Conceptual User Tracking

Daniel Oberle[1], Bettina Berendt[2], Andreas Hotho[1], and Jorge Gonzalez[1]

[1] Institute of Applied Informatics and Formal Description Methods AIFB
University of Karlsruhe
D–76128 Karlsruhe, Germany
{oberle,hotho,gonzalez}@aifb.uni-karlsruhe.de
[2] Institute of Information Systems
Humboldt University Berlin
D–10178 Berlin, Germany
berendt@wiwi.hu-berlin.de

Abstract. Web usage mining applies data mining techniques to records of Web site visits. To better understand patterns of usage, analysis should take the semantics of visited URLs into account. This paper presents a framework for enhancing Web usage records with formal semantics based on an ontology underlying the site. Besides, it elicits automated methods of mapping URLs to application events. Using the ontology's taxonomy, we describe user actions at different levels of abstractions. Using the ontology's concepts and relations, we capture the multitude of user interests expressed by a visit to one page. We employ our ideas in an application of SEAL, a framework for semantic portals that uses Semantic Web technologies to support communities of interest. Different realizations of semantically enriched user tracking are discussed and related to other approaches. We describe first results from a prototypical system, and discuss benefits of Conceptual User Tracking for Web usage mining.

Keywords. web (usage) mining, user tracking, semantic portal, ontology

1 Introduction

Web usage mining applies data mining techniques to the usage of Web resources, as recorded in Web server logs or other logs of requested URLs (plus, possibly, further parameters) [24].

Web logs were initially designed to help site administrators identify traffic and possible bandwidth problems, broken links, etc., and analyzed using simple statistics like hit and pageview counts. More and more, their value for understanding site users' behavior is also being recognized, and techniques like association rule mining, clustering, or sequential pattern discovery are being used to identify co-occurring items in browsing and shopping histories, different user segments, navigation strategies, etc. This knowledge can be exploited to improve site design and navigation opportunities [5,7,12], to develop marketing strategies including recommender systems [18,15], etc.

However, because the primary focus of this kind of usage recording is technical, an interpretation of URLs in terms of user behavior, interests, and intentions, is not always straightforward. For example, the site owners of an online bookstore will not be interested

E. Menasalvas et al. (Eds.): AWIC 2003, LNAI 2663, pp. 155–164, 2003.

in an association rule like “If `http://www.the_shop.com/show.html?item=123`, then `http://www.the_shop.com/show.html?item=456`, support = 0.05, confidence = 0.4”, but in a statement like “Users who bought *Hamlet* also tended to buy *How to stop worrying and start living*.”. In other words, Web usage analysis is not interested in *patterns of URLs*, but rather in *patterns of application events*, where application events are usually described by actions (such as “buying”) or content (such as “(showing interest in) *Hamlet*”).

So in order to obtain meaningful results, Web usage mining must exploit the semantics of the pages visited along user paths. In this process, meaningful application events have to be identified, URLs have to be mapped to application events, and different levels of abstraction (taxonomical knowledge) have to be taken into account.

A number of recent studies have shown the usefulness of exploiting semantics for mining. After the preprocessing steps in which access data have been mapped into taxonomies, different approaches are taken in subsequent mining steps. Particularly interesting are mining algorithms that can deal flexibly with taxonomical knowledge (a simple form of ontological knowledge). Srikant and Agrawal [23] search for associations in given taxonomies, using support and confidence thresholds to guide the choice of level of abstraction. Dai and Mobasher [8] present a scheme for aggregating towards more general concepts when an explicit taxonomy is missing. They apply clustering to sets of sessions; this clustering identifies related concepts at different levels of abstraction. They are thus able to identify common interests of users at a more abstract level than that of the invidividual pages (e.g., that they all liked films in which a certain actor played an important role), which allows them to circumvent the "new item problem" in a recommender system.

However, for all these mining techniques to work, the semantics of the pages have to be extracted first in order to perform the mapping from requested URLs to requested contents/services. Also, an ontology has to be defined (we use the term "ontology" in the sense of [10]).

In this paper, we describe a novel scheme for providing semantics. After an introduction to the underlying Semantic Web technology and community portals as an important application area (Sects. 2.1 and 2.2), we present a framework for the semantic enrichment of Web logs (Sect. 2.3). We show how our approach, described in Sects. 2.4 and 2.5, extends current proposals (Sect. 2.4). In Sect. 3, we briefly describe a case study that shows the kinds of knowledge that can be discovered using Conceptual User Tracking. Section 4 concludes our study.

2 Semantic Enrichment of Web Logs

2.1 SEmantic portAL

A Web portal is the center of the information needs of a certain interest group in the WWW. On the conceptual, knowledge sharing level it was found that "people can’t share knowledge if they don’t speak a common language" is utterly crucial for the case of community Web portals. Besides people, the knowledge has to be shared between machines, too. This approach, called SEmantic portAL [16], demands a conceptual

structuring for the representation of information. This is achieved by an ontology, agreed upon by the community and used as the portal's backbone.

Such portals benefit from the Semantic Web, one of today's hot topics which brings "[...] structure to the meaningful content of Web pages." [6]. To build the Semantic Web, pages are supplemented with semi-structured meta-data that provide the formal semantics for Web content by referring to ontologies. In the case of SEAL, an ontology would not only be used to structure the portal itself, but also for annotating the portal's resources with meta-data. The technological basis for representing data in the Semantic Web is RDF [14], a semi-structured data format that resembles directed labelled graphs. Ontologies are represented in RDF Schema (RDFS) which defines primitives similar to object-oriented data models. Web pages are equipped with such meta-data by XML-serializations of RDF.

2.2 The KA^2-Portal

For demonstration purposes, we use the so-called KA^2 Portal [1] (Knowledge Annotation Initiative of the Knowledge Acquisition Community, http://ka2portal.aifb.uni-karlsruhe.de) as a particular application of SEAL. The KA^2 initiative was conceived for obtaining knowledge out of annotated resources belonging to the Knowledge Acquisition Community. The ontology was created in international collaboration; it describes a universe of relevant concepts for research, as shown in Fig. 1.

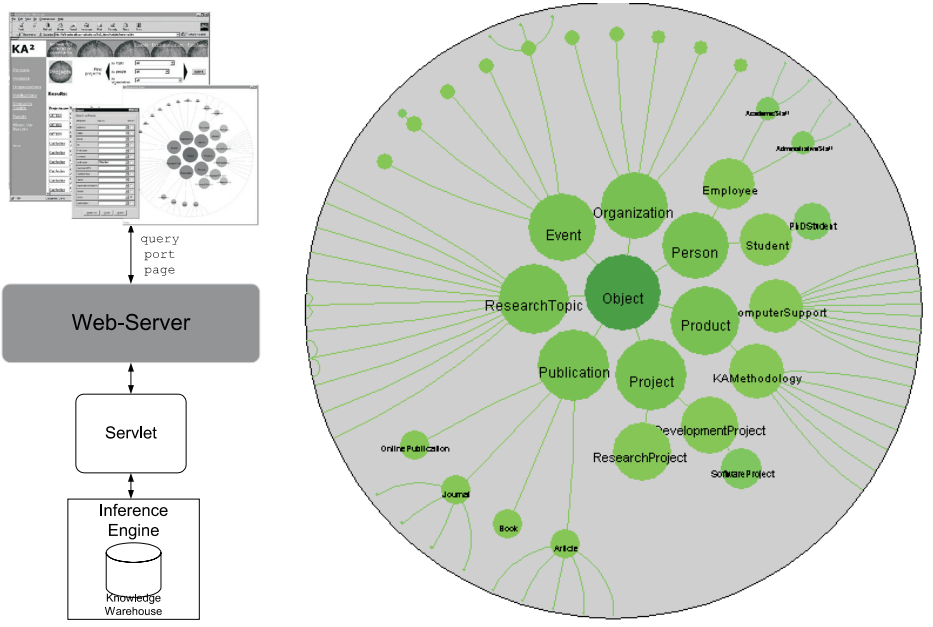

Fig. 1. KA^2 Portal architecture and its ontology (visualized by the Hyperbolic View Applet)

Ontobroker [9], an inference engine based on F-Logic, hosts the ontology and allows the generation of dynamic pages as discussed below. It is important to note that all of the static pages are annotated with RDF meta-data according to the same ontology. As shown in Fig. 1, the interface between the portal and Ontobroker is implemented via a servlet carrying out queries to the inference engine. The servlet is referenced with the parameters `query`, `page` and `port`. The first one contains the query in F-Logic syntax, which is routed to the inference engine listening on `port`. Finally, the results are written in an HTML template specified via `page`. Explicit queries can be stated by HTML and JavaScript templates as well as the so-called Hyperbolic View Applet (cf. Fig. 1). Besides, queries are implicitly issued while navigating through the portal.

2.3 A General Framework for the Semantic Enrichment of Web Logs

In this section, we propose a formal framework that investigates how a request to a portal and its associated semantics are represented. We use this framework to analyze existing methods, and to introduce our new method. Any kind of semantic Web usage mining consists of three steps:

1. **description of raw data**: the description of the transaction, which is generally a user visit/session, as a set or sequence T of URLs u: $T = \{u_i\}$ or $T = [u_i]$,
2. **mapping**: the mapping of URLs to objects that are meaningful within the application domain (e.g., concepts from an ontology), with $m(u) = O$ describing the mapping of URL u to a set of objects O, and
3. **mining**: the identification of patterns in the set or sequence of these meaningful objects, i.e., in the transformed transaction $T' = \{\bigcup_i O(u_i)\}$ or $T' = [O(u_i)]$.

The set of application objects O that a URL is mapped to may contain only one object, for example, *"home page"*. More often, however, the multi-faceted nature of Web pages is reflected by O being a set of objects. For example, on a given page, a researcher R may describe both herself and her projects *PR*, so in this case that page would be about the application objects $O = \{R, PR\}$. In the following, $O(u_i)$ will be described by $\{o_{ij}\}$ to emphasize that it is a set.

Two central assumptions underly this process: (a) "requesting the URL u" signals "an interest in the object(s) $m(u)$", and (b) the mining algorithm neither destroys existing structure nor creates a spurious structure on the transaction T.

Different realizations of the three steps have been proposed. In the following, we will describe two existing procedures, and contrast them with our new proposal.

2.4 Existing Approaches to the Semantic Enrichment of Web Logs

This section describes two existing proposals for adding semantics to Web log files.

Information Extraction. Mobasher et al. (e.g., [19,8]) treat sessions as sets of URL requests. They map URLs to sets of concepts, where each concept is a term extracted from a Web page's text. This allows them to represent each page as a feature vector, with (in the simplest case) the j^{th} component being "1" if the j^{th} term is present, and "0"

otherwise. A session is defined as a feature vector with the j^{th} component being "1" if the j^{th} term was present in any of the pages visited in that session, and "0" otherwise. Vectors with non-binary weights express the relative importance of a term in a page or session. The authors then apply clustering algorithms (e.g., k-means) on these vectors of term weights to identify similar sessions. These algorithms as well as other methods used for further processing rely on the assumption that the set-based interpretation of both sessions and pages is applicable.

This assumption appears to be justified for the analysis of a number of sites, e.g. information sites like the fictitious movie site or the real-estate site analyzed in [8]. Another good example are portal sites that contain lists of topics and pointers to them. A disadvantage of this approach is that the essentially syntactic method of keyword extraction has some inherent problems in achieving a truly semantic mapping of URLs.

Information Dimensions. Berendt and Spiliopoulou [5,3] treat sessions as sequences of URL requests. They map URLs to sets of concepts, with each concept being one value along one information dimension. For example, a listing of objects from a database may be both an *Events* page (because it is a listing of event names) and an *Entities-in-Germany* page (because it shows only the names of events taking place in Germany). These two values are values along the information dimensions "kind of entity" and "location of entity". Depending on the question of the analysis, one information dimension is then chosen, which means that each visited page is mapped to only one concept. So mining is performed on $T' = [o_{ij}]$, with some j identifying the dimension. This approach identifies each URL with one elementary action, and subsequent mining aims at identifying browsing strategies in this sequential choice of concepts. The analysis in [5,3] shows that some actions can only be understood if sequence is considered. For example, a search for events with a user-specified topic (mapped to the concept "events-by-topic"), followed by a search for events with a user-specified topic and a user-specified project (mapped to the concept "events-by-topic-and-project"), signals a *refinement* of search.

This approach is particularly suited to the analysis of strategic behavior, or more generally decision processes, in a Web site, where a user can be assumed to "bundle" his current interests and intentions into the request for the next page. For example, the approach has been successfully applied to information search and online shopping behavior [5,2,22,17]. A disadvantage of this approach is that the restriction to an analysis dimension may cause the oversight of unexpected structure in user visits.

The classification of URLs themselves, using the URL stem and the query string, is particularly interesting for dynamically generated pages. Dynamically generated pages do not exist before they are requested, so their semantics cannot be fully contained in existing HTML text or annotations.

However, the query string that leads to the generation of a page, together with a possibly existing page template, determines the contents and thus the semantics of a page. An additional determinant is the current content of the database / knowledge base being queried. When focusing on user *interests*, we can ignore this: The query shows what a user *wanted and expected* to see. So query strings are particularly suited for understanding user interests.

The kind of mapping introduced in the previous paragraphs can be said to *aggregate* the query string. In contrast to this, our new approach *decomposes* query strings. This can

be regarded as a combination of the *information extraction* approach described above with the *information dimensions approach*.

2.5 Conceptual User Tracking: Semantic Log Files Created by Meta-Data and Ontology-Based Query Decomposition

Within our exemplary KA^2 portal, we exploit the RDF annotations of static sites to map a URL into the set of ontological entities, i.e. concepts, attributes and/or relations, it deals with. This has the advantage that the semantics are decoupled from URL technicalities, and that the mapping is likely to remain correct across updates because the annotations are within the Web pages and are updated together with them. RDF annotations also ensure a better mapping to a page's semantics than syntax-based methods like keyword extraction.

Dynamically generated pages are mapped to semantics by analyzing their query strings[1]. The advantage of the KA^2 architecture is that it provides independent logging of the full query strings which are expressions in F-logic [13] and thus refer only to concepts and relations in the ontology. For details see [21].

In a first approach, we have treated static and dynamic pages as a set, i.e. $m(u) = \{o_{ij}\}$. In the following, this set will be represented as a feature vector, which contains "1" or a weight if a feature o_{ij} is present in the page, and "0" otherwise. The Semantic Log File thus produced contains, for each request, the time stamp, the URL or query-string, and a feature vector. In addition, a user ID may be contained for non-anonymous sessions. In subsequent mining, we could then use sets or sequences. It would also be possible to restrict the mapping of URLs to a subset of the concepts contained in them, analogously to the information dimensions approach.

The next question to ask is what information the feature vector should contain. In the simplest case, the feature vector only contains the extracted concepts. For example, assume that user 4711 asks for a list of all *Publication*s, querying the inference engine with "FORALL x,y,z <- x:Person[name->>y] AND z:Publication[author->>x]". Then, the entry in the Semantic Log File would look like this[2]:

```
Authenticated UserID Timestamp Querystring Feature Vector
                                           Person,Publication...
   YES        4711    ...       FORALL ...    1,      1,      ...
```

The advantage of this option is its great simplicity, and the ease of transformations. For example, the above notation, which shows requests as vectors of concepts, can easily be transformed into a representation that shows concepts as vectors of requests. This allows the analyst to compute the total number of times a concept has been referred to, and a grouping by users. Using the timestamp information, we could in addition analyze users' sequential and temporal navigation behavior.

[1] Query strings can be recorded in a number of ways; e.g., in the standard Web server log when they are sent in GET requests, or in additional logs when they are sent in POST requests or when – as in our case – a servlet handles the queries.

[2] All other feature vector components contain 0.

However, this method ignores attributes and relations in general. This can lead to imprecise results, particularly when queries address complex concepts with many attributes and relations.

Hence, another alternative would be to extract concepts, attributes, and relations from the query string into a feature vector. The resulting Semantic Log File entry can again contain one line per request. Alternatively, further processing can aggregate this by time intervals, in which the references made to each of the C concepts, the A attributes, and the R relations during that interval are recorded. As before, the references can be coded in binary form (whether the ontological entity was referenced or not) or using weights (for example, the number of occurrences of that entity in the user's requests). References are collected for all U users and for all T time intervals. All requests from anonymous users can be aggregated into one pseudo-user u_{anon}. The resulting matrix can be directly input to several mining algorithms such as clustering:

UserID	Time Int.	Person	Publication	...	name	...	author	...
4711	$t_{4711,1}$	1	1	...	1	...	1	...
4711	$t_{4711,2}$	1	0	...	1	...	0	...
0815	$t_{0815,1}$	0	1	...	0	...	1	...
...								
u_{anon}	t_{anon}	1	1	...	1	...	1	...

2.6 Semantic Analysis and Enrichment of Preprocessing

The simplest way to analyze user behavior is to use the Semantic Log File described in the previous section. To achieve even better results, we can take the concept hierarchy into account, or we can use query subsumption to generalize queries. This can be used either for instant analysis or for further transformation of the feature vector enriching the input for the mining step.

Consideration of the Concept Hierarchy. Taking the preprocessed logging matrix as a basis, generalizations and specializations of the inherent taxonomy can be taken into account. The following example serves as a motivation. Assume that a user issues the query "FORALL x, y <- x:AcademicStaff[name->>y]." resulting in all the staff's names. According to the concept hierarchy in the KA^2 ontology (cf. Fig. 1), the user isn't only interested in *AcademicStaff*, but also in its super-concepts *Employee* and *Person*. By such generalizations, the feature vector will have additional entries that could result in better mining results (see [23,8] for examples of the use of subconcepts and superconcepts to derive more meaningful mining results).

Query Lattice. Finally, the most powerful method is the use of a query lattice. Consider the following example, where Q_1 yields all existent projects and Q_2 only those dealing with Data Mining. Q_2 is more general than Q_1 according to the ϑ-subsumption known from Inductive Logic Programming [20], as its output features an additional attribute. The most general query would be the one retrieving *Project*s with all their attributes and relations.

```
Q1 = FORALL x,y   <- x:Project[title->>y].
Q2 = FORALL x,y,z <- x:Project[title->>y] AND
                     x:Project[subject->>"Data Mining"].
```

It is obvious that the log file will contain many different queries which have to be generalized to detect a certain interest. Thus, the advantage of this approach is that queries as a whole can be generalized. So far, we only considered the concepts, attributes and relations disjointly. With this method, we are able to grasp the user's interest in its most concise semantical way.

3 Using Conceptual User Tracking for Web Usage Mining

In order to use advanced mining schemes like those of Dai and Mobasher [8], we need both a reliable identification of ontological entities and a reliable mapping from URLs to ontological entities. Our approach is advantageous because it ensures that we retain the full information on each user request, and that this information is already described in semantic terms. Thus, both information loss due to incomplete logging, and the possible errors when extracting semantics from syntax are avoided. In addition, an ontology-based site usually provides a cleaner mapping from URLs to semantics because this problem is addressed *during site design*, rather than *during later analysis*.

We expect that these features of Conceptual User Tracking will help improve mining results. It can be utilized across the whole range of mining techniques applied to Web usage logs.

As mentioned above, we implemented our ideas in the KA^2 Portal (cf. Sect. 2.2), and we used the Semantic Log File for a first clustering analysis. The log file ranged from 04-25-2001 to 09-25-2002 and contained 140394 entries. We converted it into the well known arff format from the WEKA system [25], counting 1098 sessions along 22 concepts from the KA^2 ontology. After the transformation of the data with $log(x+1)$ to achieve a distribution more similar to the normal distribution, we applied the EM clustering algorithm and found 4 clusters (containing 88, 529, 246, and 197 sessions, respectively) with the following characteristics.

For each cluster we investigated the number of accesses and the number of accesses to the different concepts and relations of our ontology. We found that cluster 1 contains the visitors with few clicks and cluster 0 the heavy users. Visitors grouped into cluster 2 are interested in *Person* and its specializations. Our preprocessing made use of concept generalization, i.e. if the visitor queried for *Researcher* or other sub-concepts of *Person*, interest in *Person* is automatically added to the feature vector. Visitors grouped into cluster 3 showed particular interest in *Project*s and *Publication*s. We were thus able to identify two specific groups of user interests. This knowledge could be used to improve the navigational structure of the portal.

Besides the EM clustering algorithm, we also applied association rules which resulted in trivial rules only. Most queries within the portal are hidden behind HTML forms and always have the same form. A query like "FORALL x,y,z <- x:Person[name ->>y] AND z:Publication [author->>x]" would associate *Person* with *Publication* for example. We are currently exploring the effects of such "artificial" associations on mining results,

and investigating which ways of mapping query strings to ontological entities are most appropriate for deriving valid results.

Nevertheless, the mining results are promising and we are working on the implementation of our logging system into larger Web sites. We are also working on the integration of a component like [11] which allows a better understanding of the clustering and browsing. We expect that usage records from a larger portal will help us show the benefits of our approach in more detail.

4 Conclusion and Outlook

We have discussed earlier work related to the semantic enrichment of Web log files and outlined some problems encountered by these approaches. We have then given an overview of a prototypical system. Besides, the paper described a case study that applies Conceptual User Tracking to the Semantic Log File of the KA^2 Portal.

The KA^2 Portal is only a prototype with a small number of hits per day. Therefore, we intend to widen our analysis to larger portals with higher traffic in order to fully assess and measure the benefits of our approach. In the future, the Semantic Log File could also be leveraged to provide recommendation and personalization functionalities within SEmantic portALs. Another area of future work is the combination of the methods described here (which rely heavily on manual and thus costly ontology definition and page markup) with (semi-)automatic methods of information extraction (cf. [4]).

Acknowledgements. We would like to thank York Sure, Alexander Maedche, Steffen Staab and Gerd Stumme for their supervision as well as their insightful ideas and comments, and two anonymous reviewers for valuable comments and questions .

References

1. R. Benjamins, D. Fensel, and S. Decker. KA^2: Building ontologies for the internet: A midterm report. *International Journal of Human Computer Studies*, 51(3):687, 1999.
2. B. Berendt. Detail and context in web usage mining: Coarsening and visualizing sequences. In R. Kohavi, B. Masand, M. Spiliopoulou, and J. Srivastava, editors, *WEBKDD 2001 – Mining Web Log Data Across All Customer Touch Points*, pages 1–24. Springer-Verlag, Berlin Heidelberg, 2002.
3. B. Berendt. Using site semantics to analyze, visualize and support navigation. *Data Mining and Knowledge Discovery*, 6:37–59, 2002.
4. B. Berendt, A. Hotho, and G. Stumme. Towards semantic web mining. In *I. Horrocks and J. Hendler (Eds.), The Semantic Web - ISWC 2002. (Proceedings of the 1st International Semantic Web Conference, June 9-12th, 2002, Sardinia, Italy)*, pages 264–278. LNCS, Heidelberg, Germany: Springer, 2002.
5. B. Berendt and M. Spiliopoulou. Analysing navigation behaviour in web sites integrating multiple information systems. *The VLDB Journal*, 9(1):56–75, 2000.
6. T. Berners-Lee. XML 2000 - Semantic Web Talk, December 2000. `http://www.w3.org/2000/Talks/1206-xml2k-tbl/Overview.html`.
7. R. Cooley. *Web Usage Mining: Discovery and Application of Interesting Patterns from Web Data*. PhD thesis, University of Minnesota, Faculty of the Graduate School, 2000.

8. H. Dai and B. Mobasher. Using ontologies to discover domain-level web usage profiles. In *Proceedings of the Second Semantic Web Mining Workshop at PKDD 2001, Helsinki, Finland, August 20, 2002*, 2002.
9. S. Decker, M. Erdmann, D. Fensel, and R. Studer. Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In R. Meersman et al., editors, *Database Semantics*, pages 351–369. Kluwer Academic Publisher, 1999.
10. T. R. Gruber. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands, 1993. Kluwer Academic Publishers.
11. A. Hotho and G. Stumme. Conceptual clustering of text clusters. In *Proceedings of FGML Workshop*, pages 37–45. Special Interest Group of German Informatics Society (FGML — Fachgruppe Maschinelles Lernen der GI e.V.), 2002.
12. H. Kato, T. Nakayama, and Y. Yamane. Navigation analysis tool based on the correlation between contents distribution and access patterns. In *WebKDD Workshop on Web Mining for E-Commerce at the 6th ACM SIGKDD*, pages 95–104, Boston, MA, 2000.
13. M. Kifer and G. Lausen. F-logic: A higher-order language for reasoning about objects, inheritance, and scheme. In J. Clifford, B. G. Lindsay, and D. Maier, editors, *Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data, Portland, Oregon, May 31 - June 2, 1989*, pages 134–146. ACM Press, 1989.
14. O. Lassila and R. Swick. Resource description framework (RDF) model and syntax specification. Technical report, W3C, 1999. `http://www.w3.org/TR/REC-rdf-syntax`.
15. W. Lin, S. Alvarez, and C. Ruiz. Efficient adaptive-support association rule mining for recommender systems. *Data Mining and Knowledge Discovery*, 6:83–105, 2002.
16. A. Maedche, S. Staab, R. Studer, Y. Sure, and R. Volz. SEAL – Tying up information integration and web site management by ontologies. *IEEE-CS Data Engineering Bulletin, Special Issue on Organizing and Discovering the Semantic Web*, March 2002.
17. A. Maedche and V. Zacharias. Clustering ontology-based metadata in the semantic web. In *Proc. of the Joint Conferences ECML and PKDD, Finland, Helsinki*, LNAI. Springer, 2002.
18. B. Mobasher, R. Cooley, and J. Srivastava. Automatic personalization based on web usage mining. *Communications of the ACM*, 43(8):142–151, 2000.
19. B. Mobasher, H. Dai, T. Luo, Y. Sun, and J. Zhu. Integrating web usage and content mining for more effective personalization. In *Proc. of ECWeb*, pages 165–176, Greenwich, UK, 2000.
20. S. Muggleton and L. de Raedt. Inductive logic programming: Theory and methods. *Journal of Logic Programming*, 19(20):629–679, 1994.
21. D. Oberle. Semantic community web portals - personalization. Technical Report 424, University of Karlsruhe, Institute AIFB, 2 2003.
22. M. Spiliopoulou, C. Pohle, and M. Teltzrow. Modelling and mining web site usage strategies. In *Proc. of the Multi-Konferenz Wirtschaftsinformatik, Nürnberg, Germany, Sept. 9-11*, 2002.
23. R. Srikant and R. Agrawal. Mining generalized association rules. In *Proc. 21st VLDB, Zurich, Switzerland, September 1995*, pages 407–419, 1995.
24. J. Srivastava, R. Cooley, M. Deshpande, and P.-N. Tan. Web usage mining: discovery and application of usage patterns from web data. *SIGKDD Explorations*, 1(2):12–23, 2000.
25. I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, 1999.