Understanding Web usage at different levels of abstraction: coarsening and visualising sequences

By Bettina Berendt
Presented by: Lisheng Sun

Contents

Introduction: Problem

- The way users navigate a Web site can be used to learn about their preferences and offer them a better adapted interface.
- Behaviour is complex and can exhibit more "local" and more "global" regularities, especially the meaningful behavioural patterns extend over longer periods of time.

Introduction: Problem(Con')

- While Web usage mining provides for the syntactic specification of structured patterns, it is less clear how to analyse and visualise usage data involving ***longer*** patterns ***with little expected struture***, ***without losing an overview*** of the whole of all paths.

Introduction: Challenge

- The mining Challenge is to combine openness and little specification to be albe to find unexpected patterns with enough structure to easily find meaninful results.

Introduction: 2 Main approach

- One approach is to select patterns, e.g. by filtering based on numerical criteria like support thresholds or more sophisticated mechanisms.
- Another approach is to **abstract** from details by classifying accessed pages or paths.

Introduction: Method used here

- The method proposed in this paper employs two ways of analysing complex Web usage data:
- ***Concept hierarchies*** are used as a basic method of aggregating Web pages.
- ***Interval-based coarsening*** is proposed as a method for representing sequences at coarser levels of abstraction.

Introductin: *stratogram*

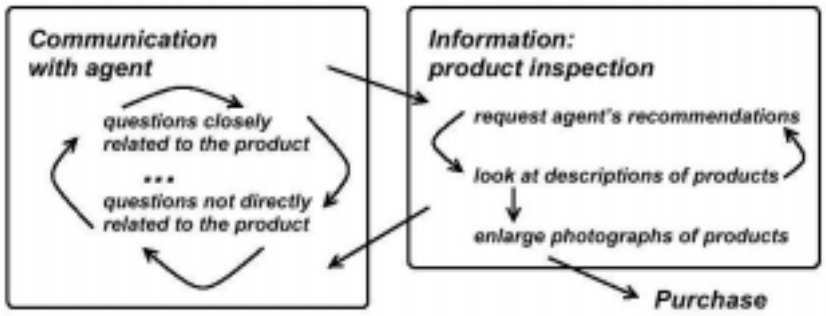
- A popular way to visualise web site: each page is assigned one place in 2D or 3D space, transitions are shown as arrows between pages.
- To visualise single user paths, pages are plotted againest time.
- ***statogram***: a way of combining the above visualisation approaches.

Contents

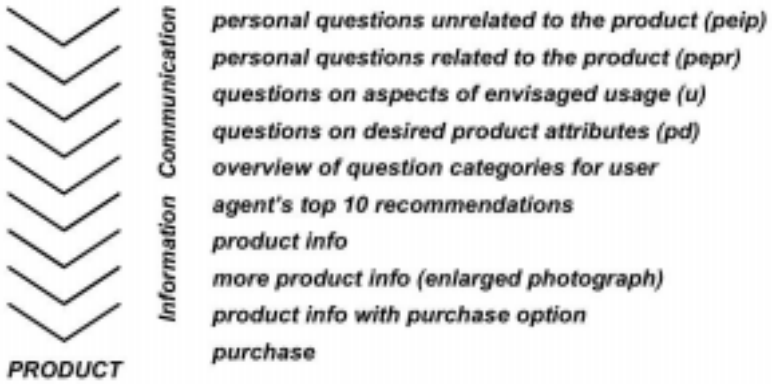
Example: *agent-supported shopping in an online store*

- After selecting a product category and going through an introductory phase, users are encouraged to answer some questions related to the product they intend to purchase.
- At any time, the agent can be asked to determine the current top 10 products out of the shop's offers, based on the user's preferences as stated in the answers.
- From the top 10 pape, users can also go back to continue answering questions or revise given answers.
- The user can exit at any time without purchasing.

Example: incomplete conceptual sketch



Example: classification of each URL



Contents

- Introduction
- Example: agent-supported shopping in an online store
- **Basic stratograms**
- Interval-based coarsening
- Pattern representation and discovery

Basic stratograms: example

- Each segment along the x axis denotes one step in the original logs.
- To find interesting patterns, pages have been abstracted using a concept hierarchy
- Each line in the diagram conects a point (t,v) with another point (t+1,v').

Basic stratograms: example

- Two phases: "communication phase" and "information phase"
- Most users have answered most of the questions.
- For jacket, "more product info" was requested more often and the information phase lasted longer.
- conversion efficiency: 55%(camera) V.S. 24%(jacket)
- conversion efficiency over short paths: 35%(camera) V.S. 10%(jacket)

Basic stratograms: example

- In many Web usage analyses, the initial analysis questions were:"What did users do in this site? When? And how often?“
- More suited to selling goods like cameras, i.e. product that may be judged by examining a range of technical details.
- For jackets, more info and interface are needed.

Basic stratograms

- A basic stratogram rests on the relative frequencies of transitions.
- $s.(o_s+t)$ denotes the t^{th} request(step) in session s after the offset o_s.
- The normalised frequency of the transition from node A_1 to node A_2 at the t^{th} step after the respective session offsets o_s is: (1).....

Basic stratograms

- function v: maps the visited pages from the set pages to numerical values N.
- Normalised frequency from node $A_1$to A_2 at the t^{th} step, after offset O_s:

$$f(A_1, A_2, t) = \frac{|\{s \mid s.(o_s+t)=A_1 \wedge s.(o_s+t+1)=A_2\}|}{S}$$

Basic stratograms visualization

Definition 1 *A* basic stratogram strat *is defined as*

$$strat = \langle pages, st, v, tr, \theta_1, \theta_2 \rangle \quad with$$
$$st = \{0, \ldots, max_s(|s| - 2)\},$$
$$v : pages \mapsto N,$$
$$tr = \{f(A_1, A_2, t) \mid A_1 \in pages, A_2 \in pages \cup \{end\}, t \in st\}$$

where the θ *are support thresholds.*

Contents

- Introduction
- Example: agent-supported shopping in an online store
- Basic stratograms
- **Interval-based coarsening**
- Pattern representation and discovery

Interval-based coarsening

- The drawback of basic stratogram: long episodes
- Interval-based coarsening summarises transitions in consecutive, disjoint intervals of a size g>= 1.

$$f_g(A_1, A_2, t) = \sum_{x:t \times g}^{(t+1)\times g-1} f(A_1, A_2, x)$$

Interval-based coarsening

Definition 2 *A* coarsened stratogram $strat_g$ *with degree of coarsening g is defined as*

$$strat_g = \langle pages, st, v, tr, \theta_1, \theta_2, g \rangle \quad with$$

$$st = \{0, \ldots, int(\frac{max_s(|s| - 2)}{g})\},$$

$$v : pages \mapsto N,$$

$$tr = \{f_g(A_1, A_2, t) \mid A_1 \in pages, A_2 \in pages \cup \{end\}, t \in st\}$$

where the θ *are support thresholds.*

Interval-based coarsening

- A coarsened stratogram visualization is defined analogously to a basic stratogram visualization.
- In a coarsened stratogram, if in at least one session, the transition under consideration has occurred more than once, this transition may be considered as more characterisitc of this interval than others. These transitions are displayed thicker and darker.

Coarsened stratogram and basic stratograms

- Basic stratograms are one limiting case of coarsened stratograms(g= 1).
- The opposite limiting case: g → unlimited

Coarsened stratogram (advantage)

- The coarsened stratograms summarise behaviour that may occur in roughly the same shape, but starting at different offset.
- The analyst can gain a summary view of the data and then "zoom in" by decreasing the value of g.
- New regularities become visible.
- A new regularity "X" become visible.
- Patterns of leaving the site become clearer.

Contents

Pattern representation and discovery

- Stratograms rest on type hierarchies that allow the discovery and statistical comparision of patterns independently of visulisation.
- A type $[A_1,p_1,A_2,p_2,.....A_n]$ matches sequences in user paths that contain A_1, then A_2,.. and then A_n, with the paths in between constrained by the path specifications p_i specifying a minimum and a maximum number of requests that lies between A_1 and A_{i+1}.

Pattern representation and discovery

- Coarsened stratograms can be derived from type hierarchies.
- classify(a,d): algorithm to classify sessions into a type hierarchy.

Algorithms to produce coarsened stratogram(1)

```
(1) determine_frequencies_1 (g)
(2)   for each session do
(3)      repeat
(4)         read(z[0]);
(5)      until ((is_offset(z[0]))
(6)          or (end-of-session));
(7)      t := 0; i := 1;
(8)      while (not end-of-session) do
(9)         (z[0],i) := classify(z[0],i,t);
(10)        t := div(i,g);
```

Algorithms to produce coarsened stratogram(2)

```
(1) determine_frequencies_2 (g)
(2)   initialise all f_g(A1,A2,t):=0;
(3)   for each A1 do
(4)      for each A2 do
(5)         x:=0; t:=0;
(6)         while (x <= t_max) do
(7)            f_g(A1,A2,t) += f(A1,A2,x)
(8)            x++;
(9)            if (x >= (t+1)*g) then
(10)              t++;
```

Conclusion

- The author proposes basic and coarsened stratograms to visualise Web usage at different degree of abstraction.
- Using the case study of a online shopping, we find that the new methods are useful and can find new interesting patterns.

The End, Thank You.

Questions or Comments ?