

Chapter II

Web Data

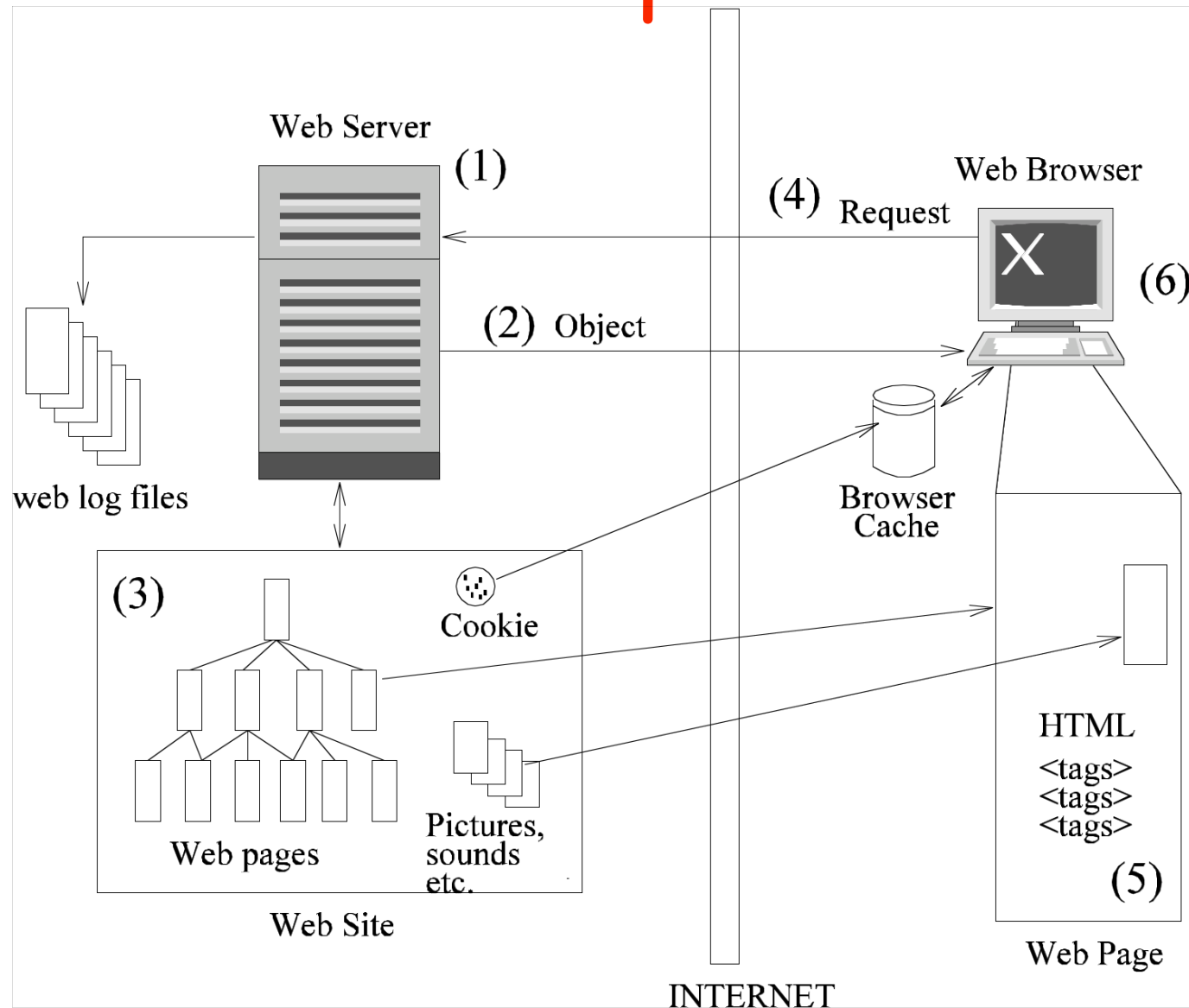
Prof. Juan D. Velásquez
Prof. Aux Pablo Román
Web intelligence Research Group
<http://wi.dii.uchile.cl/>

Outline

- Web's Operation
- The Information Behind the Clicks
- Session reconstruction process
- Finding real sessions
- The information contained in a web page
- Web page content
- Web page links

2.1 Web's Operation

The web process



Web server and web browser

- Once the document has been read by the browser, the specific tags inside are interpreted.
- When the browser interpreting the tags find a reference about an object, for instance an image, the HTTP gets it and transfers it to the browser.
- The process finishes when the last tag is interpreted and the page is shown to the visitor.

The web process

- The transport operation use the HTTP protocol.
- Web pages are written in the HTML language.
- A web page contains tags that reference other object to ask to the server or to be download to the user browser.
- Content are usually more complex than they appears:
 - Applets
 - Javascripts
 - Dynamics HTML
 - Flash

Web server and web browser (2)

- The web log registers contain information about the visitor browsing behavior, in particular the page navigation sequence and the time spent in each page visited.
- When a web page is accessed, the HTML code, with web page tags referring to various web objects, is interpreted in the browser.
- A register is created for the accessed page as well as for each object referred in the page.
- Depending on the web activity, these logs can contain millions of registers and most of them may not hold relevant information.

The web Server

- A Server is a program not a machine.
- They usually serve not only plain web pages , they also serve web applications with HTML front-end.
- Web application architecture: Multiple layer Model.
 - Interface Layer: Perform html rendering.
 - Logic Layer: core business application.
 - Data Layer: Storage/Retrieval data process.
- Web server also maintain data logs about user action in the web: Some site could have of the order of Gb/day of logs.

The web Client

- The browser perform all the required management of the connection with the server.
- there are also allowed to connect to proxy server that are cache server of statics pages.
- Also execute local program (client side application)
- Maintain a repository of client side data called "Cookies", that allows to web application:
 - retrieve particular information about a particular client.
 - maintain a session ID with the client.
- These cookies also are present in the server in order to identify the correct client.

2.2 The Information Behind the clicks

Understanding the visitor behavior in a web site

- Visitor browsing behavior: Web logs.
- Visitor preferences: Web pages
- Problems:
 - Web logs contain a lot of irrelevant data.
 - A Web site is a huge collection of heterogeneous, unlabelled, distributed, time variant, semi-structured and high dimensional data.

The dream

"Transform the visitors into customers and retain the existing ones"

- Some solutions:

- Continuous improvement of the web site structure and content.
- Personalization of the relationship between the user and the web site.
- Understanding the user behavior in the web site.

The Server log File

- Usually this repository was used to perform web server tuning and other system administration task.
- But they acquire some unexpected VALUE to the marketing researcher.
- iTHEY CONTAIN IN A IMPLICIT WAY ALL THE CLIENT BEHAVIOR: WE DON'T NEED A SURVEY, WE ALREADY HAVE THE INFORMATION!
- The Log file write a line with this precious information for each request of a client browser.

The Server Log File: an extract

#	IP	Id	Acces	Time	Method/URL/Protocol	Status	Bytes	Referer	Agent
1	165.182.168.101	-	-	16/06/2002:16:24:06	GET p1.htm HTTP/1.1	200	3821	out.htm	Mozilla/4.0 (MSIE 5.5; Wi
2	165.182.168.101	-	-	16/06/2002:16:24:10	GET A.gif HTTP/1.1	200	3766	p1.htm	Mozilla/4.0 (MSIE 5.5; Wi
3	165.182.168.101	-	-	16/06/2002:16:24:57	GET B.gif HTTP/1.1	200	2878	p1.htm	Mozilla/4.0 (MSIE 5.5; Wi
4	204.231.180.195	-	-	16/06/2002:16:32:06	GET p3.htm HTTP/1.1	304	0	-	Mozilla/4.0 (MSIE 6.0; Wi
5	204.231.180.195	-	-	16/06/2002:16:32:20	GET C.gif HTTP/1.1	304	0	-	Mozilla/4.0 (MSIE 6.0; Wi
6	204.231.180.195	-	-	16/06/2002:16:34:10	GET p1.htm HTTP/1.1	200	3821	p3.htm	Mozilla/4.0 (MSIE 6.0; Wi
7	204.231.180.195	-	-	16/06/2002:16:34:31	GET A.gif HTTP/1.1	200	3766	p1.htm	Mozilla/4.0 (MSIE 6.0; Wi
8	204.231.180.195	-	-	16/06/2002:16:34:53	GET B.gif HTTP/1.1	200	2878	p1.htm	Mozilla/4.0 (MSIE 6.0; Wi
9	204.231.180.195	-	-	16/06/2002:16:38:40	GET p2.htm HTTP/1.1	200	2960	p1.htm	Mozilla/4.0 (MSIE 6.0; Wi
10	165.182.168.101	-	-	16/06/2002:16:39:02	GET p1.htm HTTP/1.1	200	3821	out.htm	Mozilla/4.0 (MSIE 5.01; V
11	165.182.168.101	-	-	16/06/2002:16:39:15	GET A.gif HTTP/1.1	200	3766	p1.htm	Mozilla/4.0 (MSIE 5.01; V
12	165.182.168.101	-	-	16/06/2002:16:39:45	GET B.gif HTTP/1.1	200	2878	p1.htm	Mozilla/4.0 (MSIE 5.01; V
13	165.182.168.101	-	-	16/06/2002:16:39:58	GET p2.htm HTTP/1.1	200	2960	p1.htm	Mozilla/4.0 (MSIE 5.01; V
14	165.182.168.101	-	-	16/06/2002:16:42:03	GET p3.htm HTTP/1.1	200	4036	p2.htm	Mozilla/4.0 (MSIE 5.01; V
15	165.182.168.101	-	-	16/06/2002:16:42:07	GET p2.htm HTTP/1.1	200	2960	p1.htm	Mozilla/4.0 (MSIE 5.5; Wi
16	165.182.168.101	-	-	16/06/2002:16:42:09	GET C.gif HTTP/1.1	200	2878	p2.htm	Mozilla/4.0 (MSIE 5.01; V

The Server log File: Structure

- **IP Address:** Client IP.
- **Identity.**
- **Authuser:** Used when SSL is activated.
- **Time:** data and time of the request
- **Request:** The object requested by the browser.
- **Status:** Integer code of the status of the request.
- **Bytes:** The number of bytes returned.
- **Referrer:** text send by the client indicating the original source of a request.
- **User-Agent:** Name and version of the web browser used.

Session reconstruction: the need

- If we want to understand the user behavior in a web site, we need to know his/her real browsing behavior.
- The quality of patterns extracted by using a mining technique depends on the input data.
- Elements like proxies servers, dynamic IP, missing references and the inability of servers to identify different users make difficult to reconstruct a real session.

The Web Logs: some problems

- The web log doesn't store the client id.
- **Proxy and Firewall:** The IP are masked, then the IP number couldn't identify uniquely a client.
- **Web Asynchronism:** Several user access simultaneously the server. Identification method like cookies or session reconstruction techniques are needed.
- **Web Crawlers or Spider Robots:** Google or Yahoo! use a automatic program that retrieve periodically each page. They have to be identified and eliminated.
- <http://www.robotstxt.org/wc/robots.html>
- **Cache:** Sometime the browser use a web cache or a proxy cache that imply that the behaviour was not stored on the logs.

Session Reconstruction Process

- We want to identify the lines in the logs file that belong to a unique valid client.
- This process is called "Sessionization"
- Usual assumption: each session has a maximum time duration.
- Strategies:
 - Proactive strategies: Identify users methods likes cookies. Privacy problem.
 - Reactive strategies: Non invasive privacy.
 - ◆ **Navigational Oriented Heuristics:** pages visited follows the hyperlink structure. If a page doesn't follows this order is a new session.
 - ◆ **Time Oriented Heuristics:** Using usually 30 min for maximum session time.

Session Reconstruction Process

- **Filtering:** Select only the relevant log register, relevant to web pages. Eliminating request for pictures, videos or errors code.
- **Grouping:** IP and Agents can be a good selector of client sessions.
- **Discriminating sessions by time stamps:** Register are selected by time windows of 30 minutes.
- **Identifying irregular session:** robot or spider that could be detected looking at the Agent field.
- **Real session conditioning:** (next slide)

RDBMS could help with the indexing to the efficiency of the calculations.

Sesionization process

IP	Agent	Date		IP	Agent	Date	Sess
165.182.168.101	MSIE 5.01	16-Jun-02 16:39:02	165.182.168.101	MSIE 5.01	16-Jun-02 16:39:02	1
165.182.168.101	MSIE 5.01	16-Jun-02 16:39:58	165.182.168.101	MSIE 5.01	16-Jun-02 16:39:58	1
165.182.168.101	MSIE 5.01		16-Jun-02 16:42:03	165.182.168.101	MSIE 5.01	16-Jun-02 16:42:03	1
165.182.168.101	MSIE 5.5		16-Jun-02 16:24:06	165.182.168.101	MSIE 5.5	16-Jun-02 16:24:06	2
165.182.168.101	MSIE 5.5		16-Jun-02 16:26:05	165.182.168.101	MSIE 5.5	16-Jun-02 16:26:05	2
165.182.168.101	MSIE 5.5	16-Jun-02 16:42:07	165.182.168.101	MSIE 5.5	16-Jun-02 16:42:07	2
165.182.168.101	MSIE 5.5		16-Jun-02 16:58:03	204.231.180.195	MSIE 6.0	16-Jun-02 16:32:06	3
204.231.180.195	MSIE 6.0	16-Jun-02 16:32:06	204.231.180.195	MSIE 6.0	16-Jun-02 16:34:10	3
204.231.180.195	MSIE 6.0		16-Jun-02 16:34:10	204.231.180.195	MSIE 6.0	16-Jun-02 16:38:40	3
204.231.180.195	MSIE 6.0		16-Jun-02 16:38:40	204.231.180.195	MSIE 6.0	16-Jun-02 17:34:20	4
204.231.180.195	MSIE 6.0		16-Jun-02 17:34:20	204.231.180.195	MSIE 6.0	16-Jun-02 17:35:45	4
204.231.180.195	MSIE 6.0	16-Jun-02 17:35:45				

Real session condition

- L set of log register, $r_{ij} \in L$
- $R = \{r_1, \dots, r_n\}$ the set of session, where $r_i = (r_{ij})$
- C1: $r_{ij}.\text{timestamp} > r_{ij-1}.\text{timestamp}$
- C2: $\bigcup \{r_{ij}\} = L$, completeness
- C3: $\exists! i' \neq i, j' / r_{ij} = r_{i'j'}$, each object in L belong to a only one session.

Enforcing these condition we obtain a much more consistent set and also better behaviour descriptions.

Some problems [Berendt01, Cooley99]

- Single IP address/Multiple Server Sessions.
- Multiple IP addresses/Single Server Sessions. For privacy reasons or ISP configuration, it is possible to assign a random IP address to a visitor request.
- Multiple IP address/Single Visitor. A visitor that accesses a web site from different machines, but has the same behavior each time.
- Multiple Agent/Single User. As before, when a visitor uses different machines that may have different agents.

User and Session Identification Issues

- Distinguish among different users to a site
- Reconstruct the activities of the users within the site
- Proxy servers and anonymizers
- Rotating IP addresses connections through ISPs
- Missing references due to caching
- Inability of servers to distinguish among different visits

Some solutions

● Remote Agent

- A remote agent is implemented in Java Applet
- It is loaded into the client only once when the first page is accessed
- The subsequent requests are captured and send back to the server

● Modified Browse

- The source code of the existing browser can be modified to gain user specific data at the client side

● Dynamic page rewriting

- When the user first submit the request, the server returns the requested page rewritten to include a session specific ID
- Each subsequent request will supply this ID to the server

● Heuristics

- use a set of assumptions to identify user sessions and find the missing cache hits in the server log

WUM - Heuristics

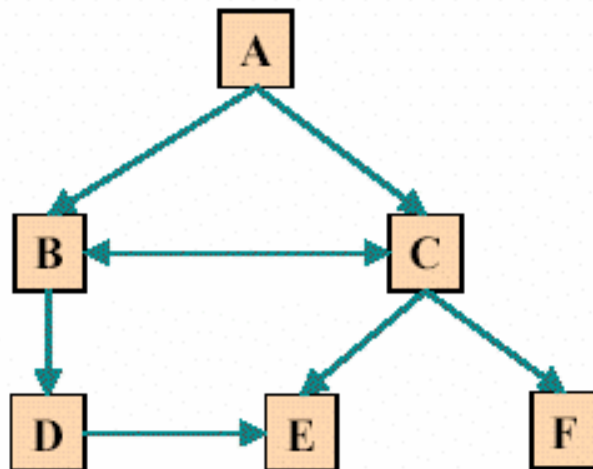
- The session identification heuristics
 - Timeout: if the time between pages requests exceeds a certain limit, it is assumed that the user is starting a new session
 - IP/Agent: Each different agent type for an IP address represents a different sessions
 - Referring page: If the referring page file for a request is not part of an open session, it is assumed that the request is coming from a different session.
 - Same IP-Agent/different sessions (Closest): Assigns the request to the session that is closest to the referring page at the time of the request.
 - Same IP-Agent/different sessions (Recent): In the case where multiple sessions are same distance from a page request, assigns the request to the session with the most recent referrer access in terms of time

WUM - Heuristics (2)

- The path completion heuristics
 - If the referring page file of a session is not part of the previous page file of that session, the user must have accessed a cached page
 - The "back" button method is used to refer a cached page.
 - Assigns a constant view time for each of the cached page file

Based on examples in tutorial
PKDD-2002 by Berendt et. al.

Sessionization- Example



Time	IP	URL	Ref	Agent
0:01	1.2.3.4	A	-	IE5;Win2k
0:09	1.2.3.4	B	A	IE5;Win2k
0:10	2.3.4.5	C	-	IE4;Win98
0:12	2.3.4.5	B	C	IE4;Win98
0:15	2.3.4.5	E	C	IE4;Win98
0:19	1.2.3.4	C	A	IE5;Win2k
0:22	2.3.4.5	D	B	IE4;Win98
0:22	1.2.3.4	A	-	IE4;Win98
0:25	1.2.3.4	E	C	IE5;Win2k
0:25	1.2.3.4	C	A	IE4;Win98
0:33	1.2.3.4	B	C	IE4;Win98
0:58	1.2.3.4	D	B	IE4;Win98
1:10	1.2.3.4	E	D	IE4;Win98
1:15	1.2.3.4	A	-	IE5;Win2k
1:16	1.2.3.4	C	A	IE5;Win2k
1:17	1.2.3.4	F	C	IE4;Win98
1:25	1.2.3.4	F	C	IE5;Win2k
1:30	1.2.3.4	B	A	IE5;Win2k
1:36	1.2.3.4	D	B	IE5;Win2k

Based on examples in tutorial
PKDD-2002 by Berendt et. al.

Sessionization- Example (2)

Sort the users (IP+Agent)

Time	IP	URL	Ref	Agent
0:01	1.2.3.4	A	-	IE5;Win2k
0:09	1.2.3.4	B	A	IE5;Win2k
0:10	2.3.4.5	C	-	IE4;Win98
0:12	2.3.4.5	B	C	IE4;Win98
0:15	2.3.4.5	E	C	IE4;Win98
0:19	1.2.3.4	C	A	IE5;Win2k
0:22	2.3.4.5	D	B	IE4;Win98
0:22	1.2.3.4	A	-	IE4;Win98
0:25	1.2.3.4	E	C	IE5;Win2k
0:25	1.2.3.4	C	A	IE4;Win98
0:33	1.2.3.4	B	C	IE4;Win98
0:58	1.2.3.4	D	B	IE4;Win98
1:10	1.2.3.4	E	D	IE4;Win98
1:15	1.2.3.4	A	-	IE5;Win2k
1:16	1.2.3.4	C	A	IE5;Win2k
1:17	1.2.3.4	F	C	IE4;Win98
1:26	1.2.3.4	F	C	IE5;Win2k
1:30	1.2.3.4	B	A	IE5;Win2k
1:36	1.2.3.4	D	B	IE5;Win2k

0:01	1.2.3.4	A	-	IE5;Win2k
0:09	1.2.3.4	B	A	IE5;Win2k
0:19	1.2.3.4	C	A	IE5;Win2k
0:25	1.2.3.4	E	C	IE5;Win2k
1:15	1.2.3.4	A	-	IE5;Win2k
1:26	1.2.3.4	F	C	IE5;Win2k
1:30	1.2.3.4	B	A	IE5;Win2k
1:36	1.2.3.4	D	B	IE5;Win2k

0:10	2.3.4.5	C	-	IE4;Win98
0:12	2.3.4.5	B	C	IE4;Win98
0:15	2.3.4.5	E	C	IE4;Win98
0:22	2.3.4.5	D	B	IE4;Win98

0:22	1.2.3.4	A	-	IE4;Win98
0:25	1.2.3.4	C	A	IE4;Win98
0:33	1.2.3.4	B	C	IE4;Win98
0:58	1.2.3.4	D	B	IE4;Win98
1:10	1.2.3.4	E	D	IE4;Win98
1:17	1.2.3.4	F	C	IE4;Win98

*Based on examples in tutorial
PKDD-2002 by Berendt et. al.*

Sessionization- Example (3)

Sessionize using heuristics (h1 with 30 min)

0:01	1.2.3.4	A	-	IE5;Win2k
0:09	1.2.3.4	B	A	IE5;Win2k
0:19	1.2.3.4	C	A	IE5;Win2k
0:25	1.2.3.4	E	C	IE5;Win2k
1:15	1.2.3.4	A	-	IE5;Win2k
1:26	1.2.3.4	F	C	IE5;Win2k
1:30	1.2.3.4	B	A	IE5;Win2k
1:36	1.2.3.4	D	B	IE5;Win2k

0:01	1.2.3.4	A	-	IE5;Win2k
0:09	1.2.3.4	B	A	IE5;Win2k
0:19	1.2.3.4	C	A	IE5;Win2k
0:25	1.2.3.4	E	C	IE5;Win2k

1:15	1.2.3.4	A	-	IE5;Win2k
1:26	1.2.3.4	F	C	IE5;Win2k
1:30	1.2.3.4	B	A	IE5;Win2k
1:36	1.2.3.4	D	B	IE5;Win2k

The h1 heuristic (timeout=30 min) will result in the two sessions

*Based on examples in tutorial
PKDD-2002 by Berendt et. al.*

Sessionization- Example (4)

Sessionize using heuristics (with href)

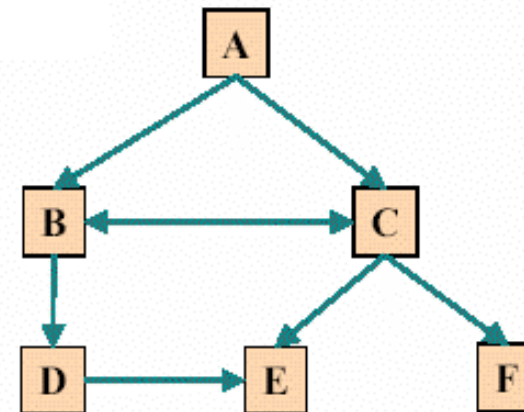
0:22	1.2.3.4	A	-	IE4;Win98
0:25	1.2.3.4	C	A	IE4;Win98
0:33	1.2.3.4	B	C	IE4;Win98
0:58	1.2.3.4	D	B	IE4;Win98
1:10	1.2.3.4	E	D	IE4;Win98
1:17	1.2.3.4	F	C	IE4;Win98

By using the reffer-based heuristics, we have only a single session

Sessionization- Example (5)

Path completion

0:22	1.2.3.4	A	-	IE4;Win98
0:25	1.2.3.4	C	A	IE4;Win98
0:33	1.2.3.4	B	C	IE4;Win98
0:58	1.2.3.4	D	B	IE4;Win98
1:10	1.2.3.4	E	D	IE4;Win98
1:17	1.2.3.4	F	C	IE4;Win98



$A \Rightarrow C$, $C \Rightarrow B$, $B \Rightarrow D$, $D \Rightarrow E$, $C \Rightarrow F$

Need to look for the shortest backwards path from E to C based on the site topology. Note, however, that the elements of the path need to have occurred in the user trail previously.

$E \Rightarrow D$, $D \Rightarrow B$, $B \Rightarrow C$

Final comments

- What happen if the web page content is changed during the study period?
- $A \rightarrow B, B \rightarrow D$ but there are two versions of D.
- If we want study the user behavior, it is necessary to consider to maintain a change register.
- Proposal solution LOGML [Punin WEBKDD'01]

Mechanisms for session identification [Berendt 2002]

Method	Description	Private Concerns	Advantages	Disadvantages
IP Address + Agent	Assume each unique IP address/Agent pair is a unique user	Low	Always available. No Additional technology required.	Not guaranteed to be unique. Defeated by rotating Ips.
Embedded Sessions Ids	Use dynamically generated pages to associate ID with every hyperlink	Low to Medium	Always available. Independent of IP address	Cannot capture repeat visitors. Additional overhead for dynamic pages
Registration	User explicitly logs into the site	Medium	Can track individuals not just browsers	Many users won't register. Not available before registration
Cookie	Save ID on the client machine	Medium to High	Can track repeat visit from same browser	Can be turned off by users
Software Agents	Program loaded into browser and sends back usage data	High	Accurate usage data for a single site	Likely to be rejected by users

2.3 The Information Contained in a Web Page

Web page content

- Another dimension of the business is the content presented by the web pages to the client.
- Web content can be represented by an object, that could be of different types: Multimedia or html Text.
- Analysis of web page content usually are made on the text content.
- Example: ¿Which word or concepts are more important to the user?

Web text content

- From different web page content, special attention receive the free text.
- For the moment, a searching is performed by using key words.
- It is necessary to represent the text information in a feature vector, before to apply a mining process.
- The representation must consider that the words in the web page don't have the same importance.

Vector Space Model

- In Information Retrieval a set of document (X) are represented by a matrix $M=(m_{ij})=M(X)$
- This matrix represent the **representativity** of a set of words for a document.
- The index i represent the different “**relevant**” word that appears on the all the document in X .
- The index j represent the different **document** that appears on X .
- The entry m_{ij} of the matrix M represent the “**importance**” or weight of this word i on the document j .
- Then a Document is represented by a column vector of this numeric matrix.

Web page: vectorial representation

- Its vectorial representation would be a matrix of $R \times Q$.
- Q is the number of pages in the web site and R is the number of different words in web page.
- Each entry of the matrix correspond to the number of word found at the respective web page.

	Word	1	2	...	Q
1	advise	1	0	...	1
2	business	0	1	...	0
.
.
.
.

Different measure of representativeness

- Q : number of documents
- Inverse Document Frequency (IDF):

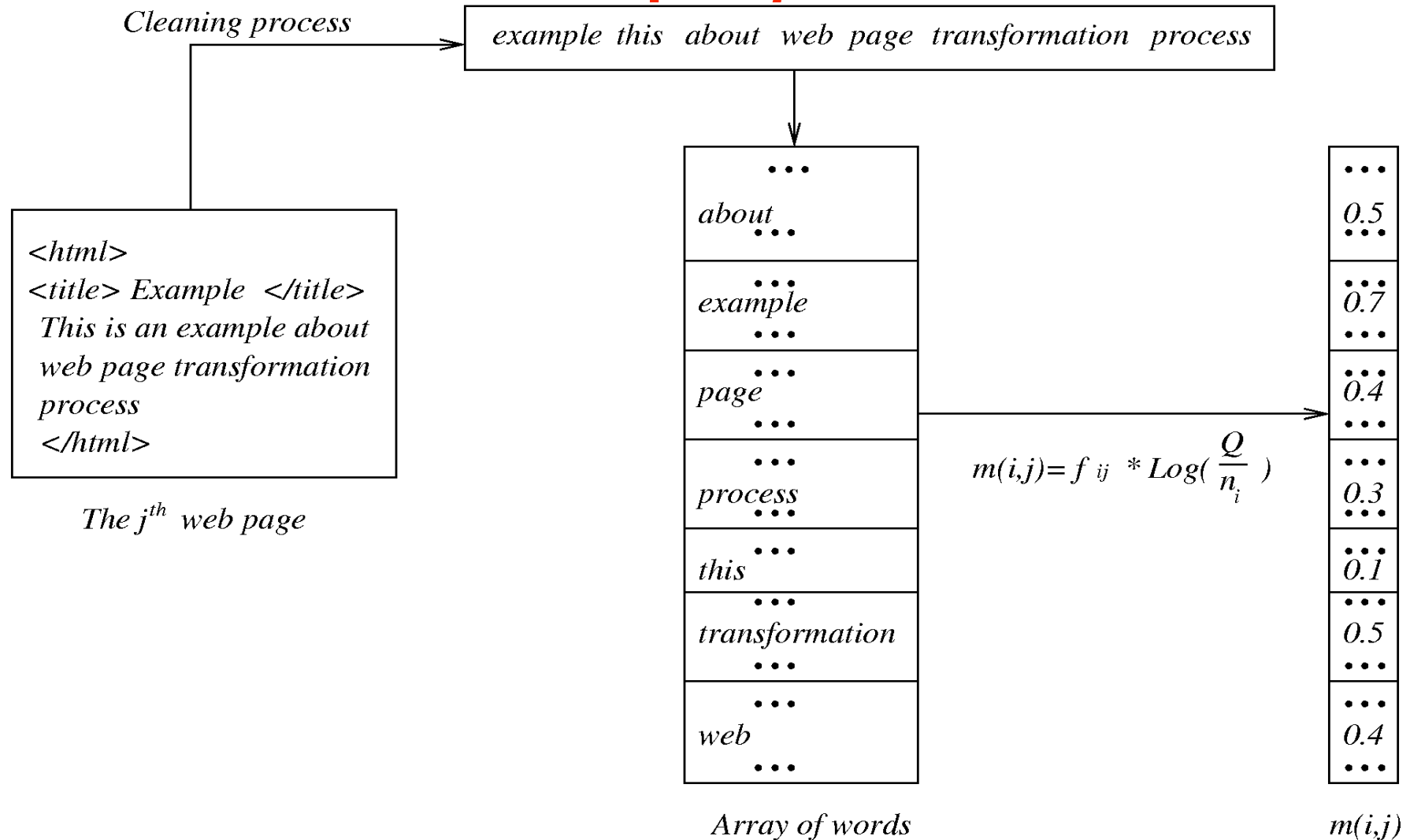
$$m_{ij} = \log(Q/n_i)$$

- Term Frequency Inverse Document

Frequency (TFIDF): $m_{ij} = f_{ij} * \log(Q/n_i)$,

where f_{ij} is the number of occurrences of the word i in the document j .

Vector representation of a web page



Vector allows us to: Compare web pages (cosine)

TFIDF: $M = (m_{ij}) = f_{ij} * \log(\frac{Q}{n_i})$

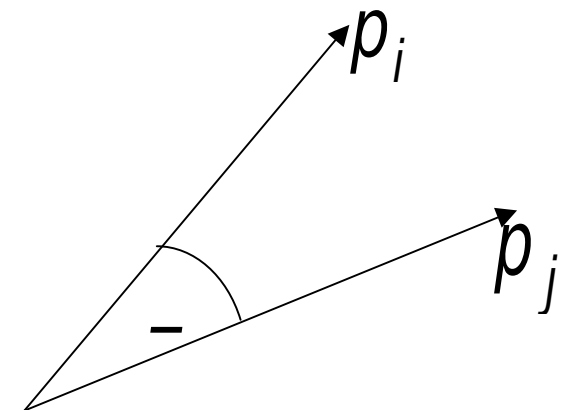
Representative
web page
vector i y j .

$$p_i = (m_{1i}, \dots, m_{Ri})$$

$$p_j = (m_{1j}, \dots, m_{Rj})$$

$$dp(p_i, p_j) = \cos \theta = \frac{\sum_{k=1}^R m_{ki} m_{kj}}{\sqrt{\sum_{k=1}^R (m_{ki})^2} \sqrt{\sum_{k=1}^R (m_{kj})^2}}$$

Similarity function



Vector Space Model: The Process (to see in details in further chapters)

- **Cleanning:** Text extraction from html.
- **Tokenization:** Use of syntactic rules, filtering stop words.
- **Stemming:** (Porter Algorithm <http://tartarus.org/martin/PorterStemmer/>) translating the word to its semantic root. (ex: writing -> write)
- **Vectorization:** The calculation of the matrix M with the appropriate weight function.

Pros & Cons of Vector Model

● Advantages:

- term-weighting improves quality of the answer set
- partial matching allows retrieval of docs that approximate the query conditions
- cosine ranking formula sorts documents according to degree of similarity to the query

● Disadvantages:

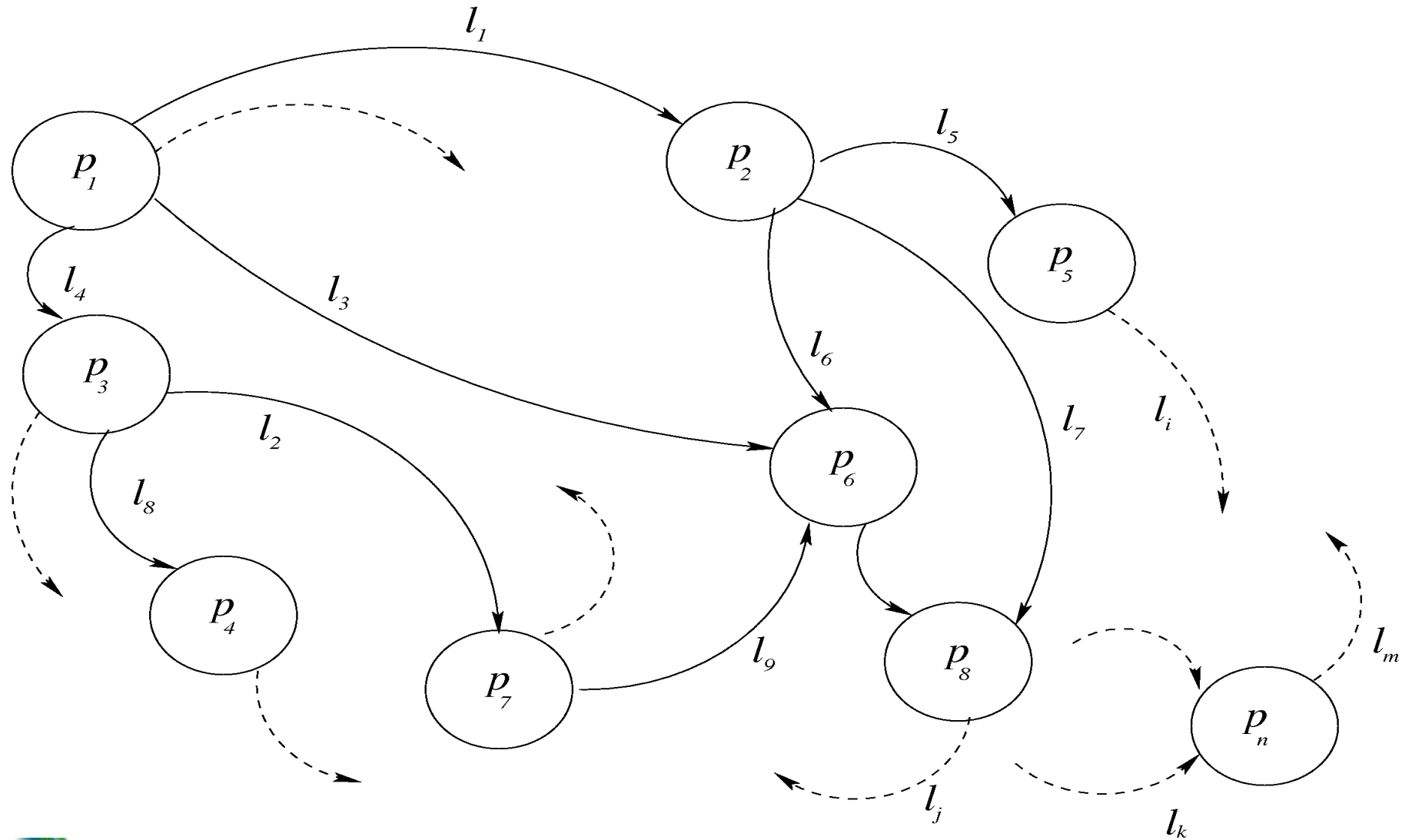
- assumes independence of index terms; not clear if this is a good or bad assumption

Web hyperlinks structure

- Why a web page point to another one?
- Link analysis: use link structure to determine credibility.
- If a web page is pointed by other ones, maybe it is because the page contains relevant information.
- We can understand the formation of a web community.
- We can improve our web site.

Web hyperlinks structure

Web page links: Structure



The hyperlink: The data

- Present on HTML tag property "href" and others like the event OnClick
- ` Buy `
- Obtaining the data:
 - Crawl a web site: obtain recursively all the pages (text) that follow by hyperlink.
 - Store the relation $q \rightarrow p$

Processing the Hyperlinks structure

- Identifying which pages contain more relevant information than others [Kleinberg99]:
 - Authorities. A natural information repository for the community. x will be a vector of weight for authorities.
 - Hub. These concentrate links to authorities web pages, for instance, "my favorite sites". y will be a vector of weight for hubs.

$$x_p = \sum_{q: \exists q \rightarrow p} y_q$$

$$y_p = \sum_{q: \exists p \rightarrow q} x_q$$

Later we will develop further in the HITS algorithm.

Processing the Hyperlinks structure

- Hits & Page Rank algorithm generate a “ranking” search result for search engine.
- Google, Yahoo!
- AdWord mechanism: give to Google a large amount of utility, based on this kind of structure mining.

Web Data: Summary

- Our FOCUS is mining an e-business datawarehouse.
- In order to analyze the data we need first to know the process that generate it.
- And the sources of them:
- Web Server Logs Files: behavior of the client.
- Web Content: The text that the client see.
- Web Link: The way that the client could browse the content.