Web Usage Mining

from Bing Liu.

"Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data", Springer

Chapter written by Bamshad Mobasher

Many slides are from a tutorial given by

B. Berendt, B. Mobasher, M. Spiliopoulou

Introduction

- Web usage mining
  - automatic discovery of patterns in clickstreams and associated data, collected or generated as a result of user interactions with one or more Web sites.
- Goal: analyze the behavioral patterns and profiles of users interacting with a Web site.
- The discovered patterns are usually represented as
  - collections of pages, objects, or resources that are frequently accessed by groups of users with common interests.

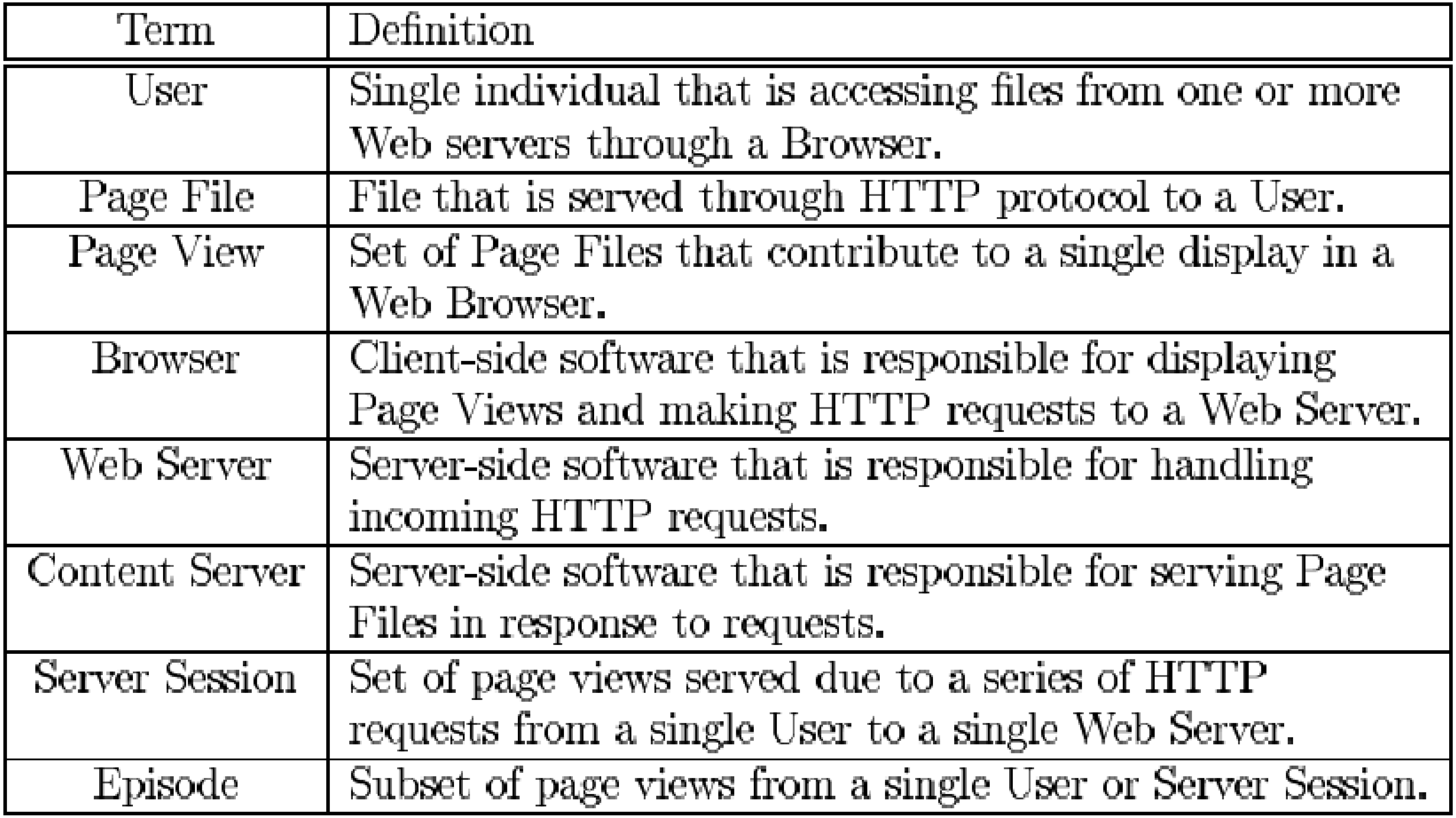
Introduction

- Data in Web Usage Mining:
  - Web server logs, integrated with site contents
  - Data about visitors, gathered from external channels
  - Further application data
- Not all these data are always available.
- When they are, they must be integrated.
- A large part of Web usage mining is about processing usage/ clickstream data.
  - After that various data mining algorithm can be applied.

Web server logs

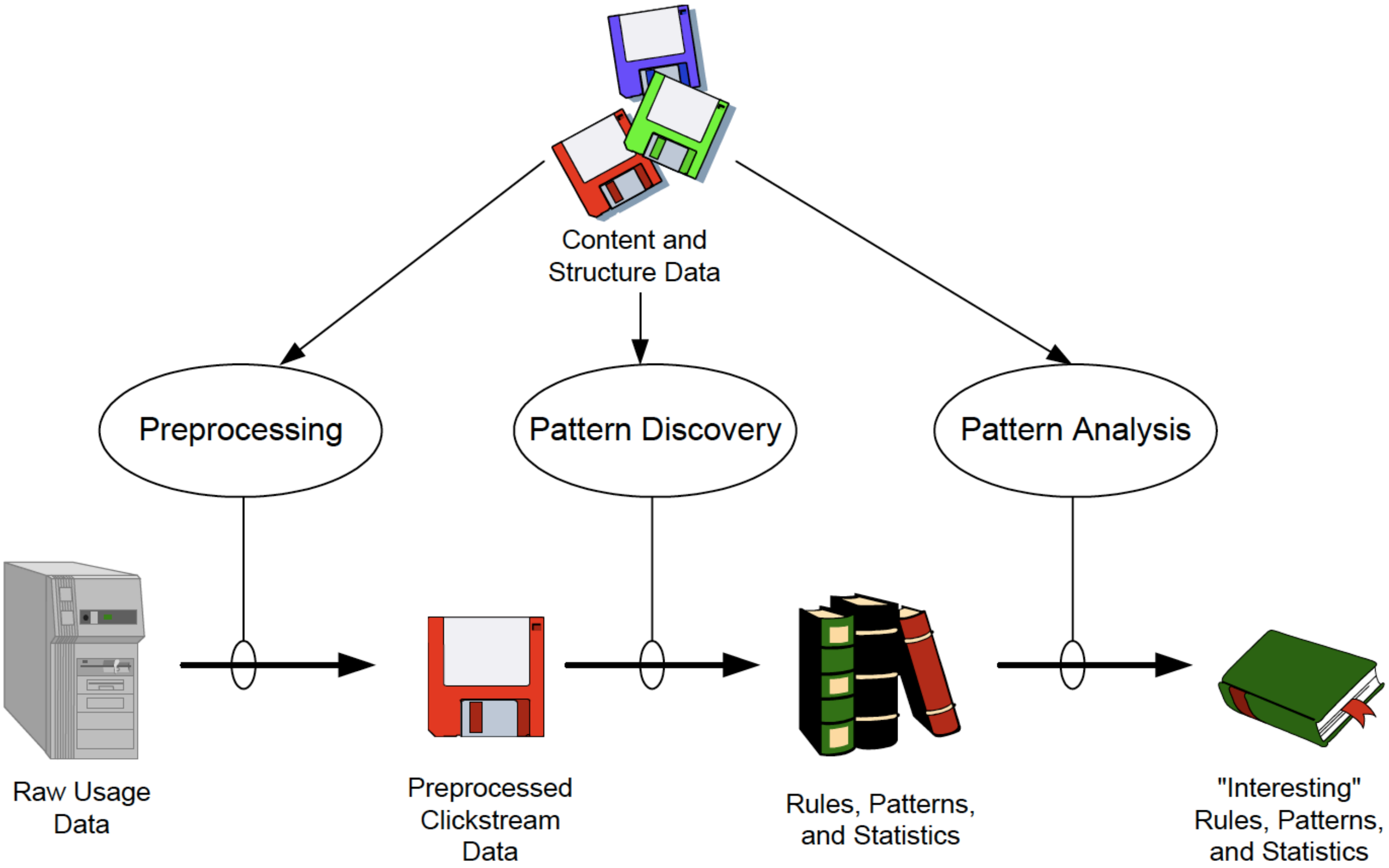
1	2006-02-01 00:08:43 1.2.3.4 - GET /classes/cs589/papers.html - 200 9221 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1;+.NET+CLR+2.0.50727) http://dataminingresources.blogspot.com/
2	2006-02-01 00:08:46 1.2.3.4 - GET /classes/cs589/papers/cms-tai.pdf - 200 4096 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1;+.NET+CLR+2.0.50727) http://maya.cs.depaul.edu/~classes/cs589/papers.html
3	2006-02-01 08:01:28 2.3.4.5 - GET /classes/ds575/papers/hyperlink.pdf - 200 318814 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1) http://www.google.com/search?hl=en&lr=&q=hyperlink+analysis+for+the+web+survey
4	2006-02-02 19:34:45 3.4.5.6 - GET /classes/cs480/announce.html - 200 3794 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1) http://maya.cs.depaul.edu/~classes/cs480/
5	2006-02-02 19:34:45 3.4.5.6 - GET /classes/cs480/styles2.css - 200 1636 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1) http://maya.cs.depaul.edu/~classes/cs480/announce.html
6	2006-02-02 19:34:45 3.4.5.6 - GET /classes/cs480/header.gif - 200 6027 HTTP/1.1 maya.cs.depaul.edu Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1) http://maya.cs.depaul.edu/~classes/cs480/announce.html

Terminology and level of abstractions

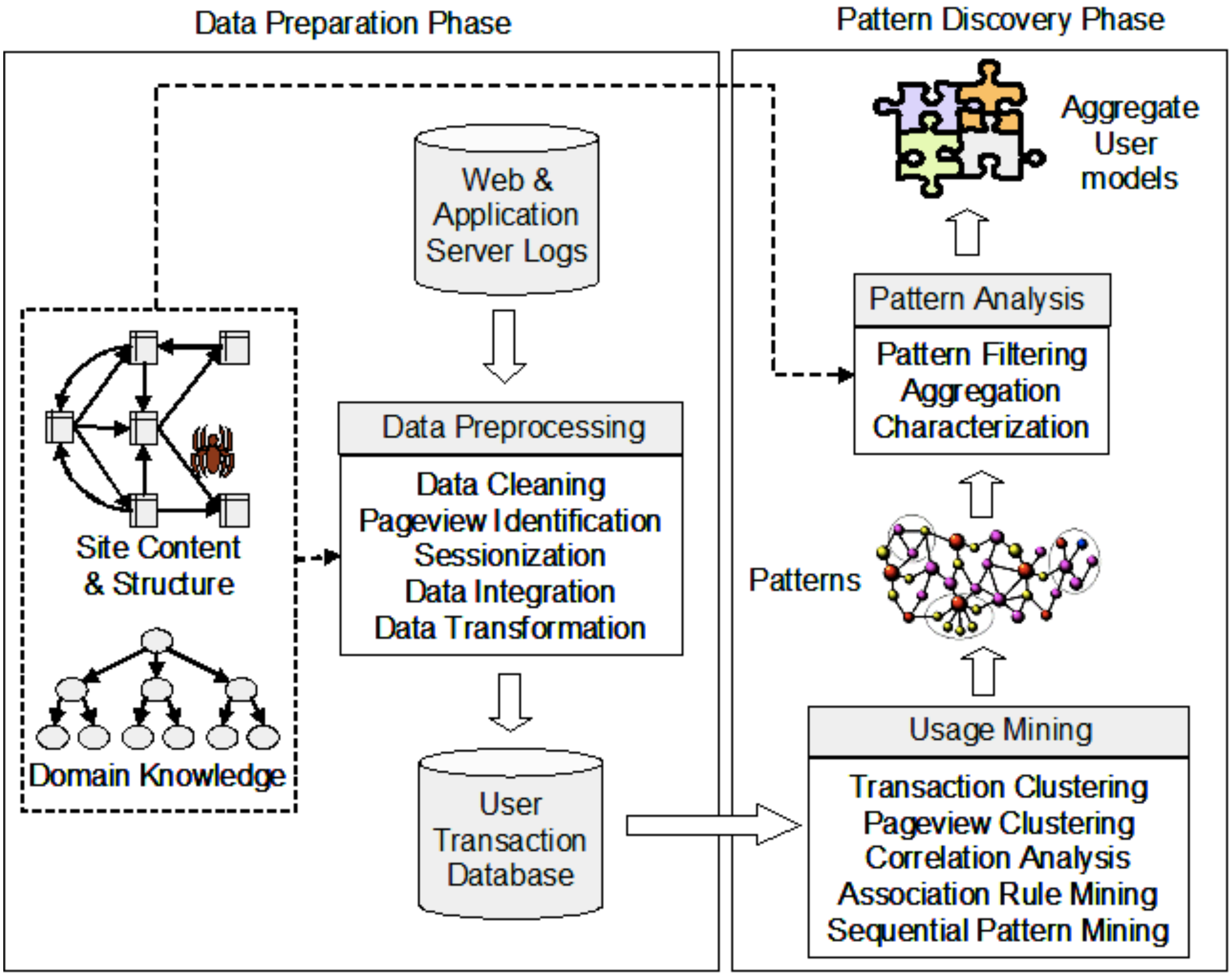


Term	Definition
User	Single individual that is accessing files from one or more Web servers through a Browser.
Page File	File that is served through HTTP protocol to a User.
Page View	Set of Page Files that contribute to a single display in a Web Browser.
Browser	Client-side software that is responsible for displaying Page Views and making HTTP requests to a Web Server.
Web Server	Server-side software that is responsible for handling incoming HTTP requests.
Content Server	Server-side software that is responsible for serving Page Files in response to requests.
Server Session	Set of page views served due to a series of HTTP requests from a single User to a single Web Server.
Episode	Subset of page views from a single User or Server Session.

Web usage mining (simplified view)


Content and Structure Data
Preprocessing
Pattern Discovery
Pattern Analysis
Raw Usage Data
Preprocessed Clickstream Data
Rules, Patterns, and Statistics
"Interesting" Rules, Patterns, and Statistics

Web usage mining process



Data Preparation Phase

Pattern Discovery Phase

Web & Application Server Logs

Site Content & Structure

Domain Knowledge

Data Preprocessing

Data Cleaning
Pageview Identification
Sessionization
Data Integration
Data Transformation

User Transaction Database

Usage Mining

Transaction Clustering
Pageview Clustering
Correlation Analysis
Association Rule Mining
Sequential Pattern Mining

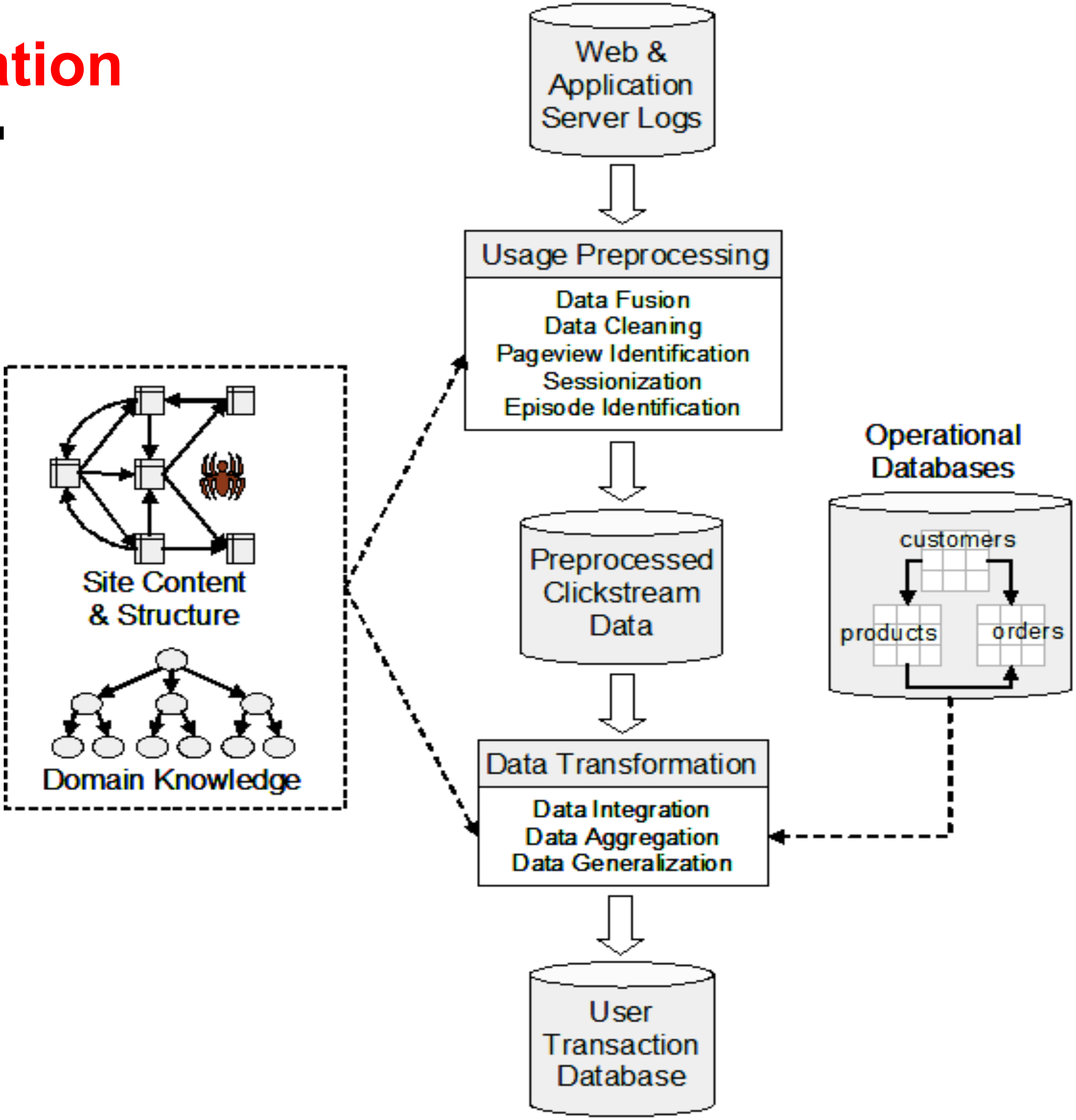
Patterns

Pattern Analysis

Pattern Filtering
Aggregation
Characterization

Aggregate User models

Data preparation



Usage preprocessing

- Data fusion
  - synchronize data from multiple server logs
- Data cleaning
  - remove irrelevant references and fields in server logs
  - remove references due to spider/robot navigation
  - remove erroneous references
  - add missing references due to caching (done after sessionization)
- Sessionization
  - user identification
  - pageview identification
    - a pageview is a set of page files and associated objects that contribute to a single display in a Web Browser
  - thus split the log and identify sessions

Data transformation

- Data integration
  - integrate e-commerce and application server data
  - integrate demographic / registration data

- Identifying User Transactions
  - i.e., sets or sequences of pageviews possibly with associated weights

Identifying sessions (Sessionization)

- **The quality of the patterns discovered in KDD depends on the quality of the data on which mining is applied.**
- **In Web usage analysis, these data are the sessions of the site visitors**
  - ***the activities performed by a user from the moment she enters the site until the moment she leaves it.***
- **Difficult to obtain reliable usage data due to**
  - **proxy servers and anonymizers,**
  - **dynamic IP addresses,**
  - **missing references due to caching, and**
  - **the inability of servers to distinguish among different visits.**

Sessionization strategies

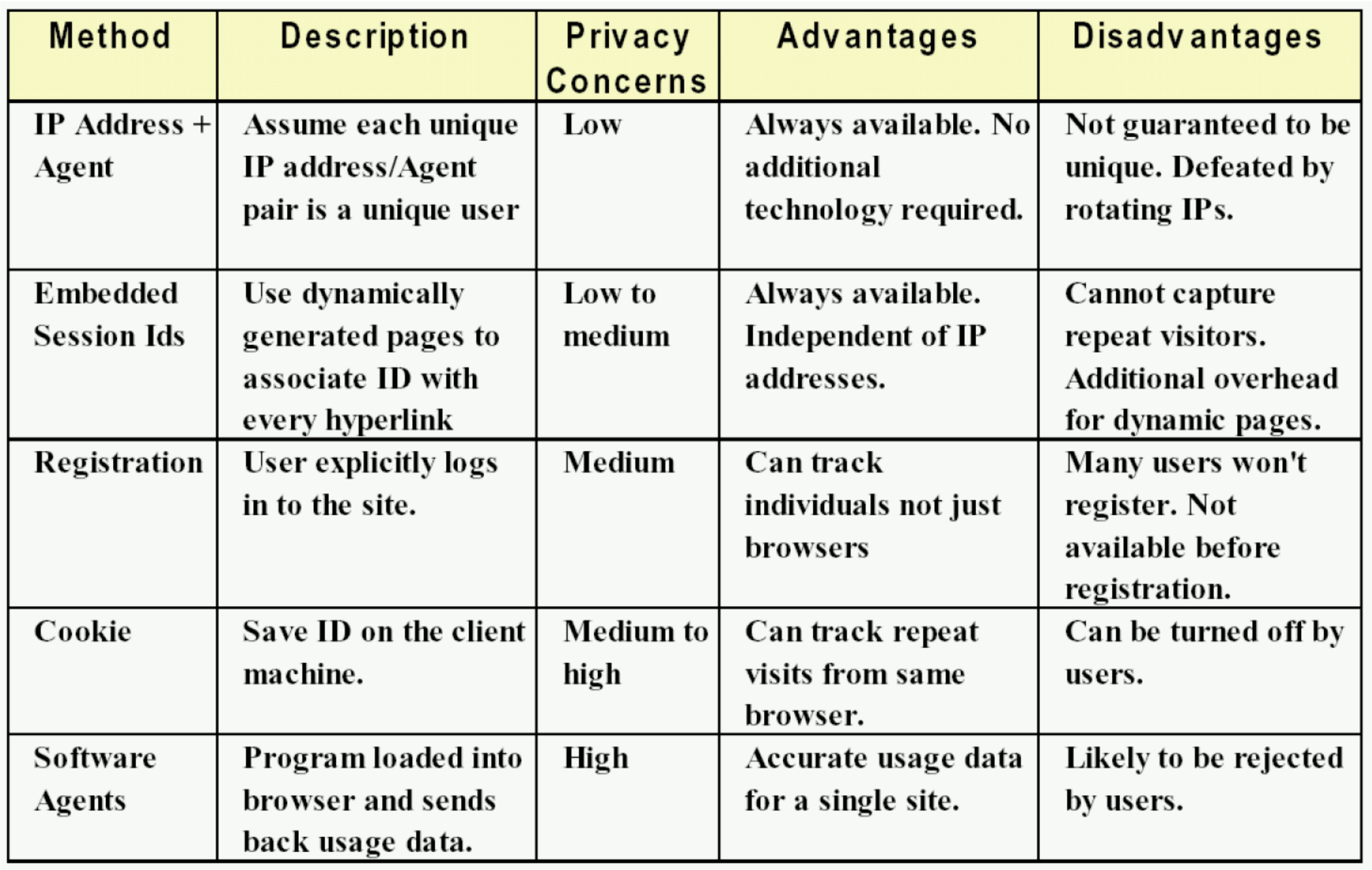
- Session reconstruction =

  correct mapping of activities to different individuals +

  correct separation of activities belonging to different visits of the same individual

While users navigate the site: identify ...		In the analysis of log files: identify …		Resulting partitioning of the log file
users by	sessions by	users by	sessions by	
—	—	IP & Agent	sessionization heuristics	constructed sessions (“**u-ipa**”)
cookies	—	—	sessionization heuristics	constructed sessions (“**cookies**”)
cookies	embedded session IDs	—	—	real sessions

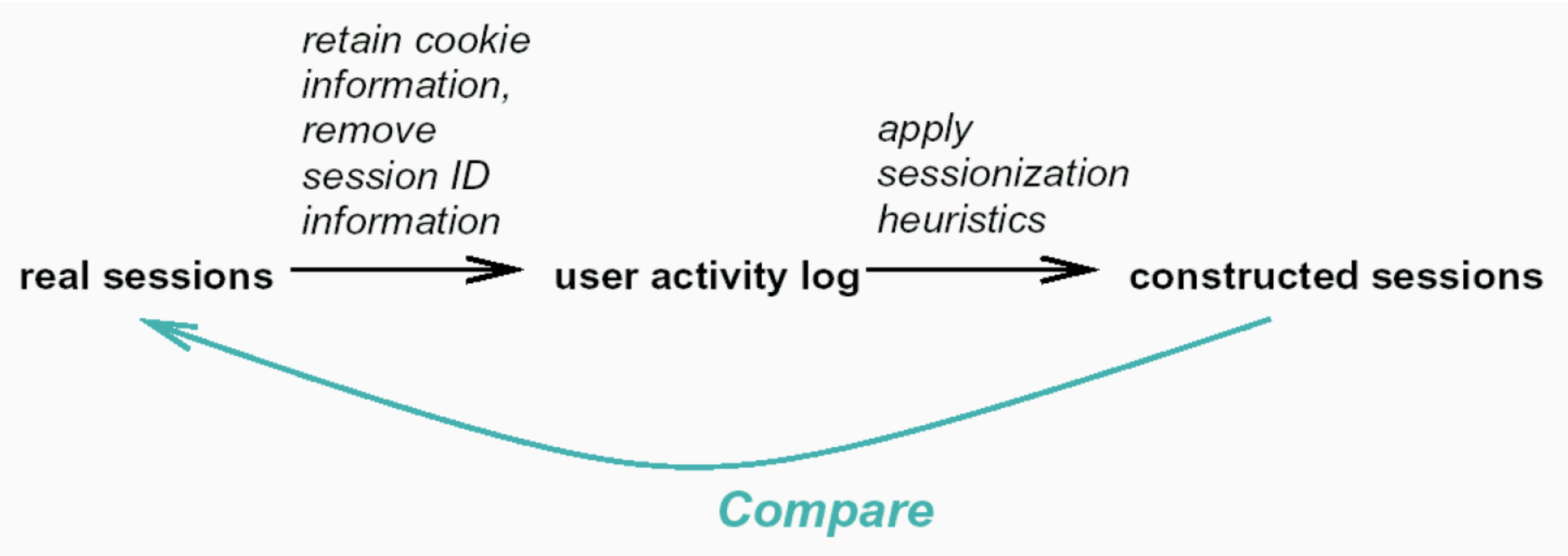
User identification



Method	Description	Privacy 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 Session Ids	Use dynamically generated pages to associate ID with every hyperlink	Low to medium	Always available. Independent of IP addresses.	Cannot capture repeat visitors. Additional overhead for dynamic pages.
Registration	User explicitly logs in to 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 visits 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.

Session uncertainty: evaluate Real vs. Re-constructed sessions

identify ...		identify ...		resulting log
users by	sessions by	users by	sessions by	partioning
cookies	—	—	sessionization heuristics	constructed sessions
cookies	session IDs	—	—	real sessions



real sessions

retain cookie information, remove session ID information

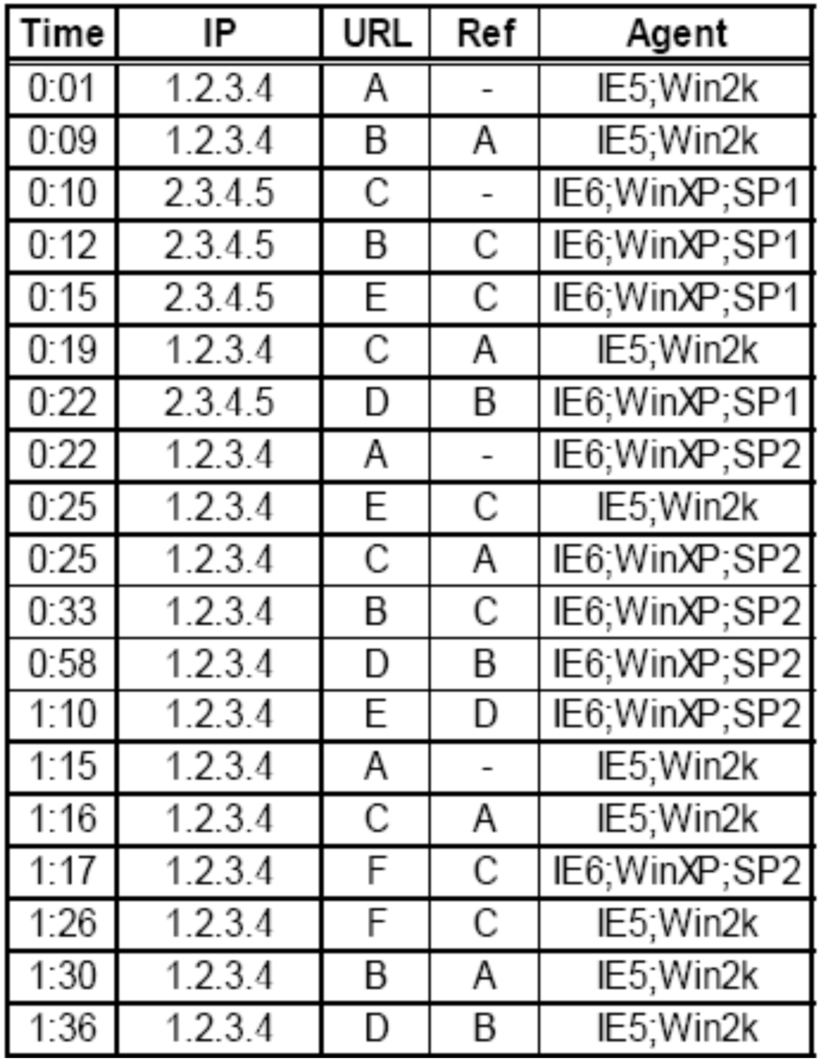
user activity log

apply sessionization heuristics

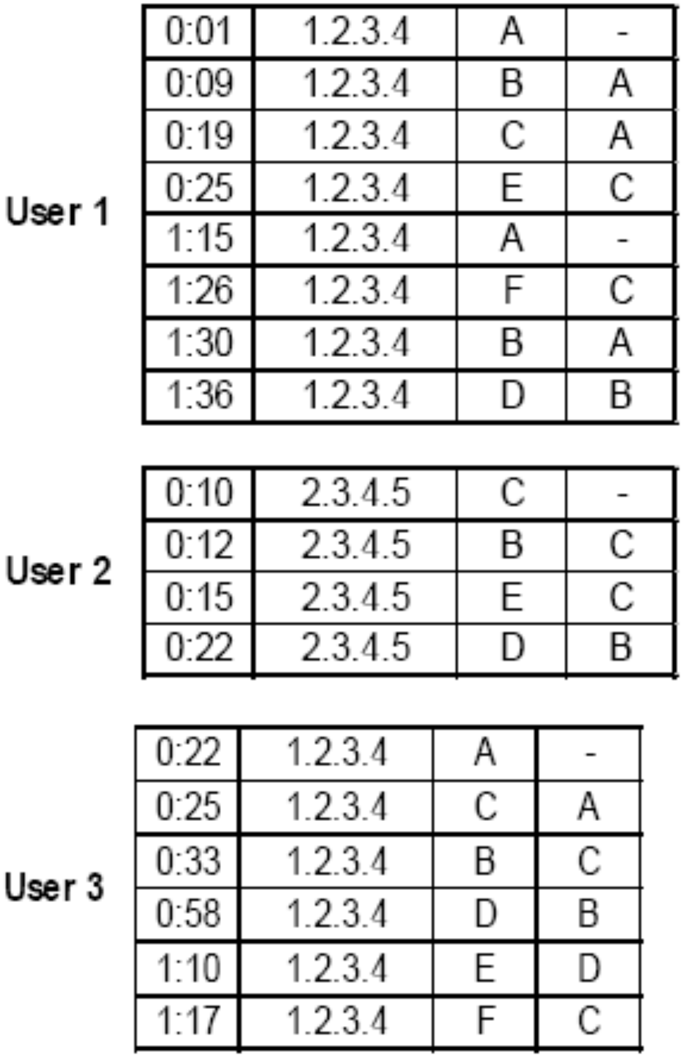
constructed sessions

Compare

User identification: an 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	-	IE6;WinXP;SP1
0:12	2.3.4.5	B	C	IE6;WinXP;SP1
0:15	2.3.4.5	E	C	IE6;WinXP;SP1
0:19	1.2.3.4	C	A	IE5;Win2k
0:22	2.3.4.5	D	B	IE6;WinXP;SP1
0:22	1.2.3.4	A	-	IE6;WinXP;SP2
0:25	1.2.3.4	E	C	IE5;Win2k
0:25	1.2.3.4	C	A	IE6;WinXP;SP2
0:33	1.2.3.4	B	C	IE6;WinXP;SP2
0:58	1.2.3.4	D	B	IE6;WinXP;SP2
1:10	1.2.3.4	E	D	IE6;WinXP;SP2
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	IE6;WinXP;SP2
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



User 1

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

User 2

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

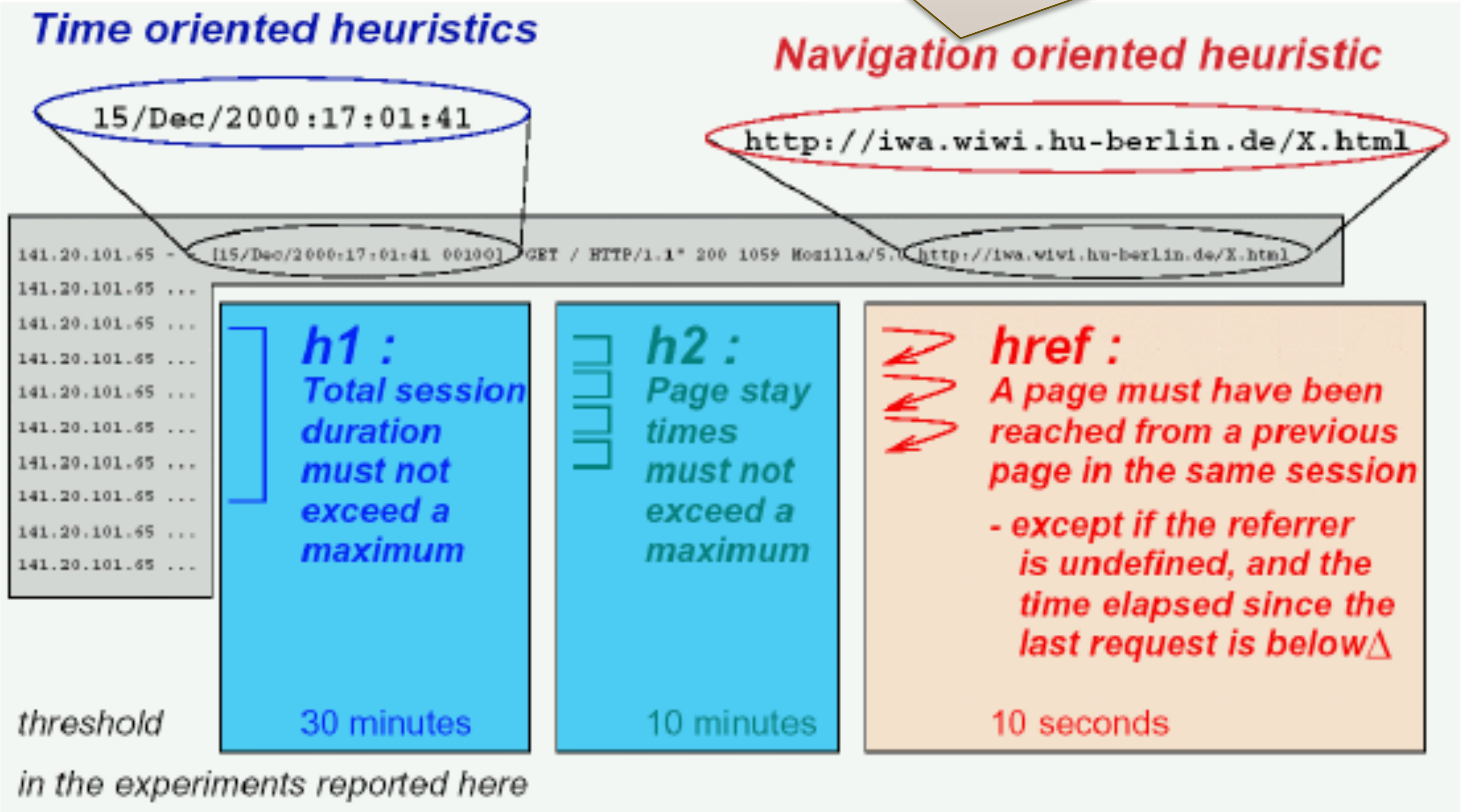
User 3

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

Combination of IP address and Agent fields in Web logs

Sessionization heuristics

**Also called structure-oriented:
use either the static structure of the site,
or the implicit linkage structure inferred
from the referrer fields**



Time oriented heuristics

Navigation oriented heuristic

15/Dec/2000:17:01:41

http://iwa.wiwi.hu-berlin.de/X.html

```
141.20.101.65 - [15/Dec/2000:17:01:41 00100] "GET / HTTP/1.1" 200 1059 Mozilla/5.0 http://iwa.wiwi.hu-berlin.de/X.html
141.20.101.65 ...
141.20.101.65 ...
141.20.101.65 ...
141.20.101.65 ...
141.20.101.65 ...
141.20.101.65 ...
141.20.101.65 ...
141.20.101.65 ...
141.20.101.65 ...
```

h1 : *Total session duration must not exceed a maximum*

h2 : *Page stay times must not exceed a maximum*

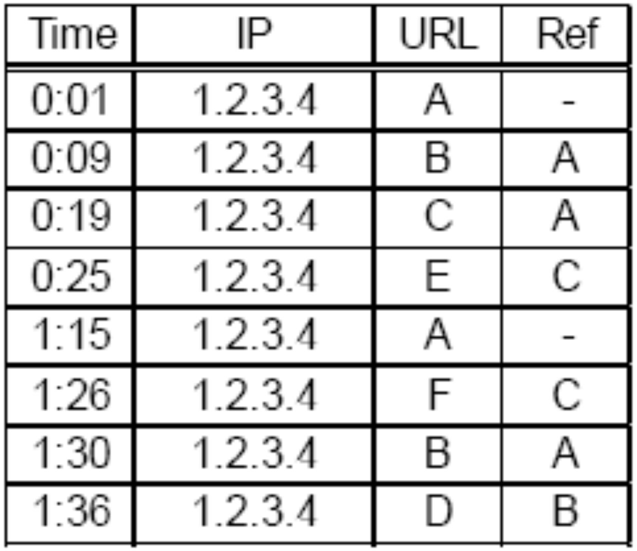
href : *A page must have been reached from a previous page in the same session*
- except if the referrer is undefined, and the time elapsed since the last request is below Δ

threshold	30 minutes	10 minutes	10 seconds

in the experiments reported here

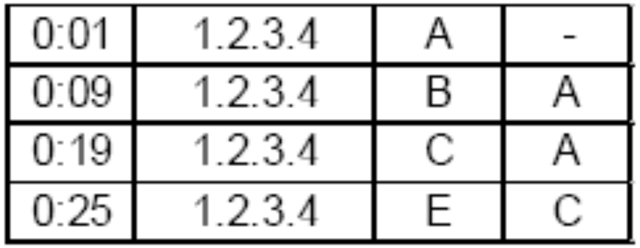
Sessionization example: time-oriented heuristic

User 1



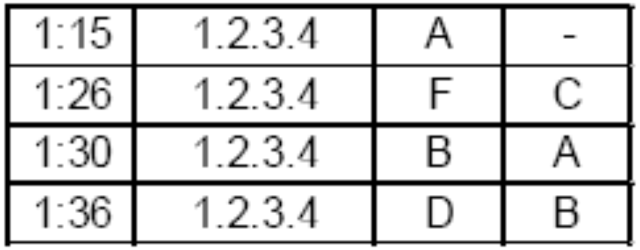
Time	IP	URL	Ref
0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C
1:15	1.2.3.4	A	-
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

Session 1



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

Session 2



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

Pageview identification

- **Pageview identification**
  - **Depends on the intra-page structure of sites**
  - **Identify the collection of Web files/objects/resources representing a specific "user event" corresponding to a click-through (e.g. viewing a product page, adding a product to a shopping cart)**
  - **In some cases it may be nice to consider pageviews at a higher level of aggregation**
    - **e.g. they may correspond to many user event related to the same concept category, like the purchase of a product on an online e-commerce site**

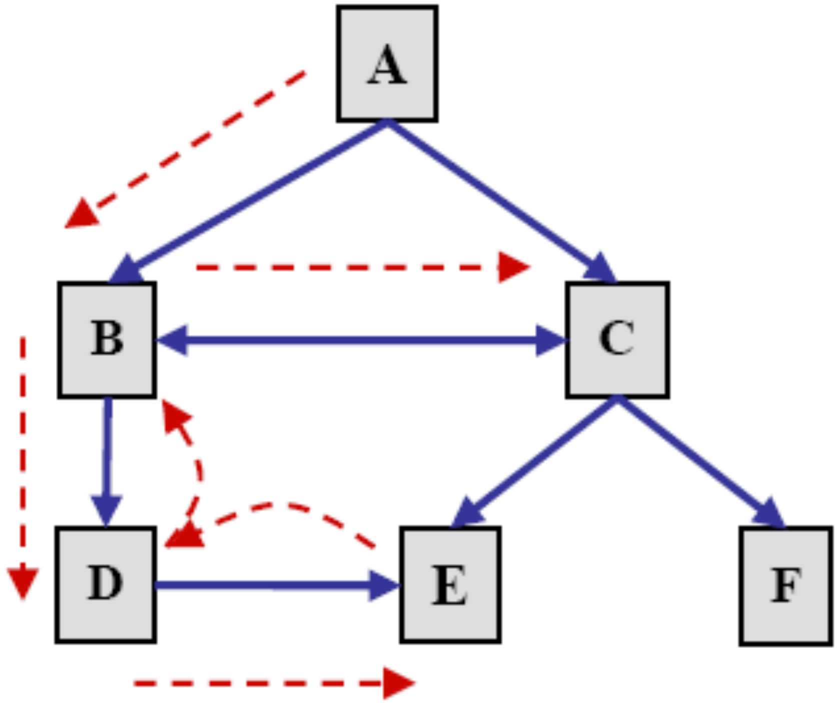
Path completion

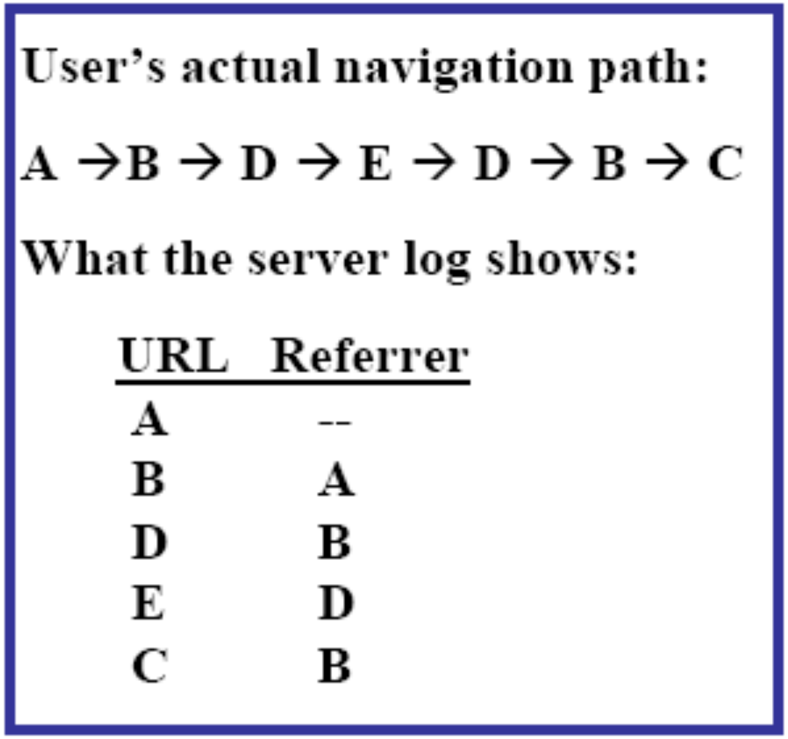
- Client- or proxy-side caching can often result in missing access references to those pages or objects that have been cached.
- For instance,
  - if a user goes back to a page A during the same session, the second access to A will likely result in viewing the previously downloaded version of A that was cached on the client-side, and therefore, no request is made to the server.
  - This results in the second reference to A not being recorded on the server logs.

Path completion

- **Path completion:**
  - **How to infer missing user references due to caching.**
- **Effective path completion requires extensive knowledge of the link structure within the site**
- **Referrer information in server logs can also be used in disambiguating the inferred paths.**
- **Problem gets much more complicated in frame-based sites.**

Missing references due to caching





User's actual navigation path:

A →B → D → E → D → B → C

What the server log shows:

URL	Referrer
A	--
B	A
D	B
E	D
C	B

- Reconstruction by using the knowledge about the site structure
  - also inferred from the the referrer fields
- Many paths are possible
  - usually the selected path is the one requiring the fewest number of “back” reference

Data modeling for Web Usage Mining

- **Data preprocessing produces**
  - **a set of pageviews:** $P=\{p_1, \ldots, p_n\}$
  - **a set of user transactions:** $T=\{t_1, \ldots, t_m\}$

    where each transaction t_i contains a subset of P
  - **Each transaction:**

$$t = \left\langle (p_1^t, w(p_1^t)), (p_2^t, w(p_2^t)), \ldots, (p_l^t, w(p_l^t)) \right\rangle$$

is a *l*-length ordered sequence of pageviews, where each *w* corresponds to a weight, e.g. the significance of the pageview

  - **In *collaborative filtering* these weights correspond to explicit user ratings**
  - **In *Web collected transactions*, the duration of the page visit in the session**

Data modeling for Web Usage Mining (cont.)

- In many mining tasks, the sequential ordering of the transactions is not important (e.g.: clustering, association rule extractions)

- In this case a transaction can be represented as an *n-length vector*:

$$t = (w_1^t, w_2^t, \ldots, w_n^t)$$

where the weight is 0 if the corresponding page is not present in t, otherwise correspond to the significance of the page in the t

	page A	page B	page C	page D	page E
user 0	15	4	1	0	0
user 1	2	0	25	0	0
user 2	200	1	0	0	3
user 3	56	0	0	4	4
user 4	0	0	23	50	0
user 5	0	0	5	3	0

$m \times n$ user-pageviews matrix (or transaction matrix)

Data modeling for Web Usage Mining (cont.)

- Given a user-pageview matrix, a number of unsupervised mining techniques can be exploited

	page A	page B	page C	page D	page E
user 0	15	4	1	0	0
user 1	2	0	25	0	0
user 2	200	1	0	0	3
user 3	56	0	0	4	4
user 4	0	0	23	50	0
user 5	0	0	5	3	0

$m \times n$ user-pageviews matrix (or transaction matrix)

- Clustering of transactions/sessions to determine important visitor segments
- Clustering of pageviews (items) expressed in terms of user judgments, to discover important relationships between pageviews (items)
- Sequential (timestamps must be maintained) and non sequential association rules, to discover important relationships between pageviews (items)

Data modeling for Web Usage Mining (cont.)

- Automatic integration of content information
  - textual features from the Web contents represent the underlying semantics of the pages
  - aiming to transform a user-pageviews matrix into a content-enhanced transaction matrix

	food	news	car	house	party	sky
page A	0	1	1	0	0	0
page B	1	0	0	1	0	0
page C	1	1	0	0	0	0
page D	0	0	1	0	0	1
page E	0	0	0	1	1	0

$n \times r$
pageviews-terms
matrix

Data modeling for Web Usage Mining (cont.)

P=

	food	news	car	house	party	sky
page A	0	1	1	0	0	0
page B	1	0	0	1	0	0
page C	1	1	0	0	0	0
page D	0	0	1	0	0	1
page E	0	0	0	1	1	0

$n \times r$ **pageviews-terms** matrix

U=

	page A	page B	page C	page D	page E
user 0	1	1	0	0	0
user 1	0	0	1	0	0
user 2	1	0	0	0	1
user 3	1	0	0	1	1
user 4	0	0	1	1	0
user 5	0	0	1	0	0

$m \times n$ **user-pageviews matrix** (or transaction matrix)

U×**P** =

	food	news	car	house	party	sky
user 0	1	1	1	1	0	0
user 1	1	1	0	0	0	0
user 2	0	1	1	1	1	0
user 3	0	1	2	1	1	1
user 4	1	1	1	0	0	1
user 5	1	1	0	0	0	0

$m \times r$ **content-enhanced transaction matrix**

Data modeling for Web Usage Mining (cont.)

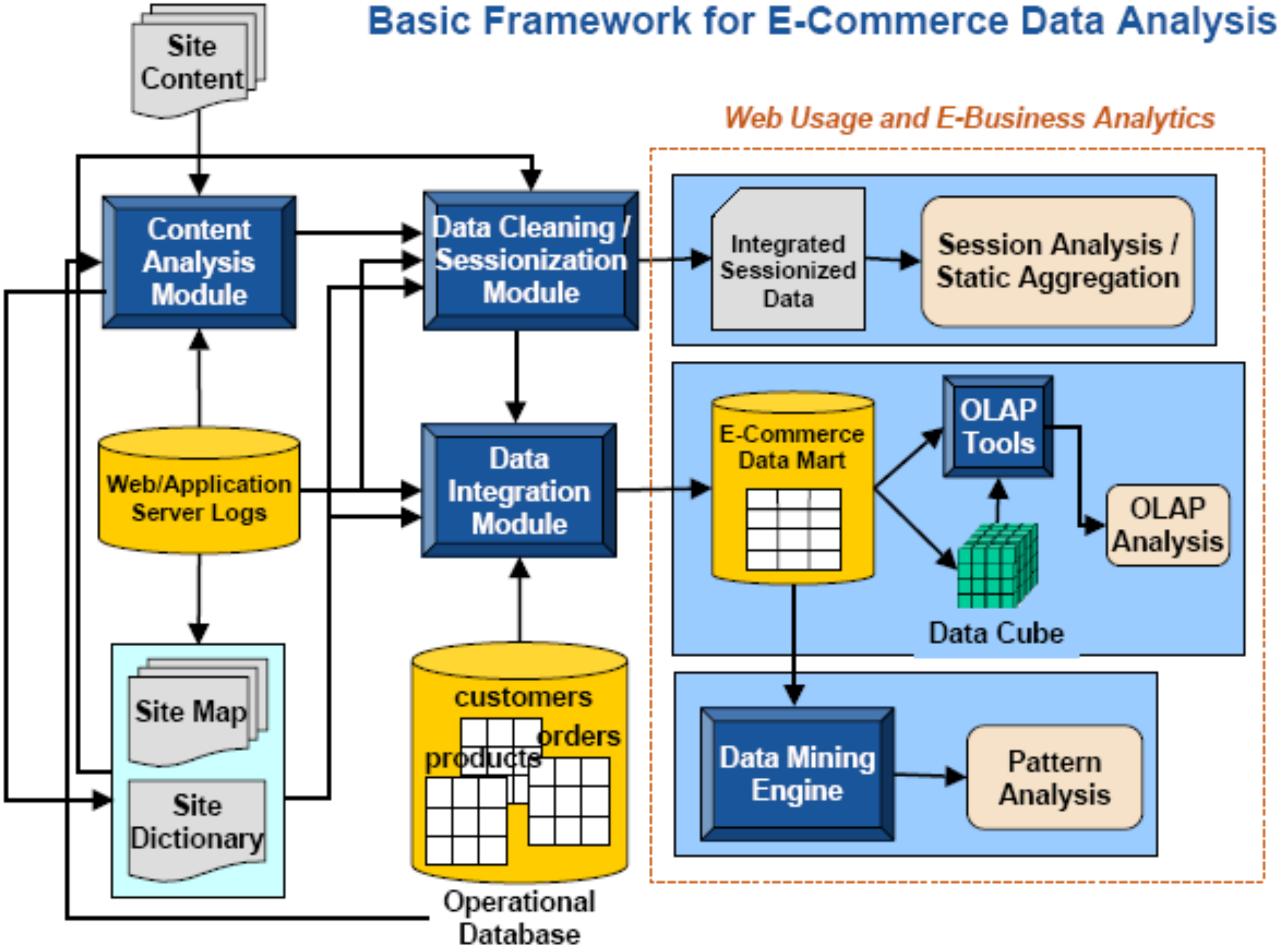
$U \times P =$

	food	news	car	house	party	sky
user 0	1	1	1	1	0	0
user 1	1	1	0	0	0	0
user 2	0	1	1	1	1	0
user 3	0	1	2	1	1	1
user 4	1	1	1	0	0	1
user 5	1	1	0	0	0	0

$m \times r$
content-enhanced transaction matrix

- Given a content-enhanced transaction matrix, a number of unsupervised mining techniques can be exploited
- For example, *clustering* the rows of the matrix may reveal users with common interests
- The following example regards *association rule* extraction from content-enhanced transaction matrix concerning a movie Website
  - pages contains information about the movies
  - possible rules could be:
    {“Romance”, “British”, “Comedy”} ⇒ {“Hugh Grant”}

E-commerce data analysis



Integrating sessions with e-commerce events

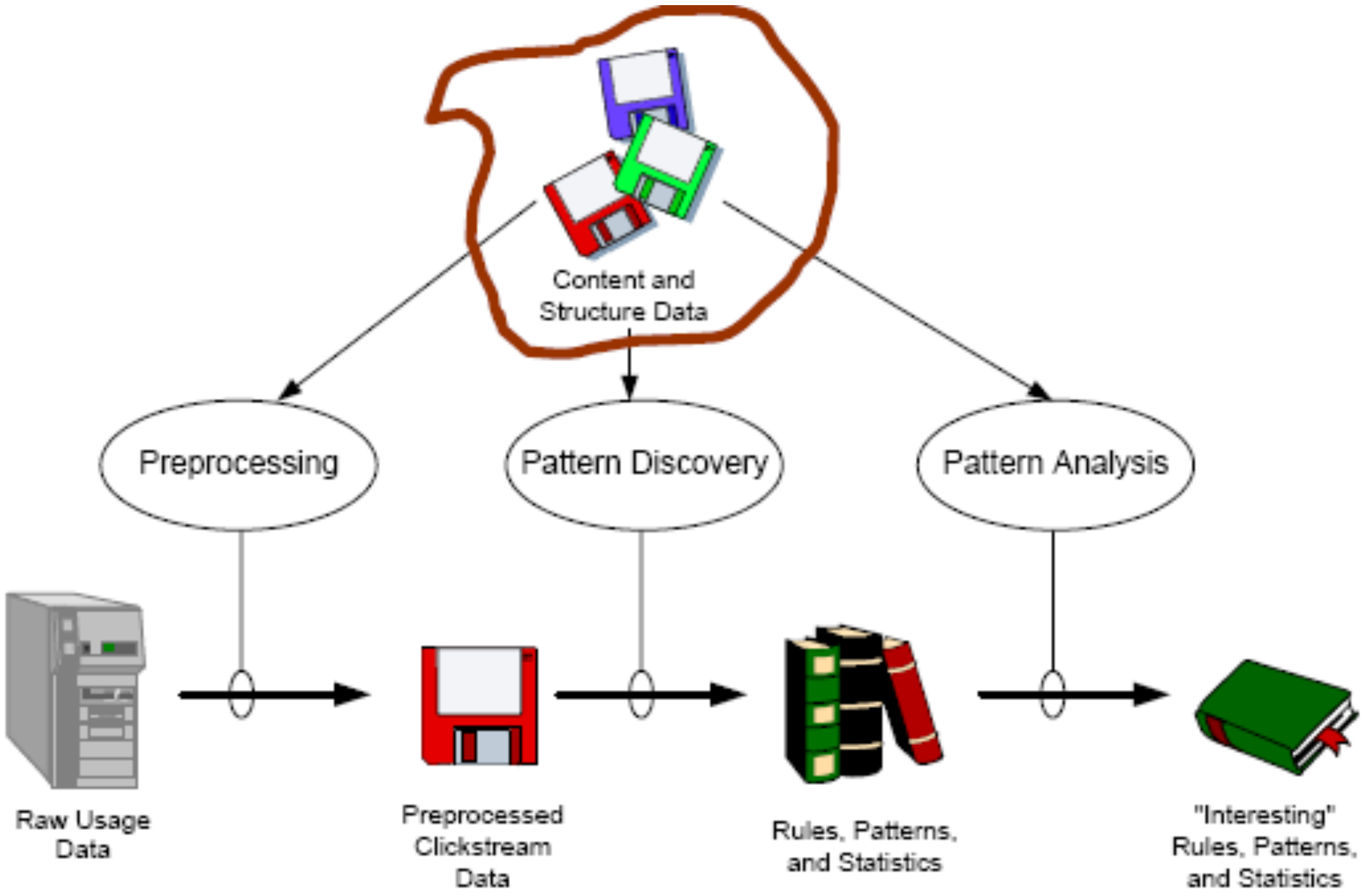
- **Either product oriented or visit oriented**
- **Used to track and analyze conversion of browsers to buyers.**
  - **major difficulty for E-commerce is defining and implementing the events for a site**
  - **however, in contrast to clickstream data, getting reliable preprocessed data is not a problem**

- **A major challenge is the successful integration with clickstream data**
  - **subsets of users' clickstream must be aggregated and appended to specific events**

Product-Oriented Events

- Product View
  - Occurs every time a product is displayed on a page view
  - Typical Types: Image, Link, Text
- Product Click-through
  - Occurs every time a user “clicks” on a product to get more information
- Shopping Cart Changes
  - Shopping Cart Add or Remove
  - Shopping Cart Change - quantity or other feature (e.g. size) is changed
- Product Buy or Bid
  - Separate buy event occurs for each product in the shopping cart
  - Auction sites can track bid events in addition to the product purchases

Integrating sessions with page content and link structure


Content and Structure Data
Preprocessing
Pattern Discovery
Pattern Analysis
Raw Usage Data
Preprocessed Clickstream Data
Rules, Patterns, and Statistics
"Interesting" Rules, Patterns, and Statistics

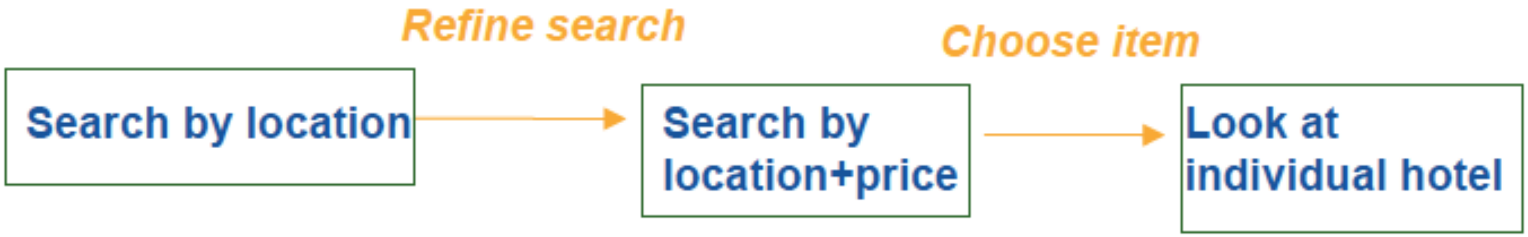
Integrating sessions with page content

- Basic idea: associate each requested page with one or more domain concepts, to better understand the process of navigation
  - *Example: a travel planning site*

- *From*

```
p3ee24304.dip.t-dialin.net - - [19/Mar/2002:12:03:51 +0100]
   "GET /search.html?l=ostsee%20strand&syn=023785&ord=asc HTTP/1.0" 200 1759
p3ee24304.dip.t-dialin.net - - [19/Mar/2002:12:05:06 +0100]
   "GET /search.html?l=ostsee%20strand&p=low&syn=023785&ord=desc HTTP/1.0" 200 8450
p3ee24304.dip.t-dialin.net - - [19/Mar/2002:12:06:41 +0100]
   "GET /mlesen.html?Item=3456&syn=023785 HTTP/1.0" 200 3478
```

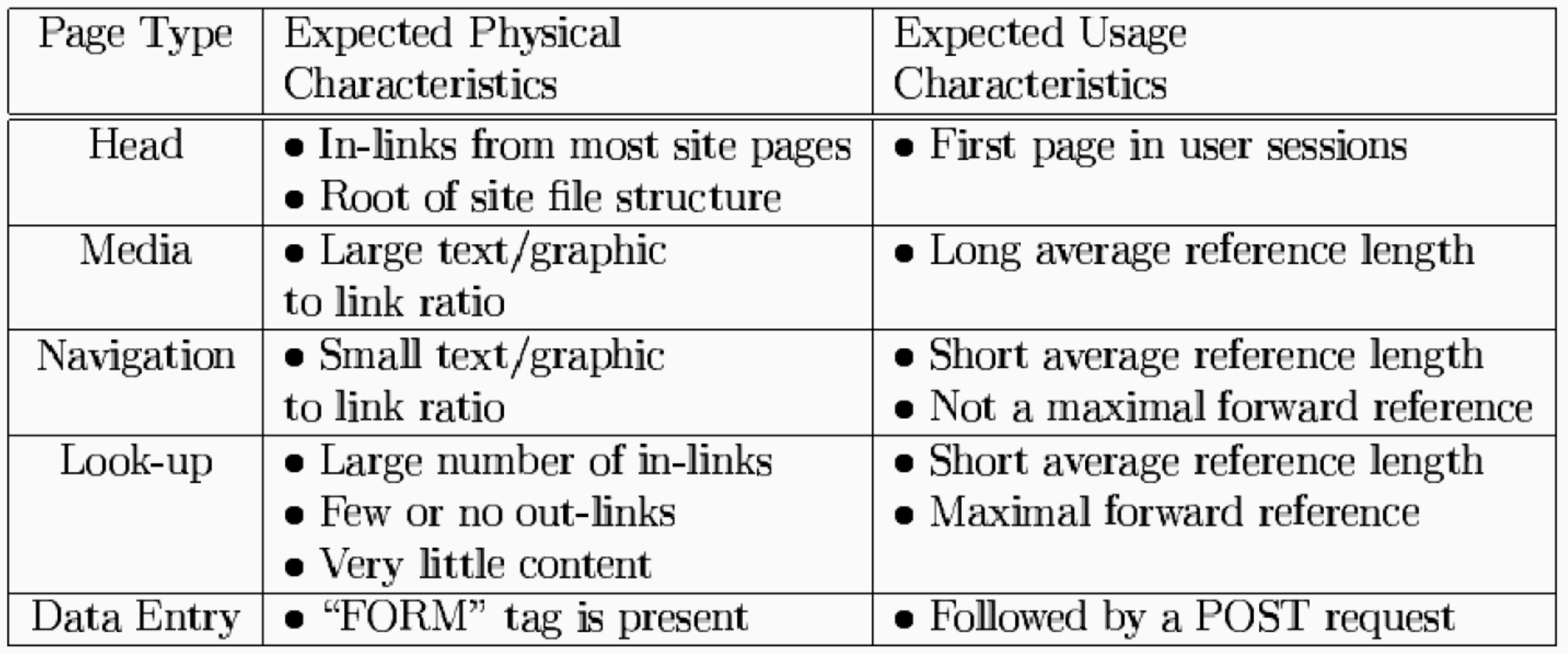
- *To*



Search by location → *Refine search* → Search by location+price → *Choose item* → Look at individual hotel

Integrating sessions with link structure

- **Page type defined by hyperlink structure bears information on function, or the designer's view, of how pages will be used**



Page Type	Expected Physical Characteristics	Expected Usage Characteristics
Head	• In-links from most site pages • Root of site file structure	• First page in user sessions
Media	• Large text/graphic to link ratio	• Long average reference length
Navigation	• Small text/graphic to link ratio	• Short average reference length • Not a maximal forward reference
Look-up	• Large number of in-links • Few or no out-links • Very little content	• Short average reference length • Maximal forward reference
Data Entry	• "FORM" tag is present	• Followed by a POST request

- **can be assigned manually by the site designer, or**
- **automatically by using classification algorithms, or**
- **a classification tag can be added to each page (e.g., using XML tags).**

Web Usage and E-Business Analytics

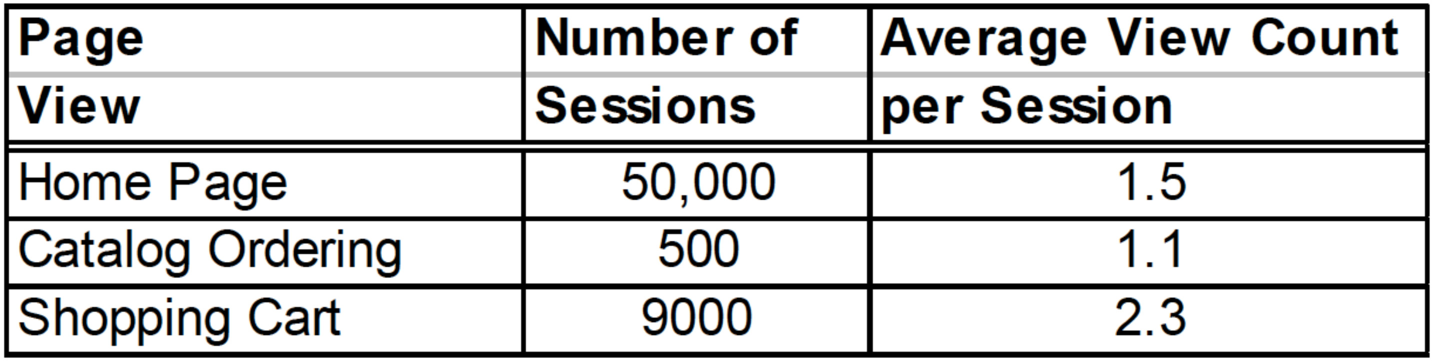
- Different types of analysis, not only data mining
  - Session Analysis
  - Static Aggregation and Statistics
  - OLAP
  - Data Mining

Session analysis

- Simplest form of analysis: examine individual or groups of server sessions and e-commerce data
- Advantages:
  - Gain insight into typical customer behaviors
  - Trace specific problems with the site
- Drawbacks:
  - LOTS of data
  - Difficult to generalize

Aggregate reports

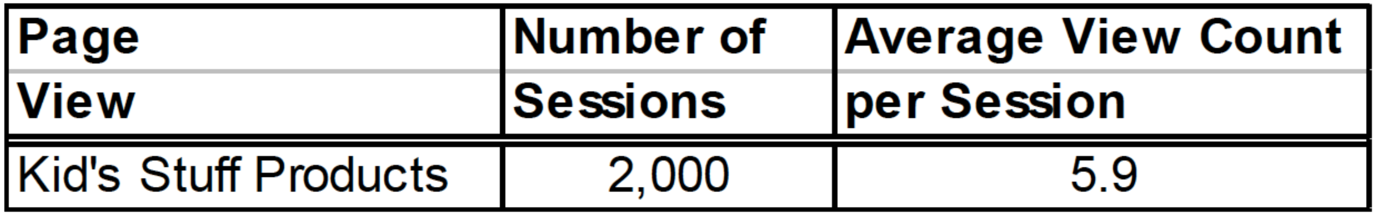
- Most common form of analysis.
- Data aggregated by predetermined units such as days or sessions.
- Advantages:
  - Gives quick overview of how a site is being used.
  - Minimal disk space or processing power required.
- Drawbacks:
  - No ability to “dig deeper” into the data.



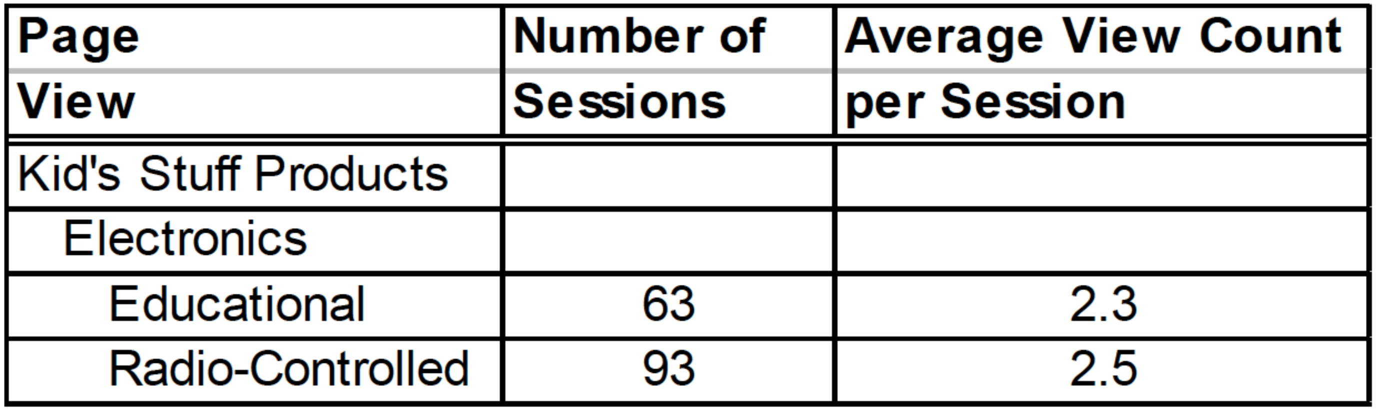
Page	Number of	Average View Count
View	Sessions	per Session
Home Page	50,000	1.5
Catalog Ordering	500	1.1
Shopping Cart	9000	2.3

Online Analytical Processing (OLAP)

- Allows changes to aggregation level for multiple dimensions.
- Generally associated with a Data Warehouse.
- Advantages & Drawbacks
  - Very flexible
  - Requires significantly more resources than static reporting



Page View	Number of Sessions	Average View Count per Session
Kid's Stuff Products	2,000	5.9



Page View	Number of Sessions	Average View Count per Session
Kid's Stuff Products		
Electronics		
Educational	63	2.3
Radio-Controlled	93	2.5

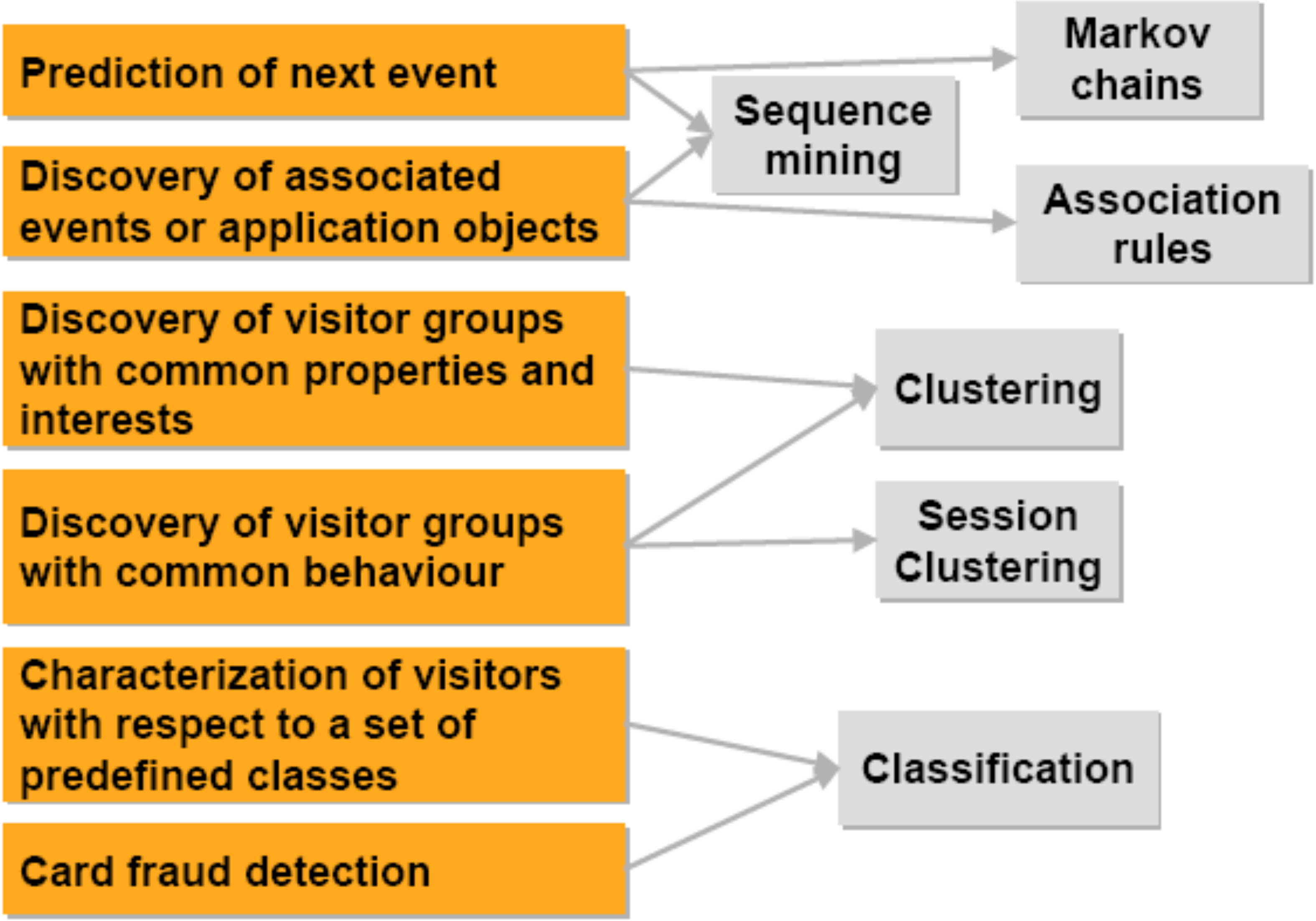
Data Mining: Going Deeper (I)

- **Frequent Itemsets**
  - The “Home Page” and “Shopping Cart Page” are accessed together in 20% of the sessions.
  - The “Donkey Kong Video Game” and “Stainless Steel Flatware Set” product pages are accessed together in 1.2% of the sessions.
- **Association Rules**
  - When the “Shopping Cart Page” is accessed in a session, “Home Page” is also accessed 90% of the time.
  - When the “Stainless Steel Flatware Set” product page is accessed in a session, the “Donkey Kong Video” page is also accessed 5% of the time.
- **Sequential Patterns**
  - add an extra dimension to frequent itemsets and association rules – time “x% of the time, when A appears in a transaction, B appears within z transactions.”
  - Example:The “Video Game Caddy” page view is accessed after the “Donkey Kong Video Game” page view 50% of the time. This occurs in 1% of the sessions.

Data Mining: Going Deeper (II)

- **Clustering: Content-Based or Usage-Based**
  - **Customer/visitor segmentation**
  - **Categorization of pages and products**
- **Classification**
  - **"Donkey Kong Video Game", "Pokemon Video Game", and "Video Game Caddy" product pages are all part of the Video Games product group.**
  - **customers who access Video Game Product pages, have income of 50+, and have 1 or more children, should get a *banner ad* for Xbox in their next visit.**

Some usage mining applications


Prediction of next event
Discovery of associated events or application objects
Discovery of visitor groups with common properties and interests
Discovery of visitor groups with common behaviour
Characterization of visitors with respect to a set of predefined classes
Card fraud detection
Markov chains
Sequence mining
Association rules
Clustering
Session Clustering
Classification

Data mining purpose: System Improvement

- Server-side caching of web pages
  - Sequential pattern mining of simple usage sessions
  - Use frequent sequence to predict candidate page, so that the proxy can prefetch pages, or the same pages are nor evicted from the cache

- Improvement of web design
  - Web mining approach for selecting hyperlinks for web portals
  - Association mining
  - Combine structure info and usage info to optimize portal page design
  - Adaptive web design

Data mining purpose: Personalization

- Web Personalization: “personalizing the browsing experience of a user by dynamically tailoring the look, feel, and content of a Web site to the user’s needs and interests.”
- Why Personalize?
  - broaden and deepen customer relationships
  - provide continuous relationship marketing to build customer loyalty
  - help automate the process of proactively market products to customers
    - lights-out marketing
    - cross-sell/up-sell products
  - provide the ability to measure customer behavior and track how well customers are responding to marketing efforts

Standard Approaches for Personalizing/ Recommending

- Rule-based filtering
  - provide content to users based on predefined rules (e.g., “if user has clicked on A and the user’s zip code is 90210, then add a link to C”)
- Collaborative filtering
  - give recommendations to a user based on responses/ratings of other “similar” users
  - see the Netflix Prize: http://www.netflixprize.com/
- Content-based filtering
  - track which pages the user visits and recommend other pages with similar content
- Hybrid Methods
  - usually a combination of content-based and collaborative

SUGGEST: Online recommender system

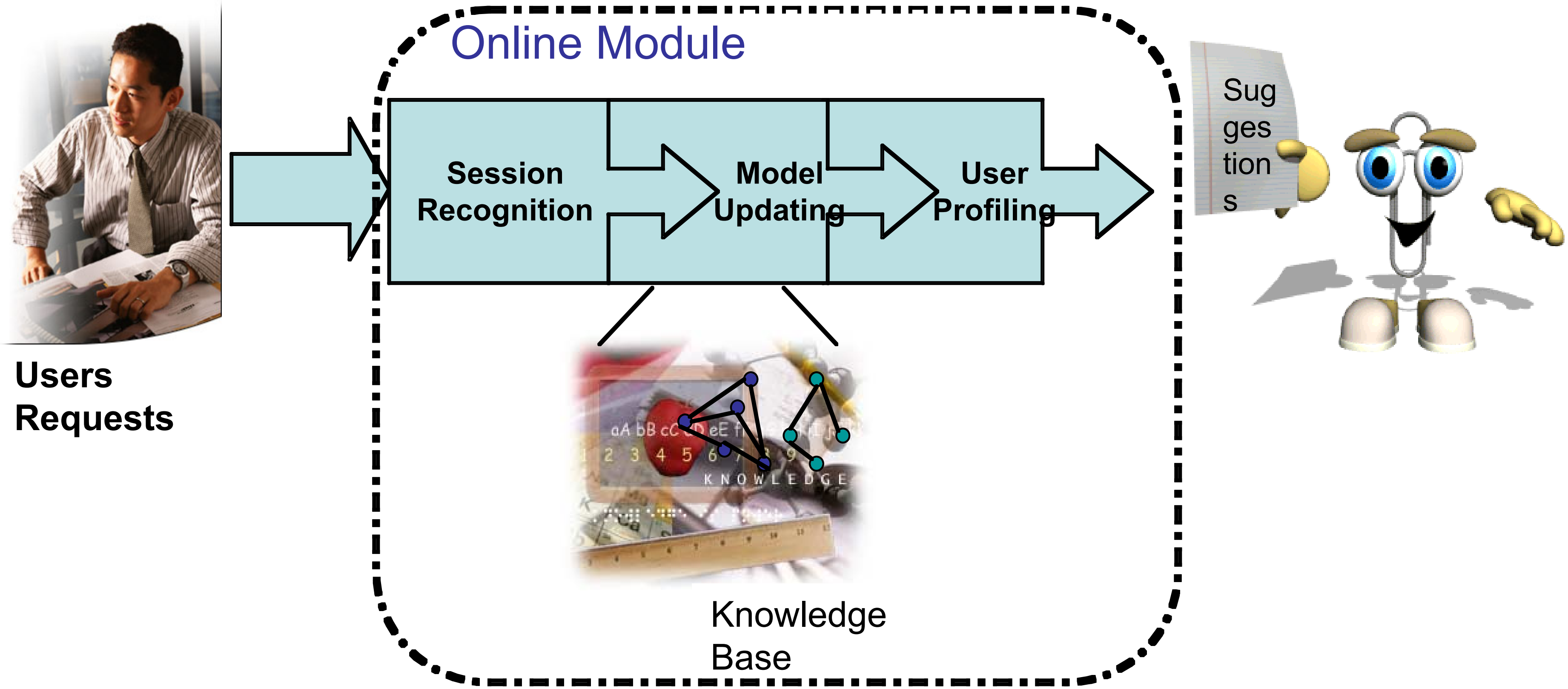
- Recommender system usually work with a model (e.g. association rules, clusters) build off-line
  - Possible out-to-date information
  - Needs for administrator to manage updates
  - Possible effectiveness issues

- SUGGEST
  - Based on an Incremental Algorithm
    - input of a recommendation: a partial user session
    - output of the recommendation: if enough info are present, it returns a list of possibly interesting pages

Ranieri Baraglia, Fabrizio Silvestri: ***Dynamic personalization of web sites without user intervention***. Commun. ACM 50(2): 63-67 (2007)

R. Baraglia, F. Silvestri, ***An Online Recommender System for Large Web Sites***, IEEE/WIC/ ACM Int.l Conf on Web Intelligence, Beijing, China, 2004.

SUGGEST: Online recommender system



Users
Requests

Online Module

Session Recognition

Model Updating

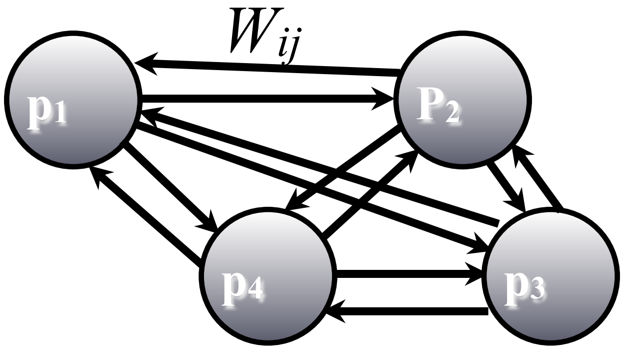
User Profiling

Knowledge
Base

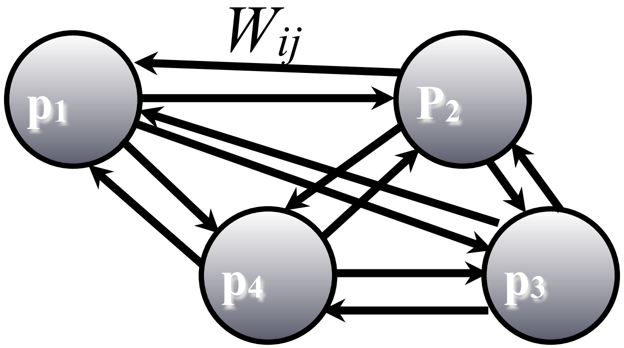
Sug
ges
tion
s

How does Suggest work

- **For building the knowledge base, no user sessionization is needed**
  - **this gives to the system some privacy preservation features**
- **Cookies stored on the client to identify users**
- **SUGGEST dynamically maintains on the web server a visit graph of a Web site**
  - **built on the basis of pairs of referrer and requested pages**



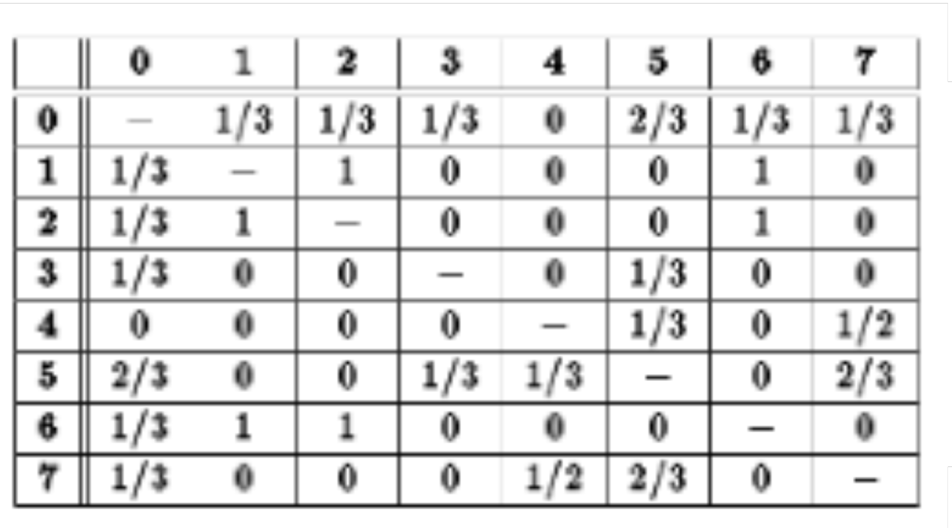
How does Suggest work



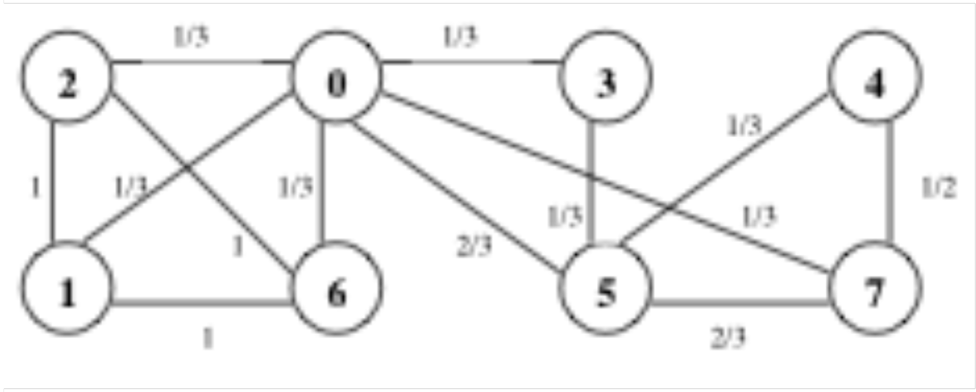
$$W_{ij} = N_{ij}/max\{N_i, N_j\}$$

- N_{ij} = the number of times pages i and j have been accessed consecutively and in any order by a user
- N_i and N_j = the numbers of times pages i and j have been visited
- Dividing by the maximum between single occurrences of the two pages has the effect of reducing the relative importance of links involving index pages, which are very likely to be visited with any other page and nevertheless are of little interest as potential suggestions

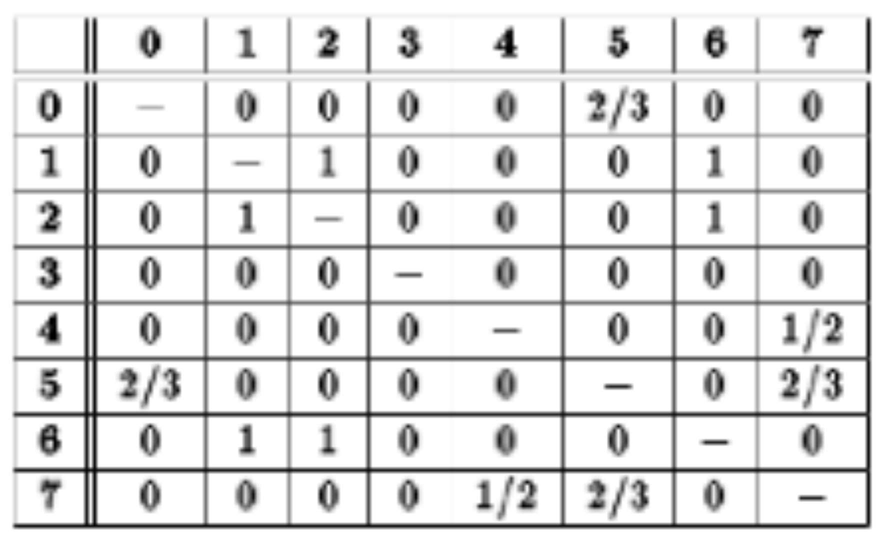
Adjacency matrix for the knowledge base



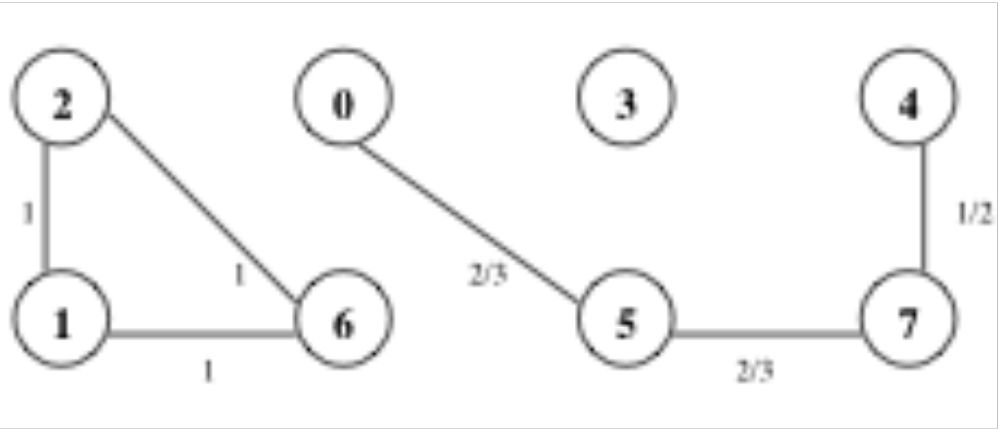
	0	1	2	3	4	5	6	7
0	–	1/3	1/3	1/3	0	2/3	1/3	1/3
1	1/3	–	1	0	0	0	1	0
2	1/3	1	–	0	0	0	1	0
3	1/3	0	0	–	0	1/3	0	0
4	0	0	0	0	–	1/3	0	1/2
5	2/3	0	0	1/3	1/3	–	0	2/3
6	1/3	1	1	0	0	0	–	0
7	1/3	0	0	0	1/2	2/3	0	–



Adjacency matrix for the knowledge base



	0	1	2	3	4	5	6	7
0	–	0	0	0	0	2/3	0	0
1	0	–	1	0	0	0	1	0
2	0	1	–	0	0	0	1	0
3	0	0	0	–	0	0	0	0
4	0	0	0	0	–	0	0	1/2
5	2/3	0	0	0	0	–	0	2/3
6	0	1	1	0	0	0	–	0
7	0	0	0	0	1/2	2/3	0	–



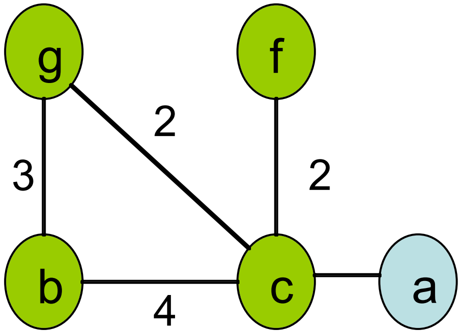
- In order to reduce the graph size, and remove noise from the obtained model, a threshold is exploited
    - MinFreq is the threshold
    - Arc labels below the threshold are deleted

The algorithm

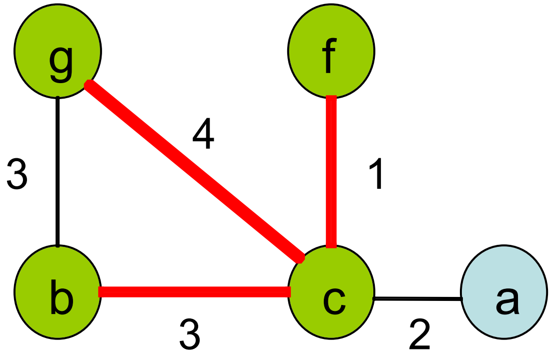
- **Model building (online)**
  - **Matrix Update**
  - **Cluster models extraction**
    - **connected components of the graph**

- **Recommendation building**
  - **Current session similarity with the stored cluster models**
  - **Suggestions are built and returned**

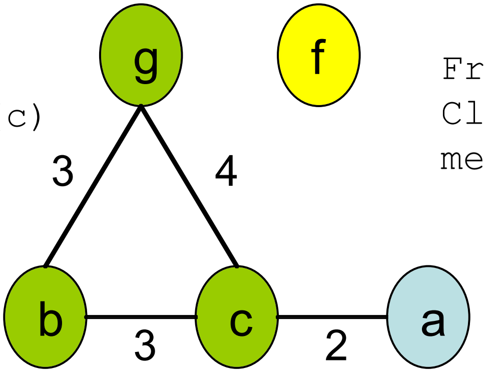
Model building



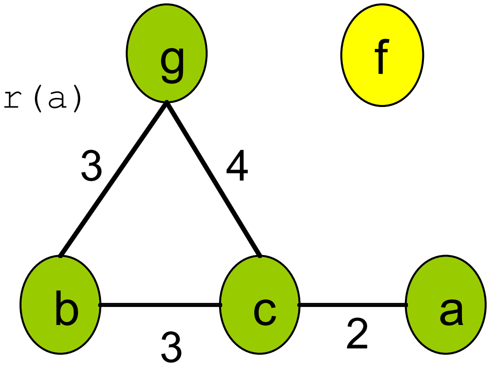
$$Freq(a,c) = \frac{\#(a,c)}{\max(\#(a),\#(c))}$$



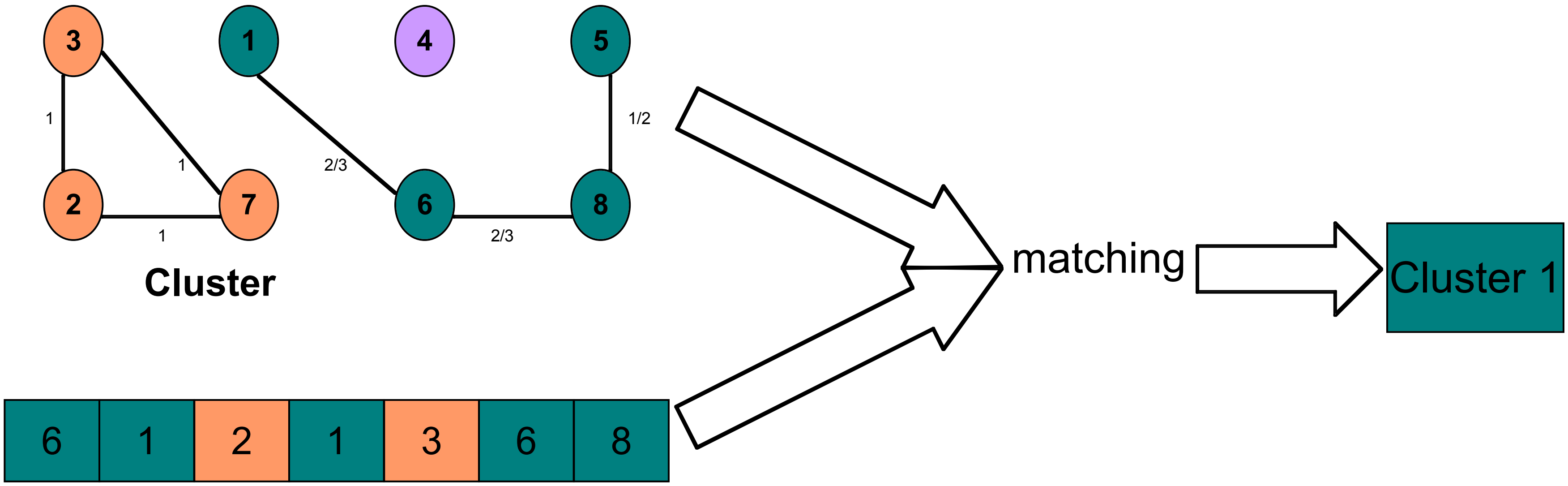
```
Freq(c,f) < Minfreq
Cluster (f) = Cluster(c)
Build_cluster(c,f)
```



```
Freq(c,a) ≥ Minfreq
Cluster (c) ≠ Cluster(a)
merge_cluster(c,a)
```



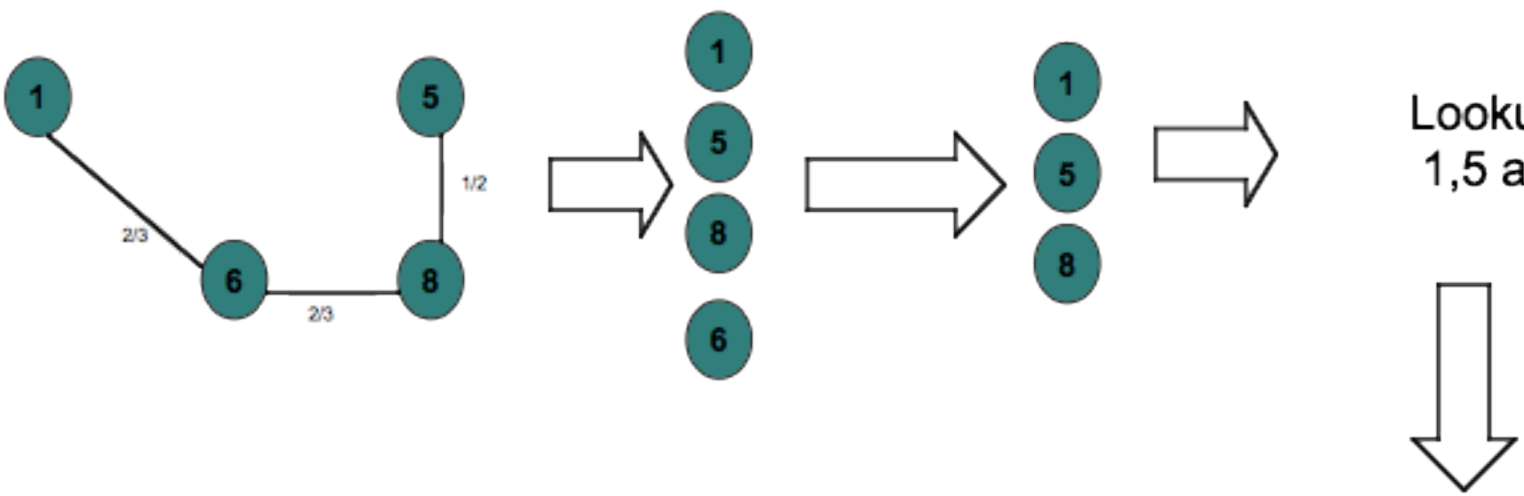
Recommendation building



6	1	2	1	3	6	8

Indexes of the visited pages
in the session of Client A

Recommendation building



cluster

Nodes as
represented
in the cluster

The three nodes
with the highest
UsageRank score

Lookup on
1,5 and 8

http://.../usr/www_root/index.html

http://.../usr/www/list/interest/index.htm

http://.../usr/www/list/interest/hobby.htm

Ranking
based on
usage based
PageRank

How to evaluate

- The evaluation of a recommender system is a big issue
  - the best way is to evaluate the relevance of suggestions with a live user trial
- The authors of SUGGEST used sessions extracted from public available logs

Dataset	Time Window	N_S
NASA	27 days	19K
USASK	180 days	10K
BERK	22 days	22K

- For a generic *i*th session, the first half-session S_i^1 is used to build the suggestion
- The second half-session S_i^2 is used to evaluate the recommendations R
- The metric used to evaluate the quality of suggestions
  - is directly proportional to the intersection $R \cap S_i^2$
  - the suggestions corresponding to page appearing late in: S_i^2 are weighted more

Summary

- Web usage mining has emerged as the essential tool for realizing more personalized, user-friendly and business-optimal Web services.
- The key is to use the user-clickstream data for many mining purposes.
- Traditionally, Web usage mining is used by e-commerce sites to organize their sites and to increase profits.
- It is now also used by search engines to improve search quality and to evaluate search results, etc, and by many other applications.