Semantic Web Mining
State of the Art and Future Directions

Gerd Stumme, Andreas Hotho, Bettina Berendt

Abstract

Semantic Web Mining aims at combining the two fast-developing research areas Semantic Web and Web Mining. This survey analyzes the convergence of trends from both areas: an increasing number of researchers is working on improving the results of Web Mining by exploiting semantic structures in the Web, and they make use of Web Mining techniques for building the Semantic Web. Last but not least, these techniques can be used for mining the Semantic Web itself.

The Semantic Web is the second-generation WWW, enriched by machine-processable information which supports the user in his tasks. Given the enormous size even of today's Web, it is impossible to manually enrich all of these resources. Therefore, automated schemes for learning the relevant information are increasingly being used. Web Mining aims at discovering insights about the meaning of Web resources and their usage. Given the primarily syntactical nature of the data being mined, the discovery of meaning is impossible based on these data only. Therefore, formalizations of the semantics of Web sites and navigation behavior are becoming more and more common. Furthermore, mining the Semantic Web itself is another upcoming application. We argue that the two areas Web Mining and Semantic Web need each other to fulfill their goals, but that the full potential of this convergence is not yet realized. This paper gives an overview of where the two areas meet today, and sketches ways of how a closer integration could be profitable.

Keywords

Web Mining, Semantic Web, Ontologies, Knowledge Discovery, Knowledge Engineering, Artificial Intelligence, World Wide Web.

I. Introduction

The two fast-developing research areas Semantic Web and Web Mining build both on the success of the World Wide Web (WWW). They complement each other well because they each address one part of a new challenge posed by the great success of the current WWW: The nature of most data on the Web is so unstructured that they can only be understood by humans, but the amount of data is so huge that they can only be processed efficiently by machines. The Semantic Web addresses the first part of this challenge by trying to make the data (also) machine-understandable, while Web Mining addresses the second part by (semi-)automatically extracting the useful knowledge hidden in these data, and making it available as an aggregation of manageable proportions.

A. Hotho and G. Stumme are with the Knowledge and Data Engineering Group, University of Kassel, D-34121 Kassel, Germany. E-mail: {hotho, stumme}@cs.uni-kassel.de . B. Berendt is with the Institute of Information Systems, Humboldt University Berlin, Spandauer Str. 1, D-10178 Berlin, Germany. E-mail: berendt@wiwi.hu-berlin.de .

Semantic Web Mining aims at combining the two areas Semantic Web and Web Mining. This vision follows our observation that trends converge in both areas: increasing numbers of researchers work on improving the results of Web Mining by exploiting (the new) semantic structures in the Web, and make use of Web Mining techniques for building the Semantic Web. Last but not least, these techniques can be used for mining the Semantic Web itself. The wording *Semantic Web Mining* emphasizes this spectrum of possible interaction between both research areas: it can be read both as *Semantic (Web Mining)* and as *(Semantic Web) Mining*.

In the past few years, there have been many attempts at "breaking the syntax barrier"[1] on the Web. A number of them rely on the semantic information in text corpora that is implicitly exploited by statistical methods. Some methods also analyze the structural characteristics of data; they profit from standardized syntax like XML. In this paper, we concentrate on markup and mining approaches that refer to an *explicit conceptualization* of entities in the respective domain. These relate the syntactic tokens to background knowledge represented in a model with *formal semantics*. When we use the term "semantic", we thus have in mind a formal logical model to represent knowledge.

The aim of this paper is to give an overview of where the two areas of Semantic Web and Web Mining meet today. In our survey, we will first describe the current state of the two areas and then discuss, using an example, their combination, thereby outlining future research topics. We will provide references to typical approaches. Most of them have not been developed explicitly to close the gap between the Semantic Web and Web Mining, but they fit naturally into this scheme.

In the next two sections, we give brief overviews of the areas Semantic Web and Web Mining. Readers familiar with these areas can skip Section II or Section III, resp. We then go on to describe how these two areas cooperate today, and how this cooperation can be improved further. First, Web mining techniques can be applied to help creating the Semantic Web. A backbone of the Semantic Web are ontologies, which at present are often hand-crafted. This is not a scalable solution for a wide-range application of Semantic Web technologies. The challenge is to learn ontologies, and/or instances of their concepts, in a (semi-)automatic way. A survey of these approaches is contained in Section IV.

Conversely, background knowledge—in the form of ontologies, or in other forms—can be used to improve the process and results of Web Mining. Recent developments include the mining of sites that become more and more Semantic Web sites and the development of mining techniques that can tap the expressive power of Semantic Web knowledge representation. Section V discusses these various techniques.

In Section VI, we then sketch how the loop can be closed: from Web Mining to the Semantic

[1]This title was chosen by S. Chakrabarti for his invited talk at the ECML/PKDD 2004 conference.

Web and back. We conclude, in Section VII, that a tight integration of these aspects will greatly increase the understandability of the Web for machines, and will thus become the basis for further generations of intelligent Web tools. We also return to the two notions of "semantics" and outline their strengths, weaknesses, and complementarity. Part of this substantially revised and extended survey was presented at the 1st International Semantic Web Conference [16].

II. Semantic Web

The Semantic Web is based on a vision of Tim Berners-Lee, the inventor of the WWW. The great success of the current WWW leads to a new challenge: a huge amount of data is interpretable by humans only; machine support is limited. Berners-Lee suggests to enrich the Web by machine-processable information which supports the user in his tasks. For instance, today's search engines are already quite powerful, but still too often return excessively large or inadequate lists of hits. Machine-processable information can point the search engine to the relevant pages and can thus improve both precision and recall.

For instance, today it is almost impossible to retrieve information with a keyword search when the information is spread over several pages. Consider, e. g., the query for Web mining experts in a company intranet, where the only explicit information stored are the relationships between people and the courses they attended on one hand, and between courses and the topics they cover on the other hand. In that case, the use of a rule stating that people who attended a course which was about a certain topic have knowledge about that topic might improve the results.

The process of building the Semantic Web is currently an area of high activity. Its structure has to be defined, and this structure then has to be filled with life. In order to make this task feasible, one should start with the simpler tasks first. The following steps show the direction where the Semantic Web is heading:

1. Providing a common syntax for machine understandable statements.
2. Establishing common vocabularies.
3. Agreeing on a logical language.
4. Using the language for exchanging proofs.

Berners-Lee suggested a layer structure for the Semantic Web. This structure reflects the steps listed above. It follows the understanding that each step alone will already provide added value, so that the Semantic Web can be realized in an incremental fashion.

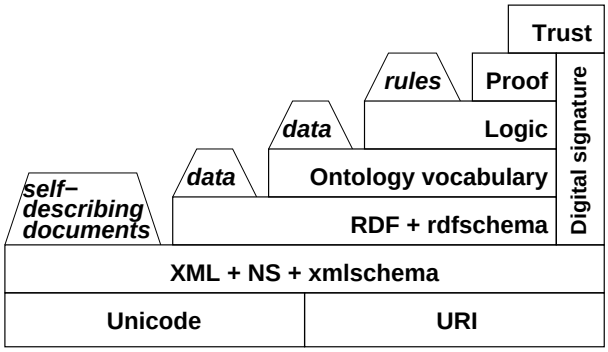

Fig. 1. The layers of the Semantic Web.

A. Layers of the Semantic Web

Figure 1 shows the layers of the Semantic Web as suggested by Berners-Lee.[2] This architecture is discussed in detail for instance in [138] and [139], which also address recent research questions.

On the first two layers, a common syntax is provided. *Uniform resource identifiers (URIs)* provide a standard way to refer to entities,[3] while *Unicode* is a standard for exchanging symbols. The *Extensible Markup Language (XML)* fixes a notation for describing labeled trees, and XML Schema allows the definition of grammars for valid XML documents. XML documents can refer to different *namespaces* to make explicit the context (and therefore meaning) of different tags. The formalizations on these two layers are nowadays widely accepted, and the number of XML documents is increasing rapidly. While XML is one step in the right direction, it only formalizes the structure of a document and not its content.

The *Resource Description Framework (RDF)* can be seen as the first layer where information becomes machine-*understandable*: According to the W3C recommendation[4], RDF "is a foundation for processing metadata; it provides interoperability between applications that exchange machine-understandable information on the Web."

RDF documents consist of three types of entities: resources, properties, and statements. Resources may be Web pages, parts or collections of Web pages, or any (real-world) objects which are not directly part of the WWW. In RDF, resources are always addressed by URIs. Properties are specific attributes, characteristics, or relations describing resources. A resource together with a property having a value for that resource form an RDF statement. A value is either a literal, a resource, or another statement. Statements can thus be considered as object–attribute–value triples.

The middle part of Figure 2 shows an example of RDF statements. Two of the authors of

[2] see `http://www.w3.org/DesignIssues/Semantic.html`

[3] *URL (uniform resource locator)* refers to a locatable URI, e.g., an `http://...` address. It is often used as a synonym, although strictly speaking URLs are a subclass of URIs, see `http://www.w3.org/Addressing`.

[4] `http://www.w3.org/TR/REC-rdf-syntax-grammar-20040210/`

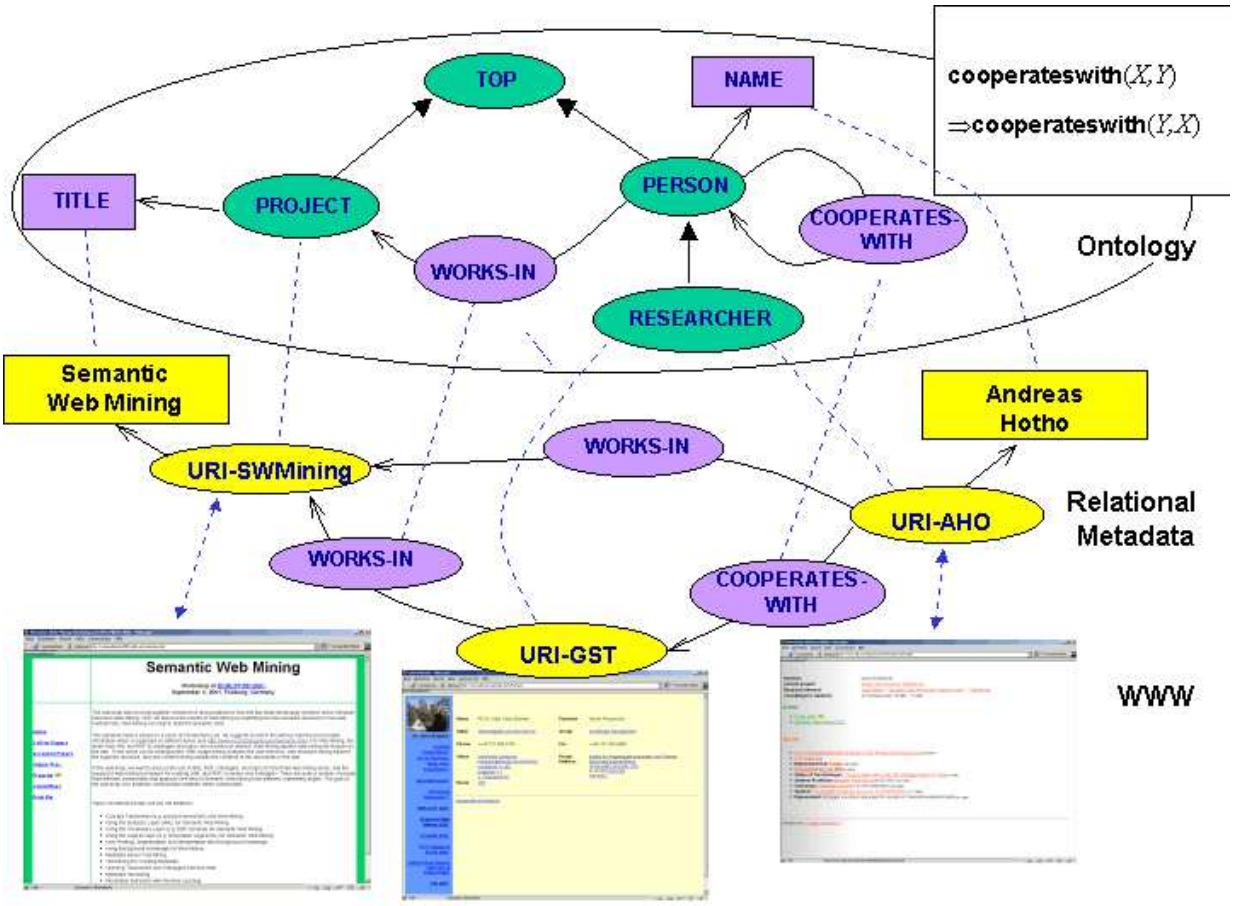

Fig. 2. The relation between the WWW, relational metadata, and ontologies.

the present paper (i. e., their Web pages) are represented as resources 'URI-GST' and 'URI-AHO'. The statement on the lower right consists of the resource 'URI-AHO' and the property 'cooperates-with' with the value 'URI-GST' (which again is a resource). The resource 'URI-SWMining' has as value for the property 'title' the literal 'Semantic Web Mining'.

The data model underlying RDF is basically a directed labeled graph. RDF Schema defines a simple modeling language on top of RDF which includes classes, is-a relationships between classes and between properties, and domain/range restrictions for properties. RDF and RDF Schema are written in XML syntax, but they do not employ the tree semantics of XML.

XML and XML schema were designed to describe the structure of text documents, like HTML, Word, StarOffice, or LaTeX documents. It is possible to define tags in XML to carry meta data but these tags don't have formally defined semantics and thus their meaning will not be well-defined. It is also difficult to convert one XML document to another one without any additionally specified semantics of the used tags. The purpose of XML is to group the objects of content, but not to describe the content. Thus, XML helps organizing documents by providing a formal syntax. This is not 'semantic' in the sense of our survey. Erdmann [53] provides a detailed analysis of the capabilities of XML, the shortcomings of XML concerning semantics, and possible solutions.

The next layer is the *ontology vocabulary*. Following [68], an ontology is "an explicit formalization of a shared understanding of a conceptualization". This high-level definition is realized differently by different research communities. However, most of them have a certain understanding in common, as most of them include a set of *concepts*, a hierarchy on them, and *relations* between concepts. Most of them also include axioms in some specific logic. We will discuss the most prominent approaches in more detail in the next subsection. To give a flavor, we present here just the core of our own definition [161], [23], as it is reflected by the Karlsruhe Ontology framework KAON.[5] It is built in a modular way, so that different needs can be fulfilled by combining parts.

Definition 1: A *core ontology with axioms* is a structure $\mathcal{O} := (\mathcal{C}, \leq_{\mathcal{C}}, \mathcal{R}, \sigma, \leq_{\mathcal{R}}, \mathcal{A})$ consisting of

- two disjoint sets C and R whose elements are called *concept identifiers* and *relation identifiers*, resp.,
- a partial order $\leq_C$ on C, called *concept hierarchy* or *taxonomy*,
- a function $\sigma\colon R \rightarrow C^+$ called *signature* (where C^+ is the set of all finite tuples of elements in C),
- a partial order $\leq_R$ on R, called *relation hierarchy*, where $r_1 \leq_R r_2$ implies $|\sigma(r_1)| = |\sigma(r_2)|$ and $\pi_i(\sigma(r_1)) \leq_C \pi_i(\sigma(r_2))$, for each $1 \leq i \leq |\sigma(r_1)|$, with π_i being the projection on the ith component, and
- a set $\mathcal{A}$ of logical axioms in some logical language $\mathcal{L}$.

This definition constitutes a core structure that is quite straightforward, well-agreed upon, and that may easily be mapped onto most existing ontology representation languages. Step by step the definition can be extended by taking into account lexicons and knowledge bases [161].

As an example, have a look at the top of Figure 2. The set C of concepts is the set {Top, Project, Person, Researcher, Literal}, and the concept hierarchy $\leq_C$ is indicated by the arrows with a filled arrowhead. The set R of relations is the set {works-in, cooperates-with, name, title}. The relation 'works-in' has (Person, Project) as signature, the relation 'name' has (Person, Literal) as signature.[6] In this example, the hierarchy on the relations is flat, i. e., $\leq_R$ is just the identity relation. (An example of a non-flat relation hierarchy will be shown below in Fig. 3.) Up to here, RDF Schema would be sufficient for formalizing the ontology. But often ontologies also contain logical axioms. The one in Figure 2 states for instance that the 'cooperates-with' relation is symmetric. This will be used for inferencing on the logic level.

The objects of the metadata level can now be seen as instances of the ontology concepts. For

[5] `http://kaon.semanticweb.org`

[6] By convention, relations with Literal as range are drawn in this way, because they are in some contexts considered as *attributes*.

example, 'URI-SWMining' is an instance of the concept 'Project', and thus by inheritance also of the concept 'Top'.

Logic is the next layer according to Berners-Lee. Today, most research treats the ontology and the logic levels in an integrated fashion because most ontologies allow for logical axioms. By applying logical deduction, one can infer new knowledge from the information which is stated implicitly. For instance, the axiom given above allows one to logically infer that the person addressed by 'URI-AHO' cooperates with the person addressed by 'URI-GST'. The kind of inference that is possible depends heavily on the logics chosen. We will discuss this aspect in the next subsection in more detail.

Proof and *trust* are the remaining layers. They follow the understanding that it is important to be able to check the validity of statements made in the (Semantic) Web, and that trust in the Semantic Web and the way it processes information will increase in the presence of statements thus validated. Therefore, the author must provide a proof which should be verifiable by a machine. At this level, it is not required that the machine of the reader finds the proof itself, it 'just' has to check the proof provided by the author. These two layers are rarely tackled in today's research. Hence we will focus our interest on the XML, RDF, ontology and logic layers in the remainder of this article.

B. Ontologies: Languages and Tools

A priori, any knowledge representation mechanism[7] can play the role of a Semantic Web language. *Frame Logic* (or *F–Logic*; [98]) is one candidate, since it provides a semantically founded knowledge representation based on the frame-and-slot metaphor. Another formalism that fits well with the structure of RDF are Conceptual Graphs [148], [42]. They also provide a visual metaphor for representing the conceptual structure.

Probably the most popular framework at the moment are Description Logics (DL). DLs are subsets of first order logic which aim at being as expressive as possible while still being decidable. The description logic $\mathcal{SHIQ}$ provides the basis for DAML+OIL, which, in its turn, is a result of joining the efforts of two projects: The DARPA Agent Markup Language DAML[8] was created as part of a research programme started in August 2000 by DARPA, a US governmental research organization. OIL (Ontology Inference Layer) is an initiative funded by the European Union programme. The latest version of DAML+OIL has been released as a W3C Recommendation under the name OWL.[9]

Several tools are in use for the creation and maintenance of ontologies and metadata, as well

[7] See [159] for a general discussion.

[8] `http://www.daml.org`

[9] `http://www.w3.org/TR/owl-features/`

as for reasoning within them. *Ontoedit* [165], [166] is an ontology editor which is connected to *Ontobroker* [58], an inference engine for F–Logic. It provides means for semantics-based query handling over distributed resources. F–Logic has also influenced the development of Triple [147], an inference engine based on Horn logic, which allows the modelling of features of UML, Topic Maps, or RDF Schema. It can interact with other inference engines, for example with FaCT or RACER.

FaCT[10] provides inference services for the Description Language $\mathcal{SHIQ}$. In [83], reasoning within $\mathcal{SHIQ}$ and its relationship to DAML+OIL are discussed. Reasoning is implemented in the FaCT inference engine, which also underlies the ontology editor OilEd [12]. RACER [69] is another reasoner for $\mathcal{SHIQ}$, with emphasis on reasoning about instances.

The Karlsruhe Ontology Framework KAON [23] is an open-source ontology management and learning infrastructure targeted for business applications. It includes a comprehensive tool suite allowing easy ontology creation supported by machine learning algorithm and management, as well as building ontology-based applications. The tool suite is also connected to databases to allow working with a large number of instances. Protégé-2000 [132] is a platform-independent environment for creating and editing ontologies and knowledge bases. Like KAON, it has an extensible plug-in structure. Sesame [91] is an architecture for efficient storage and expressive querying of large quantities of RDF(S) data. It provides support for concurrency control, independent export of RDF(S) information, and a query engine for RQL, a query language for RDF. An extensive overview of ontology tools can be found in [64].

C. Related research areas and application areas

One of the many research areas related to the Semantic Web are databases. In the last few years, most commercial database management systems have included the possibility of storing XML data in order to also accommodate semi-structured data. As the database community has worked on data mining techniques for a long time now, it can be expected that sooner or later 'XML mining' will become an active research topic. Indeed, there are first approaches in that direction [111]. From our point of view, it can be seen as a special case of Semantic Web Mining.

More general, several problems (and solutions) within the database domain are also found within ontology engineering, for instance schema mapping, or the integration of heterogeneous, distributed data sources. This is addressed in more detail in Section IV-A, where we also discuss ways of deriving ontologies from database schemas.

Another related research area are Topic Maps[11] that represent the structure of relationships between subjects. Most of the software for topic maps uses the syntax of XML, just as RDF

[10] `http://www.cs.man.ac.uk/~horrocks/FaCT`
[11] `http://www.topicmaps.org/`

does. In fact, Topic Maps and RDF are closely related. In [8], a formal framework is provided for Topic Maps, which can also be applied to RDF. Semantic Web Mining with Topic Maps has for instance been discussed in [66]. Commercial tools like "theBrain"[12] provide very similar features like named relations, but without an underlying formal semantics.

Different application areas benefit from the Semantic Web and a (re-)organisation of their knowledge in terms of ontologies. Among them are Web Services [57], [56], [140], [136], [27] and Knowledge Management (see [157] for a framework and tool suite, and [108] for an application example). In E-Learning, metadata standards have a long tradition (in particular, Dublin Core[13] and LOM, the Learning Objects Metadata[14]). They are employed in educational portals[15], and a general move towards XML and/or RDF notation can be observed.

Many types of sites can profit from a (re-)organisation as Semantic Web Sites. Knowledge portals[16] provide views onto domain-specific information on the World Wide Web for helping their users to find relevant information. Their maintenance can be greatly improved by using an ontology-based backbone architecture and tool suite, as provided by SEAL [117] and SEAL-II [85].

While metadata are useful on the Web, they are essential for finding resources in peer-to-peer networks. Examples include EDUTELLA [130] (which transfers the educational LOM standard mentioned above to a P2P architecture) and POOL [77].

III. Web Mining

Web mining is the application of data mining techniques to the content, structure, and usage of Web resources. It is thus "the nontrivial process of identifying valid, previously unknown, and potentially useful patterns" [55] in the huge amount of these Web data, patterns that describe them in concise form and manageable orders of magnitude. Like other data mining applications, Web mining can profit from given structure on data (as in database tables), but it can also be applied to semi-structured or unstructured data like free-form text. This means that Web mining is an invaluable help in the transformation from human-understandable content to machine-understandable semantics.

Three areas of Web mining are commonly distinguished: content mining, structure mining, and usage mining [177], [106], [155]. In all three areas, a wide range of general data mining techniques, in particular association rule discovery, clustering, classification, and sequence mining, are employed and developed further to reflect the specific structures of Web resources and

[12] `http://www.thebrain.com/`
[13] `http://dublincore.org`
[14] see `http://ltsc.ieee.org/wg12`
[15] e.g., `http://www.eduserver.de`
[16] An example is `http://www.ontoweb.org`.

the specific questions posed in Web mining. For reasons of space, we will introduce Web content, structure, and usage mining only briefly here; for in-depth overviews of methods and/or applications, see [74], [171], [72], [29], [9].

A. Content/text of Web pages

Web content mining analyzes the content of Web resources. Today, it is mostly a form of text mining (for overviews, see [28], [145]). Recent advances in multimedia data mining promise to widen the access also to image, sound, video, etc. content of Web resources. Multimedia data mining can produce semantic annotations that are comparable to those obtained from text mining; we therefore do not consider this field further (see [146], [175], and the references cited there). The primary Web resources that are mined in Web content mining are individual pages.

Information Retrieval is one of the research areas that provides a range of popular and effective, mostly statistical methods for Web content mining. They can be used to group, categorize, analyze, and retrieve documents, cf. [149] for a survey of IR and [106] for a survey of the relation between IR and Web content mining. These techniques form an excellent basis for more sophisticated approaches. A prime example is Latent Semantic Analysis (LSA) [48]. LSA and other factor-analytic methods have proven valuable for analyzing Web content and also usage, e.g. [26], [92]. However, LSA refers to a looser notion of "semantic"; a lot of effort is needed to identify an explicit conceptualization from the calculated relations.

In addition to standard text mining techniques, Web content mining can take advantage of the semi-structured nature of Web page text. HTML tags and XML markup carry information that concerns not only layout, but also logical structure. Taking this idea further, a "database view" of Web content mining [106] attempts to infer the structure of a Web site in order to transfer it into a database that allows better information management and querying than a pure "IR view".

Web content mining is specifically tailored to the characteristics of text as it occurs in Web resources. Therefore, it focuses on the discovery of patterns in large document collections, and in frequently changing document collections. An application is the detection and tracking of topics [5]. This can serve to detect critical events (that become reflected as a new topic in the evolving document corpus) and trends that indicate a surge or decline in interest in certain topics.

Further content mining methods which will be used for Ontology learning, mapping and merging ontologies, and instance learning are described in Section IV-A. In section VI, we will further set them in relation to the Semantic Web.

B. Structure between Web pages

Web structure mining usually operates on the hyperlink structure of Web pages (for a survey,

see [29]). Mining focuses on sets of pages, ranging from a single Web site to the Web as a whole. Web structure mining exploits the additional information that is (often implicitly) contained in the structure of *hyper*text. Therefore, an important application area is the identification of the relative relevance of different pages that appear equally pertinent when analyzed with respect to their content in isolation.

For example, hyperlink-induced topic search [100] analyzes hyperlink topology by discovering authoritative information sources for a broad search topic. This information is found in *authority* pages, which are defined in relation to *hubs*: Hubs are pages that link to many related authorities. Similarly, the search engine Google[17] owes its success to the PageRank algorithm, which states that the relevance of a page increases with the number of hyperlinks to it from other pages, and in particular from other relevant pages [135].

Web structure mining and Web content mining are often performed together, allowing to exploit simultaneously the content and the structure of hypertext. Indeed, some researchers subsume both under the notion of Web content mining [39].

C. Usage of Web pages

In *Web usage mining*, mining focuses on records of the requests made by visitors to a Web site, most often collected in a Web server log [155], [156]. The content and structure of Web pages, and in particular those of one Web site, reflect the intentions of the authors and designers of the pages and the underlying information architecture. The actual behavior of the users of these resources may reveal additional structure.

First, relationships may be induced by usage where no particular structure was designed. For example, in an online catalog of products, there is usually either no inherent structure (different products are simply viewed as a set), or one or several hierarchical structures given by product categories, etc. Mining the visits to that site, however, one may find that many of the users who were interested in product A were also interested in product B. "Interest" may be measured by requests for product description pages, or by the placement of that product into the shopping cart. Such correspondences between user interest in various items can be used for *personalization*, for example by recommending product B when product A has been viewed ("cross-selling/up-selling" in E-commerce), or by treating a visitor according to which "customer segment" his behavior indicates. Examples of algorithms and applications can be found in [127], [112], [104] and in the recommendations made by online bookstores and other online shops.

Second, relationships may be induced by usage where a different relationship was intended [38]. For example, sequence mining may show that many of the users who went from page

[17] `http://www.google.com`

C to page D did so along paths that indicate a prolonged search (frequent visits to help and index pages, frequent backtracking, etc.). This relation between topology and usage may indicate *usability* problems: Visitors wish to reach D from C, but need to search because there is no direct hyperlink [96], or because it is hard to find [19]. These insights can be used to improve the site's information architecture as well as page design.

Third, usage mining may reveal events in the world faster than content mining. Topic detection and tracking can identify events when they become reflected in texts, i.e. in Web authors' writing behaviour. However, information seeking often precedes authoring, and there are more Web users than Web authors. An example is the detection of the onset of epidemics (or the fear of epidemics) in the usage of medical information sites [173], [78]. Pattern monitoring [10] allows the analyst to go beyond the analysis of simple time series and to track evolutions in more complex access patterns like association rules or sequences.

D. Combined approaches

It is useful to combine Web usage mining with content and structure analysis in order to "make sense" of observed frequent paths and the pages on these paths. This can be done by using a variety of methods. Early approaches have relied on pre-built taxonomies [176] and/or on IR-based keyword extraction methods [41]. Many methods rely on a mapping of pages into an ontology; this will be discussed in Sections V-B and VI.

In the following section, we will first look how ontologies and their instances can be learned. We will then go on to investigate how the use of ontologies, and other ways of identifying the meaning of pages, can help to make Web Mining go semantic.

IV. Extracting Semantics from the Web

The effort behind the Semantic Web is to add machine-understandable, semantic annotation to Web documents in order to access knowledge instead of unstructured material. The purpose is to allow knowledge to be managed in an automatic way. Web Mining can help to learn structures for knowledge organization (e. g., ontologies) and to provide the population of such knowledge structures.

All approaches discussed here are semi-automatic. They assist the knowledge engineer in extracting the semantics, but cannot completely replace her. In order to obtain high-quality results, one cannot replace the human in the loop, as there is always a lot of tacit knowledge involved in the modeling process [24]. A computer will never be able to fully consider background knowledge, experience, or social conventions. If this were the case, the Semantic Web would be superfluous, since then machines like search engines or agents could operate directly on conventional

Web pages. The overall aim of our research is thus not to replace the human, but rather to provide her with more and more support.

A. Semantics created by Content and Structure

A.1 Ontology Learning

Extracting an ontology from the Web is a challenging task. One way is to engineer the ontology by hand, but this is expensive. In [116], the expression *Ontology Learning* was coined for the semi-automatic extraction of semantics from the Web. There, machine learning techniques were used to improve the ontology engineering process and to reduce the effort for the knowledge engineer. An example is given in Section VI.

Ontology learning exploits many existing resources including texts, thesauri, dictionaries, and databases (see [134] as an example of the use of WordNet). It builds on techniques from Web content mining, and it combines machine learning techniques with methods from fields like information retrieval [113] and agents [170], applying them to discover the 'semantics' in the data and to make them explicit. The techniques produce intermediate results which must finally be integrated in a machine-understandable format, e. g., an ontology. Mining can supplement existing (Web) taxonomies with new categories (cf. [4] for an extension of Yahoo[18]), and it can help build new taxonomies [105].

A growing number of sites deliver pages that are generated dynamically in an interaction of an underlying database, information architecture, and query capabilities. For many sites and analysis questions an ontology can be compiled from internal sources such as database schemas, query options, and transaction models. This "reverse engineering" typically involves a large amount of manual work, but it can be aided by (semi-)automatic ontology learning schemes. For example, many retailing and information sites have similarly structured product catalogs [19], [152]. Thus, a tourism site may contain the URL stems `search_hotel.html`, `search_yacht_club.html`, `...` which allows the deduction of the product categories `hotel`, `yacht_club`, etc.[19]

A.2 Mapping and Merging Ontologies

The growing use of ontologies leads to overlaps between knowledge in a common domain. Domain-specific ontologies are modeled by multiple authors in multiple settings. These ontologies lay the foundation for building new domain-specific ontologies in similar domains by

[18] `http://www.yahoo.com`

[19] This is part of a running example, to be used throughout the paper, describing a fictitious tourism Web site. It is based on the Getess project (`http://www.getess.de/index_en.html`), which provides ontology-based access to tourism Web pages for the German region Mecklenburg-Vorpommern (`http://www.all-in-all.de`).

assembling and extending multiple ontologies from repositories.

The process of *ontology merging* takes as input two (or more) source ontologies and returns a merged ontology. Manual ontology merging using conventional editing tools without support is difficult, labor-intensive, and error-prone. Therefore, several systems and frameworks for supporting the knowledge engineer in the ontology merging task have recently been proposed [88], [32], [131], [121]. These approaches rely on syntactic and semantic matching heuristics which are derived from the behavior of ontology engineers confronted with the task of merging ontologies. Another method is FCA-MERGE, which operates bottom-up and offers a global structural description of the process [163]. It extracts instances of source-ontology concepts from a given set of domain-specific text documents by applying natural language processing techniques. Based on the extracted instances, it uses the TITANIC algorithm [164] to compute a concept lattice. The concept lattice provides a conceptual clustering of the concepts of the source ontologies. It is explored and interactively transformed into the merged ontology by the ontology engineer.

Ontology mapping is the assignment of the concepts of one ontology and their instances to the concepts of another ontology. This could be useful, for example, when one of several ontologies has been chosen as the right one for the task at hand. The instances can simply be classified from scratch into the target ontology; alternatively, the knowledge inherent in the source ontology can be utilized by relying on the heuristic that instances from one source concept are likely to also be classified together in one concept of the target ontology [178].

An alternative to merging/mapping ontologies is to simply collect them in parallel and to select the right one according to the task at hand. This vision of a 'corpus of representations' is presented in [71], which opens a new domain of interesting research questions.

A.3 Instance Learning

Even if ontologies are present and users manually annotate new documents, there will still be old documents containing unstructured material. In general, the manual markup of every produced document is impossible. Also, some users may need to extract and use different or additional information from the one provided by the creator. To build the Semantic Web, it is therefore essential to produce automatic or semi-automatic methods for extracting information from Web-related documents as instances of concepts from an ontology, either for helping authors to annotate new documents or for extracting additional information from existing unstructured or partially structured documents.

A number of studies investigate the use of content mining to enrich existing conceptualizations behind a Web site. For example, in [126], Mladenic used text categorization techniques to assign HTML pages to categories in the Yahoo hierarchy. This can reduce the manual effort for

maintaining the Yahoo Web index.

Information Extraction from texts (IE) is one of the most promising areas of Natural Language Technologies (see, e. g., [43]). IE is a set of automatic methods for locating important facts in electronic documents for subsequent use. IE techniques range from the extraction of keywords from pages' text using the *tf.idf* method known from Information Retrieval, via techniques that take the syntactic structures of HTML or natural language into account, to techniques that extract with reference to an explicitly modeled target structure such as an ontology (for a survey, see [107]).

Information Extraction is the perfect support for knowledge identification and extraction from Web documents as it can — for example — provide support in documents analysis either in an automatic way (unsupervised extraction of information) or in a semi-automatic way (e. g., as support for human annotators in locating relevant facts in documents, via information highlighting). One such system for IE is FASTUS [81]. Another system is GATE.[20] With the rise of the Semantic Web, it has been extended to ontology support, and in particular for instance learning [21]. The OntoMat Annotizer [75] has been developed directly for the Semantic Web. It complements IE with authoring functionality. The approach of Craven et al. [44] is discussed in Section VI. In [79], [80], machine learning techniques have been used for the semi-automatic annotation of Web services.

A.4 Using existing conceptualizations as ontologies and for automatic annotation

For many sites, an explicit domain model for the generation of Web pages already exists. These existing formalizations can be (re-)used for semantic markup and mining.

For example, many Content Management Systems generate Web pages from a product catalog, at URLs that reflect the path to the product in the catalog hierarchy. In the running example, this might lead to URLs like `Hotels/WellnessHotels/BeachHotel.html` (similar URLs can be found in popular Web indices). Classification by product hierarchy is a commonly used technique for Web usage mining, see e.g. [154], [7], [59] and the KDD Cup 2000 dataset available for testing algorithms.[21] Alternatively, pages may be generated from a full-blown ontology and its inference engine [133], [125]. The adaptation of this basic idea to dynamic URLs is described in Section V-B.1.

To achieve a common ontology and markup scheme, pages can be generated centrally by one application server. In the case of distributed authorship, the use of the common ontology can be ensured by interactive tools that help individual authors to mark up their pages. This has proven

[20] `http://gate.ac.uk/`

[21] `http://www.ecn.purdue.edu/KDDCUP`

to be a successful strategy for developing community-based portals.[22]

Another way of using existing information is described in [76]: "Deep annotation" derives mappings between information structures from databases. These mappings are used for querying semantic information stored in the database underlying the Web site. This combines capabilities of conventional Web page annotation and automatic Web page generation from databases.

A.5 Semantics created by structure

As we have discussed in Section III-B, the results of the analysis of Web page linkage by Web usage mining create a certain kind of knowledge, a ranking of relevance. Another kind of knowledge that may be inferred from structure is a similarity between pages, useful for the popular browser application "Find similar pages" (to one that has been retrieved by browsing or search): Based on the observation that pages which are frequently cited together from other pages are likely to be related, Dean and Henzinger [47] propose two algorithms for finding similar pages based on hyperlink structure. These techniques structure the set of pages, but they do not classify them into an ontology.

In contrast, the hyperlink structure within pages lends itself more directly to classification. Cooley, Mobasher, and Srivastava [40], based on [141], propose an ontology of page functions, where the classification of a single page with respect to this ontology can be done (semi)-automatically. For example, "navigation" pages designed for orientation contain many links and little information text, whereas "content" pages contain a small number of links and are designed to be visited for their content. This can be used to compare intended usage with actual usage [38]. For example, a content page that is used as a frequent entry point to a site signals a challenge for site design: First, the *intended* entry point, which is probably the home page, should be made better-known and easier to locate. Second, additional links for navigation could be provided on the page that is currently the *actual* entry point. Its content may become a candidate for a new top-level content category on various "head" pages.

The structure of within-page markup may also help in extracting page content: concentrating on page segments identified by reference to the page's DOM (document object model, or tag tree) can serve to identify the main content of a page [29, pp. 228ff.] and to separate it from "noise" like navigation bars, advertisements, etc. [172].

B. Semantics created by Usage

The preceding discussion has implicitly assumed that content exists independently of its usage. However, a large proportion of knowledge is socially constructed. Thus, navigation is not only

[22] See `http://www.ontoweb.org` and `http://www.eduserver.de`

driven by formalized relationships or the underlying logic of the available Web resources. Rather, it "is an information browsing strategy that takes advantage of the behavior of like-minded people" ([33, p.18]). Recommender systems based on "collaborative filtering" have been the most popular application of this idea. In recent years, the idea has been extended to consider not only ratings, but also Web usage as a basis for the identification of like-mindedness ("People who liked/bought this book also looked at ..."; cf. Section III-C and [94] for a classic application).

Extracting such relations from usage can be interpreted as a kind of ontology learning, in which the binary relation "is related to" on pages (and thus concepts) is learned. Can usage patterns reveal further relations to help build the Semantic Web? This field is still rather new, so we will only describe an illustrative selection of research approaches.

Ypma and Heskes [174] propose a method for learning content categories from usage. They model navigation in terms of hidden Markov models, with the hidden states being page categories, and the observed request events being instances of them. Their main aim is to show that a meaningful page categorization may be learned simultaneously with the user labeling and inter-category transitions; semantic labels (such as "sports pages") must be assigned to a state manually. The resulting taxonomy and page classification can be used as a conceptual model for the site, or used to improve an existing conceptual model.

Chi et al. [35], [34] identify frequent paths through a site. Based on the keywords extracted from the pages along the path, they compute the likely "information scent" followed, i.e. the intended goal of the path. The information scent is a set of weighted keywords, which can be inspected and labeled more concisely by using an interactive tool. Thus, usage creates a set of information goals users expect the site to satisfy.[23] These goals may be used to modify or extend the content categories shown to the users, employed to structure the site's information architecture, or employed in the site's conceptual model.

Stojanovic, Maedche, Motik, and Stojanovic [158] propose to measure user interest in a site's concepts by the frequency of accesses to pages that deal with these concepts. They use these data for *ontology evolution*: extending the site's coverage of high-interest concepts, and deleting low-interest concepts, or merging them with others.

The combination of implicit user input (usage) and explicit user input (search engine queries) can contribute further to conceptual structure. User navigation has been employed to infer topical relatedness, i.e. the relatedness of a set of pages to a topic as given by the terms of a query to a search engine ("collaborative crawling" [2]). A classification of pages into "satisfying the user defined predicate" and "not satisfying the predicate" is thus learned from usage, structure, and

[23] An empirical validation showed that this kind of content analysis does indeed group paths that have the same information goal [36].

content information. An obvious application is to mine user navigation to improve search engine ranking [93], [97].

Many approaches use a combination of content and usage mining to generate recommendations. For example, in content-based collaborative filtering, textual categorization of documents is used for generating pseudo-rankings for every user-document pair [122]. In [137], ontologies, IE techniques for analyzing single pages, and a user's search history together serve to generate recommendations for query improvement in a search engine.

V. Using Semantics for Web Mining and Mining the Semantic Web

Semantics can be utilized for Web Mining for different purposes. Some of the approaches presented in this section rely on a comparatively *ad hoc* formalization of semantics, while others can already exploit the full power of the Semantic Web. The Semantic Web offers a good basis to enrich Web Mining: The types of (hyper)links are now described explicitly, allowing the knowledge engineer to gain deeper insights in Web structure mining; and the contents of the pages come along with a formal semantics, allowing her to apply mining techniques which require more structured input. Because the distinction between the use of semantics for Web mining and the mining of the Semantic Web itself is all but sharp, we will discuss both in an integrated fashion.

The first major application area is content mining, i. e., the explicit encoding of semantics for mining the Web content. The hyperlinks and anchors in a page are part of that page's text, and in a semantically marked-up page they are page elements in the same way that text is. So content and structure are strongly intertwined (the two fields are sometimes treated as one [39]). In the Semantic Web, the distinction between content and structure mining disappears completely, as the content of the page is explicitly turned into the structure of the annotation. However, it should be noted that the distribution of the semantic annotations within a page and across pages may provide additional implicit knowledge.

A. Content and Structure Mining

In [84], ontologies are used as background knowledge during preprocessing, with the aim of improving clustering results. We preprocess the input data (e. g., text) and apply ontology-based heuristics for feature selection and feature aggregation. Based on these representations, we compute multiple clustering results using k-Means. Using the ontology, we can select the result which is most approporiate to our task at hand. In [87], we demonstrate the improvement in clustering that arises from the use of WordNet for preprocessing the Reuters corpus. An analogous study showed improvements in classification [20].

Another current project aims at facilitating the customized access to courseware material which

is stored in a peer-to-peer network[24] by means of conceptual clustering. We employ techniques from Formal Concept Analysis, which have been applied successfully in the Conceptual Email Manager CEM [37]. CEM provides an ontology-based search hierarchy of concepts (clusters) with multiple search paths. A combination of this approach with text clustering and a visualization method for analyzing the results are presented in [86].

Knowledge-rich approaches in automatic text summarization (cf. [118], [119], [89]) aim at maximizing the information within a minimal amount of resulting text. They are closely related to Web content mining using semantics because in both Web content mining and text summarization, natural language text needs to be mapped into an abstract representation. This abstract is often represented in some logic, and it is used to improve the results of text summarization. We expect that techniques for automatic text summarization will play an important role in Semantic Web Mining.

Web structure mining can also be improved by taking content into account. The PageRank algorithm mentioned in Section III co-operates with a keyword analysis algorithm, but the two are independent of one another. So PageRank will consider any much-cited page as 'relevant', regardless of whether that page's content reflects the query. By also taking the hyperlink anchor text and its surroundings into account, CLEVER [30] can more specifically assess the relevance for a given query. The Focused Crawler [31] improves on this by integrating topical content into the link graph model, and by a more flexible way of crawling. The learning Intelligent Crawler [3] extends the Focused Crawler, allowing predicates that combine different kinds of topical queries, keyword queries, or other constraints on the pages' content or meta-information (e.g., URL domain). Ontology-based focused crawling is proposed by [114].

An important group of techniques which can easily be adapted to Semantic Web content/structure mining are the approaches discussed as *(Multi-)Relational Data Mining* (formerly called *Inductive Logic Programming* / *ILP*) [51]. Relational Data Mining looks for patterns that involve multiple relations in a relational database. It comprises techniques for classification, regression, clustering, and association analysis. The algorithms can be transformed to deal with data described in RDF or by ontologies. A starting point for such transformations is described in [67] that analyzes different logics and develops a new knowledge representation format closely related to Horn logic, one of the logics that are common in ILP. Making Relational Data Mining amenable to Semantic Web Mining faces two major challenges. The first is the size of the datasets to be processed and the second is the distribution of the data over the Semantic Web. The scalability to huge datasets has always been a major concern for ILP algorithms. With the expected growth of the Semantic Web, this problem increases as well. Therefore, the performance of the mining

[24] `http://edutella.jxta.org`

algorithms has to be improved by methods like sampling (e.g., [144]). To process distributed data, algorithms have to be developed that perform mining in a distributed manner, such that instead of whole datasets, only (intermediate) results have to be transmitted.

B. Usage Mining

Web usage mining benefits from including semantics into the mining process for the simple reason that the application expert as the end user of mining results is interested in *events in the application domain*, in particular user behavior, while the data available—Web server logs—are technically oriented sequences of *HTTP requests*.[25] A central aim is therefore to map HTTP requests to meaningful units of application events.

In this section, we will first introduce a framework for the modeling of user behavior, and then discuss how this background knowledge is used in mining. To illustrate the framework, we will use it to describe a number of existing studies of Web usage. We will concentrate on the semantic aspects of the framework. The studies we describe use a number of different syntactical conventions for representing the semantics; we expect that in the future, XML-based (and thus syntactically standardized) notations will allow a better exchange and re-use of these models [143], [101].

B.1 Application events

Application events are defined with respect to the application domain and the site, a non-trivial task that amounts to a detailed formalization of the site's business/application model (for details, see [17]). For example, relevant E-business events include product views and product click-throughs in which a user shows specific interest in a specific product by requesting more detailed information (e.g., from the Beach Hotel to a listing of its prices in the various seasons). Related events include click-throughs to a product category (e.g., from the Beach Hotel, to the category of All Wellness Hotels), click-throughs from a banner ad, shopping cart changes, and product purchases or bids.

These events are examples of what we call *atomic application events*; they generally correspond to a user's request for one page(view). They can be characterized by their content (e.g., the Beach Hotel, or more generally All Wellness Hotels or All Hotels, see Fig. 3) and the service requested when this page is invoked (e.g., the "search hotels by location" function) [19]. One page may be mapped to one or to a set of application events. For example, it may be mapped to

[25]Note that this discussion assumes that some other issues affecting data quality, e.g., the assignment of requests to users and/or sessions, have either been solved or do not affect the inferences based on the semantics of the requested Web pages. This is an idealization, see [18] for an investigation of the effect of sessionization heuristics on mining results. The use of application server logs can help to circumvent some of these problems [102]. In the following discussion, we also assume that other standard preprocessing steps have been taken [40].

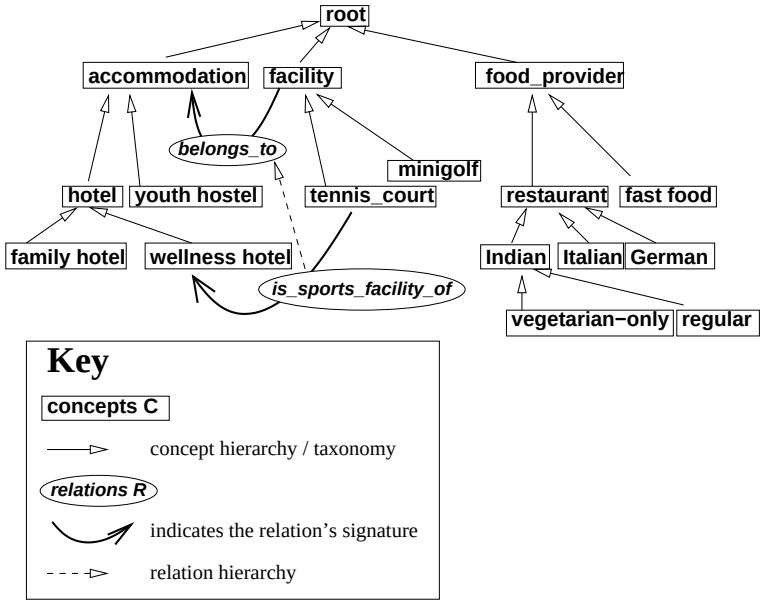

Fig. 3. Parts of the ontology of the content of a fictitious tourism Web site.

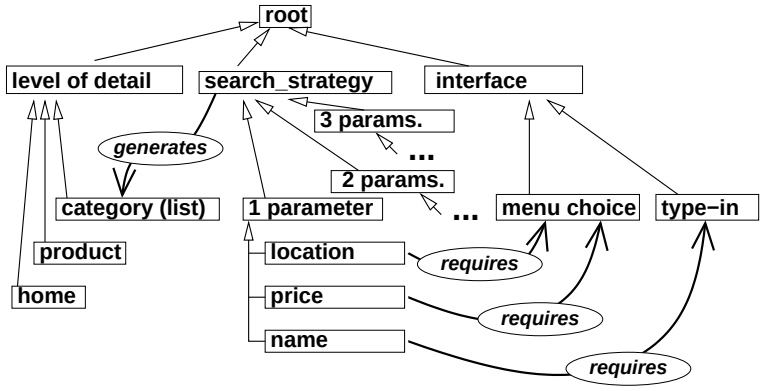

Fig. 4. Parts of the ontology of the services of the fictitious example site.

the set of all concepts and relations that appear in its querystring [133]. Alternatively, keywords from the page's text and from the pages linked with it may be mapped to a domain ontology, with a general-purpose ontology like WordNet serving as an intermediary between the keywords found in the text and the concepts of the ontology [52].

Figure 4 shows a service ontology for the fictitious example site, modeled after the one used in a real-world example in [19]. The site shows accommodation-related information at different levels of detail: as a home (or start) page, on product category pages (lists of hotels or facilities), and on invidividual product pages. Search strategies consist of the specification of one or more of the parameters location, price, and name. Parameters and their values are specified by choice from a menu or by typing. In response, the server generates a category page with all hotels or facilities that satisfy the given specifications.

Atomic application events are usually part of larger meaningful units of activities in the site, which we call *complex application events*. Examples include (a) "directed buying events" in which a user enters an E-commerce store, searches for a product, views that product, puts it into

the shopping cart and purchases it, and then leaves, or (b) "knowledge building events", in which a user browses and searches repeatedly through categories and product views, and usually leaves without purchasing (but may use the knowledge built to return and acquire something later) [153]. Complex application events are usually described by regular expressions whose alphabet consists of atomic application events [153], or by an order structure on atomic application events [14].

B.2 How is knowledge about application events used in mining?

Once requests have been mapped to concepts, the transformed data are ready for mining. We will investigate the treatment of atomic and of complex application events in turn.

In many applications (cf. the examples in Figures 3 and 4), concepts partake in multiple taxonomies. Mining using multiple taxonomies is related to OLAP data cube techniques: objects (in this case, requests or requested URLs) are described along a number of dimensions, and concept hierarchies or lattices are formulated along each dimension to allow more abstract views (cf. [176], [99], [90], [160]).

Taxonomic abstraction is often essential for generating meaningful results: First, in a site with dynamically generated pages, each individual page will be requested so rarely that no regularities may be found in an analysis of navigation behavior. Rather, regularities may exist at a more abstract level, leading to rules like "people who stay in Wellness Hotels also tend to eat in restaurants". Second, patterns mined in past data are not helpful for applications like recommender systems when new items are introduced into product catalog and/or site structure: The new Pier Hotel cannot be recommended simply because it was not in the tourism site until yesterday and thus could not co-occur with any other item, be recommended by another user, etc. A knowledge of regularities at a more abstract level could help to generate a recommendation of the Pier Hotel because it is a Wellness Hotel, and there are criteria for recommending Wellness Hotels.

After the preprocessing steps in which access data have been mapped into taxonomies, subsequent mining techniques can use these taxonomies statically or dynamically. In static approaches, mining operates on concepts at a chosen level of abstraction; each request is mapped to exactly one concept or exactly one set of concepts (see the above examples). This approach is usually combined with interactive control of the software, so that the analyst can re-adjust the chosen level of abstraction after viewing the results (e.g., in the miner WUM; see [19] for a case study). When the investigated complex application events have a sequential structure, sequence mining is required. This is usually the case in investigations of searching, shopping, etc. strategies, as the examples above show.

In dynamic approaches, algorithms identify the most specific level of relationships by choosing concepts dynamically. This may lead to rules like "People who stay in Wellness Hotels tend to

eat at vegetarian-only Indian restaurants"—linking hotel-choice behavior at a comparatively high level of abstraction with restaurant-choice behavior at a comparatively detailed level of description.

For example, Srikant and Agrawal [154] search for associations in given taxonomies, using support and confidence thresholds to guide the choice of level of abstraction. The subsumption hierarchy of an existing ontology is also used for the simultaneous description of user interests at different levels of abstraction, and this description is used to guide the association rule and clustering algorithms in methods that link Web pages to an underlying ontology in a more fine-grained and flexible way [133], [52], [125]. When an explicit taxonomy is missing, mining can provide aggregations towards more general concepts [45].

Semantic Web Usage Mining for complex application events involves two steps of mapping requests to events. As discussed in Section V-B.1 above, complex application events are usually defined by regular expressions in atomic application events (at some given level of abstraction in their respective hierarchies). Therefore, in a first step, URLs are mapped to atomic application events at the required level of abstraction. In a second step, a sequence miner can then be used to discover sequential patterns in the transformed data. The shapes of sequential patterns sought, and the mining tool used, determine how much prior knowledge can be used to constrain the patterns identified. They range from largely unconstrained first-order or k-th order Markov chains [22], via combinations of Markov processes and content taxonomies for a data-driven modelling of content [1], to regular expressions that specify one or a set of atomic activities [150], [11].

Examples of the use of regular expressions describing application-relevant courses of events include search strategies [19], a segmentation of visitors into customers and non-customers [152], and a segmentation of visitors into different interest groups based on the customer buying cycle model from marketing [153].

To date, few commonly agreed-upon models of Semantic Web behavior exist. The still largely exploratory nature of the field implies that highly interactive data preparation and mining tools are of paramount importance: They give the best support for domain experts working with analysts to contribute their background knowledge in an iterative mining cycle. A central element of interactive tools for exploration is visualization. In the STRATDYN tool [14], [13], we propose a semantic Web usage visualization that enables the analyst to detect visual patterns that can be interpreted in terms of application domain behaviors.

With the increasing standardization of many Web applications, and the increasing confluence of mining research with application domain research (e.g., marketing), the number of standard courses of events is likely to grow. Examples are the predictive schemes of E-commerce sites (see the example from [124] mentioned in Section V-B.1 above), and the description of browsing

strategies given by [128].

The representational power of models that capture user behaviour only in terms of a sequence of states identified by page requests is limited. In the future, we expect more explorations of the meaning of viewing time (e.g., [60], [11]) and of the transitions between states [14].

In the analysis and evaluation of user behavior, it must be kept in mind that different stakeholders have different perspectives on the usage of a site, which leads them to investigate different processes (complex application events) and also makes them consider different user actions 'correct' or 'valuable'. Recently, frameworks have been proposed for capturing different processes [110], [168], [6] and perspectives [123].

In summary, a central challenge for future research in Semantic Web Usage Mining lies in the development, provision, and testing of ontologies of application events.

VI. Closing the Loop

In the previous two sections, we have analyzed how to establish Semantic Web data by data mining, how to exploit formal semantics for Web Mining, and how to mine the Semantic Web. In this section, we sketch one out of many possible combinations of these approaches. The example shows how different combinations of Semantic Web and Web Mining can be arranged in a feedback loop.

Our goal is to take a set of Web pages from a site and to improve them for both human and machine users: (a) to generate metadata that reflect a semantic model underlying the site, (b) to identify patterns both in the pages' text and in their usage, and, based on these insights, to improve information architecture and page design. To achieve these goals, we will proceed through several steps in which we

- employ mining methods on Web resources to generate semantic structure (steps 1 and 2: learning and filling the ontology),
- employ mining methods on the resulting semantically structured Web resources to generate further structure (steps 3 and 4),
- at the end of each step, feed these results back into the content and design of the Web pages themselves (visible to human users) and/or their metadata and the underlying ontology (visible to machine users).

We will only give a rough sketch in order to illustrate our ideas, using the running example of the fictitious tourism Web site used throughout this paper.

One may split the first step, *ontology learning*, into two sub-steps. First a concept hierarchy is established using the OTK methodology for modeling ontologies [167]. It may be supported by the formal ontology modeling method OntEx (Ontology Exploration, [62]) which relies on the

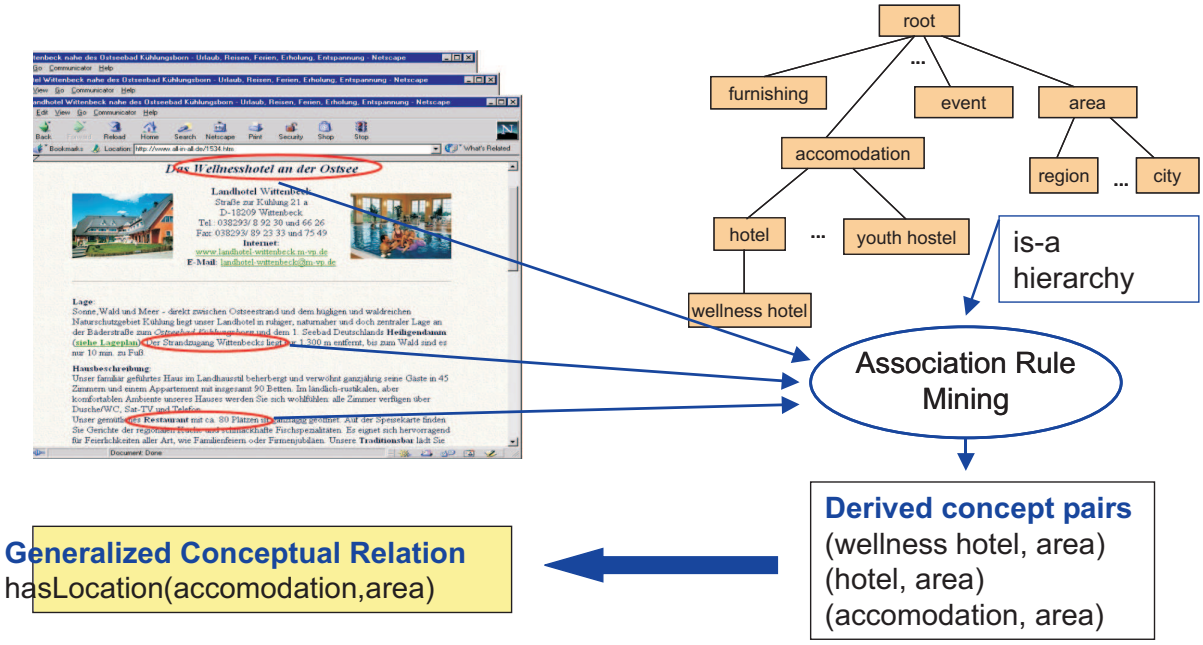

Fig. 5. Step 1: Mining the Web for learning ontologies.

knowledge acquisition technique of Attribute Exploration [61] as developed in the mathematical framework of Formal Concept Analysis [63]; and guarantees that the knowledge engineer considers all relevant combinations of concepts while establishing the subsumption hierarchy. ONTEX takes as input a set of concepts, and provides as output a hierarchy on them. This output is then the input to the second sub-step, together with a set of Web pages. Maedche and Staab [115] describe how association rules are mined from this input, which lead to the generation of relations between the ontology concepts (see Fig. 5). The association rules are used to discover combinations of concepts which frequently occur together. These combinations hint at the existence of conceptual relations. They are suggested to the analyst. As the system is not able to automatically generate names for the relations, the analyst is asked to provide them.

In the example shown in the figure, automatic analysis has shown that three concepts frequently co-occur with the concept “area”. Since the ontology bears the information that the concept “wellness hotel” is a subconcept of the concept “hotel”, which in turn is a subconcept of “accommodation”, the inference engine can derive that only one conceptual relation needs to be inferred based on these co-occurrences: the one between “accommodation” and “area”. Human input is then needed to specify a meaningful name such as “hasLocation” for the generalized conceptual relation.

In the second step, *the ontology is filled*. In this step, instances are extracted from the Web pages, and the relations from the ontology are established between them using techniques described in [44] (see Fig. 6), or any other technique described in Section IV-A.3. Beside the ontology, the approach needs tagged training data as input. Given this input, the system learns to

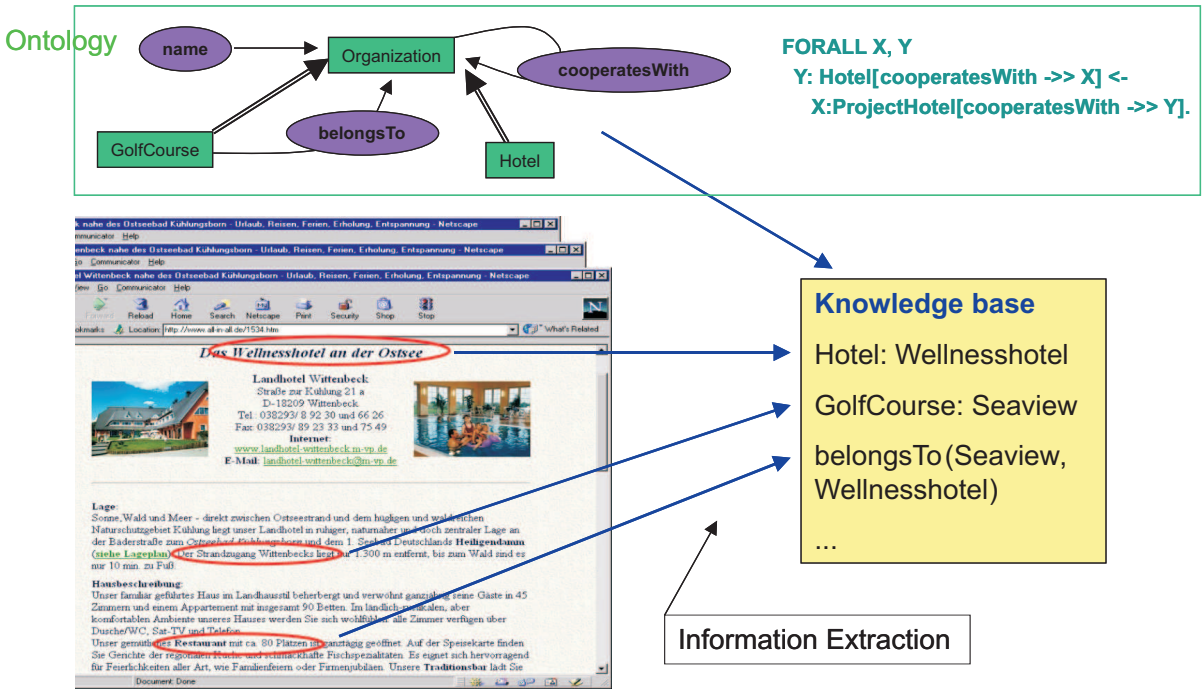

Fig. 6. Step 2: Mining the Web for filling the ontology.

extract instances and relations from other Web pages and from hyperlinks.

In the example shown in the figure, the relation "belongsTo" between the concepts "golf course" and "hotel" is instantiated by the pair (SeaView, Wellnesshotel), i. e., by the fact derived from the available Web pages that the golf course named "SeaView" belongs to the "Wellness Hotel".

After the second step, we have an ontology and a knowledge base, i. e., instances of the ontology concepts and relations between them. These data are now input to the third step, in which *the knowledge base is mined*. Depending on the purpose, different techniques may be applied. One can for instance compute relational association rules, as described in detail in [49] (see Fig. 7). Another possibility is to conceptually cluster the instances [164].

In the example shown in Fig. 7, a combination of knowledge about instances like the Wellnesshotel and its SeaView golf course, with other knowledge derived from the Web pages' texts, produces the rule that hotels with golf courses often have five stars. More precisely, this holds for 89 % of hotels with golf courses, and 0.4 % of all hotels in the knowledge base are five star hotels owning a golf course. The two values are the rule's confidence and support, standard measures for mining association rules.

The resulting semantic structure can now be used to better understand usage patterns. In our example, the clustering of user sessions may identify a cluster of users who visit and closely examine the pages of the "Wellnesshotel", the "Schlosshotel", and the "Hotel Mecklenburg". While this information, on its own, may be sufficient to generate a dynamic recommendation "You might

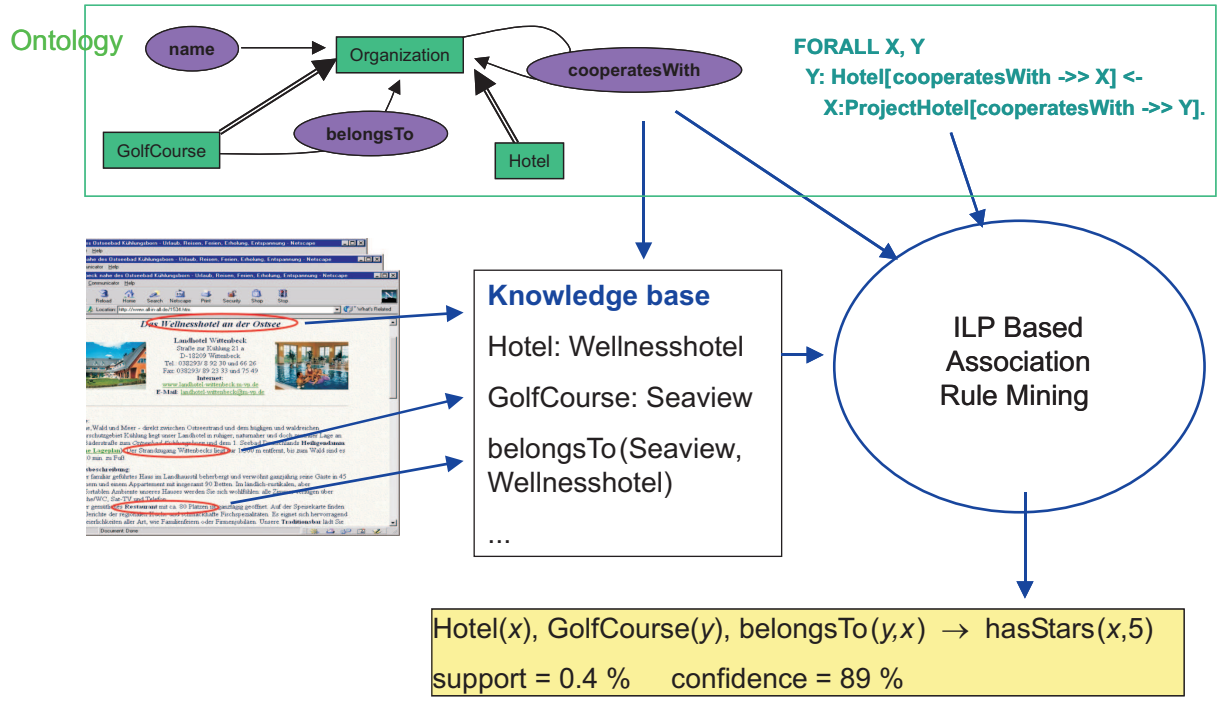

Fig. 7. Step 3: Using the ontology for mining again.

want to also look at the Castle Hotel at the Lake" for new users who visit the "Wellnesshotel" and the "Hotel Mecklenburg", it remains unclear *why* this cluster of hotels may be interesting to a sizeable group of users. This problem can be solved by using our ontology to compute *domain-level usage profiles* [45]: We find that all these hotels are characterized by having a golf course.

This understanding of usage patterns can help us to achieve our initial goal (b), the generation of recommendations for site re-design. We propose to introduce a new category "golf hotels" into both the site's ontology and its information architecture and page design. Instance learning for this category is simple: All "hotels" for which there is a "golf course" that "belongs to" the hotel, and only these, become instances of the new category. Site and page design could, for example, be modified by adding a new value "golf hotel" for the search criterion "hotel facilities" in the site's search/browse navigation bar. Also, when new hotels with golf courses are entered into the knowledge base, these may be dynamically recommended to visitors of the pages of the "Wellnesshotel", the "Schlosshotel", and the "Hotel Mecklenburg".

Our initial goal (a), the generation of a semantic model and metadata that reflect this model, has also been achieved. Among other benefits, this allows a page of the site that describes the "Fischerhotel" in the town of "Zingst" to be returned in answer to a a search engine query for "accommodation in Ahrenshoop", because "hotels" are known to be a subclass of "accommodation" and the towns "Ahrenshoop" as well as "Zingst" are known to be located on the "Fischland-Darß" peninsula. The former piece of knowledge is taken from our ontology (see Fig. 5); the latter is

retrieved by the search engine from a general-purpose geography ontology available from another semantically enriched site.

As we have seen, the results of steps 3 and 4 may lead to further modifications of the ontology and/or knowledge base. When new information is gained, it can be used as input to the first steps in the next turn of the site and ontology life cycle.

Of course, ultimate quality control, the decision to maintain or drop concepts from the ontology, and the transformation of ideas obtained from interpreting usage patterns into site design changes, remain a human responsibility. Nonetheless, in the achievement of both our initial goals, the combination of Semantic Web and Web mining methods has saved a considerable amount of manual effort that is necessary when both goals are worked on in isolation: the work of creating and instantiating an ontology of tourism facilities from a huge number of dynamic page templates, database schemas, and raw HTML, as well as the work of interpreting patterns of co-occurring URLs found in user sessions.

Semantic Web Mining and other feedback loops. The feedback loop described in this section shares a number of features with approaches to discovering knowledge from the Web that relies on a looser notion of semantics. A prime example of the latter is the recently proposed KNOWITALL system [54]. It is based on a bootstrapping approach similar to the one we have described in this paper: Instances of concepts and relations are extracted from the Web, and the reliability of these instances is then judged by the amount of support that these assertions receive from the Web. Due to the size of the Web, fully automated approaches like this one seem to be the premier route for gaining instantaneous access to the knowlegde implicit in the whole Web, in particular its fast-changing, ad-hoc parts.

However, such syntax-based systems rely on the massive redundancy of the Web and can therefore gain access only to information that can be found in a large number of Web pages (and that can be identified by the necessarily limited natural-language templates used for information extraction). In the nineties, Voorhees [169] claimed that, as a matter of principle, Artificial Intelligence (and in particular NLP) is inapt to provide significantly better IR results than such pure syntactical approaches. Since then, however, progress has been made. For instance, [129] won with significant distance the TREC contest in the ‘open domain question answering task’ in 2002 by combining NLP with logical knowledge representation and reasoning.

We therefore expect a preference for Semantic-Web-Mining solutions when the knowledge sought must cover as many (or all) information items available and cannot rely on the redundancy and the “majority votes” implicit in mining schemes like KNOWITALL or PageRank. Because of the extra work required at least by authors, participants in suitable application areas should be

dedicated to information quality, a dedication induced by high intrinsic or extrinsic motivation. Application contexts with a higher-than-usual tolerance for being observed by usage mining will benefit from the added advantages of Semantic Web usage mining. Prominent examples of application areas that exhibit this combination of features are science (where exhaustive literature lists are important), voluntary communities joined by common interests, and business (where transaction costs have to be minimized). Currently, systems for leveraging these application areas' need for coverage and information quality range from WWW-based architecture proposals [142] via operational P2P networks [70] to long-established frameworks that might profit tremendously from being transferred from proprietary systems to the open architecture of the (Semantic) Web (for instance the EDI system of modeling business transactions[26]).

Besides coverage and quality, the form of semantics described in this paper has two further advantages that make it suitable for high-commitment domains. Both advantages derive from the differences in opaqueness between syntactical and semantic information processing approaches. First, the information processing of statistical methods that operate exclusively on syntactic tokens remains opaque to most human users, in particular when proprietary algorithms are employed. There is usually no way to explain, in user-understandable terms, why an algorithm arrived at a particular result. In contrast, an explicit conceptualization enables people and programs to explain, reason, and argue about meaning and thus rationalize their trust, or lack of trust, in a system. Second, their relative opaqueness forces purely statistical-syntactical methods to rely on the individual user's sense-making abilities. Experience shows that users do make sense of the results, but generally in an ad hoc manner that does not encourage reflection or externalization. In contrast, Semantic Web Mining supports the development of principled feedback loops that consolidates the knowledge extracted by mining into information available for the Web at large.

VII. Conclusion and Outlook

In this paper, we have studied the combination of the two fast-developing research areas Semantic Web and Web Mining. We discussed how Semantic Web Mining can improve the results of Web Mining by exploiting the new semantic structures in the Web; and how the construction of the Semantic Web can make use of Web Mining techniques. The example provided in the last section shows the potential benefits of further research in this integration attempt.

Further investigating this interplay will give rise to new research questions and stimulate further research both in the Semantic Web and in Web Mining—towards the ultimate goal of Semantic Web Mining: "a better Web" for all of its users, a "better usable Web". One important focus is

[26]EDI (Electronic Data Interchange) is a standard format for exchanging business data, internationally standardized in ISO 9735.

to enable search engines and other programs to better understand the content of Web pages and sites. This is reflected in the wealth of research efforts that model pages in terms of an ontology of the content, the objects described in these pages.

We expect that, in the future, Web mining methods will increasingly treat content, structure, and usage in an integrated fashion in iterated cycles of *extracting* and *utilizing* semantics, to be able to understand and (re)shape the Web. Among those iterated cycles, we expect to see a productive complementarity between those relying on semantics in the sense of the Semantic Web, and those that rely on a looser notion of semantics.

REFERENCES

[1] S. Acharyya and J. Ghosh. Context-sensitive modeling of web-surfing behaviour using concept trees. In *Proc. of the WebKDD Workshop on Web Mining and Web Usage Analysis*, pages 1–8, 2003.

[2] C.C. Aggarwal. Collaborative crawling: Mining user experiences for topical resource discovery. In *[73]*, pages 423–428, 2002.

[3] C.C. Aggarwal, F. Al-Garawi, and P.S. Yu. Intelligent crawling on the world wide web with arbitrary predicates. In *Proceedings of the WWW Conference*, 2001.

[4] C.C. Aggarwal, S.C. Gates, and P.S. Yu. On the merits of building categorization systems by supervised clustering. In *KDD'1999 – Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 352–356, 1999.

[5] J. Allan, editor. *Topic Detection and Tracking: Event-based Information Organization*. Kluwer Academic Publishers, Norwell, MA, 2002.

[6] S.S. Anand, M. Mulvenna, and K. Chevalier. On the deployment of web usage mining. In *[15]*, pages 23–42. 2004.

[7] C.R. Anderson, P. Domingos, and D.S. Weld. Relational Markov models and their application to adaptive web navigation. In *[73]*, pages 143–152, 2002.

[8] Pascal Auillans, Patrice Ossona de Mendez, Pierre Rosenstiehl, and Bernard Vatant. A formal model for topic maps. In *[82]*, pages 69–83, 2002.

[9] P. Baldi, P. Frasconi, and P. Smyth, editors. *Modeling the Internet and the Web. Probabilistic Methods and Algorithms*.

[10] S. Baron and M. Spiliopoulou. Monitoring the evolution of web usage patterns. In *[15]*, pages 181–200. 2004.

[11] M. Baumgarten, A.G. Büchner, S.S. Anand, M.D. Mulvenna, and J.G. Hughes. User-driven navigation pattern discovery from internet data. In *[151]*, pages 74–91. 2000.

[12] Sean Bechhofer, Ian Horrocks, Carole Goble, and Robert Stevens. OilEd: A reason-able ontology editor for the semantic Web. *LNCS*, 2174:396ff, 2001.

[13] B. Berendt. Detail and context in web usage mining: Coarsening and visualizing sequences. In *[104]*, pages 1–24. 2002.

[14] B. Berendt. Using site semantics to analyze, visualize and support navigation. *Data Mining and Knowledge Discovery*, 6(1):37–59, 2002.

[15] B. Berendt, A. Hotho, D. Mladenic, M. van Someren, M. Spiliopoulou, and G. Stumme, editors. *Web Mining: From Web to Semantic Web. First European Web Mining Forum, EWMF 2003. Invited and Selected Revised Papers*, volume 3209 of *LNAI*. Springer, Berlin, 2004.

[16] B. Berendt, A. Hotho, and G. Stumme. Towards semantic web mining. In *[82]*, pages 264–278, 2002.

[17] B. Berendt, A. Hotho, and G. Stumme. Usage mining for and on the semantic web. In *[95]*. 2003.

[18] B. Berendt, B. Mobasher, M. Nakagawa, and M. Spiliopoulou. The impact of site structure and user environment on session reconstruction in web usage analysis. In *[120]*, pages 115–129, 2002.

[19] B. Berendt and M. Spiliopoulou. Analys of navigation behaviour in web sites integrating multiple information systems. *The VLDB Journal*, 9(1):56–75, 2000.

[20] Stephan Bloehdorn and Andreas Hotho. Text classification by boosting weak learners based on terms and concepts. In *Proceedings of the Fourth IEEE International Conference on Data Mining*. IEEE Computer Society Press, 2004.

[21] K. Bontcheva, V. Tablan, D. Maynard, and H. Cunningham. Evolving gate to meet new challenges in language engineering. natural language engineering. *Natural Language Engineering*, 10(3/4):349–373, 2004.

[22] J. L. Borges and M. Levene. Data mining of user navigation patterns. In *[151]*, pages 92–111. 2000.

[23] E. Bozsak, M. Ehrig, S. Handschuh, A. Hotho, A. Maedche, B. Motik, D. Oberle, C. Schmitz, S. Staab, L. Stojanovic, N. Stojanovic, R. Studer, G. Stumme, Y. Sure, J. Tane, R. Volz, and V. Zacharias. Kaon - towards a large scale semantic web. In K. Bauknecht, A. Min Tjoa, and G. Quirchmayr, editors, *E-Commerce and Web Technologies, Third International Conference, EC-Web 2002, Proceedings*, volume 2455 of *LNCS*, pages 304–313, Berlin, 2002. Springer.

[24] B. Buchanan. Informed knowledge discovery: Using prior knowledge in discovery programs. In *KDD 2000 – Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Boston, MA, August 20-23, 2000*, page 3, New York, 2000. ACM.

[25] P. Buitelaar, J. Franke, M. Grobelnik, G. Paaß, and V. Svátek, editors. *Proceedings of the Workshop on Knowledge Discovery and Ontologies at ECML/PKDD 2004*, 2004.

[26] W. Buntine, S. Perttu, and V. Tuulos. Using discrete PCA on web pages. In *[65]*, pages 99–110, 2004.

[27] Mark H. Burstein, Jerry R. Hobbs, Ora Lassila, David Martin, Drew V. McDermott, Sheila A. McIlraith, Srini Narayanan, Massimo Paolucci, Terry R. Payne, and Katia P. Sycara. Daml-s: Web service description for the semantic web. In *[82]*, pages 348–363, 2002.

[28] S. Chakrabarti. Data mining for hypertext: A tutorial survey. *SIGKDD Explorations*, 1(2):1–11, 2000.

[29] S. Chakrabarti. *mining the Web*. Morgan Kaufmann, San Francisco, CA, 2003.

[30] S. Chakrabarti, B. Dom, D. Gibson, J. Kleinberg, P. Raghavan, and S. Rajagopalan. Automatic resource compilation by analyzing hyperlink structure and associated text. In *Proceedings of the 7th World-wide Web conference (WWW7),30(1-7)*, pages 65–74, 1998.

[31] S. Chakrabarti, M. van den Berg, and B. Dom. Focused crawling: A new approach to topic-specific web resource discovery. *Computer Networks*, 31:1623–1640, 1999.

[32] Hans Chalupsky. Ontomorph: A translation system for symbolic knowledge. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Seventh International Conference (KR2000)*, pages 471–482, 2000.

[33] C. Chen. *Information Visualisation and Virtual Environments*. Springer, London, 1999.

[34] E.H. Chi, P. Pirolli, K. Chen, and J. Pitkow. Using information scent to model user information needs and actions on the web. In *Proceedings of the ACM CHI 2001 Conference on Human Factors in Computing Systems*, pages 490–497, Amsterdam: ACM Press, 2001.

[35] E.H. Chi, P. Pirolli, and J. Pitkow. The scent of a site: a system for analyzing and predicting information scent, usage, and usability of a web site. In *Proceedings of the ACM CHI 2000 Conference on Human Factors in Computing Systems*, pages 161–168, Amsterdam: ACM Press., 2000.

[36] E.H. Chi, A. Rosien, and J. Heer. Intelligent discovery and analysis of web user traffic composition. In *[120]*, pages 1–15, 2002.

[37] R. Cole and G. Stumme. Cem - a conceptual email manager. In B. Ganter and G. W. Mineau, editors, *Proc. ICCS 2000*, volume 1867 of *LNAI*, pages 438–452. Springer, 2000.

[38] R. Cooley. *Web Usage Mining: Discovery and Application of Interesting Patterns from Web Data*. PhD thesis, University of Minnesota, May 2000.

[39] R. Cooley, B. Mobasher, and J. Srivastava. Web mining: Information and pattern discovery on the world wide web. In *Proceedings of the Ninth IEEE International Conference on Tools with Artificial Intelligence (ICTAI'97)*. IEEE Computer Society, Nov 1997.

[40] R. Cooley, B. Mobasher, and J. Srivastava. Data preparation for mining world wide web browsing patterns. *Journal of Knowledge and Information Systems*, 1(1):5–32, 1999.

[41] R. Cooley, P.-N. Tang, and J. Srivastava. Discovery of interesting usage patterns from web data. In *[151]*, pages 163–182. 2000.

[42] O. Corby, R. Dieng, and C. Hébert. A conceptual graph model for w3c resource description framework. In B. Ganter

and G. W. Mineau, editors, *Conceptual Structures: Logical, Linguistic, and Computational Issues, 8th International Conference on Conceptual Structures, ICCS 2000, Darmstadt, Germany, August 14-18, 2000, Proceedings*, volume 1867 of *LNCS*, pages 468–482. Springer, 2000.

[43] J. Cowie and Y. Wilks. Handbook of natural language processing. chapter Information Extraction. Marcel Dekker, New York, 2000.

[44] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. Learning to construct knowledge bases from the world wide web. *Artificial Intelligence*, 118(1-2):69–113, 2000.

[45] 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*, km.aifb.uni-karlsruhe.de/semwebmine2002/papers/full/bamshad.pdf, Aug 2002.

[46] J. Davies, D. Fensel, and F. van Harmelen, editors. *On-To-Knowledge: Semantic Web enabled Knowledge Management*. J. Wiley and Sons, 2002.

[47] J. Dean and M.R. Henzinger. Finding related pages in the world wide web. In *Proceedings of the Eighth International World Wide Web Conference WWW-1999*, Toronto, May 1999.

[48] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[49] L. Dehaspe and H. Toivonen. Discovery of frequent datalog patterns. *Data Mining and Knowledge Discovery*, 3(1):7–36, 1999.

[50] P. Domingos, C. Faloutsos, T. Senator, H. Kargupta, and L. Getoor, editors. *KDD'2003 – Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2003. ACM.

[51] Saso Dzeroski and Nada Lavrac, editors. *Relational Data Mining*. Springer, 2001.

[52] M. Eirinaki, M. Vazirgiannis, and I. Varlamis. Sewep: Using site semantics and a taxonomy to enhance the web personalization process. In *[50]*, pages 99–108, 2003.

[53] Michael Erdmann. *Ontologien zur konzeptuellen Modellierung der Semantik von XML*. Isbn: 3831126356, University of Karlsruhe, 10 2001.

[54] O. Etzioni, M. Cafarella, D. Downey, A.-M. Popescu, T. Shaked, S. Soderland, D.S. Weld, and A. Yates. Methods for domain-independent information extraction from the web: An experimental comparison. In *Proc. of the Nineteenth National Conference on Artificial Intelligence (AAAI-04)*, pages 391–398, Menlo Park, CA, 2004. AAAI/MIT Press.

[55] U.M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pages 1–34. AAAI / MIT Press, Cambridge, MA, 1996.

[56] Alexander Felfernig, Gerhard Friedrich, Dietmar Jannach, and Markus Zanker. Semantic configuration web services in the cawicoms project. In *[82]*, pages 192–205, 2002.

[57] D. Fensel, C. Bussler, and A. Maedche. Semantic web enabled web services. In *[82]*, pages 1–2, 2002.

[58] D. Fensel, S. Decker, M. Erdmann, and R. Studer. Ontobroker in a nutshell. In *European Conference on Digital Libraries*, pages 663–664, 1998.

[59] M. Fernández, D. Fiorescu, A. Levi, and D. Sucin. Declarative specification of web sites with strudel. *The VLDB Journal*, 9:38–55, 2000.

[60] J. Forsyth, T. McGuire, and J. Lavoie. *All visitors are not created equal*. McKinsey Marketing Practice, McKinsey & Company, April 2000, 2000.

[61] B. Ganter. Attribute exploration with background knowledge. *TCS*, 217(2):215–233, 1999.

[62] B. Ganter and G. Stumme. Creation and merging of ontology top-levels. In *(in preparation)*, 2002.

[63] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, Berlin – Heidelberg, 1999.

[64] Asun Gomez-Perez, Juergen Angele, Mariano Fernandez-Lopez, V. Christophides, Athur Stutt, and York Sure. A survey on ontology tools. OntoWeb deliverable 1.3, Universidad Politecnia de Madrid, 2002.

[65] M. Gori, M. Ceci, and M. Nanni, editors. *Proceedings of the Workshop on Statistical Approaches for Web Mining at ECML/PKDD 2004*, 2004.

[66] B. Le Grand and M. Soto. Xml topic maps and semantic web mining. In *[162]*, pages 67–83, 2001.

[67] Benjamin Grosof, Ian Horrocks, Raphael Volz, and Stefan Decker. Description Logic Programs: Combining Logic Programs with Description Logics. In *Proc. of WWW-2003*, Budapest, Hungary, 05 2003.

[68] 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, Netherlands, 1993. Kluwer.

[69] Volker Haarslev and Ralf Moller. Description of the RACER system and its applications. In D. L. McGuinness et al, editor, *Proceedings of the 2001 International Workshop on Description Logics (DL-2001)*. CEUR Workshop Proceedings, 2001.

[70] P. Haase, M. Ehrig, A. Hotho, and B. Schnizler. Personalized information access in a bibliographic peer-to-peer system. In *Proc. of the Semantic Web Personalization Workshop at AAAI'2004*, pages 1–12, 2004.

[71] Alon Y. Halevy and Jayant Madhavan. Corpus-based knowledge representation. In Georg Gottlob and Toby Walsh, editors, *IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 9-15, 2003*, pages 1567–1572. Morgan Kaufmann, 2003.

[72] Han and Kamber. *Data Mining. Concepts and Techniques*. Morgan Kaufmann, San Francisco, LA, 2001.

[73] D. Hand, D. Keim, and R. Ng, editors. *KDD - 2002 – Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2002. ACM.

[74] D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining*. Cambridge, MA: MIT Press, 2001.

[75] Siegfried Handschuh and Steffen Staab. Authoring and annotation of web pages in cream. In *Proceedings of the Eleventh International World Wide Web Conference, WWW2002*, pages 462–473. ACM, 2002.

[76] Siegfried Handschuh, Steffen Staab, and Raphael Volz. On deep annotation. In *Proc. of WWW-2003*, Budapest, Hungary, 05 2003.

[77] Marek Hatala and Griff Richards. Global vs. community metadata standards: Empowering users for knowledge exchange. In *[82]*, pages 292–306, 2002.

[78] J. Heino and H. Toivonen. Automated detection of epidemics from the usage logs of a physicians' reference database. In *[109]*, pages 180–191, 2003.

[79] Andreas Hess, Eddie Johnston, and Nicholas Kushmerick. ASSAM: A tool for semi-automatically annotating web services with semantic metadata. In Sheila A. McIlraith, Dimitris Plexousakis, and Frank van Harmelen, editors, *The Semantic Web - ISWC 2004: Third International Semantic Web Conference*, volume 3298 of *Lecture Notes in Computer Science*, pages 320–334. Springer, 2004.

[80] Andreas Hess and Nicholas Kushmerick. Learning to attach semantic metadata to web services. In *The Semantic Web – Proc. Intl. Semantic Web Conference (ISWC 2003)*, pages 258–273. Springer, 2003.

[81] Jerry Hobbs, Douglas Appelt, John Bear, David Israel, Megumi Kameyama, Mark Stickel, and Mabry Tyson. Fastus: A cascaded finite-state transducer for extracting information from natural-language text. In E. Roche and Y. Schabes, editors, *Finite State Devices for Natural Language Processing*. MIT Press, Cambridge MA, 1996.

[82] I. Horrocks and J. A. Hendler, editors. *The Semantic Web – ISWC 2002, First International Semantic Web Conference, Proceedings*, volume 2342 of *LNCS*. Springer, 2002.

[83] I. Horrocks and S. Tessaris. Querying the web: A formal approach. In *[82]*, pages 177–191, 2002.

[84] A. Hotho, A. Maedche, and S. Staab. Ontology-based text clustering. In *Proceedings of the IJCAI-2001 Workshop "Text Learning: Beyond Supervision", August, Seattle, USA*, 2001.

[85] A. Hotho, A. Maedche, S. Staab, and R. Studer. SEAL-II — the soft spot between richly structured and unstructured knowledge. *Journal of Universal Computer Science*, 7(7):566–590, 2001.

[86] A. Hotho, S. Staab, and G. Stumme. Explaining text clustering results using semantic structures. In *[109]*, pages 217–228, 2003.

[87] A. Hotho, S. Staab, and G. Stumme. Ontologies improve text document clustering. In *Proc. of the ICDM 03, The 2003 IEEE International Conference on Data Mining*, pages 541–544, 2003.

[88] E.H. Hovy. Combining and standardizing large-scale, practical ontologies for machine translation and other uses. In *Proc. 1st Intl. Conf. on Language Resources and Evaluation (LREC)*, Granada, 1998.

[89] M. Hu and B. Liu. Mining and summarizing customer reviews. In *[103]*, pages 695–700, 2004.

[90] J.Z. Huang, M. Ng, W.-K. Ching, J. Ng, and D. Cheung. A cube model and cluster analysis for web access sessions. In *[104]*, pages 48–67. 2002.

[91] Frank van Harmelen Jeen Broekstra, Arjohn Kampman. Sesame: A generic architecture for storing and querying rdf and rdf schema. In *[82]*, pages 54–68, 2002.

[92] X. Jin, Y. Zhou, and B. Mobasher. Web usage mining based on probabilistic latent semantic analysis. In *[103]*, pages 197–205, 2004.

[93] T. Joachims. Optimizing search engines using clickthrough data. In *[73]*, pages 133–142, 2002.

[94] T. Joachims, D. Freitag, and T. Mitchell. Webwatcher: A tour guide for the world wide web. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 770–777, San Francisco, CA, 1997. Morgan Kaufmann.

[95] H. Kargupta, A. Joshi, K. Sivakumar, and Y. Yesha, editors. *Data Mining: Next Generation Challenges and Future Directions*. AAAI/MIT Press, Menlo Park, CA, 2004.

[96] H. Kato, T. Nakayama, and Y. Yamane. Navigation analysis tool based on the correlation between contents distribution and access patterns. In *Working Notes of the Workshop on Web Mining for E-Commerce - Challenges and Opportunities (WebKDD 2000) at KDD 2000*, pages 95–104, Boston, MA, 2000.

[97] C. Kemp and K. Ramamohanarao. Long-term learning for web search engines. In *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD 2002)*, pages 263–274, Berlin, 2002. Springer.

[98] M. Kifer, G. Lausen, and J. Wu. Logical foundations of object-oriented and frame-based languages. *Journal of the ACM*, 42(4):741–843, 1995.

[99] R. Kimball and R. Merx. *The Data Webhouse Toolkit – Building Web-Enabled Data Warehouse*. Wiley Computer Publishing, New York, 2000.

[100] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[101] N. Koeppen, K. Polkehn, and H. Wandke. A toolset to support logfile examinations. In Noldus I.T. AG, editor, *Measuring Behavior 2002, 4th International Cofnerence on Methods and Techniques in Behavioral Research, 27-30 August 2002, Amsterdam*, 2002.

[102] R. Kohavi. Mining e-commerce data: The good, the bad, and the ugly. In *KDD 2001 – Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, August 26-29, 2002*, pages 8–13, New York, 2001. ACM.

[103] R. Kohavi, J. Gehrke, W. DuMouchel, and J. Ghosh, editors. *KDD'2004 – Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2004. ACM.

[104] R. Kohavi, B.M. Masand, M. Spiliopoulou, and J. Srivastava, editors. *WEBKDD 2001 – Mining Web Log Data Across All Customer Touch Points*, volume 2356 of *LNAI*. Springer, Berlin / Heidelberg, 2002.

[105] D. Koller and M. Sahami. Hierarchically classifying documents using very few words. In *Proceedings of the 14th International Conference on Machine Learning (ML), Nashville, Tennessee, July 1997*, pages 170–178, 1997.

[106] R. Kosala and H. Blockeel. Web mining research: A survey. *SIGKDD Explorations*, 2(1), 2000.

[107] A.H.F. Laender, B.A. Ribeiro-Neto, A.S. da Silva, and J.S. Teixeira. A brief survey of web data extraction tools. *SIGMOD Record*, 31(2):84–93, 2002.

[108] T. Lau and Y. Sure. Introducing ontology-based skills management at a large insurance company. In *Proceedings of the Modellierung 2002*, Tutzing, Germany, March 2002.

[109] N. Lavrač, D. Gamberger, L. Todorovski, and H. Blockeel, editors. *Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases: PKDD 2003*, volume 2838 of *LNAI*, Berlin Heidelberg, 2003. Springer.

[110] J. Lee, M. Podlaseck, E. Schonberg, and R. Hoch. Visualization and analysis of clickstream data of online stores for understanding web merchandising. *Data Mining and Knowledge Discovery*, 5(1/2):59–84, 2001.

[111] Jung-Won Lee, Kiho Lee, and Won Kim. Preparations for semantics-based xml mining. In Nick Cercone, Tsau Young Lin, and Xindong Wu, editors, *Proceedings of the 2001 IEEE International Conference on Data Mining, 29 November - 2 December 2001, San Jose, California, USA*, pages 345–352. IEEE Computer Society, 2001.

[112] W. Lin, S.A. Alvarez, and C. Ruiz. Efficient adaptive-support association rule mining for recommender systems. *Data Mining and Knowledge Discovery*, 6(1):83–105, 2002.

[113] A. Maedche. *Ontology Learning for the Semantic Web*. Kluwer, 2002.

[114] A. Maedche, M. Ehrig, S. Handschuh, L. Stojanovic, and R. Volz. Ontology-focused crawling of documents and relational metadata. In *Proceedings of the Eleventh International World Wide Web Conference WWW-2002*, Hawaii, 2002.

[115] A. Maedche and S. Staab. Discovering conceptual relations from text. In *ECAI-2000 – Proceedings of the 13th European Conference on Artificial Intelligence*, pages 321–325. IOS Press, Amsterdam, 2000.

[116] A. Maedche and S. Staab. Ontology learning for the semantic web. *IEEE Intelligent Systems*, 16(2):72–79, 2001.

[117] 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.

[118] I. Mani and M. Maybury. Advances in automatic text summarization. pages 123–136. The MIT Press, 1999.

[119] Inderjeet Mani. *Automatic Summarization*, volume 3 of *Natural Language Processing*. John Benjamins Publishing Company, Amsterdam/Philadelphia, 2001.

[120] B. Masand, M. Spiliopoulou, J. Srivastava, and O.R. Zaïane, editors. *Workshop Notes of the Fourth WEBKDD Web Mining for Usage Patterns & User Profiles at KDD'2002*, Edmonton, Alberta, Canada, July 23 2002. ACM.

[121] D. McGuinness, R. Fikes, J. Rice, and S. Wilder. An environment for merging and testing large ontologies. In *In the Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000)*, pages 483–493, Breckenridge, Colorado, USA, 2000.

[122] P. Melville, R.J. Mooney, and R. Nagarajan. Content-boosted collaborative filtering. In *Proceedings of the ACM SIGIR Workshop on Recommender Systems*, Sep 2001.

[123] E. Menasalvas, S. Millán, M.S. Pérez, E. Hochsztain, and A. Tasistro. An approach to estimate the value of user sessions using multiple viewpoints and goals. In *[15]*, pages 164–180. 2004.

[124] D.A. Menascé, V. Almeida, R. Fonseca, and M.A. Mendes. A methodology for workload characterization of e-commerce sites. In *Proceedings of the ACM Conference on Electronic Commerce*, New York, 1999. ACM.

[125] Rosa Meo, Pier Luca Lanzi, Maristella Matera, and Roberto Esposito. Integrating web conceptual modeling and web usage mining. In *Proc. of the WebKDD Workshop on Web Mining and Web Usage Analysis*, pages 105–115, 2004.

[126] Dunja Mladenic. Turning yahoo to automatic web-page classifier. In *European Conference on Artificial Intelligence*, pages 473–474, 1998.

[127] B. Mobasher, R. Cooley, and J. Srivastava. Automatic personalization based on web usage mining. *Communications of the ACM*, 43(8):142–151, 2000.

[128] W. Moe. Buying, searching, or browsing: Differentiating between online shoppers using in-store navigational clickstream. *Journal of Consumer Psychology*, 13(1&2), 2002.

[129] D. Moldovan, S. Harabagiu, R. Girju, P. Morarescu, F. Lacatusu, A. Novischi, A. Badulescu, and O. Bolohan. Lcc tools for question answering. In Voorhees and Buckland, editors, *Proceedings of the 11th Text REtrieval Conference (TREC-2002)*, NIST, Gaithersburg.

[130] Wolfgang Nejdl, Wolf Siberski, Bernd Simon, and Julien Tane. Towards a modification exchange language for distributed rdf repositories. In *[82]*, pages 236–249, 2002.

[131] N. Noy and M. Musen. Prompt: Algorithm and tool for automated ontology merging and alignment. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, pages 450–455, Austin, Texas, 2000.

[132] N. F. Noy, M. Sintek, S. Decker, M. Crubezy, R. W. Fergerson, and M. A. Musen. Creating semantic web contents with protégé-2000. *IEEE Intelligent Systems*, 16(2):60–71, 2001.

[133] D. Oberle, B. Berendt, A. Hotho, and J. Gonzalez. Conceptual user tracking. In *Web Intelligence, First International Atlantic Web Intelligence Conference, AWIC 2003, Madrid, Spain, May 5-6, 2003, Proceedings*, volume 2663 of *LNCS*, pages 155–164, Berlin, 2003. Springer.

[134] G. Paaß, J. Kindermann, and E. Leopold. Learning prototype ontologies by hierarchical latent semantic analysis. In *[25]*, pages 49–60, 2004.

[135] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. In *Proceedings of the 7th International World Wide Web Conference*, pages 161–172, Brisbane, Australia, 1998.

[136] Massimo Paolucci, Takahiro Kawamura, Terry R. Payne, and Katia P. Sycara. Semantic matching of web services capabilities. In *[82]*, pages 333–347, 2002.

[137] S. Parent, B. Mobasher, and S. Lytinen. An adaptive agent for web exploration based of concept hierarchies. In *Proceedings of the 9th International Conference on Human Computer Interaction*, New Orleans, LA, 2001.

[138] P. Patel-Schneider and D. Fensel. Layering the semantic web: Problems and directions. In *[82]*, pages 16–29, 2002.

[139] P. Patel-Schneider and J. Siméon. Building the semantic web on xml. In *[82]*, pages 147–161, 2002.

[140] Joachim Peer. Bringing together semantic web and web services. In *[82]*, pages 279–291, 2002.

[141] Ramana Rao Peter Pirolli, James Pitkow. Silk from a sow's ear: Extracting usable structures from the web. In *Proc. ACM Conf. Human Factors in Computing Systems, CHI*, pages 118–125, New York, NY, 1996. ACM Press.

[142] D. Pierrakos, G. Paliouras, C. Papatheodorou, V. Karkaletsis, and M. Dikaiakos. Web community directories: A new approach to web personalization. In *[15]*, pages 113–129. 2004.

[143] J.R. Punin, M.S. Krishnamoorthy, and M.J. Zaki. Logml: Log markup language for web usage mining. In *[104]*, pages 88–112. 2002.

[144] Tobias Scheffer and Stefan Wrobel. A sequential sampling algorithm for a general class of utility criteria. In *Knowledge Discovery and Data Mining*, pages 330–334, 2000.

[145] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[146] S.J. Simoff. Variations on multimedia data mining. In S.J. Simoff and O.R. Zaïane, editors, *Proceedings of the MDKM/KDD2000 Workshop on Multimedia Data Mining*, pages 104–109, www.cs.ualberta.ca/ zaiane/mdm_kdd2000/mdm00-15.pdf, 2000.

[147] M. Sintek and S. Decker. Triple — a query, inference, and transformation language for the semantic web. In *[82]*, pages 364–378, 2002.

[148] J. F. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley Publishing Company, Reading, MA, 1984.

[149] K. Sparck-Jones and P. Willett, editors. *Readings in Information Retrieval*. Morgan Kaufmann, 1997.

[150] M. Spiliopoulou. The laborious way from data mining to web mining. *International Journal of Computer Systems, Science, & Engineering*, 14:113–126, 1999.

[151] M. Spiliopoulou and B.M. Masand, editors. *Advances in Web Usage Analysis and User Profiling*, volume 1836 of *LNAI*. Springer, Berlin / Heidelberg, 2000.

[152] M. Spiliopoulou and C. Pohle. Data mining for measuring and improving the success of web sites. *Data Mining and Knowledge Discovery*, 5:85–14, 2001.

[153] M. Spiliopoulou, C. Pohle, and M. Teltzrow. Modelling and mining web site usage strategies. In *Proceedings of the Multi-Konferenz Wirtschaftsinformatik*, Sep 2002.

[154] R. Srikant and R. Agrawal. Mining generalized association rules. In *Proceedings of the 21st International Conference on Very Large Databases*, pages 407–419, Sep 1995.

[155] 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.

[156] J. Srivastava, P. Desikan, and V. Kumar. Web mining – concepts, applications & research directions. In *[95]*. 2003.

[157] S. Staab, H.-P. Schnurr, R. Studer, and Y. Sure. Knowledge processes and ontologies. *IEEE Intelligent Systems, Special Issue on Knowledge Management*, 16(1), January/Febrary 2001.

[158] N. Stojanovic, A. Maedche, B. Motik, and N. Stojanovic. User-driven ontology evolution management. In *Proceedings of the 13^{th} European Conference on Knowledge Engineering and Knowledge Management EKAW'02*, 2002.

[159] R. Studer, V. R. Benjamins, and D. Fensel. Knowledge engineering: Principles and methods. *Data Knowledge Engineering*, 25(1–2):161–197, 1998.

[160] G. Stumme. Conceptual on-line analytical processing. In K. Tanaka, S. Ghandeharizadeh, and Y. Kambayashi, editors, *Information Organization and Databases*, chapter 14, pages 191–203. Kluwer, Boston, MA, 2002.

[161] G. Stumme. Using ontologies and formal concept analysis for organizing business knowledge. In J. Becker and R. Knackstedt, editors, *Wissensmanagement mit Referenzmodellen – Konzepte für die Anwendungssystem- und Organisationsgestaltung*, pages 163–174, Heidelberg, 2002. Physica.

[162] G. Stumme, A. Hotho, and B. Berendt, editors. *Semantic Web Mining*, Freiburg, September 3rd 2001. 12th Europ. Conf.

on Machine Learning (ECML'01) / 5th Europ. Conf. on Principles and Practice of Knowledge Discovery in Databases (PKDD'01).

[163] G. Stumme and A. Maedche. FCA–Merge: Bottom-Up Merging of Ontologies. In *IJCAI-2001 – Proceedings of the 17th International Joint Conference on Artificial Intelligence, Seattle, USA, August, 1-6, 2001*, pages 225–234, San Francisco, 2001. Morgen Kaufmann.

[164] G. Stumme, R. Taouil, Y. Bastide, N. Pasqier, and L. Lakhal. Computing iceberg concept lattices with titanic. *J. on Knowledge and Data Engineering*, 42(2):189–222, 2002.

[165] Y. Sure, M. Erdmann, J. Angele, S. Staab, R. Studer, and D. Wenke. OntoEdit: Collaborative ontology development for the semantic web. In *[82]*, pages 221–235, 2002.

[166] Y. Sure, S. Staab, and J. Angele. OntoEdit: Guiding ontology development by methodology and inferencing. In *Proceedings of the International Conference on Ontologies, Databases and Applications of SEmantics ODBASE 2002*, University of California, Irvine, USA, 2002.

[167] Y. Sure and R. Studer. On-To-Knowledge methodology. In *[46]*, chapter 3, pages 33–46. 2002.

[168] M. Teltzrow and B. Berendt. Web-usage-based success metrics for multi-channel businesses. In *Proc. of the WebKDD Workshop on Web Mining and Web Usage Analysis*, pages 17–27, 2003.

[169] E. Voorhees. Natural language processing and information retrieval. In M.T. Pazienza, editor, *Information Extraction: Towards Scalable, Adaptable Systems*, pages 32–48. Springer, Berlin etc., 1999.

[170] A.B. Williams and C Tsatsoulis. An instance-based approach for identifying candidate ontology relations within a multi-agent system. In *Proceedings of the First Workshop on Ontology Learning OL'2000*, Berlin, Germany, 2000. Fourteenth European Conference on Artificial Intelligence.

[171] I.H. Witten and E. Frank. *Data Mining. Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, San Francisco, CA, 2000.

[172] L. Yi, B. Liu, and X. Li. Eliminating noisy information in web pages for data mining. In *[50]*, pages 296–305, 2003.

[173] G. Yihune. *Evaluation eines medizinischen Informationssystems im World Wide Web*. PhD thesis, Medizinische Fakultät der Ruprecht-Karls-Universität Heidelberg, Germany, 2003.

[174] A. Ypma and T. Heskes. Categorization of web pages and user clustering with mixtures of hidden markov models. In *[120]*, pages 31–43, 2002.

[175] O.R. Zaïane and S.J. Simoff. Mdm/kdd: Multimedia data mining for the second time. *SIGKDD Explorations*, 3(2), 2003.

[176] O.R. Zaïane, M. Xin, and J. Han. Discovering web access patterns and trends by applying olap and data mining technology on web logs. In *Proceedings of Advances in Digital Libraries Conference (ADL'98)*, pages 19–29, Apr 1998.

[177] Osmar R. Zaïane. From resource discovery to knowledge discovery on the internet. Technical Report TR 1998-13, Simon Fraser University, 1998.

[178] D. Zhang and W.S. Leee. Learning to integrate web taxonomies. *Journal of Web Semantics*, 2(2):131–151, 2004.