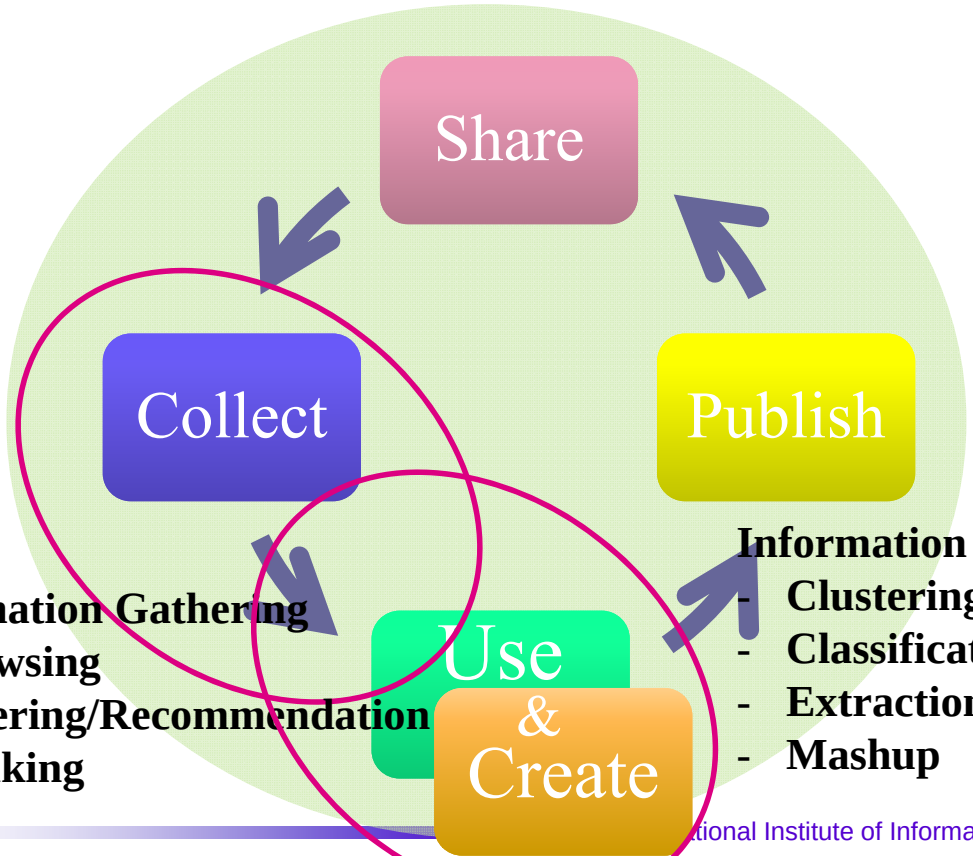
Information Cycle
1. Information Level

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

Information Cycle on Information Level



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

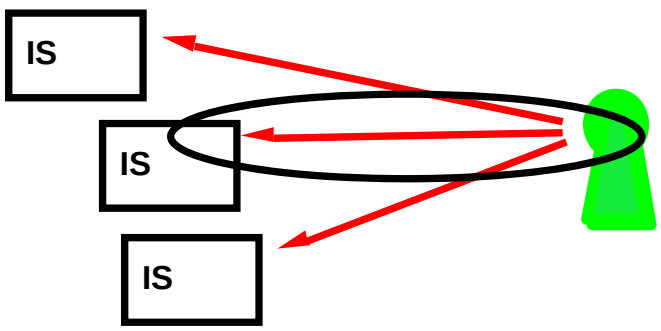
Information Integratio
- **Clustering**
- **Classification**
- **Extraction**
- **Mashup**

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

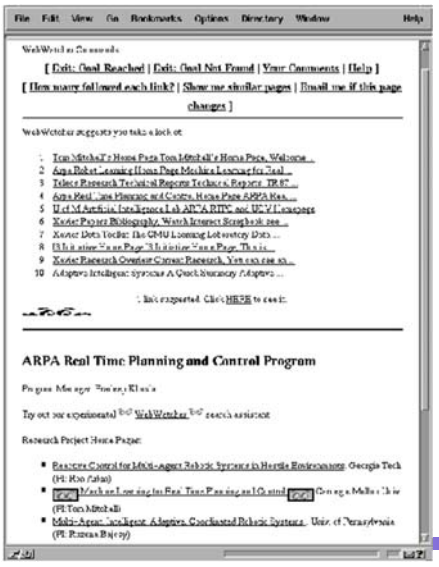
- Characteristics as Information Gathering
  - Pros:
    - Users initiative
    - Applicable even with vague purpose
  - Cons: No warranty to reach the goal
    - Human habit: up to down, side trip
    - Human limitation: Sequential access
- Problems
  - How to support users with keeping users initiative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

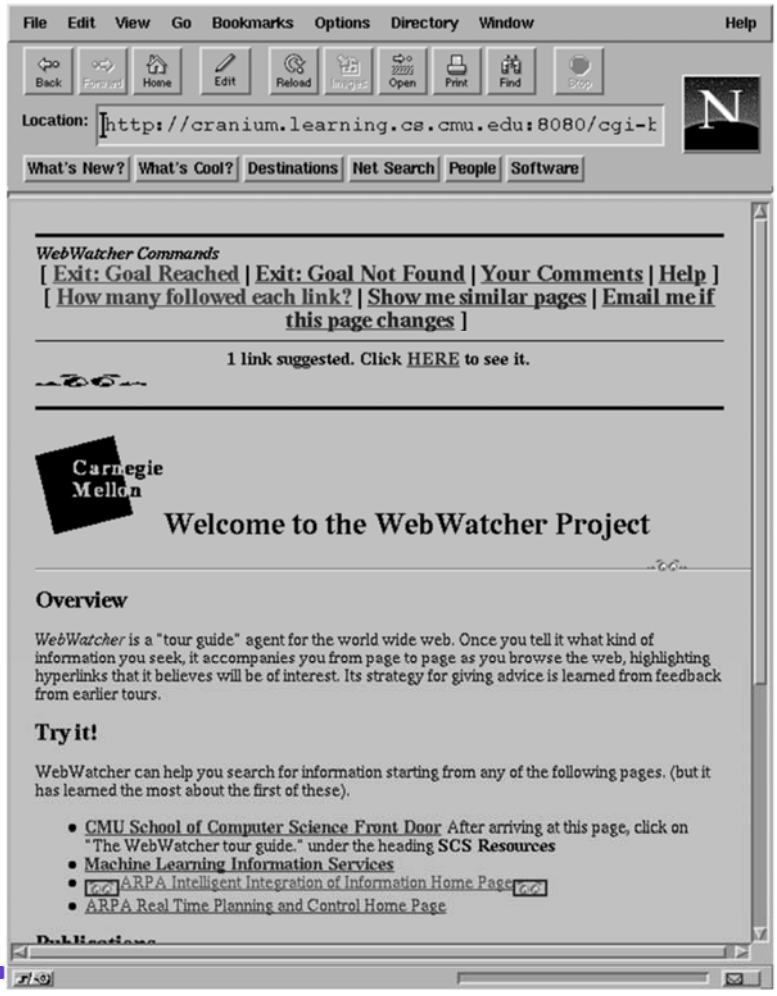
Browsing

- 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





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

Web Watcher

- Learning Method

  TFIDF(term frequency times inverse document frequency) [Salton] and other two methods

- Results

  2 or 3 times better than random recommendation

TF/IDF

- To measure importance of document ***d*** within document set ***D*** by word ***t***

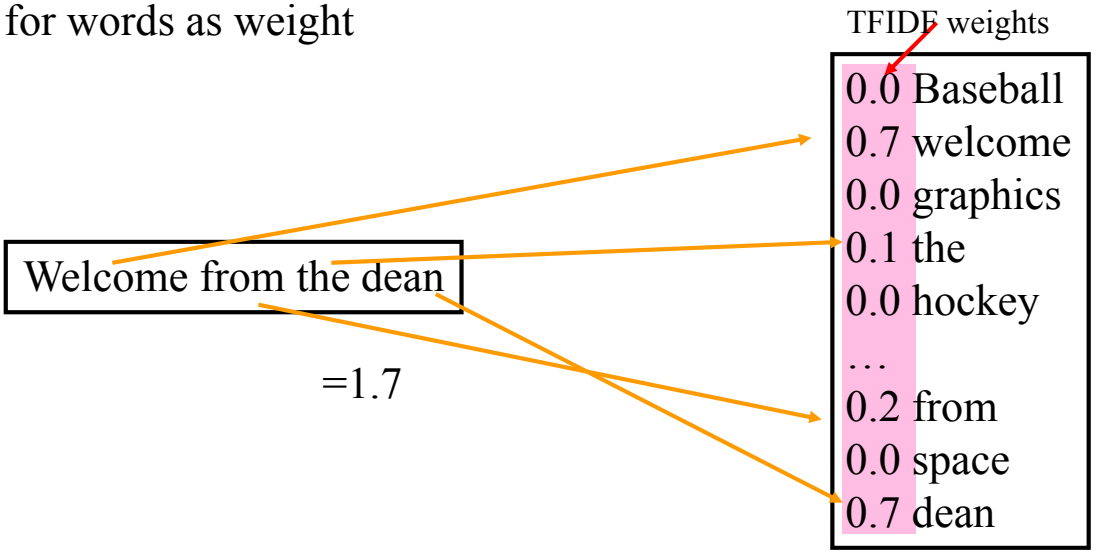
$$W(d,t,D) = TF(d,t) \cdot IDF(t,D)$$

  - ***TF(d,t)***: frequency of word ***t*** in document ***d***
  - $IDF(t,D) = \log(|D|/freq(D,t))$
  - ***freq(D, t)***: number of document in ***D*** in which ***t*** appears

- Intuitively
  - More frequent the specific word appears, more important
  - But, more documents have the specific words, less important

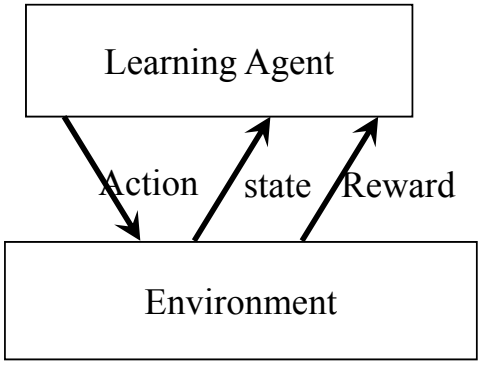
Use of TF/IDF in WebWatcher

- TF/IDF is calculated for the goal page
- Then each vector for “userchoice” is calculated with TF/IDF values for words as weight



Reinforcement Learning

- A model that an agent can learn its behavior through trial-and-error in a dynamic environment

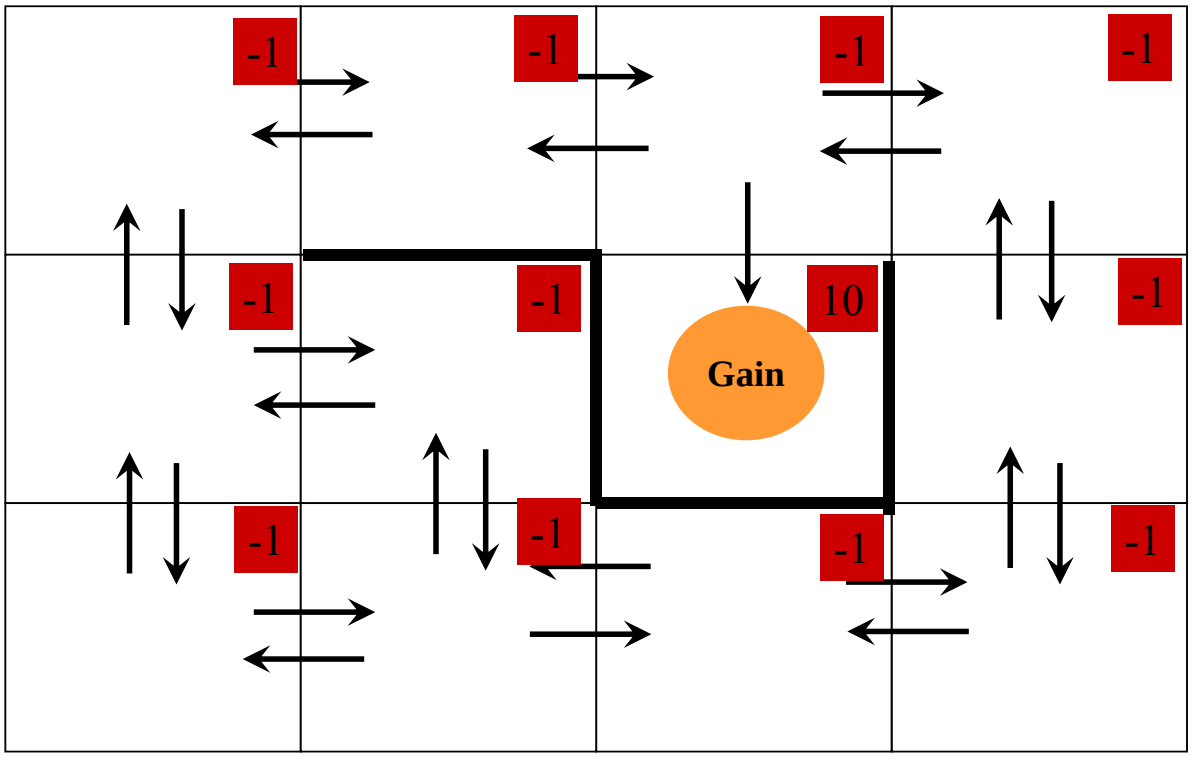


Basic elements in reinforcement learning

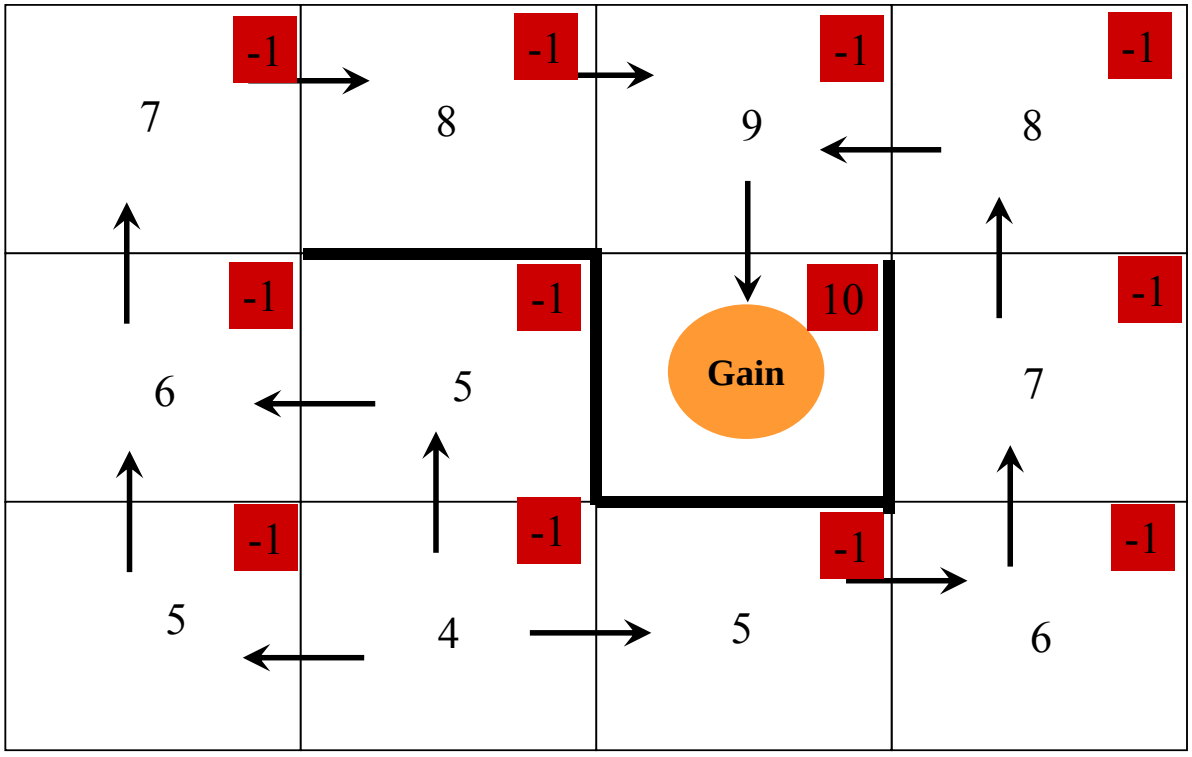
- Environment
  - The environment is (partially) observable by sensors
  - A set of states described by some parameters
- Reinforcement (Reward) function
  - Reinforcement or reward is feedback from the environment to the agent
  - The goal of reinforcement learning is to find a function to maximize rewards that it will receive in the future
  - Examples for reward: delayed rewards, minimal time …

- value function
  - policy
    - Decide next action in each state
    - Mapping from state to action
    - Optimal policy: mapping from any state to the state where maximal reward is given
  - state value
    - Sum of rewards from the initial state to the state
  - Value function
    - Mapping from a state to a state value
    - It depends on policy

The given environment



Optimal state value



Q-Learning

- R(s): reward at state s
- Q(s, a): evaluation function for action a at state s
- Delayed rewards

$$Q(S_t, a) = \sum_{i=0}^{\infty} \gamma^i R(S_{t+1+i})$$

$$Q_{n+1}(s, a) = R(s') + \lambda \max_{a' \in action} [Q_n(s', a')]$$

Q-learning in WebWatcher

- Page: state
- Hyperlink: action
- Reward for page: how much it is interesting for user
  - It is measured by sum of words weighted by TFIDF

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)

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

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)

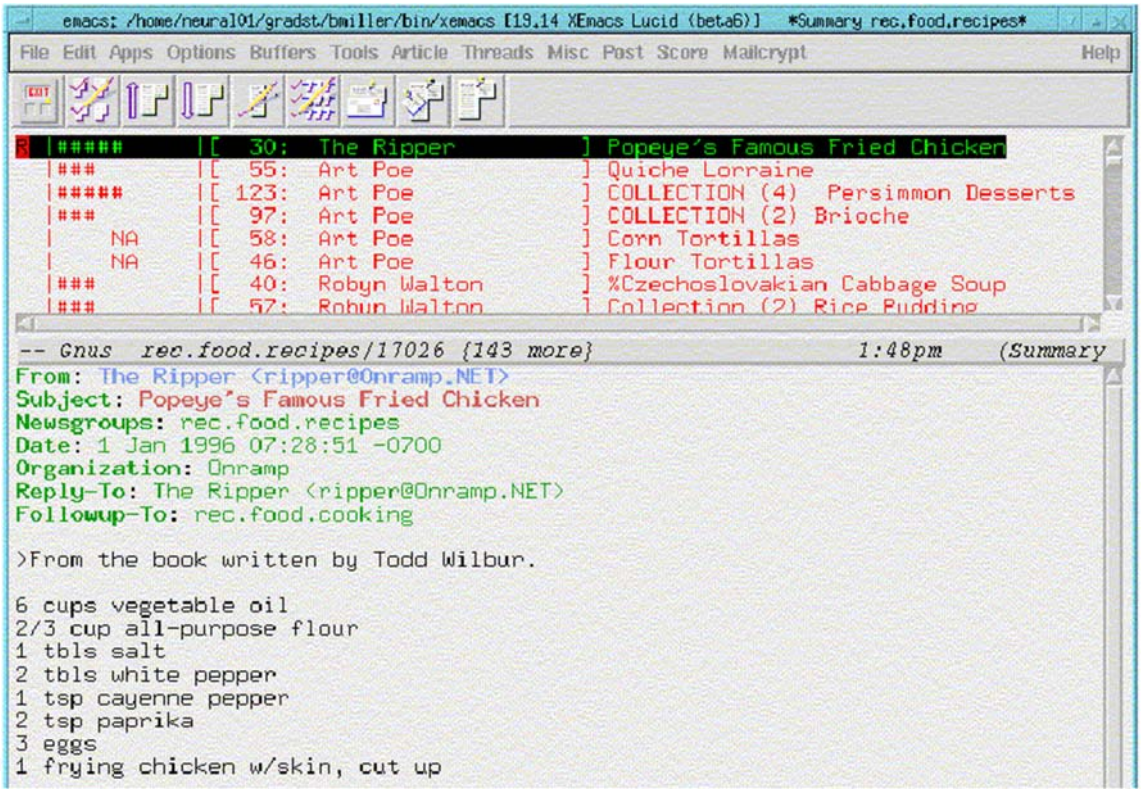
$$x'_{kl} = \bar{x}_k + \frac{\sum_{k' \neq k} (x_{k'l} - \bar{x}_{k'}) r_{kk'}}{\sum_{k' \neq k} |r_{kk'}|}$$

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

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

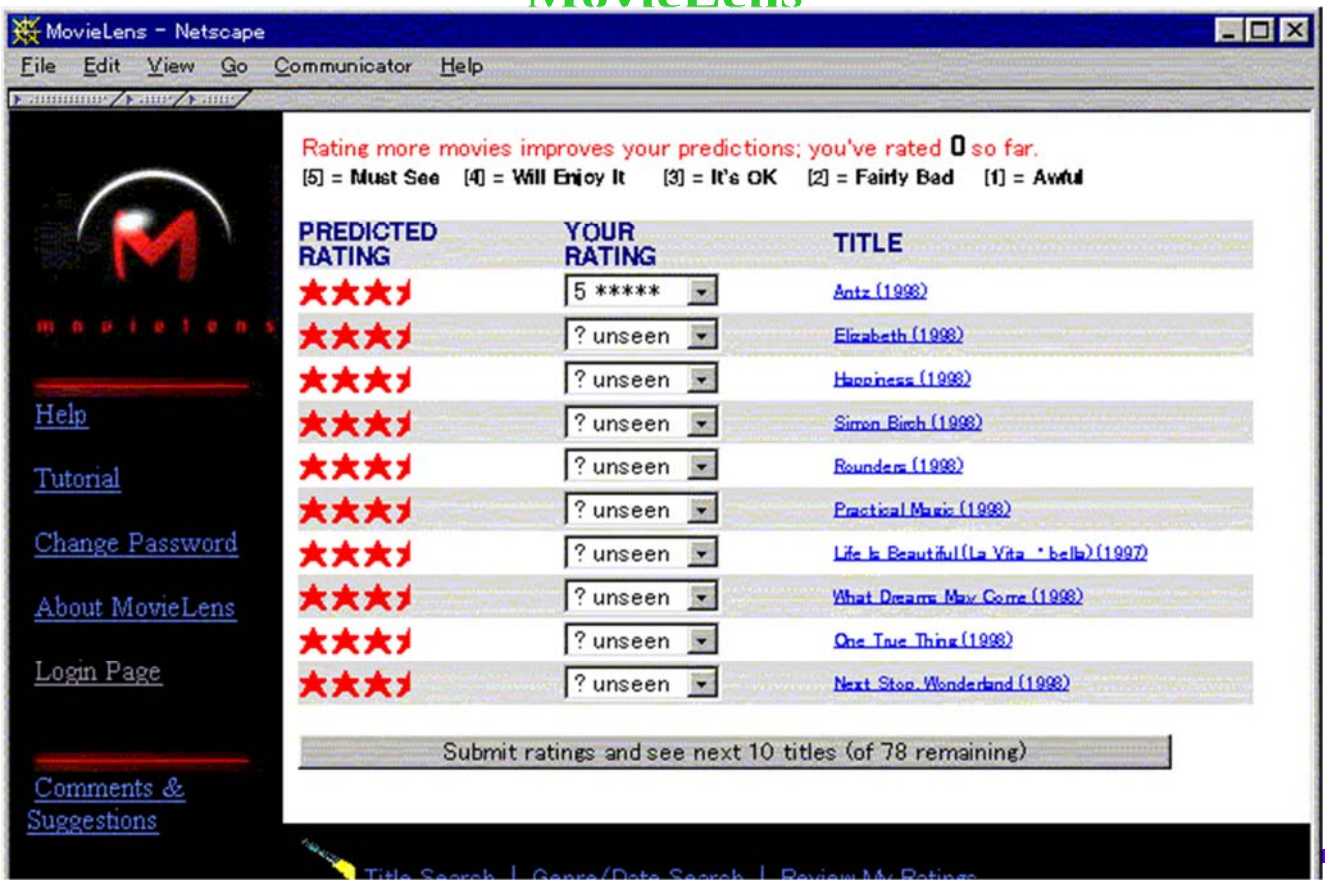
- Collaborative filtering system for NetNews



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



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 Alogrithm
    - …

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

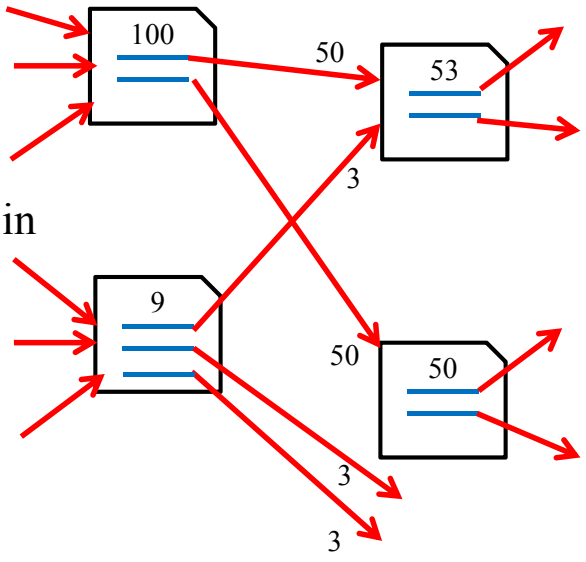
- The Simplified Model
  - If link (v-> u) exist,

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

where L is the number of links in Page v

- Dumping factor

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



PageRank

- $\boldsymbol{R} = \begin{bmatrix} PR(p_1) \\ PR(p_2) \\ \vdots \\ PR(p_N) \end{bmatrix}$
- $\boldsymbol{R} = \begin{bmatrix} (1-d)/N \\ (1-d)/N \\ \vdots \\ (1-d)/N \end{bmatrix} + d \begin{bmatrix} l(p_1,p_1) & l(p_1,p_2) & \cdots & l(p_1,p_N) \\ l(p_2,p_1) & & \ddots & \vdots \\ \vdots & & & \\ l(p_N,p_1) & & \cdots & l(p_N,p_N) \end{bmatrix}$
- $l(p_i,p_j) = \begin{cases} 1/\mathrm{L}\left(\mathrm{p_j}\right) & \text{link from } p_j \text{ to } p_i \text{ exists} \\ 0 & otherwise \end{cases}$
- $\sum_{i=1}^{N} l(p_i,p_j) = 1$
- **R**(t+1)= dM**R**(t) + (1-d) **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
  - Euclidean distance: $\|a-b\| = \sqrt{\sum_i (a_i - b_i)^2}$
  - Manhattan distance: $\|a-b\| = \sum_i |a_i - b_i|$
  - cosine similarity: $\frac{\boldsymbol{a}\cdot\boldsymbol{b}}{\|\boldsymbol{a}\|\|\boldsymbol{b}\|} = \sin\theta$
- Linkage criteria: the distance between two sets of data
  - Maximum: $\max\{d(a,b): a \in A, b \in B\}$
  - Minimum: $min\{d(a,b): a \in A, b \in B\}$
  - Mean: $\frac{1}{|A||B|}\sum_{a\in A}\sum_{b\in B} d(a,b)$

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”

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

Mashup

- Data integration with Web API
  - Web API (application program interface)
    - Access web via program not by human browsing
    - Interface:
      - SOAP (Simple Object Access Protocol)
      - REST (Representational State Transfer)
        - Just like accessing conventional web pages
  - Mashup with Web API
    - Creating a new service combing multiple Web API

Mashup examples

- **Woozor http://woozor.com/**
  - A Google Maps / Weather.com mashup providing 10 day weather forecasts all around the world. Google map, weather.com

- **Portwiture http://portwiture.com/**
  - Grabs photography from Flickr that matches the content of your most recent Twitter updates.
  - **Flickr, twitter**

Information Extraction

- プラットフォーム
  - *UIMA*, Unstructured Information Management Architecture
  - U-Compare: All-in-one NLP system