RECOMMENDATION MODELS FOR WEB USERS

Dr. Şule Gündüz Öğüdücü
sgunduz@itu.edu.tr

1

What is Web Mining?

- The use of data mining techniques to automatically discover and extract information from Web documents and services (Etzioni, 1996)
- Web mining research integrate research from several research communities (Kosala and Blockeel, 2000) such as:
  - Database (DB)
  - Information Retrieval (IR)
  - The sub-areas of machine learning (ML)
  - Natural language processing (NLP)

2

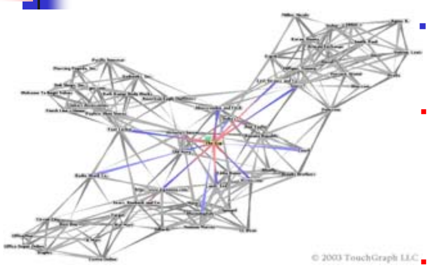
World-wide Web

- Initiated at CERN (the European Organization for Nuclear Research)
  - By Tim Berners-Lee
- GUIs
  - Berners-Lee (1990)
  - Erwise and Viola(1992), Midas (1993)
- Mosaic (1993)
  - a hypertext GUI for the X-window system
  - HTML: markup language for rendering hypertext
  - HTTP: hypertext transport protocol for sending HTML and other data over the Internet
  - CERN HTTPD: server of hypertext documents
- 1994
  - Netscape was founded
  - 1st World Wide Web Conference
  - World Wide Web Consortium was founded by CERN and MIT
    http://www.w3.org/

Mining the Web — Chakrabarti and Ramakrishnan

3

WWW: Incentives



http://www.touchgraph.com/TGGoogleBrowser.html

- WWW is a huge, widely distributed, global information source for:
  - Information services: news, advertisements, consumer information, financial management, education, government, e-commerce, health services, etc.
  - Hyper-link information
  - Web page access and usage information
  - Web site contents and organizations

4

Mining the World Wide Web

- Growing and changing very rapidly
  - 6 December 2006 : 12.52 billion pages
    http://www.worldwidewebsize.com/
- Only a small portion of information on the Web is truly relevant or useful to Web user
- WWW provides rich sources for data mining
- Goals include:
  - Target potential customers for electronic commerce
  - Enhance the quality and delivery of Internet information services to the end user
  - Improve Web server system performance
  - Facilitates personalization/adaptive sites
  - Improve site design
  - Fraud/intrusion detection
  - Predict user's actions

5

Application Examples: e-Commerce

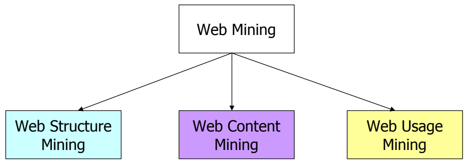
6

Challenges on WWW Interactions

- Searching for usage patterns, Web structures, regularities and dynamics of Web contents
- Finding relevant information
  - 99% of info of no interest to 99% of people
- Creating knowledge from information available
  - Limited query interface based on keyword – oriented search
- Personalization of the information
  - Limited customization to individual users

7

Web Mining Taxonomy



8

Web Usage Mining

- Web usage mining also known as Web log mining
  - The application of data mining techniques to discover usage patterns from the secondary data derived from the interactions of users while surfing the Web
- Information scent

  Information: Food

  WWW: Forest

  User: Animal

  Understanding the behavior of an animal that looks for food in the forest

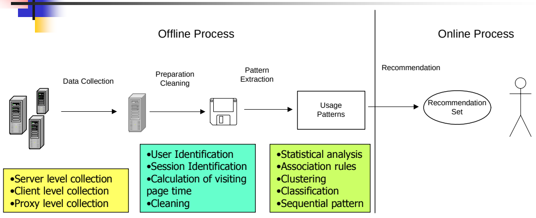
9

Terms in Web Usage Mining

- User: A single individual who is accessing files from one or more Web servers through a browser.
- Page file: The file that is served through a HTTP protocol to a user.
- Page: The set of page files that contribute to a single display in a Web browser constitutes a Web page.
- Click stream: The sequence of pages followed by a user.
- Server session (visit): A set of pages that a user requested from a single Web server during her single visit to the Web site.

10

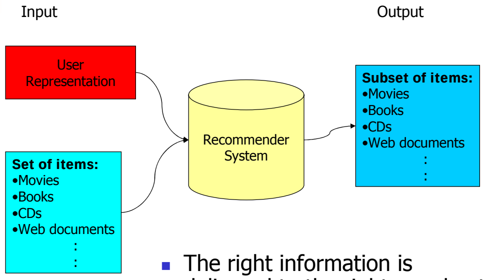
Methodology



- Data collection
- Data preparation and cleaning
- Pattern extraction
- Recommendation

11

Recommendation Process



- The right information is delivered to the right people at the right time.

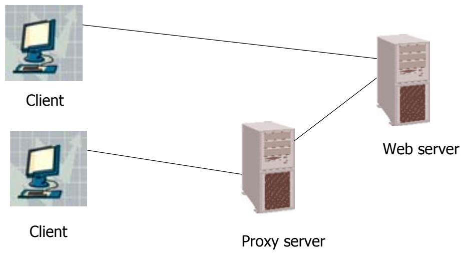
12

What information can be included in User Representation?

- The order of visited Web pages
- The visiting page time
- The content of the visited Web page
- The change of user behavior over time
- The difference in usage and behavior from different geographic areas
- User profile

Data Collection

- Server level collection
- Client level collection
- Proxy level collection



Web Server Log File Entries

IP Address	User ID	Timestamp	Method/ URL	Status	Size

Source of Request	User ID	Date and Time of Request	Method,URL, (HTTP Protocol)	Status Code	Num. of Bytes
216.35.116.28	-	[11/Jan/2002:00:58:43 -0500]	"GET / HTTP/1.0"	200	6557
216.35.116.28	-	[11/Jan/2002:00:58:53 -0500]	"GET a.gif HTTP/1.0"	200	5478
216.35.116.28	-	[11/Jan/2002:00:59:25 -0500]	"GET b.gif HTTP/1.0"	200	6057
216.35.116.28	-	[11/Jan/2002:00:59:54 -0500]	"GET B.html HTTP/1.1"	200	59825
216.35.116.28	-	[11/Jan/2002:00:59:54 -0500]	"GET B.gif HTTP/1.1"	200	2050
24.102.227.6	-	[11/Jan/2002:00:59:55 -0500]	"GET index.html HTTP/1.1"	200	6557
216.35.116.28	-	[11/Jan/2002:00:59:55 -0500]	"GET C.html HTTP/1.1"	200	2560
24.102.227.6	-	[11/Jan/2002:00:59:56 -0500]	"GET a.gif HTTP/1.1"	200	5478
24.102.227.6	-	[11/Jan/2002:00:59:56 -0500]	"GET b.gif HTTP/1.1"	200	6057
24.102.227.6	-	[11/Jan/2002:00:59:57 -0500]	"GET D.HTML HTTP/1.1"	200	12800
24.102.227.6	-	[11/Jan/2002:00:59:58 -0500]	"GET G.gif HTTP/1.1"	200	1500
24.102.227.6	-	[11/Jan/2002:00:59:58 -0500]	"GET e.gif HTTP/1.1"	200	1230
24.102.227.6	-	[11/Jan/2002:00:59:59 -0500]	"GET c.jpg HTTP/1.1"	200	3345
216.35.116.28	-	[11/Jan/2002:00:59:59 -0500]	"GET c.jpg HTTP/1.1"	200	2247
216.35.116.28	-	[11/Jan/2002:01:00:00 -0500]	"GET E.jpg HTTP/1.1"	200	2247
216.35.116.28	-	[11/Jan/2002:01:00:00 -0500]	"GET D.html HTTP/1.1"	200	32768
216.35.116.28	-	[11/Jan/2002:01:00:01 -0500]	"GET D.gif HTTP/1.1"	200	7977
216.35.116.28	-	[11/Jan/2002:01:00:01 -0500]	"GET d.jpg HTTP/1.1"	200	6121
216.35.116.28	-	[11/Jan/2002:01:00:02 -0500]	"GET e.jpg HTTP/1.1"	200	3567
24.102.227.6	-	[11/Jan/2002:01:00:02 -0500]	"GET C.html HTTP/1.1"	200	32768

Use of Log Files

- Questions
  - Who is visiting the Web site?
  - What is the path users take through the Web pages?
  - How much time users spend on each page?
  - Where and when visitors are leaving the Web site?

Data Preprocessing (1)

- Data cleaning
  - Remove log entries with filename suffixes such as gif, jpeg, GIF, JPEG, jpg, JPG
  - Remove the page requests made by the automated agents and spider programs
  - Remove the log entries that have a status code of 400 and 500 series
  - Normalize the URLs: Determine URLs that correspond to the same Web page
- Data integration
  - Merge data from multiple server logs
  - Integrate semantics (meta-data)
  - Integrate registration data

Data Preprocessing (2)

- Data transformation
  - User identification
    - Users with the same client IP are identical
  - Session idendification
    - A new session is created when a new IP address is encountered or if the visiting page time exceeds 30 minutes for the same IP-address
- Data reduction
  - sampling
  - dimensionality reduction

Discovery of Usage Patterns

- Pattern Discovery is the key component of the Web mining, which converges the algorithms and techniques from data mining, machine learning, statistics and pattern recognition etc research categories
- Separate subsections:
  - Statistical analysis
  - Association rules
  - Clustering
  - Classification
  - Sequential pattern

Statistical Analysis

- Different kinds of statistical analysis (frequency, median, mean, etc.) of the session file, one can extract statistical information such as:
  - The most frequently accessed pages
  - Average view time of a page
  - Average length of a path through a site
- Application:
  - Improving the system performance
  - Enhancing the security of the system
  - Providing support for marketing decisions
- Examples:
  - PageGather (Perkowitz et al., 1998)
  - Discovering of user profiles (Larsen et al., 2001)

Association Rules

- Sets of pages that are accessed together with a support value exceeding some specified threshold
- Application:
  - Finding related pages that are accessed together regardless of the order
- Examples:
  - Web caching and prefetching (Yang et al., 2001)

Clustering

- A technique to group together objects with the similar characteristics
  - Clustering of sessions
  - Clustering of Web pages
  - Clustering of users
- Application:
  - Facilitate the development and execution of future marketing strategies
- Examples:
  - Clustering of user sessions (Gunduz et al., 2003)
  - Clustering of individuals with mixture of Markov models (Sarukkai, 2000)

Classification

- The technique to map data item into one of several predefined classes
- Application
  - Developing a usage profile belonging to a particular class or category
- Examples:
  - WebLogMiner (Zaiane et al., 1998)

Sequential Pattern

- Discovers frequent subsequences as patterns
- Applications:
  - The analysis of customer purchase behavior
  - Optimization of Web site structure
- Examples:
  - WUM (Spiliopoulou and Faulstich, 1998)
  - Longest Repeated Subsequence (Pitkow and Pirolli, 1999)

Online Module: Recommendation

- The discovered patterns are used by the online component of the model to provide dynamic recommendations to users based on their current navigational activity
- The produced recommendation set is added to the last requested page as a set of links before the page is sent to the client browser

Probabilistic models of browsing behavior

- Useful to build models that describe the browsing behavior of users
- Can generate insight into how we use Web
- Provide mechanism for making predictions
- Can help in pre-fetching and personalization

Markov models for page prediction

- General approach is to use a finite-state Markov chain
  - Each state can be a specific Web page or a category of Web pages
  - If only interested in the order of visits (and not in time), each new request can be modeled as a transition of states
- Issues
  - Self-transition
  - Time-independence

Markov models for page prediction

- For simplicity, consider order-dependent, time-independent finite-state Markov chain with M states
- Let s be a sequence of observed states of length L. e.g. s = ABBCAABBCCBBAA with three states A, B and C. s_t is state at position t (1<=t<=L). In general,

$$P(s) = P(s_1)\prod_{t=2}^{L} P(s_t \mid s_{t-1},\ldots,s_1)$$

- Under a first-order Markov assumption, we have

$$P(s) = P(s_1)\prod_{t=2}^{L} P(s_t \mid s_{t-1})$$

- This provides a simple generative model to produce sequential data

Markov models for page prediction

- If we denote $T_{ij} = P(s_t = j | s_{t-1} = i)$, we can define a M x M *transition matrix*
- Properties
  - Strong first-order assumption
  - Simple way to capture sequential dependence
- If each page is a state and if W pages, O(W²), W can be of the order 10^5 to 10^6 for a CS dept. of a university
- To alleviate, we can cluster W pages into M clusters, each assigned a state in the Markov model
- Clustering can be done manually, based on directory structure on the Web server, or automatic clustering using clustering techniques

Markov models for page prediction

- $T_{ij} = P(s_t = j | s_{t-1} = i)$ now represent the probability that an individual user's next request will be from category *j*, given they were in category *i*
- We can add E, an end-state to the model
- E.g. for three categories with end state: -

$$T = \begin{pmatrix} P(1|1) & P(2|1) & P(3|1\_) & P(E|1) \\ P(1|2) & P(2|2) & P(3|2) & P(E|2) \\ P(1|3) & P(2|3) & P(3|3) & P(E|3) \\ P(1|E) & P(2|E) & P(3|E) & 0 \end{pmatrix}$$

- E denotes the end of a sequence, and start of a new sequence

Markov models for page prediction

- First-order Markov model assumes that the next state is based only on the current state
- Limitations
  - Doesn't consider 'long-term memory'
- We can try to capture more memory with *k*th-order Markov chain

$$P(s_t \mid s_{t-1},\ldots,s_1) = P(s_t \mid s_{t-1},\ldots,s_{t-k})$$

- Limitations
  - Inordinate amount of training data $O(M^{k+1})$

Fitting Markov models to observed page-request data

- Assume that we collected data in the form of N sessions from server-side logs, where i^{th} session s_i, $1 <= i <= N$, consists of a sequence of L_i page requests, categorized into M – 1 states and terminating in E. Therefore, data $D = \{s_1, \ldots, s_N\}$
- Let θ denote the set of parameters of the Markov model, θ consists of M^2 -1 entries in T
- Let θ_{ij} denote the estimated probability of transitioning from state *i* to *j*.

Fitting Markov models to observed page-request data

- The *likelihood* function would be

$$L(\theta) = P(D \mid \theta) = \prod_{i=1}^{N} P(s_i \mid \theta)$$

- This assumes conditional independence of sessions.
- Under Markov assumptions, likelihood is

$$L(\theta) = \prod \theta_{ij}^{n_{ij}}, 1 \le i, j \le M$$

- where n_{ij} is the number of times we see a transition from state *i* to state j in the observed data D.

Fitting Markov models to observed page-request data

- For convenience, we use log-likelihood

$$l(\theta) = \log L(\theta) = \sum_{ij} n_{ij} \log \theta_{ij}$$

- We can maximize the expression by taking partial derivatives *wrt* each parameter and incorporating the constraint (via Lagrange multipliers) that the sum of transition probabilities out of any state must sum to one $\sum_j \theta_{ij} = 1$
- The maximum likelihood (ML) solution is $\theta_{ij}^{ML} = \frac{n_{ij}}{n_i}$

Bayesian parameter estimation for Markov models

- In practice, M is large ($\sim 10^2$-10^3), we end up estimating M^2 probabilities
- D may contain potentially millions of sequences, so some $n_{ij} = 0$
- Better way would be to incorporate prior knowledge – prior probability distribution $P(\theta)$ and then maximize , the posterior distribution on $P(\theta \mid D)$ given the data (rather than $P(D \mid \theta)$)
- Prior distribution reflects our prior belief about the parameter set θ
- The posterior reflects our posterior belief in the parameter set now informed by the data D

Bayesian parameter estimation for Markov models

- For Markov transition matrices, it is common to put a distribution on each row of T and assume that each of these priors are independent $P(\theta) = \prod_i P(\{\theta_{i1},\ldots,\theta_{iM}\})$

  where $\sum_j \theta_{ij} = 1$
- Consider the set of parameters for the i_{th} row in T, a useful prior distribution on these parameters is the Dirichlet distribution defined as

$$P(\{\theta_{i1},\ldots,\theta_{iM}\}) = D_{\alpha q_i} = C \prod_{j=1}^{M} \theta_{ij}^{\alpha(q_{ij}-1)}$$

- where $\alpha, q_{ij} > 0, \sum_j q_{ij} = 1$, and C is a normalizing constant

Bayesian parameter estimation for Markov models

- The MP posterior parameter estimates are

$$\theta_{ij}^{MP} = \frac{n_{ij} + \alpha q_{ij}}{n_i + \alpha}$$

- If n_{ij} = 0 for some transition (*i, j*) then instead of having a parameter estimate of 0 (ML), we will have $\alpha q_{ij} / (n_i + \alpha)$ allowing prior knowledge to be incorporated
- If n_{ij} > 0, we get a smooth combination of the data-driven information (n_{ij}) and the prior

37

Bayesian parameter estimation for Markov models

- One simple way to set prior parameter is
  - Consider alpha as the effective sample size
  - Partition the states into two sets, set 1 containing all states directly linked to state *i* and the remaining in set 2
  - Assign uniform probability e/K to all states in set 2 (all set 2 states are equally likely)
  - The remaining (1-e) can be either uniformly assigned among set 1 elements or weighted by some measure
  - Prior probabilities in and out of E can be set based on our prior knowledge of how likely we think a user is to exit the site from a particular state

38

Predicting page requests with Markov models

- Many flavors of Markov models proposed for next page and future page prediction
- Useful in pre-fetching, caching and personalization of Web page
- For a typical website, the number of pages is large – Clustering is useful in this case
- First-order Markov models are found to be inferior to other types of Markov models
- *k*th-order is an obvious extension
  - Limitation: $O(M^{k+1})$ parameters (combinatorial explosion)

39

Predicting page requests with Markov models

- Deshpande and Karypis (2001) propose schemes to prune kth-order Markov state space
  - Provide systematic but modest improvements
- Another way is to use empirical smoothing techniques that combine different models from order 1 to order k (Chen and Goodman 1996)
- Cadez *et al.* (2003) and Sen and Hansen (2003) propose mixtures of Markov chains, where we replace the first-order Markov chain:

$$P(s_t \mid s_{t-1}, \ldots, s_1) = P(s_t \mid s_{t-1})$$

40

Predicting page requests with Markov models

with a mixture of first-order Markov chains

$$P(s_t \mid s_{t-1}, \ldots, s_1) = \sum_{k=1}^{K} P(s_t \mid s_{t-1}, c = k) P(c = k)$$

- where c is a discrete-value hidden variable taking K values Sum_k P(c = k) = 1and

  $P(s_t \mid s_{t-1}, c = k)$ is the transition matrix for the *k*th mixture component
- One interpretation of this is user behavior consists of K different navigation behaviors described by the K Markov chains
- Cadez *et al.* use this model to cluster sequences of page requests into K groups, parameters are learned using the EM algorithm

41

Predicting page requests with Markov models

- Consider the problem of predicting the next state, given some number of states t
- Let $s_{[1,t]} = \{s_1, \ldots, s_t\}$ denote the sequence of *t* states
- The predictive distribution for a mixture of K Markov models is

$$P(s_{t+1} \mid s_{[1,t]}) = \sum_{k=1}^{K} P(s_{t+1}, c = k \mid s_{[1,t]})$$

$$= \sum_{k=1}^{K} P(s_{t+1} \mid s_{[1,t]}, c = k) P(c = k \mid s_{[1,t]})$$

$$= \sum_{k=1}^{K} P(s_{t+1} \mid s_t, c = k) P(c = k \mid s_{[1,t]})$$

- The last line is obtained if we assume conditioned on component c = k, the next state s_{t+1} depends only on s_t

42

Predicting page requests with Markov models

- Weight based on observed history is

$$P(c = k \mid s_{[1,t]}) = \frac{P(s_{[1,t]} \mid c = k)P(c = k)}{\sum_j P(s_{[1,t]} \mid c = j)P(c = j)}, 1 \le k \le K$$

where $P(s_{[1,t]} \mid c = k) = P(s1 \mid c = k)\prod_{t=2}^{L} P(s_t \mid s_{t-1}, c = k)$

- Intuitively, these membership weights 'evolve' as we see more data from the user
- In practice,
  - Sequences are short
  - Not realistic to assume that observed data is generated by a mixture of K first-order Markov chains
- Still, mixture model is a useful approximation

43

Predicting page requests with Markov models

- K can be chosen by evaluating the out-of-sample predictive performance based on
  - Accuracy of prediction
  - Log probability score
  - Entropy
- Other variations of Markov models
  - Sen and Hansen 2003
  - Position-dependent Markov models (Anderson *et al.* 2001, 2002)
  - Zukerman *et al.* 1999

44

Collaborative Filtering

- Method
  - Correlation between the item contents (such as term frequency inverse document frequency and weighting)
  - Correlation between user's interests (such as votes and trails)
    - Results are captured in a generally sparse matrix (users x items)
- Problems
  - Sparsity
  - Cold-start

45

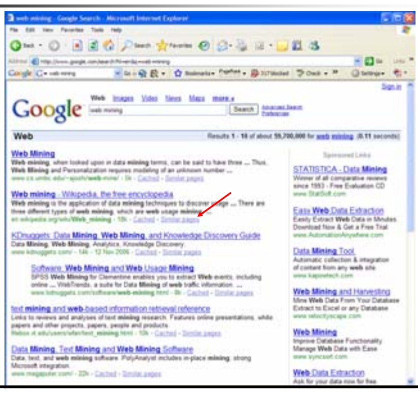
Examples of Collaborative Filtering

- Customer X
  - Buys Falcon CD
  - Buys Stret Dogs CD

- Customer Y
  - Searchs for Falcon
  - Street Dogs is recommended

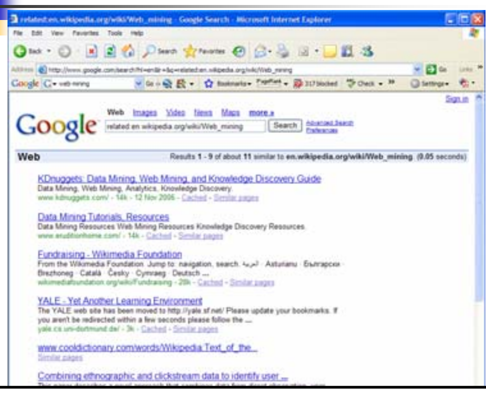
46

Examples of Collaborative Filtering



47

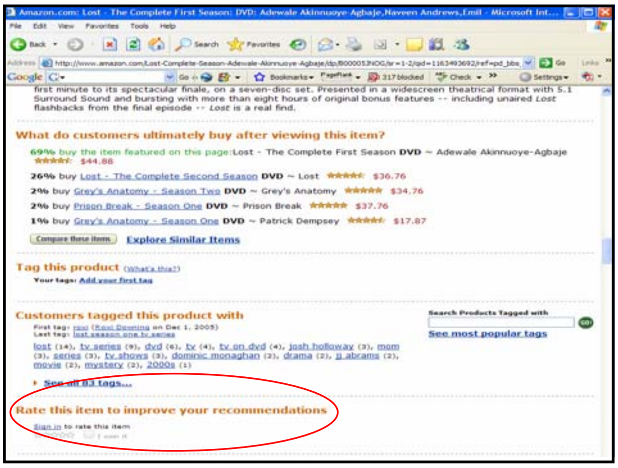
Examples of Collaborative Filtering



48

Examples of Collaborative Filtering

49



Networks & Recommendation

- Word-of-Mouth
  - Needs little explicit advertising
  - Products are recommended to friends, family, co-workers, etc.
  - This is the primary form of advertising behind the growth of Google

51

Email Product Recommendation

- Hotmail
  - Very little direct advertising in the beginning
  - Launched in July 1996
    - 20,000 subscribers after a month
    - 100,000 subscribers after 3 months
    - 1,000,000 subscribers after 6 months
    - 12,000,000 subscribers after 18 months
    - By April 2002 Hotmail had 110 million subscribers

52

Email Product Recommendation

- What was Hotmail's primary form of advertising?
  - Small link to the sign up page at the bottom of every email sent by a subscriber
    - 'Spreading Activation'
    - Implicit recommendation

53

Spreading Activation

- Network effects
  - Even if a small number of people who receive the message subscribe (~0.1%), the service will spread rapidly
  - This can be contrasted with the current practice of SPAM
    - SPAM is not sent by friends, family, co-workers
    - No implicit recommendation
    - SPAM is often viewed as not providing a good service

54

Page Prediction

- Click-straem Tree Model (Gunduz et. al. 2003)
  - Pair-wise similarities of user sessions are calculated
    - the order of pages
    - the distance between identical pages
    - the time spent on these pages
  - A graph is constructed
    - Vertices: User sessions
    - Edges: Pair-wise similarities
  - A graph-based clustering approach is applied
  - Each cluster is then represented by a click-stream tree whose nodes are pages of user sessions of that cluster
  - When a request is received from an active user, a recommendation set consisting of three different pages that the user has not yet visited, is produced using the best matching user session

Problems

- Evaluation of recommender systems
- Poor data
- Lack of data
- Privacy control

References

- http://ibook.ics.uci.edu/slides.html
- http://www-users.cs.umn.edu/~kumar/dmbook/
- O. Etzioni. *The world wide web: Quagmire or gold mine*. Communications of the ACM, 39(11):65-68, 1996.
- R. Kosala and H. Blockeel. *Web Mining Research: A summary*. SIGKDD Explorations, 2(1):1-15, 2000
- L. Page, S. Brin, R. Motwani, T. Winograd, *The PageRank Citation Ranking: Bringing Order to the Web*. In Proc. of the 7th Intenational World Widw Web Conference, 161-172, 1998

References

- L. V.S. Lakshmanan, F. Sadri, and I. N. Subramanian. *A Declarative Language for Querying and Restructing the World Wide web*. Post-ICDE IEEE Workshop on Research Issues in Data Engineering (RIDE-NDS'96). New Orleans, February 1996.
- G. O. Arocena and A. O. Mendelzon. *WebOQL: Restructuring Documents, Databases and Webs*. In Proc. ICDE'98, 1998
- M. Perkowitz and O. Etzioni, *Adaptive Web Sites: Automatically Synthesizing Web Pages*. In Proc. of Fifteenth National Conference on Artificial Intelligence, 1998
- J. Larsen, L. K.Hansen, A. Szymkowiak, T. Christiansen and T. Kolenda, *Webmining: Learning from the World Wide Web*. Computational Statistics and Data Analysis, 2001
- Q. Yang, H. H. Zhang and I. T. Yi Li, *Mining web logs for prediction models in WWW caching and prefetching*. Knowledge Discovery and Data Mining, 473-478, 2001

References

- Y. Yan, M. Jacobsen and H. G. Molina and U. Dayal, *From User Access Patterns to Dynamic Hypertext Linking*. In Proc. 5th Int. World Wide Web Conference, 1007-1014, 1996 C. Shahabi, A. Zarkesh, J. Adibi and V. Shah, Knowledge Discovery from Users Web-Page Navigation. In Proc. 7th Int. Workshop on Research Issues in Data Engineering, 20-29, 1997
- R. R. Sarukkai, *Link Prediction and Path Analysis Using Markov Chains*. In Proc. 9th Int. World Wide Web Conference, 377-386, 2000.
- O. R. Zaiane, M. Xin and J. Han, *Discovering Web Access Patterns and ternds by applying OLAP and data mining technology on Web logs*. Advances in Digital Libraries, 19-29, 1998
- M. Spiliopoulou and L. C. Faulstich, *WUM: A Tool for Web Utilization Analysis*, extended version of Proc. EDBT Workshop WebDB'98, 184-203, 1998

References

- J.Pitkow and P.Pirolli, *Mining Longest Repeating Subsequences to Predict World Wide Web Surfing*. In Proc. USENIX Symp. on Internet Technologies and Systems (USITS'99), 1999.
- B. Mobasher and H. Dai and T. Luo and M. Nakagawa, *Discovery of Aggregate Usage Profiles for Web Personalization*. In roc. International {WEBKDD} Workshop -- Web Mining for E-Commerce: Challenges and Opportunities, 2000.
- B. Mobasher and H. Dai and T. Luo and M. Nakagawa, *Effective Personalization Based on Association Rule Discovery* from Web Usage Data. In Proc. 3rd ACM Workhop on Web Information and Data Management, 2001.