Ontology research and development. Part 1 - a review of ontology generation

Ying Ding and Schubert Foo

Journal of Information Science 2002; 28; 123
DOI: 10.1177/016555150202800204

Ontology research and development. Part 1 – a review of ontology generation

Ying Ding and Schubert Foo

Vrije Universiteit, Amsterdam and Nanyang Technological University, Singapore

Received 25 June 2001
Revised 8 October 2001

Abstract.

Ontology is an important emerging discipline that has the huge potential to improve information organization, management and understanding. It has a crucial role to play in enabling content-based access, interoperability, communications, and providing qualitatively new levels of services on the next wave of web transformation in the form of the Semantic Web. The issues pertaining to ontology generation, mapping and maintenance are critical key areas that need to be understood and addressed. This survey is presented in two parts. The first part reviews the state-of-the-art techniques and work done on semi-automatic and automatic ontology generation, as well as the problems facing such research. The second complementary survey is dedicated to ontology mapping and ontology 'evolving'. Through this survey, we have identified that shallow information extraction and natural language processing techniques are deployed to extract concepts or classes from free-text or semi-structured data. However, relation extraction is a very complex and difficult issue to resolve and it has turned out to be the main impediment to ontology learning and applicability. Further research is encouraged to find appropriate and efficient ways to detect or identify relations through semi-automatic and automatic means.

Correspondence to: Ying Ding, Division of Mathematics and Computer Science, Vrije Universiteit, Amsterdam, The Netherlands. E-mail: ying@cs.vu.nl

1. Introduction

Ontology is the term referring to the shared understanding of some domains of interest, which is often conceived as a set of classes (concepts), relations, functions, axioms and instances [1]. In the knowledge representation community, the commonly used or highly cited ontology definition is adopted from Gruber [1] where an 'ontology is a formal, explicit specification of a shared conceptualization. "*Conceptualization"* refers to an abstract model of phenomena in the world by having identified the relevant concepts of those phenomena. "*Explicit"* means that the type of concepts used, and the constraints on their use are explicitly defined. "*Formal"* refers to the fact that the ontology should be machine readable. "*Shared"* reflects that ontology should capture consensual knowledge accepted by the communities'.

Ontology is a complex multi-disciplinary field that draws upon the knowledge of information organization, natural language processing, information extraction, artificial intelligence, knowledge representation and acquisition. Ontology is gaining popularity and is touted as an emerging technology that has a huge potential to improve information organization, management and understanding. In particular, when ontology provides a shared framework of the common understanding of specific domains that can be communicated between people and application systems, then it can have a significant impact on areas dealing with vast amounts of distributed and heterogeneous computer-based information, such as those residing on the world wide web and intranet information systems, complex industrial software applications, knowledge management, electronic commerce and e-business. For instance, ontology has played a strategic role for agent communication [2]; ontology mapping has the capability to break the bottleneck of the business-to-business marketplace [3]; and ontology is the enabler

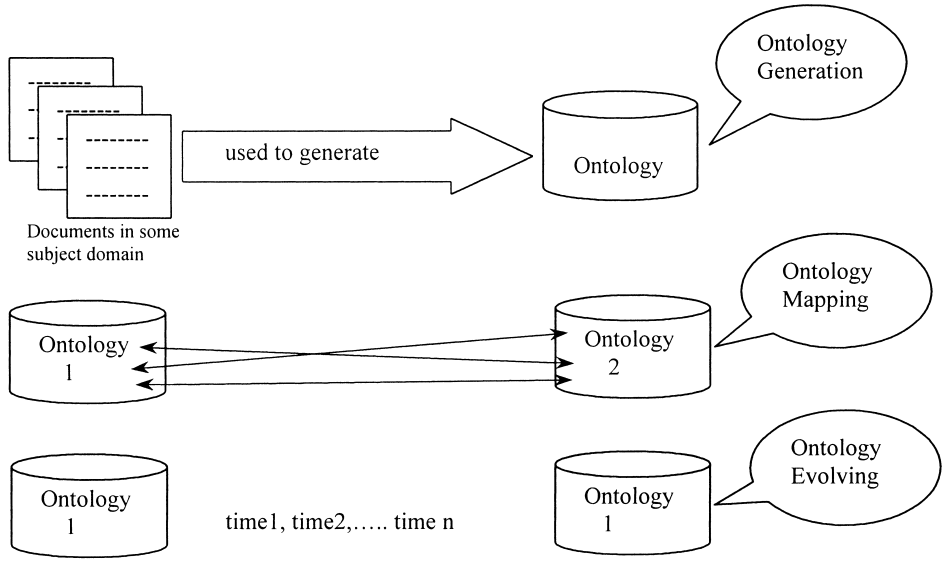

Fig. 1. General overview of ontology generation, mapping and evolving.

to improve intranet-based knowledge management systems [4]. Ontology itself is an explicitly defined reference model of application domains with the purpose of improving information consistency and reusability, systems interoperability and knowledge sharing. It describes the semantics of a domain in both a human-understandable and computer-processable way.

Developments at the World Wide Web Consortium (W3C) indicate that the first generation of the world wide web will make transition in future into the second generation, the Semantic Web [5, 6]. The term 'Semantic Web' was coined by Tim Berners-Lee, the inventor of the world wide web, to describe his vision of the next generation web that provides services that are much more automated based on machine-processable semantics of data and heuristics [7]. Ontologies that provide shared and common domain theories will become a key asset for this to happen. They can be seen as metadata that explicitly represent semantics of data in a machine-processable way. Ontology-based reasoning services can operationalize such semantics and be used for providing various forms of services (for instance, consistency checking, subsumption reasoning, query answering, etc.). Ontologies help people and computers to access the information they need, and effectively communicate with each other. They therefore have a crucial role to play in enabling content-based access, interoperability and communication across the web, providing it with a qualitatively new level of service such as proposed in the Semantic Web [8].

Ontology learning is starting to emerge as a sub-area of ontology engineering due to the rapid increase of web documents and the advanced techniques shared by the information retrieval, machine learning, natural language processing and artificial intelligence communities. The majority of existing ontologies have been generated manually. Generating ontologies in this manner has been the normal approach undertaken by most ontology engineers. However, this process is very time-intensive and error-prone, and poses problems in maintaining and updating ontologies. For this reason, researchers are looking for other alternatives to generate ontologies in more efficient and effective ways. This survey aims to provide an insight into this important emerging field of ontology, and highlights the main contributions of ontology generation, mapping and evolving,* whose inter-relationships are shown in Fig. 1.

The survey is carried out over two parts, namely a state-of-the-art survey on ontology generation and a state-of-the-art survey on ontology mapping and evolving. In this first part of the survey on ontology generation, the areas of semi-automatic or automatic ontology generation will be covered. A subsequent paper will report on ontology mapping and evolving.

*Ontology evolving, although an incorrect use of English language, has become an accepted buzzword in the ontology field, as in ontology mapping and ontology versioning.

2. Ontology generation in general

Although there already exist large-scale ontologies, ontology engineers are still needed to construct the ontology and knowledge base for a particular task or domain, and to maintain and update the ontology to keep it relevant and up-to-date. Manually constructed ontologies are time-consuming, labour-intensive and error-prone. Moreover, a significant delay in updating ontologies, causing currency problems, actually hinders the development and application of the ontologies.

The starting point for creating an ontology could arise from different situations. An ontology can be created from scratch, from existing ontologies (whether global or local ontologies) only, from a corpus of information sources only; or a combination of the latter two approaches [9]. Various degrees of automation could be used to build ontologies, ranging from fully manual through semi-automated to fully automated. At present, the fully automated method only functions well for very lightweight ontologies under very limited circumstances.

Normally, methods to generate an ontology could be summarized as: bottom-up – from specification to generalization; top-down – from generalization to specification (e.g. KACTUS ontology); and middle-out – from the most important concepts to generalization and specialization (e.g. Enterprise ontology and Methodology ontology [10]). Most often, lifting algorithms are used to lift and derive different levels of ontologies from a basic ontology [11].

There are also a number of general ontology design principles that have been proposed by different ontology engineers over a period of time:

- Guarino [12] was inspired by philosophical research and proposed a methodology for ontology design known as 'Formal Ontology' [13]. This design principle contains a theory of parts, a theory of wholes, a theory of identity, a theory of dependence and a theory of universals. He summarized the basic design principles that include the need to: (1) be clear about the domain; (2) take identity seriously; (3) isolate a basic taxonomic structure; and (4) identify roles explicitly.
- Uschold and Gruninger [14] proposed a skeletal methodology for building ontologies via a purely manual process – (1) identify purpose and scope; (2) build the ontology via a three-step process – *ontology capture* (identification of the key concepts and relationships and provision of the definitions of such concepts and relationships); *ontology coding* (committing to the basic terms for ontology (class, entity, relation); choosing a representation language; writing the code); and *integrating existing ontologies*; (3) evaluation [15]; (4) documentation; and (5) guidelines for each of the previous phases. The final resulting ontology should be clear (definitions should be maximally clear and unambiguous), consistent and coherent (an ontology should be internally and externally consistent), extensible and reusable (an ontology should be designed in such a way as to maximize subsequent reuse and extensibility).
- Ontological design patterns (ODPs) [16] were used to abstract and identify ontological design structures, terms, larger expressions and semantic contexts. These techniques can separate the construction and definition of complex expressions from its representation to change them independently. This method was successfully applied in the integration of molecular biological information [16].

Hwang [2] proposed a number of desirable criteria for the final generated ontology to be (1) open and dynamic (both algorithmically and structurally for easy construction and modification), (2) scalable and interoperable, (3) easily maintained (ontology should have a simple, clean structure as well as being modular), and (4) context independent.

The remaining sections highlight the major contributions and projects that have been reported with respect to ontology generation. In each project, an introduction and background is first provided. This is followed by a description of the methods employed and concludes with a summary of problems that have surfaced during the process of ontology generation.

2.1. InfoSleuth (MCC)

InfoSleuth is a research project at MCC (Microelectronics and Computer Technology Corporation) to develop and deploy new technologies for finding information available both in corporate networks and external networks. It focuses on the problems of locating, evaluating, retrieving and merging information in an environment in which new information sources are constantly being added. It is a project aiming to build up the ontology-based agent architecture. It has been successfully implemented in different application areas that include Knowledge Management, Business Intelligence, Logistics, Crisis Management, Genome Mapping, environmental data exchange networks, and so on.

Methods. The procedure for automatic generation of the ontology adopted in InfoSleuth is as follows [2]:

- Human experts provide the system with a small number of seedwords that represent high-level concepts. Relevant documents will be collected from the web (with POS-tagged or otherwise unmarked text) automatically.
- The system processes the incoming documents, extracts phrases containing seedwords, generates corresponding concept terms and places them in the 'right' place in the ontology. At the same time, it also collects candidates for seedwords for the next round of processing. The iteration continues until a set of results is reached.
- Several kinds of relations are extracted – 'is-a', 'part-of', 'manufactured-by', 'owned-by', etc. The 'assoc-with' relation is used to define all relations that are not an 'is-a' relation. The distinction between 'is-a' and 'assoc-with' relations is based on a linguistic property of noun compounds.
- In each iteration, a human expert is consulted to ascertain the correctness of the concepts. If necessary, the expert has the right to make the correction and reconstruct the ontology. Figure 2 shows an example of the structure of a generated ontology.

In Fig. 2, the indentation shows the hierarchy (class and subclass relationships). Thus, 'field emission display', 'flat panel display' and 'display panel' are subclasses of 'display'. Here one obvious rule to generate this hierarchy is that if the phrase has 'display' as the last word in the phrase, it will become the subclass of 'display'. Likewise, the same rule is applied for 'image'. Another rule is that if the phrase has 'display' as the first word in the phrase, this phrase will also become the subclass of the 'display' with the indication of '*', such as 'display panel', 'video image retrieval' and so on.

This system has a number of special features and characteristics:

- Discover-and-alert – the system expands the ontology with new concepts it learns from new documents and alerts the human experts of the changes.
- Attribute-relation-discovery – this approach can discover some of the attributes associated with certain concepts. For instance, the method can discover the attributes of physical dimensions or number of pixels and can even learn the range of their possible values. Based on the linguistic characters, the 'assoc-with' relations can be identified automatically. Since the ontology is organized as hierarchies, attributes are automatically inherited following 'is-a' links.
- Indexing-documents – while constructing the ontology, this method also indexes documents for future retrieval, optionally saving the results in a relational database. It collects 'context lines' for each concept generated, showing how the concept was used in the text, as well as frequency and co-occurrence statistics for word association discovery and data mining.
- This system allows the users to decide between the precision completeness through browsing the ontology and inferences based on the is-a relation and assoc-with relations.

The system uses simple part-of-speech (POS) tagging to conduct superficial syntactic analysis (shallow information extraction techniques). The relationship of the concepts is detected from the linguistic features. As in any other corpus-based approach, the richer and the more complete the data set, the higher will be the reliability of the results achieved as a direct result of the applicability of machine learning techniques.

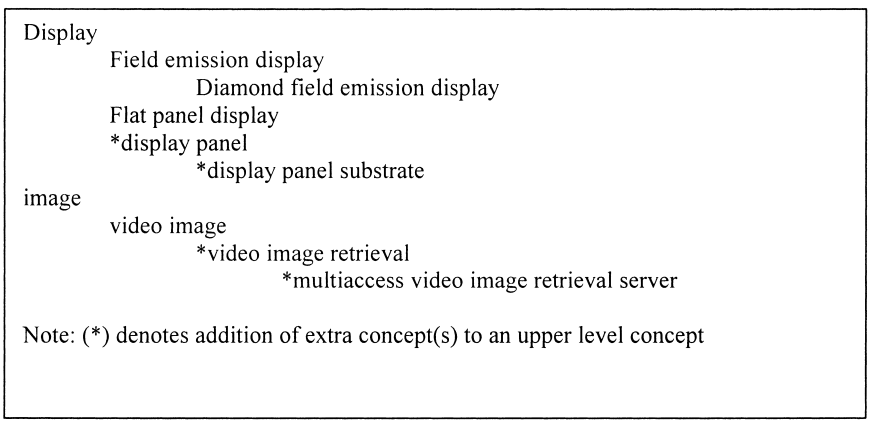

Fig. 2. Example of the automatic generated ontology from InfoSleuth.

Problems encountered. Generating automatic ontologies in this manner involves several challenges and problems such as:

- Syntactic structural ambiguity – a correct structural analysis of a phrase is important because the decision whether to regard a certain subsequence of a phrase as a concept often depends on the syntactic structure. For example, the concept 'image processing software' when extracted sequentially takes the varying forms of image (wrong concept), image processing (wrong concept), and image processing software (right concept). Likewise, 'low-temperature polysilicon TFT panel' becomes 'low-temperature' (wrong concept), 'low-temperature polysilicon' (wrong concept), 'low-temperature polysilicon TFT panel' (right concept).
- Recognition of different phrases that refer to the same concept. For example, 'quartz crystal oscillator' is actually the same as 'crystal quartz oscillator'. One possible solution to this problem is to differentiate them via the co-occurrence frequency of these two phrases.
- Proper attachment of adjectival modifiers is a possible way to avoid creating non-concepts.
- Word sense problem – a possible way to solve this problem is to make reference to some general linguistic ontologies (such as SENSUS or WordNet [17]) so as to disambiguate different word senses (using human involvement to select the word sense from SENSUS or WordNet, or through machine learning techniques to learn the patterns).
- Heterogeneous resources as the data source for generating ontology – terminological inconsistencies are common when document sources are diverse, which is also a common information retrieval problem. A possible way to solve this is to find the synonyms or similar terms from general linguistic ontologies (such as SENSUS or WordNet) or use the co-occurrence techniques to cluster the concepts based on the high similarities to detect the inconsistency.
- The automatically constructed ontology can be too prolific and deficient at the same time. Excessively prolific ontologies could hinder a domain expert's browsing and correction (reasonable choice of seedwords and initial cleaning and training data should limit this risk). On the other hand, automatically generated ontologies could be deficient since they rely on seedwords only. One promising technique could be synonym learning [18].

2.2. SKC (University of Stanford)

The Scalable Knowledge Composition (SKC) project aims to develop a novel approach to resolving semantic heterogeneity in information systems. The project attempts to derive general methods for ontology integration that can be used in any application area so that it is basically application neutral. An ontology algebra has been developed therefore to represent the terminologies from distinct, typically autonomous domains. This research effort is funded by the United States Air Force Office of Scientific Research (AFOSR), with the cooperation of the United States Defense Advanced Research Project Agency (DARPA) High-Performance Knowledge Base (HPKB) program.

In this project, Jannink and Wiederhold [19] and Jannink [20] converted the Webster's dictionary data to a graph structure to support the generation of a domain or task ontology. The resulting text is tagged to mark the parts of the definitions, similar to the XML (eXtensible Markup Language) structure. According to their research purpose, only head words (<hw> . . . </hw>) and definitions (<def> . . . </def>) having many-to-many relationships were considered. This resulted in a directed graph that had the two properties that each head word and definition grouping was a node; and each word in a definition node was an arc to the node having that head word.

Methods. Jannink and Wiederhold [19] did not describe the adopted technique in detail in their publication. However, Jannink [20] mentioned that they used a novel algebraic extraction technique to generate the graph structure and create the thesaurus entries for all words defined in the structure including some stop words (e.g. a, the, and). Ideas from the PageRank algorithm [21] were also adopted. This is a flow algorithm over the graph structure of the WWW that models the links followed during a random browsing session through the web. The ArcRank from the PageRank model was chosen to extract relationships between the dictionary words and the strength of the relationship. The attraction of using the dictionary as a structuring tool is that headwords are terms distinguished from the definition text, which provides the extra information allowing for types of analysis that are not currently performed in traditional data mining and information retrieval. This method could also be applied to document classification and the relevance ranking of mining queries. The basic hypothesis for this work is that structural relationships between terms are relevant to their meaning. The methodology to extract the relations (the important component of ontology) is achieved through

a new iterative algorithm, based on the Pattern/Relation extraction algorithm as follows [22]:

- compute a set of nodes that contain arcs comparable to seed arc set;
- threshold them according to ArcRank value;
- extend seed arc set, when nodes contain further commonality;
- if the node set increased in size repeat from the first step.

The algorithm outputs a set of terms that are related by the strength of the associations in the arcs that they contain. These associations were computed according to local hierarchies of subsuming and specializing relationships, and sets of terms are related by the kinship relation. The algorithm is naturally self-limiting via the thresholds. This approach can also be used to distinguish senses. For instance, the senses of a word such as *hard*, are distinguished by the choice of association with *tough* and *severe*. Also, ranking the different senses of a term by the strength of its associations with other terms could uncover the principal sense of a term.

Problems encountered. A number of problems have been highlighted during the process of ontology generation in this project:

- syllable and accent markers in head words;
- mis-spelled head words;
- mis-tagged fields;
- stemming and irregular verbs (e.g. hopelessness);
- common abbreviations in definitions (e.g. etc.);
- undefined words with common prefixes (e.g. un-);
- multi-word head words (e.g. water buffalo);
- undefined hyphenated and compound words (e.g. sea-dog).

The interested reader can refer to Jannink [20] for a more detailed account of the methodology and problems encountered.

2.3. Ontology learning (AIFB, University of Karlsruhe)

The Ontology Learning Group in AIFB (Institute of Applied Informatics and Formal Description Methods, University of Karlsruhe, Germany) is active in the ontology engineering area. They have developed various tools to support ontology generation that include OntoEdit (the ontology editor) and Text-To-Onto (an integrated environment for the task of learning ontologies from text [23, 24]).

Extracting ontology from domain data, especially domain-specific natural-language free texts, turns out to be very important. Common approaches usually extract relevant domain concepts based on shallow information retrieval techniques and cluster them into a hierarchy based on statistics and machine learning algorithmal analysis. However, most of these approaches have only managed to learn the taxonomic relations in ontologies. Detecting the non-taxonomic conceptual relationships, for example, the 'has Part' relations between concepts, is becoming critical for building good-quality ontologies [23, 24].

Methods. AIFB's approach to ontology generation contains two parts: shallow text processing and learning algorithms. In shallow text processing, techniques have been implemented on top of SMES (Saarbrucken Message Extraction System). SMES is a shallow text processor for the German language developed by DFKI (German Research Centre for Artificial Intelligence, Germany [25]). It comprises techniques for tokenizer, lexicon, lexical analysis (morphological analysis, recognition of name entities, retrieval of domain-specific information, part-of-speech tagging) and Chunk parser. SMES uses weighted finite state transducers to efficiently process phrasal and sentential patterns.

The outputs of SMES are dependency relations found through lexical analysis. These relations are treated as the input of the learning algorithms. Some of the dependency relations do not hold the meaningful relations of the two concepts which can be linked together (co-occurrence) by some mediator (i.e. proposition, etc.). SMES also returns some phrases without any relations. Some heuristic rules have been defined to increase the high recall of the linguistic dependency relations, for instance, the NP-PP-heuristic (attaching all prepositional phrases to adjacent noun phrases), sentence-heuristic (relating all concepts contained in one sentence), and title-heuristic (linking the concepts appearing in the title with all the concepts contained in the overall document [23, 24]).

The learning mechanism is based on the algorithm for discovering generalized association rules proposed by Srikant and Agrawal [26]. The learning module contains four steps: (1) selecting the set of documents; (2) defining the association rules; (3) determining confidence for these rules; and (4) outputting association rules exceeding the user-defined confidence.

AIFB also built a system to facilitate the semi-automatic generation of the ontologies known as Text-To-Onto [24], as shown in Fig. 3. The system includes a number of components: (1) text and processing management component (for selecting domain texts exploited for the further discovery process); (2) text processing server (containing a shallow text processor based on the core system SMES – the result of text processing is

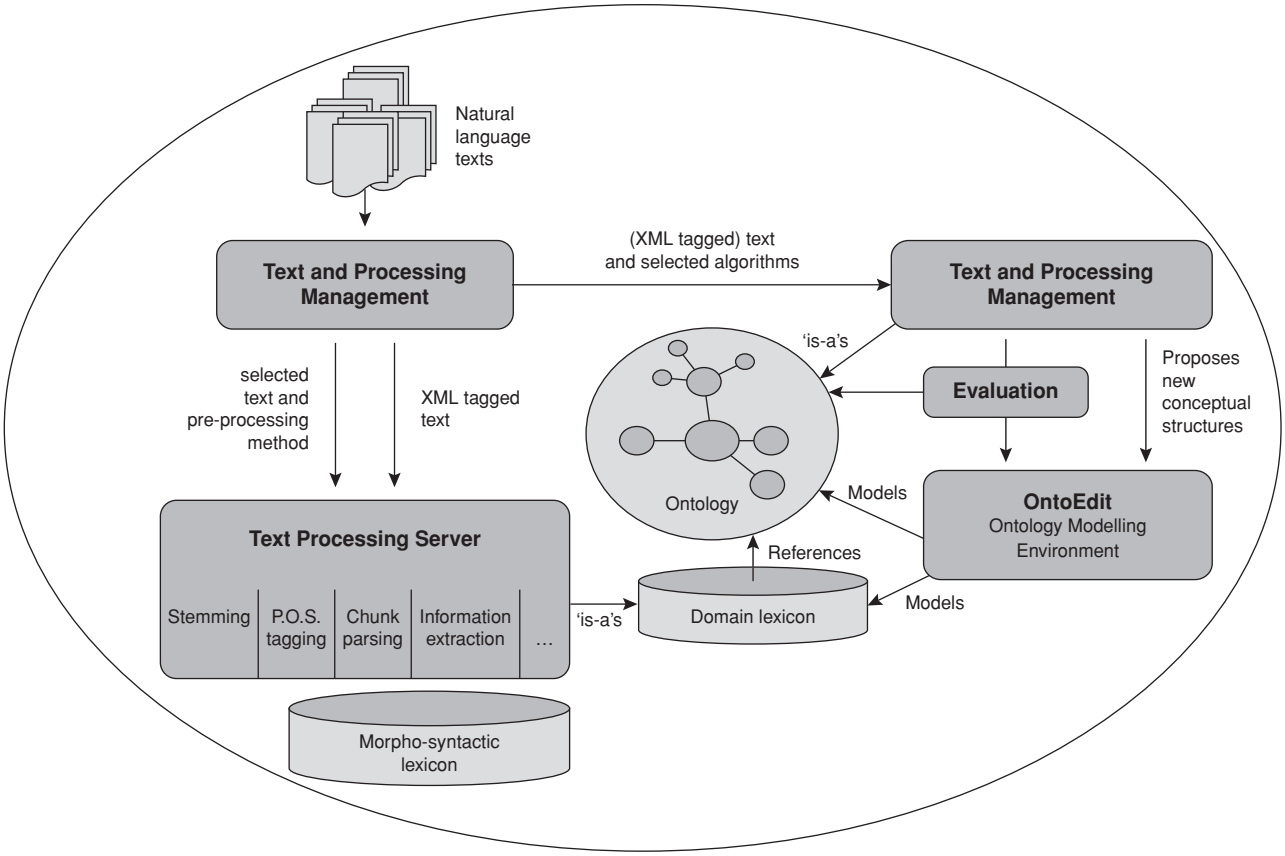

Fig. 3. General overview of ontology learning system Text-to-Onto (http://ontoserver.aifb.uni-karlsruhe.de/texttoonto/).

stored in annotations using XML-tagged text); (3) lexical DB and Domain Lexicon (facilitating syntactic processing based on the lexical knowledge); (4) learning and discovering component (using various extraction methods, e.g. association rules, for concept acquisition); and (5) ontology engineering environment (support in semi-automatically adding newly discovered conceptual structures to the ontology by the ontology engineers, OntoEdit).

Kietz *et al.* [4] adopted the above method to build an insurance ontology from a corporate intranet. First, the GermaNet was chosen as the top-level ontology for the domain-specific goal ontology. Then, domain-specific concepts were acquired using a dictionary that contains important corporate terms that were described in natural language. Then, a domain-specific and a general corpus of texts were used to remove concepts that were not domain-specific through some heuristic rules. Relations between concepts were learned by analyzing the corporate intranet documents based on a multi-strategy learning algorithm, either from the statistical approach (frequent coupling of concepts in sentences can be regarded as relevant relations between concepts) or from the pattern-based approach.

Problems encountered. A number of problems have been highlighted in the process of ontology generation in this project:

- The lightweight ontology contains much noisy data. For instance, not every noun-phrase or single term can be considered as the concept or class. The word sense problem generates lots of ambiguity.
- The refinement of these lightweight ontologies is a tricky issue that needs to be resolved. For instance, domain experts should be involved or semi-automatic techniques based on the recursive machine learning algorithms should be used.
- The learning relationship is not trivial. For instance, general relations can be detected from the hierarchical structure of the extracted concepts (or terms). How to identify and regroup the specific or concrete relationships becomes the main hurdle for ontology generation.

2.4. ECAI 2000

Some of the automatic ontology learning research reported in the *Ontology Learning Workshop of ECAI 2000 (European Conference on Artificial Intelligence)* is important and appropriate to this survey. A number of shallow natural language processing techniques such as Part-of-Speech tagging, word sense disambiguation, tokenization etc., are directly relevant. These are used to extract important (high frequency) words or phrases that could be used to define concepts. In the concept formation step, it is usual for some form of general top-level ontologies (WordNet, SENSUS) to be used to assist correct extraction of the final terms and disambiguation of the word senses.

Methods. Wagner [27] addressed the automatic acquisition of selectional preferences of verbs by means of statistical corpus analysis for automatic ontology generation. Such preference is essential for inducing thematic relations, which link verbal concepts to nominal concepts that are selectionally preferred as their complements. Wagner [27] introduced a modification of Abe and Li's [28] method (based on the well-founded principle: minimum description length) and evaluated it by employing a gold standard. It aimed to find the appropriate level of generation of concepts that can be linked to some specific relations. The EuroWordNet database provided information that can be combined to obtain a gold standard for selectional preferences. With this gold standard, lexicographic appropriateness can be evaluated automatically. However, one of the drawbacks of this method was that the learning algorithms were fed with word forms rather than word senses.

Chalendar and Grau [29] conceived a system SVETLAN, which was able to learn categories of nouns from domain-free texts. In order to avoid general classes, they also considered the contextual use of words. The input data of this system were the semantic domains with the thematic units (TU). Domains were sets of weighted words, relevant to represent the same specific topic. The first step of SVETLAN was to retrieve text segments of the original texts associated with the different TUs, then all the triplets constituted by a verb, the head noun of a phrase and its syntactic role from the parser results in order to produce syntactic thematic units (STUs). The STUs belonging to a same semantic domain were aggregated together to learn about a structured domain.

Bisson *et al.* [30] described Mo'K (a configurable workbench) to support the development of conceptual clustering methods for ontology building. Mo'K was intended to assist ontology developers to define the most suitable learning methods for a given task. It provides facilities for evaluation, comparison, characterization and elaboration of conceptual clustering methods.

Faure and Poibeau [31] discussed how semantic knowledge learned from a specific domain can help the creation of a powerful information extraction system. They combined two systems, SYLEX [32] and ASIUM together, and termed it the 'double regularity model', to eliminate the negative impacts of individual systems to yield good results. For instance, this combination could easily avoid a very time-consuming manual disambiguation step. The special part of their work is that the conceptual clustering process does not only identify the lists of nouns but also augments this list by induction.

Todirascu *et al.* [33] used the shallow natural language processing parsing techniques to semi-automatically build up the domain ontology (conceptual hierarchy) and represent it in description logics (DL), which provides a powerful inference mechanism and is capable of dealing with incomplete, erroneous data. Different small French *corpora* have been tested in the prototype. The system was capable of identifying relevant semantic issues (semantic chunks) using minimal syntactic knowledge and the complex concepts were inferred by the DL mechanism. Several tools were employed in the model:

- A POS tagging identifies the content words (nouns, adjectives, verbs) and functional words (prepositions, conjunctions, etc.). The tagger uses a set of contextual and lexical rules (based on prefixes and suffixes identification) learned from annotated texts.
- The sense tagger contains a pattern matcher, a set of patterns (words, lexical categories and syntagms) and their sense assigned by a human expert. The sense is represented by DL concepts. The set of conceptual descriptions was established by a human expert from a list of the most frequently repeated segments and words extracted from a set of representative texts. The pattern matcher annotates each sequence of words matching the pattern with its semantic description.
- The chunk border identification identifies the words and the syntactic constructions delimiting the semantic chunks. This module uses the output of the POS tagger, as well as a set of manually built cue phrases (syntactic phrases containing auxiliaries, composed prepositions, etc.). The borders of noun and prepositional phrases (determiners, prepositions) are best candidates for the chunk border.

This research has basically automated the process of creating a domain hierarchy based on a small set of primitive concepts defined by the human expert. The expert also has to define the relations of these concepts. As part of future research, emphasis is needed on the use of document summaries as indexes and integration of the system into XML documents.

Problems encountered. The main problem arising from these researches pertains to relation extraction. Such relations were defined manually or induced from the hierarchical structure of the concept classes. A potential solution proposed is to have provision of very general relations, such as 'assoc-with', 'is-a' and so on. How to efficiently extract concrete relations for the concept class remains an important and interesting topic for ontology learning and research.

2.5. Inductive logic programming (University of Texas at Austin)

The Machine Learning Group of the University of Texas (UT) applied inductive logic programming (ILP) to learn relational knowledge from different examples. Most machine learning algorithms are restricted to feature-based examples or concepts and therefore limit themselves for learning complex relational and recursive knowledge. The applications of ILP by this group are extended to various problems in natural language and theory refinement.

Methods. Thompson and Mooney [34] described a system WOLFIE (Word Learning From Interpreted Examples) that learns a semantic lexicon from a *corpus* of sentences. The lexicon learned consists of words paired with representations of their meanings, and allows for both synonymy and polysemy. WOLFIE is part of an integrated system that learns to parse novel sentences into their meaning representations. The overview of the WOLFIE algorithm is to 'derive possible phrase-meaning pairs by sampling the input sentence–representation pairs that have phrases in common, and deriving the common substructure in their representations, until all input representation can be composed from their phrase meanings, then add the best phrase-meaning pair to the lexicon, constrain the remaining possible phrase-meaning pairs to reflect the pair just learned, return the lexicon of learned phrase-meaning pairs'.

ILP is a growing topic in machine learning to study the induction of logic programs from examples in the presence of background knowledge [35].

Problems encountered. UT's systems (ILP) have the potential to become a workbench for ontological concept extraction and relation detection. They combine the techniques from information retrieval, machine learning and artificial intelligence for concept and rule learning. However, how to deploy UT's methods for ontology concept and rule learning is still an open question that needs to be resolved to make this workbench a feasible possibility.

2.6. Library science and ontology

Traditional techniques deployed in the library and information science area have been significantly challenged by the huge amount of digital resources. Ontologies, explicit specification of the semantics and relations in a machine-processable way, have the potential to suggest a solution to such issues. The digital library and Semantic Web communities are at present working hand-in-hand and collaborating to address such special needs in the digital age. The recent European thematic network on digital libraries (DELOS, www.ercim.org/delos/) and Semantic Web (ONTOWEB, www.ontoweb.org) proposed a joint sponsorship of a working group on 'Content standardization for cultural repositories' within the OntoWeb SIG on Content Management (www.ontoweb.org/sig.htm), which attests to the cooperation and future collaborations of these communities.

Methods. In digital library projects, ontologies are specified or simplified to take the form of various vocabularies, including cataloguing codes such as machine readable cataloguing (MARC), thesauri or subject heading lists, and classification schemes. Thesauri and classifications, on the one hand, are used to represent the subject content of a book, journal article, a file, or any form of recorded knowledge. Semantic relationships among different concepts are reflected through broader terms, narrower terms and related terms in thesauri, and a hierarchical structure in classification schemes. On the other hand, a thesaurus does not handle descriptive data (title, author, publisher, etc.). In this instance, separate representational vocabularies for the descriptive data such as the Anglo-American Cataloguing Rules (AACR2) to meet the need for descriptive representation, or the Dublin Core Metadata (www.dublincore.org) have been used.

The fundamental difference between an ontology and a conventional representational vocabulary is the level of abstraction, relationships among concepts, the machine-understandability and, most important, the expressiveness that can be provided by ontologies. For instance, an ontology can be layered according to

different requirements (similar to the design model of an object-oriented programming language such as the unified modelling language, UML). Thus, we may have an upper-level ontology to define the general and generic concept or the schema of the lower-level ontology. Moreover, an ontology can function as a database schema to define various tasks or applications. In a nutshell, an ontology aims to achieve communication between humans and computers, while the conventional vocabulary in the library and information science field fulfils the requirement for communication among human beings. Ontology promotes standardization and reusability of information representation through identifying common and shared knowledge. Ontology adds values to traditional thesauri through deeper semantics in digital objects, conceptually, relationally and through machine understandability [36]. Deeper semantics can imply deeper levels of hierarchy, enriched relationships between classes and concepts, conjunction and disjunction of various classes and concepts, formulation of inference rules, etc.

The University of Michigan Digital Library (UMDL) [37] maps the UMDL ontology to MARC with either 'has' or 'of' or 'kind-of' or 'extended' relationships. In another study, Kwasnik [36] converted a controlled vocabulary scheme into an ontology, citing it as an added-value contribution between ontology and the knowledge representation vocabularies used in libraries and information industries for the following reasons:

- higher levels of conception of descriptive vocabulary;
- deeper semantics for class/subclass and cross-class relationships;
- ability to express such concepts and relationships in a description language;
- reusability and 'share-ability' of the ontological constructs in a heterogeneous system; and
- strong inference and reasoning functions.

Qin and Paling [38] used the controlled vocabulary at the Gateway to Educational Materials (GEM) as an example and converted it into an ontology. The major difference between the two models is the value added through deeper semantics both conceptually and relationally. The demand to convert the controlled vocabulary into an ontology is due to the limited expressive power of controlled vocabulary, the emerging popularity of agent communication (ontology-driven communication), semantic searching through the intranet and the internet, and the content and context exchanges existing in various marketplaces of e-commerce. The purpose of such conversions not only reduces the duplication of effort involved in building an ontology from scratch by using the existing vocabularies, but also establishes a mechanism for allowing differing vocabularies to be mapped onto the ontology.

Problems encountered. Three main problems remain in the area of development of ontology in this respect:

- different ways of modelling the knowledge (Library Science and Ontology) due to the 'shallow' and 'deeper' semantics that are inherent in these two disciplines;
- different ways of representing the knowledge, for instance, the librarian uses the hierarchical tree (more lexically flavoured) to represent the thesaurus or catalogues, while the ontology engineer uses mathematical and formal logics to enrich and represent ontologies (more mathematically and logically flavoured);
- there is a long way to go to achieve consensus to merge the two to create a standardized means to organize and describe information.

2.7. Others

Borgo *et al.* [39] used lexical semantic graphs to create an ontology (or annotate the knowledge) based on the WordNet. They pointed out some special nouns which are always used to represent relations, for instance, part (has-part), function (function-of), etc. They called these special nouns 'relational nouns' which would facilitate the identification of the relations between two concepts.

Yamaguchi [40] focused on how to construct domain ontologies based on a machine-readable dictionary (MRD). He proposed a domain ontology rapid development environment (called DODDLE) to manage concept drift (word sense changes due to different domains). However, no detailed information about the basic theory adopted was made available. Nonetheless, it implies some form of concept graph mapping plus user-involved word sense disambiguation based on the WordNet to trim and deal with the concept shift so as to get the very specific small domain ontology from the user input containing several seed words for the domain.

Kashyap [41] proposed an approach for designing an ontology for information retrieval based on the schemas of the databases and a collection of queries of interest to the users. Ontology construction from database schema involves many issues, such as determining primary keys, foreign keys, inclusion dependencies, abstracting details, grouping information from multiple

Table 1
Summary of state-of-the-art ontology generation studies and projects

	InfoSleuth (MCC)	SKC (University of Stanford)	Ontology learning (AIFB)	ECAI2000	Inductive logic programming (UT)	Library science and ontology	Others
Source data (tagged)	Domain thesaurus Seedwords from expert Free-text doc from the internet (POS tagged automatically)	Webster's online dictionary (in XML or SGML format), semi-structured source data	Free-text, natural language documents from the web	Domain-free text Semantic domain with TU Annotated texts Primitive concepts from the human experts	Annotated documents A corpus of sentences	Controlled vocabulary	Machine readable dictionary Schema of database User queries
Methods for concept extraction	Superficial syntactic analysis: pattern matching + local context (noun phrases) Word sense disambiguation is needed	Tag extraction Pattern matching (wrapper or script) PageRank algorithm	Tokenzier Morphological analysis Name recognition Part-of-speech tagging Chunk parser	Category of nouns Conceptual clustering and induction Shallow natural language processing POS tagging (contextual and lexical rules)	Slots fillers (rules learning from C4.5 and Rapier) Pattern matching POS Token	Subject headings from controlled vocabulary Manually refined concepts	
Methods for relation extraction	Relations were automatically acquired based on the linguistic property of noun components and the inheritance hierarchy Two kinds: is-a and assoc-with	ArcRank An iterative algorithm based on pattern/relation extraction algorithm Relations could be learned and refined based on the local graphical hierarchies of subsuming and specializing	Co-occurrence clustering of concepts Mediator (proposition, verb) Heuristic rules based on the linguistic dependency relations General association rules by machine learning	Selectional preferences of verbs (minimum description length with a gold standard)	Inductive logic programming (machine learning)	Relations from controlled vocabulary (broad term, narrow term, etc.) Manually refined relations	Relational nouns to represent relations
Ontology reuse	Yes (unit of measure, geographic metadata etc.)	Yes (online dictionary)	Yes (Lexicon)	Yes (EuroWordNet)		Yes (controlled vocabulary)	Yes (WordNet, data dictionary, controlled vocabulary)
Ontology representation	Hierarchical structure	Conceptual graph	XML	Conceptual hierarchy Description logic			Conceptual graph
Tool or system associated	None		SMES Text-To-Onto OntoEditor	SVETLAN Mo'K SYLEX ASIUM	WOLFIE		DODDLE
Other	Corpus-based learning		Non-taxonomy relation learning				

tables, identifying relationships, and incorporating concepts suggested by new database schema. A set of user queries expressing their information needs could be used to further refine the ontology created, which could result in the design of new entities, attributes and class–subclass relationships that are not directly presented in existing database schemas. The ontology generated can be further enhanced by the use of a data dictionary and a controlled vocabulary. The approach for ontology construction in this instance is therefore to make full use of the existing sources, such as database schemas, user queries, data dictionaries and standardized vocabularies for proposing an initial set of entities, attributes and relationships for an ontology.

3. Summary and conclusions

As the first part of the survey of ontology generation, research has been examined that is related to semi-automatic or automatic ontology generation. Table 1 summarizes the general pattern and characteristics of the various methods adopted by different research groups or researchers along the dimensions of source data, methods for concept extraction and relation extraction, ontology reuse, ontology representation, associative tools and systems and other special features.

The following salient points and features in ontology generation to date can be observed in general:

- Source data are more or less semi-structured and some seed-words are provided by the domain experts not only for searching for the source data but also as the backbone for ontology generation. Learning ontology from free-text or heterogeneous data sources is still within the area of the research laboratory and far from real applications.
- For concept extraction, there already exist some relatively mature techniques (such as POS, word sense disambiguation, tokenizer, pattern matching, etc.) that have been employed in the field of information extraction, machine learning, text mining and natural language processing. The results of these individual techniques are promising as basic entities and should prove most useful in the formation of concepts in ontology building.
- Relation extraction is a very complex and difficult problem to resolve. It has turned out to be the main impediment to ontology learning and applicability. Further research is encouraged to find appropriate and efficient ways to detect or identify the relations either semi-automatically or automatically.
- Ontologies are highly reused and reusable. Based on a basic ontology, other forms of ontologies may be lifted off to cater for specific application domains. This is important because of the cost of generation, abstraction and reusability.
- Ontologies can be represented as graph (conceptual graph), logic (description logic), web standards (XML), or a simple hierarchy (conceptual hierarchy). Currently there is the standard ontology representation language called DAML+OIL (www.daml.org/2001/03/daml+oil-index), which combines the merits of description logic, formal logic and web standards.
- A number of tools have been created to facilitate ontology generation in a semi-automatic or manual way. For instance, the University of Karlsruhe (Germany) has developed and commercialized the semi-automatic ontology editor called OntoEdit (now owned by the Spin-off Company called Ontoprice). Stanford University has exploited and provided an ontology-editing environment called Protégé with many users. The University of Manchester owns OilEd, an ontology editor for supporting DAML+OIL (for details of the state-of-the-art ontology editor, see www.ontoweb.org/workshop/amsterdamfeb13/index.html).

It is evident that much needs to be done in the area of ontology research before any viable large scale system can emerge to demonstrate ontology's promise of superior information organization, management and understanding. Far beyond ontology generation, evolving and mapping existing ontologies will form another challenging area of work in the ontology field.

References

[1] T.R. Gruber, A translation approach to portable ontology specifications, *Knowledge Acquisition* 5 (1993) 199–220.

[2] C.H. Hwang, Incompletely and imprecisely speaking: using dynamic ontologies for representing and retrieving information. In: *Proceedings of the 6th International Workshop on Knowledge Representation meets Databases (KRDB'99)*, Linköping, Sweden, 29–30 July 1999.

[3] D. Fensel, J. Hendler, H. Lieberman and W. Wahlster (eds), *Semantic WebTechnology* (MIT Press, Boston, MA, 2002).

[4] J.-U. Kietz, A. Maedche and R. Volz, Extracting a domain-specific ontology Learning from a corporate intranet. In: *Second 'Learning Language In Logic' LLL Workshop,* co-located with the *International Conference in Grammere Inference (ICGI'2000) and Conference on Natural Language Learning (CoNLL'2000),* Lisbon, Portugal, 13–14 September 2000.

[5] T. Berners-Lee, J. Hendler and O. Lassila, The Semantic Web. *Scientific American* May (2001) 216–218.
[6] D. Fensel, B. Omelayenko, Y. Ding, E. Schulten, G. Botquin, M. Brown and A. Flett, Product data integration in B2B e-commerce, *IEEE Intelligent Systems* 16(4) (2001) 54–59.
[7] T. Berners-Lee and M. Fischetti, *Weaving the Web* (Harper, San Francisco, CA, 1999).
[8] D. Fensel, *Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce* (Springer, Berlin, 2001).
[9] M. Uschold, Creating, integrating and maintaining local and global ontologies. In: *Proceedings of the First Workshop on Ontology Learning (OL-2000)* in conjunction with the *14th European Conference on Artificial Intelligence (ECAI 2000),* August 2001, Berlin.
[10] M. Fernandez-Lopez, Overview of methodologies for building ontologies. In: *Proceedings of IJCAI-99 Workshop on Ontologies and Problem-Solving Methods: Lessons Learned and Future Trends*, in conjunction with the *Sixteenth International Joint Conference on Artificial Intelligence*, Stockholm, August 1999.
[11] J. McCarthy, Notes on formalizing context. In: *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence* (AAAI, 1993).
[12] N. Guarino, Some ontological principles for designing upper level lexical resources. In: *Proceedings of the First International Conference on Lexical Resources and Evaluation*, Granada, 28–30 May 1998.
[13] N.B. Cocchiarella, Formal ontology. In: H. Burkhardt and B. Smith (eds), *Handbook of Metaphysics and Ontology* (Philosophia, Munich, 1991), pp. 640–647.
[14] M. Uschold and M. Gruninger, Ontologies: principles, methods, and applications, *Knowledge Engineering Review* 11(2) (1996) 93–155.
[15] A. Gomez-Perez, N. Juristo and J. Pazos, Evaluation and assessment of knowledge sharing technology. In: N.J. Mars (ed.), *Towards very Large Knowledge Bases – Knowledge Building and Knowledge Sharing* (IOS Press: Amsterdam, 1995), pp. 289–296.
[16] J.R. Reich, Ontological design patterns for the integration of molecular biological information. In: *Proceedings of the German Conference on Bioinformatics GCB'99*, 4–6 October, Hannover, 1999, pp.156–166.
[17] G.A. Miller, WORDNET: a lexical database for English. *Communications of ACM* (11) (1995) 39–41.
[18] E. Riloff and J. Shepherd, A corpus-based approach for building semantic lexicons. In: *Proceedings of International Symposium on Cooperative Database Systems for Advanced Applications*, 1999.
[19] J. Jannink and G. Wiederhold, Ontology maintenance with an algebraic methodology: a case study. In: *Proceedings of AAAI Workshop on Ontology Management*, July 1999.
[20] J. Jannink, Thesaurus entry extraction from an on-line dictionary. In: *Proceedings of Fusion '99*, Sunnyvale, CA, July 1999.
[21] L. Page and S. Brin, The anatomy of a large-scale hypertextual web search engine. In: *Proceedings of the 7th Annual World Wide Web Conference*, 1998.
[22] S. Brin, Extracting patterns and relations from the world wide web. In: *WebDB Workshop at EDBT '98*, 1998.
[23] A. Maedche and S. Staab, Discovering conceptual relations from text. In: *ECAI 2000. Proceedings of the 14th European Conference on Artificial Intelligence* (IOS Press, Amsterdam, 2000).
[24] A. Maedche and S. Staab, Mining ontologies from text. In: R. Dieng and O. Corby (eds), *EKAW-2000 – 12th International Conference on Knowledge Engineering and Knowledge Management*, Juan-les-Pins, France, 2–6 October (LNAI, Springer, Berlin, 2000).
[25] G. Neumann, R. Backofen, J. Baur, M. Becker and C. Braun, An information extraction core system for real world German text processing. In: *ANLP'97 – Proceedings of the Conference on Applied Natural Language Processing*, Washington, DC, pp. 208–215.
[26] R. Srikant and R. Agrawal, Mining generalized association rules. In: *Proceedings of VLDB'95* 1995, pp. 407–419.
[27] A. Wagner, Enriching a lexical semantic net with selectional perferences by means of statistical corpus analysis. In: *Proceedings of the First Workshop on Ontology Learning (OL-2000)* in conjunction with the *14th European Conference on Artificial Intelligence (ECAI 2000)*, Berlin, August 2000.
[28] N. Abe and H. Li, Learning word association norms using tree cut pair models. In: *Proceedings of 13th International Conference on Machine Learning*, 1996.
[29] G. Chalendar and B. Grau, SVETLAN: a system to classify nouns in context. In: *Proceedings of the First Workshop on Ontology Learning (OL-2000)* in conjunction with the *14th European Conference on Artificial Intelligence (ECAI 2000)*, Berlin, August 2000.
[30] G. Bisson, C. Nedellec and D. Canamero, Designing clustering methods for ontology building: the Mo'K workbench. In: *Proceedings of the First Workshop on Ontology Learning (OL-2000)* in conjunction with the *14th European Conference on Artificial Intelligence (ECAI 2000)*, Berlin, August 2000.
[31] D. Faure and T. Poibeau, First experiments of using semantic knowledge learned by ASIUM for information extraction task using INTEX. In: *Proceedings of the First Workshop on Ontology Learning (OL-2000)* in conjunction with the *14th European Conference on Artificial Intelligence (ECAI 2000)*, Berlin, August 2000.
[32] P. Constant, *Reducing the Complexity of Encoding Rule-based Grammars*. December 1996.
[33] A. Todirascu, F. Beuvron, D. Galea and F. Rousselot, Using description logics for ontology extraction. In: *Proceedings of the First Workshop on Ontology Learning (OL-2000)* in conjunction with the *14th European Conference on Artificial Intelligence (ECAI 2000)*, Berlin, August 2000.

[34] C.A. Thompson and R.J. Mooney, *Semantic Lexicon Acquisition for Learning Parsers*, Unpublished Technical Note, January 1997.

[35] N. Lavrac and S. Dzeroski (eds), *Inductive Logic Programming: Techniques and Applications* (Ellis Horwood, Chichester, 1994).

[36] B. Kwasnik, The role of classification in knowledge representation and discovery. *Library Trends* 48 (1999) 22–47.

[37] P. Weinstein, Ontology-based metadata: transforms the MARC legacy. In: F. Akscyn and F.M. Shipman (eds), *Digital Libraries 98, Third ACM Conference on Digital Libraries* (New York: ACM Press, 1998), 254–263.

[38] J. Qin and S. Paling, Converting a controlled vocabulary into an ontology: the case of GEM. *Information Research* 6(2) (2000). Available at: http:\\InformationR.net/ir/6-2/infres62.html

[39] S. Borgo, N. Guarino, C. Masolo and G. Vetere, Using a large linguistic ontology for internet-based retrieval of object-oriented components. In: *Proceedings of 1997 Conference on Software Engineering and Knowledge Engineering*, Madrid (Knowledge Systems Institute, Snokie, IL, 1997).

[40] T. Yamaguchi, Constructing domain ontologies based on concept drift analysis. In: *Proceedings of IJCAI-99 Workshop on Ontologies and Problem-Solving Methods: Lessons Learned and Future Trends*, in conjunction with the *Sixteenth International Joint Conference on Artificial Intelligence*, Stockholm, August 1999.

[41] V. Kashyap, Design and creation of ontologies for environmental information retrieval. In: *Proceedings of Twelfth Workshop on Knowledge Acquisition, Modeling and Management*, Alberta, October, 1999.