill & Webert

Web Watcher



File Edit View Go Bookmarks Options Directory Window Help
[Exit: Goal Reached | Exit: Goal Not Found | Your Comments | Help]
[How many followed each link? | Show me similar pages | Email me if this page changes]
WebWatcher suggests you take a look at
1. Tom Mitchell's Home Page Tom Mitchell's Home Page, Welcome ...
2. Arpa Robot Learning Home Page Machine Learning for Real ...
3. Telecs Research Technical Reports Technical Reports TR 87 ...
4. Arpa Real Time Planning and Control Home Page ARPA Rea ...
5. U of M Artificial Intelligence Lab ARPA RIPS and UCV Homepage
6. Xavier Papers Bibliography, Watch Interact Scrapbook see ...
7. Xavier Data Toolkit The CMU Learning Laboratory Data ...
9. Xavier Research Overview Current Research, You can see an ...
10. Adaptive Intelligent Systems A Quick Summary Adaptive ...
1 link suggested. Click HERE to see it.
ARPA Real Time Planning and Control Program
Try out our experimental WebWatcher search assistant
Research Project Home Pages:
Reactive Control for Multi-Agent Robotic Systems in Hostile Environments. Georgia Tech (PI: Ron Arkin)
Machine Learning for Real Time Planning and Control Carnegie Mellon Univ (PI: Tom Mitchell)
Multi-Agent Intelligent Adaptive Coordinated Robotic Systems. Univ. of Pennsylvania (PI: Ruzena Bajcsy)

File Edit View Go Bookmarks Options Directory Window Help
Back Forward Home Edit Reload Images Open Print Find Stop
Location: http://cranium.learning.cs.cmu.edu:8080/cgi-b
What's New? What's Cool? Destinations Net Search People Software
WebWatcher Commands
[Exit: Goal Reached | Exit: Goal Not Found | Your Comments | Help]
[How many followed each link? | Show me similar pages | Email me if this page changes]
1 link suggested. Click HERE to see it.
Carnegie Mellon
Welcome to the WebWatcher Project
Overview
WebWatcher is a "tour guide" agent for the world wide web. Once you tell it what kind of information you seek, it accompanies you from page to page as you browse the web, highlighting hyperlinks that it believes will be of interest. Its strategy for giving advice is learned from feedback from earlier tours.
Try it!
WebWatcher can help you search for information starting from any of the following pages. (but it has learned the most about the first of these).
CMU School of Computer Science Front Door After arriving at this page, click on "The WebWatcher tour guide." under the heading SCS Resources
Machine Learning Information Services
ARPA Intelligent Integration of Information Home Page
ARPA Real Time Planning and Control Home Page

Web Watcher

- Use of machine learning
  - Learn users preference from browsing process
- Functions
  - Recommendation of links which the system infers useful to the user among links in browsing pages
  - Recommendation of links which the systems infers useful to the user among all links

Web Watcher

- Learning target

  LinkUtility: Page × Goal × User × Link→[0,1]

  UserChoice: Page × Goal × Link→[0,1]

  Page: Keyword vector of 200 words extracted from pages

  link: Keyword vector of 200 words from links and 100 words from texts surrounding links

  Goal: Keyword vector of 30 words

Information Cycle on Information Level

- Information Gathering
  - Browsing
  - Filtering/Recommendation
  - Ranking
- Information Integration
  - Clustering
  - Classification
  - Extraction
  - Mashup

Information Filtering / Recommendation system

- Content-based filtering
  - Estimate users preference by comparing keywords in pages and users profiles
- Social filtering / Collaborative filtering
  - Estimate users preference by collecting and analyzing preferences of many users

Problem definition

- How to estimate missing information from the given matrix?
  - Each vector represents preference of each person
  - Some values are missing because she has not experience them
  - Estimate these values
- Solution: Use similarity between users

Article	Person A	Person B	Person C	Person D
1	1	4	2	2
2	5	2	4	4
3			3	
4	2	5		5
5	4	1		1
6	?	2	5	

Algorithm for Social filtering (correlation coefficient)

Calculate relation between user *k* and *k'* by correlation coefficient (相関係数)

$$r_{kk'} = \frac{Cov(k,k')}{\sigma_k \sigma_{k'}} \quad (k = 1 \dots m, k' = 1 \dots m)$$

Standard deviation $\sigma_k = \sqrt{\sum_{l=1}^{n'} (x_{kl} - \bar{x}_k)^2}$

Covariance $Cov(k,k') = \sum_{l=1}^{n'} (x_{kl} - \bar{x}_k)(x_{k'l} - \bar{x}_{k'})$

$$r_{kk'} = \frac{\sum_{l=1}^{n'} (x_{kl} - \bar{x}_k)(x_{k'l} - \bar{x}_{k'})}{\sqrt{\sum_{l=1}^{n'} (x_{kl} - \bar{x}_k)^2} \sqrt{\sum_{l=1}^{n'} (x_{k'l} - \bar{x}_{k'})^2}}$$

$$r_{AB} = \frac{-2 \cdot 1 + 2 \cdot (-1) + (-1) \cdot 2 + 1 \cdot (-2)}{\sqrt{4+4+1+1}\sqrt{1+1+4+4}} = -0.8$$

$$r_{AC} = \frac{-2 \cdot (-1) + 2 \cdot 1}{\sqrt{4+4}\sqrt{1+1}} = 1$$

Article	Person A	Person B	Person C	Person D
1	1	4	2	2
2	5	2	4	4
3			3	
4	2	5		5
5	4	1		1
6	?	2	5	

Algorithm for Cooperative filtering (correlation coefficient)

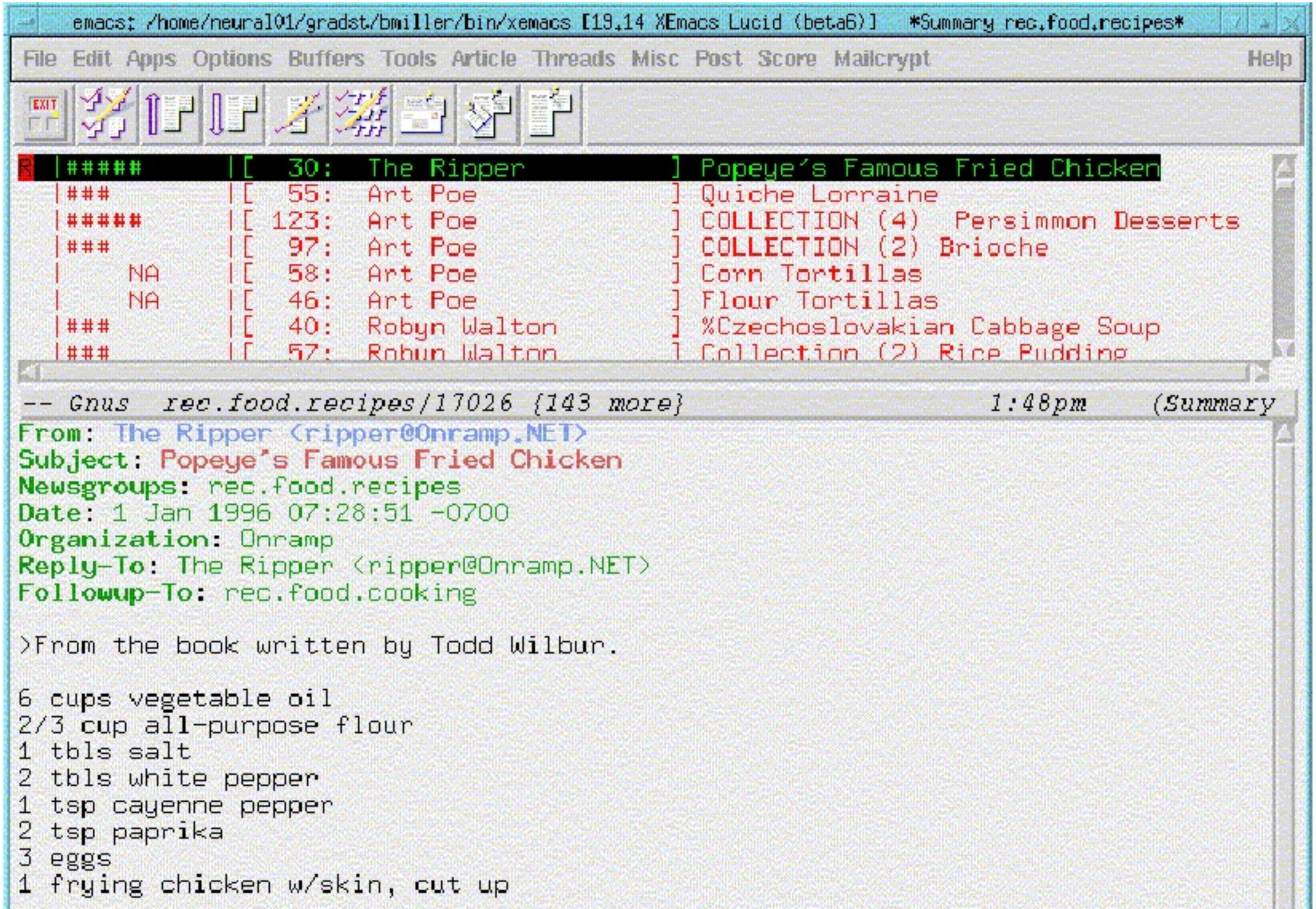
$$x'_{kl} = \bar{x}_k + \frac{\sum_{k' \neq k} (x_{k'l} - \bar{x}_{k'}) r_{kk'}}{\sum_{k' \neq k} |r_{kk'}|}$$

$$x'_{A6} = 3 + \frac{(-1)\cdot(-0.8) + 2\cdot 1}{0.8 + 1} = 4.56$$

Article	Person A	Person B	Person C	Person D
1	1	4	2	2
2	5	2	4	4
3			3	
4	2	5		5
5	4	1		1
6	?	2	5	

GroupLens

- Collaborative filtering system for NetNews



Collaborative Filtering: Pros and Cons

- Pros
  - Robust for content change
    - No need for content analysis
    - Applicable for non-text data
  - Few users actions
    - Just only evaluate items
- Cons
  - "Cold start" problem
    - Massive evaluation data is need before reliable recommendation
  - No evaluation, no recommendation
    - Items without evaluation are never recommended

MovieLens

MovieLens - Netscape
File Edit View Go Communicator Help
Help
Tutorial
Change Password
About MovieLens
Login Page
Comments & Suggestions
Rating more movies improves your predictions; you've rated 0 so far.
[5] = Must See [4] = Will Enjoy It [3] = It's OK [2] = Fairly Bad [1] = Awful
PREDICTED RATING
YOUR RATING
TITLE
5 *****
Antz (1998)
? unseen
Elizabeth (1998)
? unseen
Happiness (1998)
? unseen
Simon Birch (1998)
? unseen
Rounders (1998)
? unseen
Practical Magic (1998)
? unseen
Life Is Beautiful (La Vita • bella) (1997)
? unseen
What Dreams May Come (1998)
? unseen
One True Thing (1998)
? unseen
Next Stop, Wonderland (1998)
? unseen
Submit ratings and see next 10 titles (of 78 remaining)

Content-based filtering: pros and cons

- Pros
  - Precise recommendation is possible
    - For users
    - For providers
- Cons
  - For providers: Difficulty to design profiles
  - For users: Difficulty for keeping users profiles
    - Input
    - Update
  - Not adaptive for new contents

Ranking

- Sort contents by some criteria
  - Relativeness to the given keywords
    - TF/IDF, NLP
    - metadata, full text
    - Early Search Engines (e.g. infoseek)
  - Importance/reliability/creditability of contents
    - PageRank (google)
    - HITS Algorithm
    - …

PageRank

- A link analysis algorithm
  - Probability distribution to represent the likelihood for random access to pages
  - Assumptions similar to academic papers:
    - More cited papers are more valuable
    - Papers cited by more Valuable papers are more valuable

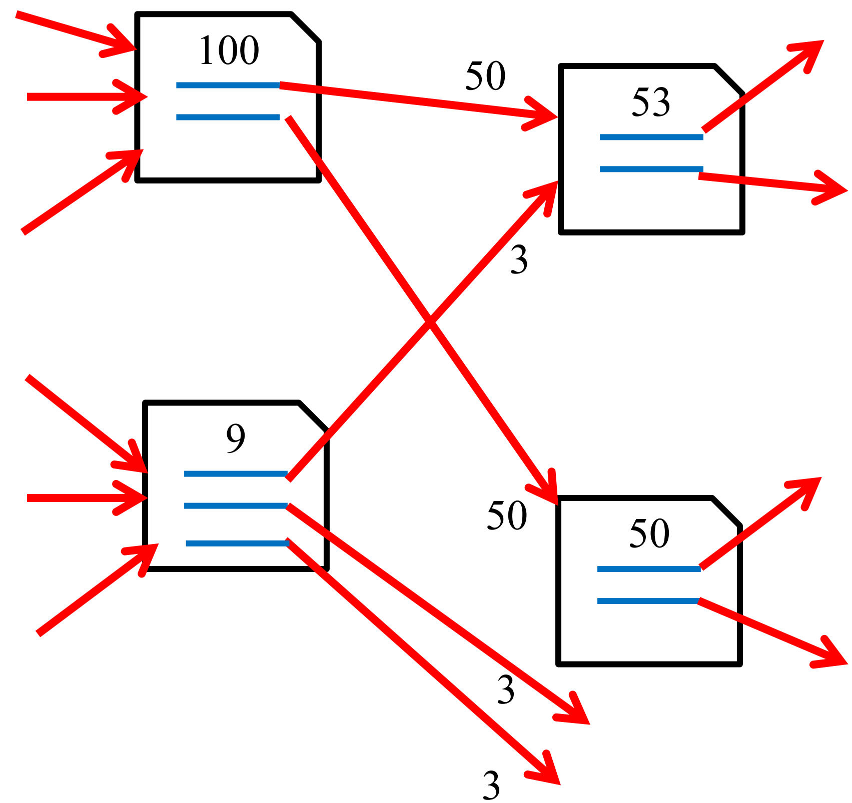
PageRank

- The Simplified Model
  - If link (v-> u) exist,

$$PR(u) = \sum \frac{PR(v)}{L(v)}$$

    - where L is the number of links in Page v
- Dumping factor

$$PR(u) = (1-d) + d\sum \frac{PR(v)}{L(v)}$$



$$R = \begin{bmatrix} PR(p_1) \\ PR(p_2) \\ \vdots \\ PR(p_N) \end{bmatrix}$$

$$R = \begin{bmatrix} (1-d)/N \\ \vdots \\ (1-d)/N \\ (1-d)/N \end{bmatrix} + d \begin{bmatrix} l(p_1, p_1) & l(p_1, p_2) & \cdots & l(p_1, p_N) \\ l(p_2, p_1) & & & \\ & & \ddots & \vdots \\ l(p_N, p_1) & & \cdots & l(p_N, p_N) \end{bmatrix} R$$

$$\sum_{i=1}^{N} l(p_i, p_j) = 1$$

PageRank

- Computation
  - Iterative

$$PR(p_i;0) = \frac{1}{N}$$

$$PR(p_i;t+1) = \frac{1-d}{N} + d\sum \frac{PR(p_j;t)}{L(p_j)}$$

$$\mathbf{R}(t+1) = d\mathbf{M}\mathbf{R}(t) + \frac{1-d}{N}\mathbf{1}$$

$$M_{ij} = \begin{cases} \frac{1}{L(p_j)}, & \text{if } i \text{ has link form } j \\ 0, & \text{otherwise} \end{cases}$$

while $|\mathbf{R}(t+1) - \mathbf{R}(t)| < \varepsilon$

  - Algebraic

$$\mathbf{R} = (\mathbf{I} - d\mathbf{M})^{-1} + \frac{1-d}{N}\mathbf{1}$$

Information Cycle on Information Level

- Information Gathering
  - Browsing
  - Filtering/Recommendation
  - Ranking
- Information Integration
  - Clustering
  - Classification
  - Extraction
  - Mashup

Clustering/Classification

- Clustering
  - Group data into some numbers of classes (not given)
  - Unsupervised learning
  - ex. Hierarchical Clustering, decision tree, C4.5, k-means clustering
- Classification
  - Divide data into the given classes
  - Supervised learning
  - ex. k-nearest neighbor, Bayesian Classification

Hierarchical Clustering

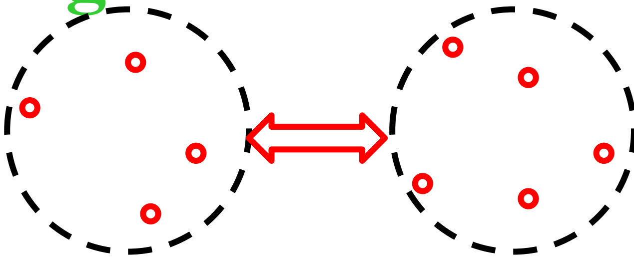
- An algorithm to build up a hierarchy of clusters
  - Agglomerative: Bottom up approach. A pair of clusters are merged into one
  - Divisive: Top down approach. A cluster is split into two.

Hierarchical Clustering

- Metric: A measure of dissimilarity $\mathbf{a} = (a_1, a_2, \ldots, a_n)$ $\mathbf{b} = (b_1, b_2, \ldots, b_n)$
  - Euclidean distance: $\sqrt{\sum_i (a_i - b_i)^2}$
  - Manhattan distance: $\sum_i |a_i - b_i|$
  - Maximum distance: $\max |a_i - b_i|$
  - Cosine similarity: $\cos\theta = (\boldsymbol{a} \cdot \boldsymbol{b}) / \|\boldsymbol{a}\| \|\boldsymbol{b}\|$
- Metric for text
  - Hamming distance: minimum number of substitution between two strings with the same length
  - Levenshtein distance: minimum number of single-character edits (insertion, deletion or substitution)

$$\frac{1}{|A||B|} \sum_{a \in A} \sum_{b \in B} d(a,b)$$

Hierarchical Clustering



- Linkage criteria: the distance between two sets of data
  - Maximum: $\max \boxed{f()} \{d(\boldsymbol{a},b) : a \in A,\ b \in B\}$
  - Minimum: $\min\{d(a,b) : a \in A,\ b \in B\}$
  - Mean: $\dfrac{1}{|A||B|}\sum_{\mathbf{a}\in A}\sum_{\mathbf{b}\in B} d(\mathbf{a},\mathbf{b})$
  - Centroid: $d(\mathbf{c}_A,\mathbf{c}_B)$, $\quad \mathbf{c}_A = \dfrac{1}{|A|}\sum_{\mathbf{a}\in A}\mathbf{a} \quad \mathbf{c}_B = \dfrac{1}{|B|}\sum_{\mathbf{b}\in B}\mathbf{b}$

An Example

$$d = \sqrt{(e_i - e_j)^2 + (m_i - m_j)^2}$$

	English	Math
St 1	5	1
St 2	4	2
St 3	1	5
St 4	5	4
St 5	5	5

	1	2	3	4
1				
2	1.41			
3	5.66	4.24		
4	3.00	2.24	4.12	
5	4.00	3.16	4.00	1.00

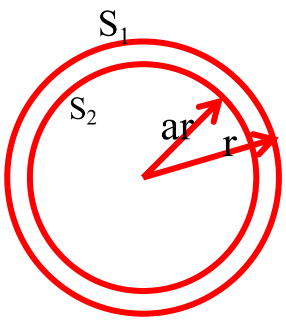
	1	2	3
1			
2	1.41		
3	5.66	4.24	
C1(4,5)	4.00	3.16	4.12

K-nearest neighbor algorithm

- Classifying objects based on closest training examples in the feature space
- Classify an object into a class to which most frequent training samples near it belong (among nearest ***k*** samples)
- Benefit: simple, often useful
- Drawback: "majority voting" the major classes may dominate classification
- Parameter
  - If k is larger, it tends to be noise tolerant but classes ambiguous
  - If k is 1, it is called "nearest neighbor algorithm"

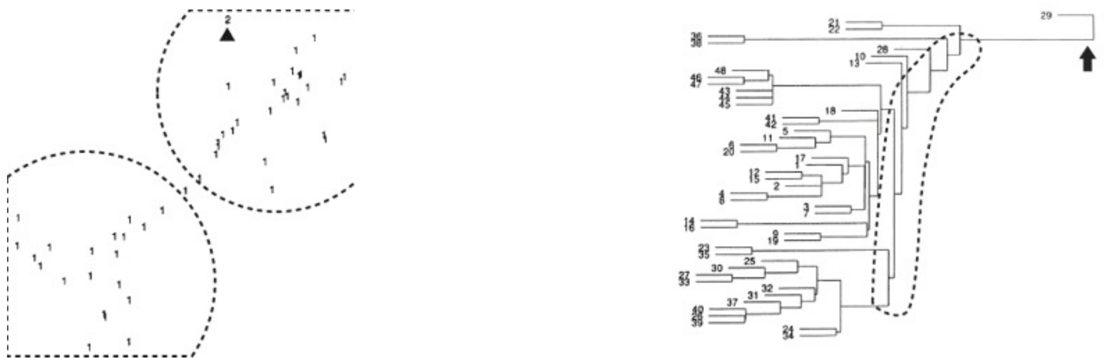
Notes for Clustering

- Curse of dimensionality / 次元の呪い
  - 0<a<1
  - $\delta V/V = 1 - a^d$
  - If d becomes bigger, $\delta V/V \to 1$



Notes for Clustering

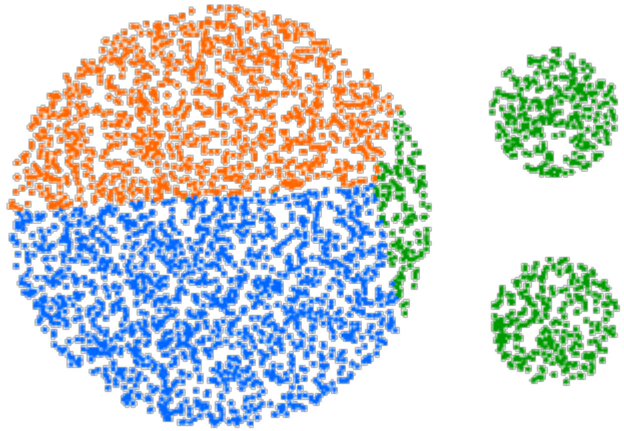
- Characteristics of the methods
  - Chaining



(a) data (b) dendrogram

B.S.Everitt: Cluster Analysis, Edward Arnold, third edition (1993)

  - k-means



S.Guha, R.Rastogi, and K.Shim: CURE: An Efficient Clustering Algorithm for Large Databases, in Proc. of the ACM SIGMOD International Conference on Management of Data, pp.73-80 (1998)

Information Extraction

- Extract the specified information from information sources.
- Natural Language Processing Techniques
  - Sentence segmentation
  - Word segmentation
    - Little problem for most Latin languages
    - Serious problem for Japanese, Chinese etc.
  - Part-of-speech tagging
  - Synthetic analysis (parsing)
- Ngram
- Keyword extraction
  - TF/IDF

Information Extraction

- Part-of-speech tagging
  - Identify a word class to each word in a sentence
    - Noun, pronoun, verb, adjective, verb, adverb, preposition, conjunction, interjection (English)
    - Verb, adjective, noun, prenominal adjective (連体詞), adverb, conjunction, interjection, auxiliary verb, postpositional particle (Japanese)
  - Tools
    - English
      - Stanford Log-linear Part-Of-Speech Tagger
      - Postagger (Tsujii lab)
      - •Lingua::EN::Tagger
    - Japanese
      - KAKASI
      - MeCab(和布蕪）, Sen
      - Chasen(茶筅)

Information Extraction

- Synthetic analysis (parsing)
  - Selection of grammar
  - Tree structure
  - Tools
    - Japanese
      - KNP
      - Cabocha
    - English
      - OPEN NLP

N-gram

- An n-gram is a substring of n item from a given string
  - 1-gram (unigram)
  - 2-gram (bigram, digram)
  - 3-gram (trigram)
- N-gram model: statistical model of n-gram occurrence
  - Indexing texts

Information Extraction

- *NLP Platform*
  - *UIMA*, Unstructured Information Management Architecture
  - U-Compare: All-in-one NLP system

Summary

- Information Gathering and Integration
  - Basic technologies for handling information
  - Knowledge is treated implicitly
    - e.g., classification reflects our knowledge how we classify information
    - Recent development of deep learning technologies is the same trend
  - But we have some explicit knowledge
    - We have categories for information for the specific aspects
    - We have typologies to represent typical information pieces
    - …
  - Need for two types of knowledge working together
    - Implicit
    - Explicit

Assignment 1

- Pick up two of the algorithms as follows. Explain them in general and make some example using the programs (find some libraries, don't make them from the scratch).
  - HITS Algorithm
  - decision tree
  - C4.5
  - k-means clustering
  - Bayesian Classification
  - *Or any algorithms you are interested in*

- Deadline: May 7, Friday, 2021. I will ask you to present in the lecture on May 10.
  - Mail to the report takeda@nii.ac.jp