Index Structures of User Profiles for Efficient Web Page Filtering Services

Y. H. Wu  A. L. P. Chen

Database Laboratory
Department of Computer Science
National Tsing Hua University

Outline

- Introduction
- Related Approaches
- Our Approaches
- Comparisons
- Conclusion
Introduction

- **Motivation**
  - Searching problem on the WWW
    - search engine
    - meta-search engine
  - The performance may get worse if the number of web pages grows rapidly

- **Goal**
  - Filtering approach
  - find the matched profiles for each web page

A Web Page Filtering Service
Introduction

- **An Example**
  - Conjunction of keywords
  - Boolean model
  - Matches \{P_1, P_4\}

<table>
<thead>
<tr>
<th>Profile</th>
<th>Keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_1</td>
<td>a b</td>
</tr>
<tr>
<td>P_2</td>
<td>a d</td>
</tr>
<tr>
<td>P_3</td>
<td>a d e</td>
</tr>
<tr>
<td>P_4</td>
<td>b f</td>
</tr>
<tr>
<td>P_5</td>
<td>c d e f</td>
</tr>
</tbody>
</table>

Related Approaches

- **The Counting Method**
  - Keyword array: inverted lists
  - Profile arrays: TOTAL, COUNT
  - Matching criteria: COUNT=TOTAL

<table>
<thead>
<tr>
<th>Profile</th>
<th>TOTAL</th>
<th>COUNT</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>P_2</td>
<td>2</td>
<td>/</td>
</tr>
<tr>
<td>P_3</td>
<td>3</td>
<td>/</td>
</tr>
<tr>
<td>P_4</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>P_5</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>P_1, P_2, P_3</td>
</tr>
<tr>
<td>b</td>
<td>P_1, P_4</td>
</tr>
<tr>
<td>c</td>
<td>P_3</td>
</tr>
<tr>
<td>d</td>
<td>P_2, P_3, P_4</td>
</tr>
<tr>
<td>e</td>
<td>P_1, P_3</td>
</tr>
<tr>
<td>f</td>
<td>P_4, P_5</td>
</tr>
</tbody>
</table>

Example page
a b c f
Related Approaches

- **The Tree Method**
  - K-nodes: internal nodes
  - P-nodes: leaf nodes
  - External path
    - root $\rightarrow$ p-node
    - a profile
  - Matches
    - root $\rightarrow$ a $\rightarrow$ b $\rightarrow$ P
    - root $\rightarrow$ b $\rightarrow$ f $\rightarrow$ P

Our Approaches

- **Method 1**
  - Index path with path signatures
  - Path signature of the example page: 110011
  - Matches
    - at node b: P AND 11 = P
    - at node f: P AND 11001 = P
**Our Approaches**

- **Method 2**
  - Index graph with path signatures
  - Matches
    - $\text{root} \rightarrow a \rightarrow b \rightarrow P_1$: 11
    - $\text{root} \rightarrow b \rightarrow f \rightarrow P_4$: 21

- **Method 3**
  - Index path with profile sets
  - Candidate set
  - Target set $\leq$ candidate set $\cap$ profile set
    - at node b: $T = \{P_1P_2P_3P_4P_5\} \cap \{P_1P_4\} = \{P_1P_4\}$
    - at node d: $T = \{P_2P_3P_4P_5\} \cap \{P_2P_3P_5\} = \{P_2P_3P_5\}$

---

**Example page**

- a b c f
Our Approaches

- **Method 4**
  - Index graph with profile sets
  - Path length
  - Matching criteria: matching of keywords $\land$ profile id $\subseteq$ target set $\land$ equal path length

- **Comparisons**

  - **Notation**
  - | Symbol | Description |
  - |-------|-------------|
  - | $P$   | The set of all profiles |
  - | $K$   | The set of all distinct keywords |
  - | $n$   | Average number of keywords in a profile |
  - | $f$   | Average number of profiles in which a specific keyword is specified |
  - | $m$   | Average number of keywords to represent a web page |
Comparisons

Summary

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Approaches</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Counting Method</td>
</tr>
<tr>
<td>Duplication of</td>
<td>profile</td>
</tr>
<tr>
<td>information</td>
<td>O(nf)</td>
</tr>
<tr>
<td>Sorting of keywords</td>
<td>no</td>
</tr>
<tr>
<td>Storage space</td>
<td>O(</td>
</tr>
<tr>
<td>Insertion/Deletion time</td>
<td>O(nf)</td>
</tr>
<tr>
<td>Matching time</td>
<td>O(mf+</td>
</tr>
<tr>
<td>Modification time</td>
<td>O(nf)</td>
</tr>
</tbody>
</table>

Conclusion

Contribution

- Four new methods for profile indexing
- Comparisons by complexity analyses
- Efficient web page filtering service

Future Work

- Prototype system for real data
- Dissemination and display of the filtered results
- More predicates for specifying the user profiles