UNIVERSITAT POLITÈCNICA DE CATALUNYA
DEPARTAMENT DE LLENGUATGES I SISTEMES INFORMÀTICS
PROGRAMA DE DOCTORAT EN INTEL·LIGÈNCIA ARTIFICIAL

TESI DOCTORAL

Domain Ontology Learning from the Web

Memòria presentada per David Sánchez Ruenes
per optar al títol de Doctor en Informàtica per
la Universitat Politècnica de Catalunya

Director: Dr. Antonio Moreno Ribas (URV)
Tutor: Dr. Ulises Cortés (UPC)

Tarragona, 2007

Chapter 7

Implementation and applications

Up to this point we have presented the main contributions of our work, describing several specially designed methods for learning domain ontologies from the Web. In order to test their viability in a real world environment, a prototype has been developed. It includes all the different learning steps in an integrated fashion. It also contains functions for the different automatic evaluation procedures described in the previous chapter. The system has been designed and implemented in a distributed way providing, as will be described later, an efficient solution. Note that its execution over several domains has provided the example results presented in the previous sections and in the publications referenced in the Annex.

So, in §7.1, we discuss the computation complexity of the developed algorithms. As a consequence of this study and the potential improvement that can be achieved using a distributed approach, we present a system architecture based on a Multi-Agent system. Next, the formal language used to represent the results and the programming libraries, tools and software used during the development are presented.

In addition, in order to proof the viability of the proposed learning methodologies and the usefulness of the potential results, we have applied them over several real world problems as will be presented in §7.2. Concretely, in §7.2.1 we introduce a way for bringing structure to the web resources analysed during the ontology learning process of a particular domain. Next, in §7.2.2, we describe a method based on our taxonomic learning proposal to structure automatically large digital libraries. Finally, in §7.2.3, we present a distributed knowledge-based system that, using our automatically constructed domain ontologies as input, is able to perform semantically grounded Web Information Retrieval.

7.1 Prototype implementation

In previous chapters, we have described from a methodological point of view the proposed learning procedures. Through the explanation we have commented several questions regarding the scalability and efficiency of the analytical procedure in order to obtain a feasible learning throughput in such an enormous repository as the Web and for general domains of knowledge involving thousands of entities.

Even considering aspects such as the lightweight analysis or snippet-based web parsing, the knowledge acquisition can be a very time consuming task. As Table 44 shows, one iteration of the learning process for one general concept can take about 1 hour using one computer. This is mainly caused by the online accessing to web resources and the querying of web search engines. However, the runtime is reduced when dealing with specific subclasses or concrete non-taxonomically related concepts (both retrieved from the initial domain's keyword during the incremental learning process), as a narrower spectrum of web resources and candidates is available.

Table 44. Summary of results obtained for one iteration of the full learning process for several domains using one computer. All test performed against MSNSearch with default parameters.

Domain	Sub classes	Instances	Non taxonomic	Queries (statistics)	Total webs	Run-time
Sensor	55	48	444	3105	848	57 min.
Cancer	73	25	497	2298	774	48 min.
Hypertension	29	11	336	1799	664	37 min.
Colon cancer	19	9	48	765	175	11 min.
Metastatic breast cancer	9	1	11	334	74	8 min.
Papiloma virus	3	0	0	35	65	3 min.

From the analysis of the results presented in Table 44, one can observe that the runtime depends linearly on the number of access to the Web, querying a web search engine or checking a web site. However, considering that, for medium to wide analyses, the number of queries overpass the number of web sites accessed, we can conclude that the runtime depends on the number of web search engine queries (see Figure 29). More details about this assumption will be discussed in §7.1.4.

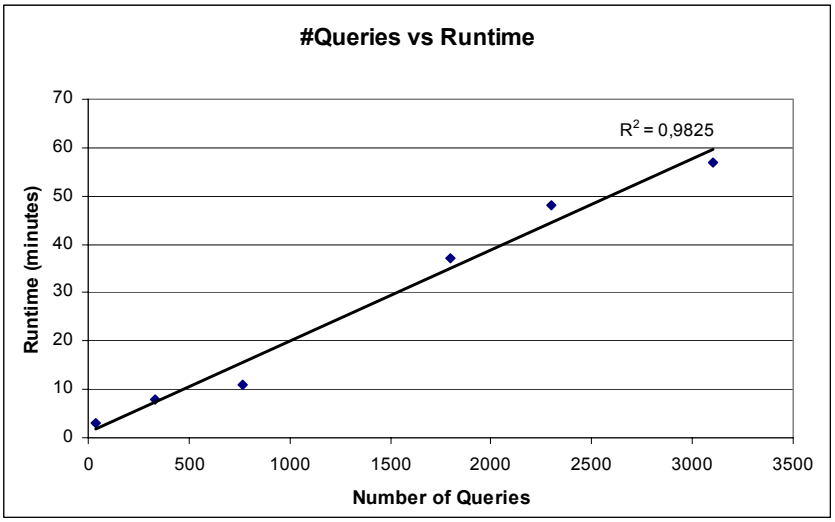

Figure 29. Runtime depends linearly on the number of Web search queries.

In any case, the incremental execution of the different instances of the learning processes (taxonomic and non-taxonomic) as new knowledge (domain concepts) is acquired may represent, for general domains, a computational cost that is hard to be assumed by one computer. For example, in the tests presented in Table 45 –considering only the taxonomic aspect- and Table 46 –restricting to 2 taxonomic levels and 1 non-taxonomic level-, one may observe the increment of runtime required to recursively analyse general domains of knowledge.

Table 45. Summary of results obtained for recursive iterations of the taxonomic learning process for several domains on one computer. All tests have been performed against MSNSearch with default parameters.

Domain	Sub classes	Instances	Queries (statistics)	Total webs	1st level runtime	Total runtime
Sensor	868	737	31455	12591	17 min.	15 hours
Cancer	1458	710	40491	11160	8 min.	21 hours
Mammal	957	1187	46308	12747	16 min.	16 hours

Table 46. Summary of results obtained the full learning process restricted to 2 taxonomic levels and 1 non-taxonomic level for several domains on one computer. All test performed against MSNSearch with default parameters.

Domain	Sub classes	Instances	Non taxonomic	Queries (statistics)	Total webs	Runtime
Equation	215	100	730	28741	12326	10 hours
Virus	919	317	1709	204450	23116	66 hours
Cpu	134	164	121	13934	4567	6 hours
Insect	668	227	236	58286	8270	20 hours
Tea	236	87	1430	57148	6471	17 hours

In order to justify this empirically observed behaviour, in the next section we analyse from a theoretical point of view, the computational complexity of the algorithms.

7.1.1 Computational complexity

As has been introduced in §7.1, the full learning process can be divided in individual tasks which correspond to the evaluation of a particular concept. The actions performed during this process are:

- The system queries a web search engine using each Hearst pattern and analyses the web snippets (grouped in sets of 50 for the case of MSNSearch). As a result of this process, a certain number of taxonomic candidates (t_1) are retrieved. They are evaluated by performing new queries into a web search engine. Considering the scores introduced in §5.2, $2ht_1$ queries are requested, where h is the number of Hearst patterns employed (all pattern-based scores are queried and the highest is used). As a result of the queries, s_1 items are selected and t_1- s_1 are rejected in function of the specified selection threshold. Then, depending on the learning threshold, the algorithm may decide to evaluate an additional set of resources (re-

sulting in t_2 candidates to evaluate) or continue with the next pattern. After performing all the iterations for all the Hearst patterns, a total of $2h\sum\sum t_{ij}$ queries for statistics have been performed (where t_{il} is the number of candidates retrieved for the *'i'* iteration and the *'l'* pattern –from a total of *'h'* patterns-). As introduced in the previous section, the number of queries for statistics is the variable that mainly defines the runtime, as it is always much higher than the queries required for web IR and requires much more runtime (several orders of magnitude higher) than web parsing of fixed size results pages.

- The taxonomic learning using Noun Phrases follows the same behaviour but, in this case, each web resource will be accessed independently, resulting in an additional number of web accesses. In addition, the parsing runtime is more nondeterministic as it depends on the size of the specific web site. In any case, queries for computing statistics typically consume most of the runtime. This results in additional $2\sum n_j$ queries, where n_j the number of noun phrase-based candidates evaluated in the iteration *'j'*.
- Additionally to the selection of taxonomic subclasses, retrieved candidates can also be evaluated as instance candidates (named entities). This process requires performing an additional number of queries (e queries are required for '*e'* named entities) and parsing fixed size snippets.
- Once the taxonomic learning is finished, the non-taxonomic phase starts by evaluating extracted verb phrases (as introduced in §5.4). Following the same philosophy, this requires new web queries for computing statistics. Concretely, $2v$ queries are needed to evaluate '*v'* verb phrase candidates.
- Each selected verb phrase is used as a pattern for learning non-taxonomic relations, similarly to the taxonomic case (search querying, snippet parsing and incremental candidate evaluation). This requires $2\sum\sum r_{kp}$ queries for statistics, where r_{kv} is the number of candidates retrieved in the *'k'* iteration for the *'p'* verb phrase.
- Other tasks such as the ontology post-processing performed offline over the obtained structure do not influence in the required runtime as they have a reduced scope (typically thousands of ontological entities).

As a conclusion, the effort applied to a particular domain concept depends on its learning productiveness: the number of taxonomic relations ($t_{ij}+n_j$), non-taxonomic relations (r_{kv}), instances (e) and verb labels (v) candidates retrieved. This result in a linearly dependant amount of web queries for statistics (Q) that finally defines the runtime (18).

$$Q = Queries\_per\_concept = 2h\sum_{l}^{h}\sum_{i} t_{il} + 2\sum_{j} n_j + e + 2v + 2\sum_{p}^{v}\sum_{k} r_{kp} \quad \textbf{(18)}$$

Arrived at this point, we have learned the immediate relations for a particular concept (C). In function of the particular domain, the available web resources and the selection and learning thresholds, we have obtained '*x'* subclasses and '*y'* non-taxonomically related terms (see Figure 30). Note that for the taxonomic case the branching factor is the same as the number of subclasses, but for the non-taxonomic case we can obtain fewer classes than relationships (*i.e.* two classes can be non-taxonomically related with different verb labels).

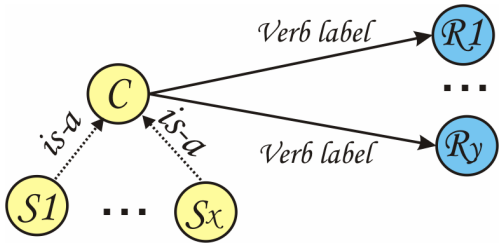

Figure 30. Learning expansion of the concept *C* with *x* taxonomic relationships and *y* non-taxonomic relationships.

The next step will consist on executing the same learning process over those *x+y* new concepts acquired. On one computer those concepts would be sequentially analysed. Considering *T* the runtime for a particular concept that mainly depends on the number of web queries (*Q*), the final runtime would be *(x+y)T*. As this is an incremental process, multilevel relationships can be further developed and the number of concepts can grow consequently. As shown in the previous section, for general concepts, the particular '*T*' has an order of magnitude of minutes, '*x*' can be dozens of subclasses and '*y*' may arrive to several hundreds.

The algorithm is responsible of finishing the less productive ontological branches in function of how the learning evolves as stated in §5.6.2. One should also note that due to the bootstrapped information added to the web queries (presented in §5.6.3), the more advanced the learning is, the more concrete and the less amount of new results are retrieved. In any case, non-taxonomic relationships are hard limited to a maximum of two links from the initial concept. The taxonomic subclass level is not limited but, in practice, the maximum depth achieved is about 3 or 4.

At the end, considering this kind of expansion, the final runtime on one computer where concepts are sequentially evaluated is a *polynomic* function (19).

$$Runtime = T(taxo\_concepts + notaxo\_concepts)^{\max(taxo\_depth,\, notaxo\_depth)} \quad \textbf{(19)}$$

It depends on the number of taxonomic and non-taxonomically related concepts retrieved at each iteration. The exponent is the maximum depth of the relationships (typically the taxonomic depth will be higher and inferior to 4). As stated, the runtime (*T*) required to perform the analysis of one concept depends linearly on the web queries for statistics that, at the same time, depend linearly on the number of retrieved candidates. Considering the orders of magnitude managed (runtime in *minutes* and number of concepts in *hundreds*) one can easily realize that a sequential execution in one computer is not computationally feasible.

However, as shown in Figure 31, taking into consideration the tree like expansion of the learning process, several tasks can be performed concurrently and independently.

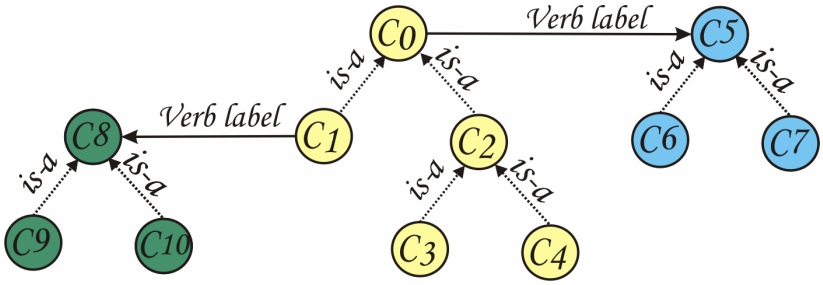

Figure 31. Basic ontological structure with tree like taxonomic a non-taxonomic relationships.

This workflow is adequate for parallel computing, as several tasks (different analyses for each new concept) can be performed at the same time without interference. In our case, the parallelisation is not only related to the computational power, but also to other resources such as the Internet bandwidth or system memory. However, the most important aspect is that the parallel execution of various learning instances through several computers can reduce the overhead of Web access, minimizing the execution waits and web search engine restrictions thanks to the distributed access from, ideally, different IP addresses. Our hypothesis is that a distributed approach of our learning methodology can represent a great improvement.

As the execution workflow is nondeterministic, as it is defined by the knowledge acquired at execution time, coordination and flexibility are fundamental. In order to tackle these execution requirements, we have used the agent paradigm. In the next section we offer an overview of this technology. Next, we provide details about the system architecture, operation and implementation, and the results representation and visualization.

7.1.2 Agents and Multi-Agent Systems

An agent [Wooldridge, 2002] is a computer system capable of flexible autonomous action in some environment. An agent has its own goals and the tools to be able to achieve them. The main properties of agents are:

- *Sociability*: an agent must be able to communicate with other agents, and cooperate with them to solve complex tasks.
- *Reactivity*: an agent is aware of the changes in the environment and responds to them in a timely fashion.
- *Autonomy*: the agent may decide whether to fulfil a given request or not, and may decide which is the best way to achieve its goals.

There are particular problems that cannot be solved by a single agent because different resources, knowledge or tools are needed. In this case, agents must cooperate, co-ordinate or negotiate with other agents to achieve their goals. This is a Multi-Agent System (MAS) [Weiss, 1999]. The main advantages of using a Multi-Agent System are:

- *Modularity*: the full problem can be divided into several tasks than can be modelled into individual agents.

- *Efficiency*: the distributed approach allows concurrent and parallel execution through several nodes of a computer network.
- *Robustness*: against failures of individual agents.
- *Flexibility*: agents can be managed (*i.e.* created, destroyed) dynamically depending on the particular execution needs of the full system.

In recent years it has been argued that MAS may be considered as the latest software engineering paradigm [Jenning, 2000; Petrie, 2001]. This is interesting for large and complex systems in several senses: (i) with geographically distributed data, (ii) with many components or entities, possibly with particular interests, (iii) with a broad scope and huge amounts of information to consider. The use of intelligent, distributed agents is an adequate approach for this type of problems.

As a conclusion, MAS provide some advantages with respect to traditional systems such as efficiency, flexibility, autonomy and highly elaborate communicative skills, and are very suitable to implement dynamic and distributed systems. Several projects applying MAS to information retrieval and knowledge acquisition such as [Gibbins *et al*., 2003; Moreno *et al*., 2004] are an indication that agents can provide benefits in this area.

7.1.3 Agent-based distributed ontology learning

In this section, the implementation of the presented knowledge acquisition methodologies for constructing domain ontologies over a distributed agent-based approach is presented.

In general, the main idea is to distribute the full ontology learning process into several independent tasks that can be executed on different computers. At the end of each execution, partial results obtained by each one are returned and incorporated into the domain ontology. Repeating iteratively this parallel execution model, the final ontology can be constructed transparently using the computational power of several nodes of a computer network.

The developed MAS (*Multi-Agent System*) is composed of several autonomous entities (agents) that can be deployed around a network. Each agent can be considered as an execution unit that follows a particularly modelled behaviour and interacts (communicates) with other ones, coordinating their execution to achieve a common goal. Those agents can be created, eliminated or modified dynamically in function of the execution requirements derived from the learning process, providing an efficient utilisation of the available computational resources.

There are three kinds of agents in the MAS:

a) User Agent (UA): allows the human user to interact with the system. It offers a web interface from which the user can asynchronously manage the learning process and visualize results. Even though the ontology construction process can be fully automatic and unsupervised, through this agent, he has the possibility of configuring, initializing and controlling the construction process. In addition, the web interface represents an invaluable help for debugging during the development phase.

b) Internet Agent (IA): implements the taxonomic and non-taxonomic learning methodology as described in chapter 5. For a specific concept, it performs a single execution of the developed learning methods, composing a partial ontology containing the new taxonomically and non-taxonomically related concepts. Those new concepts can be recursively analysed using new instances of IAs. The coordinated and parallel execution of several IAs with different concepts allows obtaining a set of partial results that can be joined and interrelated in order to build the final domain ontology. As this construction process is very time consuming, in order to provide an efficient solution, this kind of agents are placed in different computer nodes from a network that provides the required hardware resources (*i.e.* available RAM and/or internet bandwidth). They also implement mobility capabilities in order to be deployed transparently and dynamically in an available computer node.

c) Coordinator Agent (CA): it coordinates the domain ontology construction process by creating and configuring IAs to explore retrieved concepts. Concretely, each concept discovered by each partial analysis is used as a seed for further analyses by creating new IAs, bootstrapping with the knowledge already acquired, as described in §5.6.3. In addition, CA joins partial results composing the final domain ontology. It also implements load balancing policies that allow it to decide, at every moment, where to deploy each IA according to the free resources available. It is also able to restore learning state of unfinished tasks (due to software or hardware errors) by continuously monitoring the MAS state. This provides the degree of robustness necessary in distributed environments. Note that, although the ontology construction is centralised by this agent, its work load in relation to the IAs (even with several machines available) is quite reduced.

As shown in Figure 32, the process starts when the UA receives from the user the concept (*e.g. cancer*) that represents the domain to explore (step 1). This is sent to the CA. It creates a first IA that is deployed in an available network node and it starts acquiring domain knowledge (new taxonomically and non taxonomically related terms) using the methodology presented in chapter 5 (step 2). Up to this moment the learning process is executed sequentially. As a result, a set of related terms (*e.g. breast*, *lung*, *colon*, *radiotherapy*) is returned to the CA (step 3). The CA incorporates this knowledge into the domain ontology as classes and relationships and, for each class, a new IA is created and deployed to explore it. At this moment, a degree of parallel execution is achieved in function of the number of tasks to execute (associated to IAs) and the available computer nodes. Concurrently, those partial results are sent to the UA in order to offer an updated visualization of the obtained results.

As the different IAs finalize their analyses, several sets of taxonomically and non-taxonomically related classes are returned asynchronously to the CA that incorporates them into the ontology (step 4). The process is repeated until the algorithm decides to stop exploring each ontology branch (as described in §5.6.2). At the end, the CA is able to construct recursively an ontology that represents the available knowledge in the Web for the domain. As a final step, the CA refines the ontology in order to detect implicit relationships and attributes for each class (*e.g. metastatic cancer*) as described in §5.5 and outputs the result.

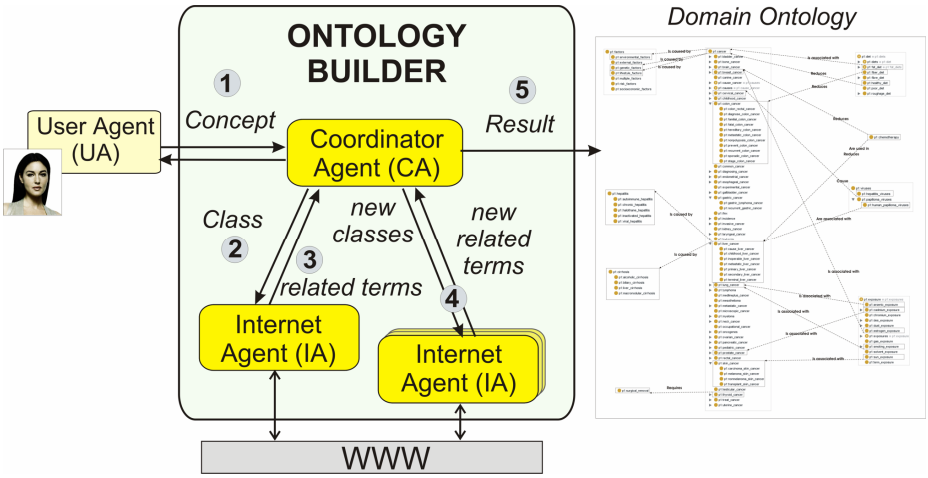

Figure 32. Multi-agent system architecture to create domain ontologies from the Web.

All the agents are deployed and executed in a computer network that provides the required computational resources and Internet bandwidth to perform the analysis (as shown in Figure 33). This network is linked to a server that manages the agent platform and provides a web interface that is managed by the UA allowing the user's access to the system from any computer with Internet connection.

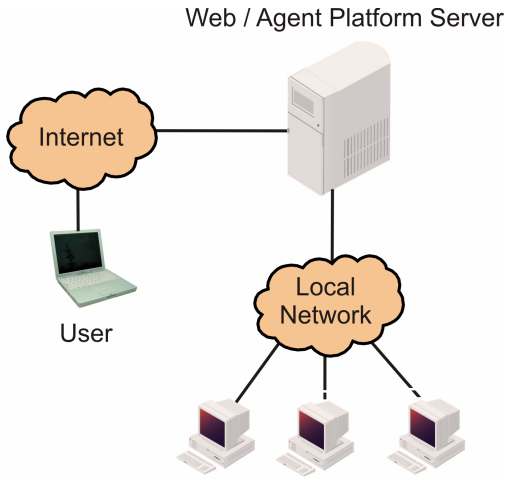

Figure 33. Agent-based knowledge acquisition physical architecture.

Note that no special requirements (computer architecture, operating system, software, computing power, *etc*) are established on the user's side as all the learning process is performed on the server's internal network and the interaction is performed via Internet. Moreover, due to the potential runtime required to finish the full learning process, the server implements a persistence mechanism to store the user's session, maintaining the state and partial results of the works currently in execution.

7.1.4 Distributed learning performance

Once the distributed agent-based system has been presented, in this section we discuss the learning performance obtained using different degrees of parallelism. In this manner we intend to show the scalability capabilities of the designed system and the performance improvement over non parallel approach (introduced in §7.1.1).

The first test consists in picking up four tasks of similar complexity (4 immediate subclasses of the *Cancer* domain) and to execute them, using the same parameters, in the following hardware configurations:

- 1 computer runs the 4 tasks: they are executed sequentially. The final runtime is computed by adding each individual runtime.
- 2 computers running 2 similar tasks: 2 tasks are modelled over an IA which are sequentially executed in one computer and in parallel with the other pair (and the other IA). The final runtime is the maximum of both sequential executions.
- 4 computers running 1 task: maximum parallelism with 4 agents. The final runtime is the maximum of the four executions.

Table 47. Performance tests for the execution of 4 similar learning tasks with different parallel conditions. Individual and total runtimes are presented.

Domain	1 node	2 nodes	4 nodes
Breast cancer	1083 s.	1093 s.	1095 s.
Colon cancer	627 s.	667 s.	705 s.
Lung cancer	980 s.	992 s.	1029 s.
Ovarian cancer	715 s.	812 s.	841 s.
Total	**3405 s.**	**2085 s.**	**1095 s.**

One can see that the improvement is very significant and proportional to the degree of parallelism (see Figure 34). It is also interesting that the execution overhead introduced by the agent and platform management is negligible in relation to the sequential approach. This is due to the complexity and heavyweight nature of tasks.

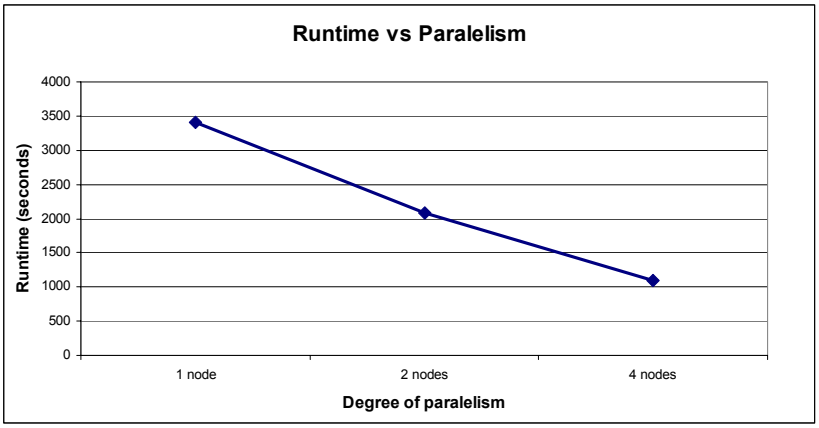

Figure 34. Increase of performance in relation to the degree of parallelism.

The following test covers the full parallel execution of a domain. In this case, we pick up one domain and execute 2 taxonomic levels sequentially (using one computer) and in parallel (using 4 computers) with automatic distribution of the work load in function of the available computational resources (following the implemented scheduling policy). From the performance obtained, we can check, in a real situation, the degree of parallelism one can except from our MAS and the behaviour of the implemented task planner.

First, we have executed the taxonomic learning (two levels) for the *Sensor* domain, which results, for the specified parameters, in 12 immediate subclasses that should be analysed. When running the full process (*sensor*+12 subclasses analyses) in one computer, it takes 6606 seconds. Next, the same test with the same search parameters is executed in a parallel environment with 4 nodes. As a result, the same amount of subclasses is obtained, but the process is finished in 2944 seconds. This represents an improvement of 224% when the hardware is increased by a 400%. Examining the execution trace and representing the task-node assignation at each moment, we can compose the Gantt diagram shown in Figure 35 (each task corresponds to each coloured interval).

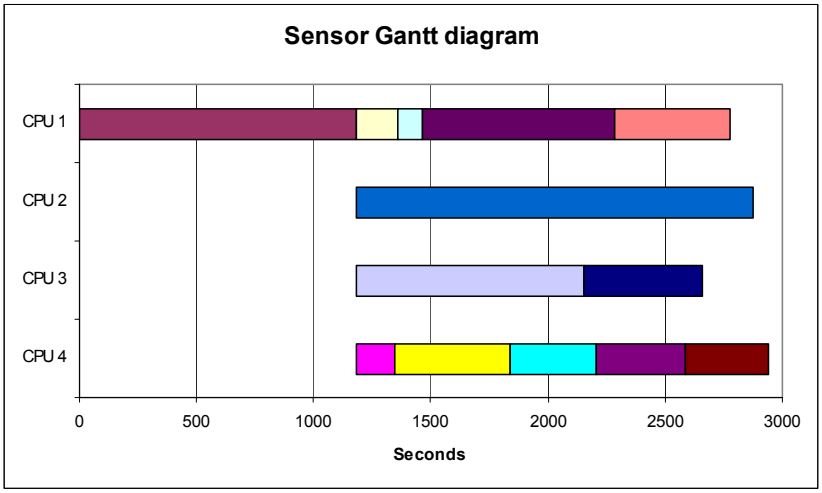

Figure 35. Distribution of taxonomic learning tasks among 4 CPUs for the *Sensor* domain.

One can see that the first task (the *Sensor* analysis) is executed alone in computer 1, as no other concepts have been discovered. Once its analysis is finished and 12 new tasks (subclasses) has been discovered, the maximum degree of parallelism is achieved, as the scheduler assigns tasks to free nodes whenever they are available. At the end, the system has to wait until all nodes have finished as no more tasks (we have limited the analysis to two taxonomic levels) remain. In consequence, the final performance is restricted by the sequential parts of the non-parallel implementation.

Regarding the runtime required for each learning task, as stated in §7.1, they depend linearly on the number of queries for statistics performed to the web search engine (see Figure 36). In this case, however, there is more variability due to the higher degree of parallelism and the finer granularity of the measured tasks.

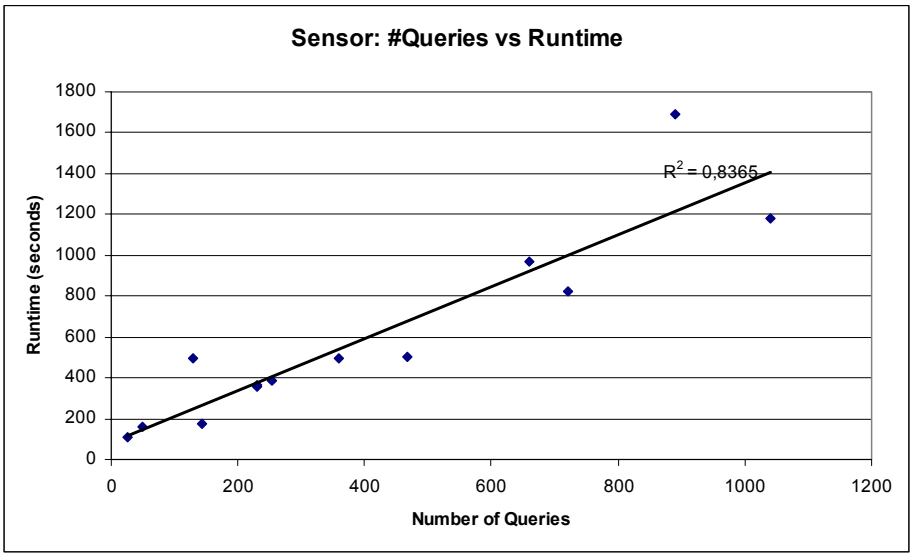

Figure 36. Number of queries *vs* runtime for each learning task (subclass) of the Sensor domain. A linear dependence can be inferred.

In the next test, we have picked a much wider domain (Cancer) and perform the same executions. This has result in 49 immediate subclasses to analyse. When executing the learning process in one computer, it lasts a total of 16505 seconds. Performing the same execution in parallel with 4 computers, the total runtime is lowered till 5634 seconds. This represents a performance improvement of 292% with and hardware increase of 400%. In this case, the task distribution among the available nodes is shown in Figure 37.

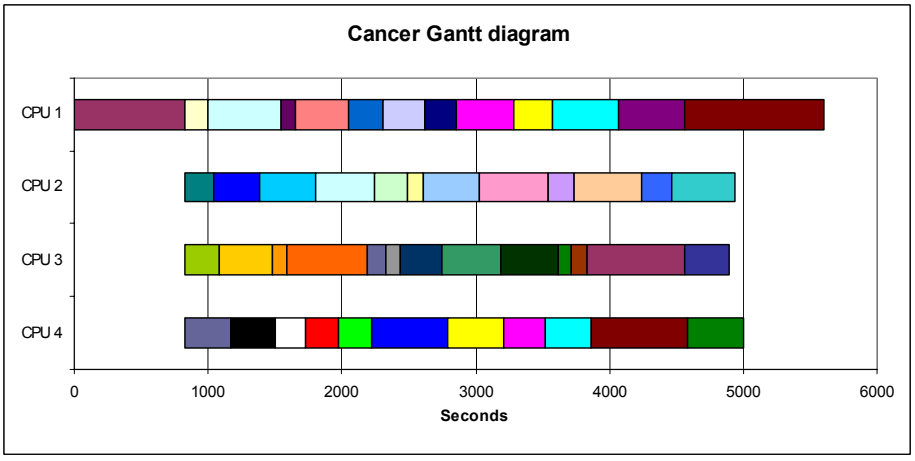

Figure 37. Distribution of taxonomic learning tasks among 4 CPUs for the *Cancer* domain.

In this case, the non fully parallel intervals are shorter than in the previous example, due to the higher amount of tasks to execute. One can see that the potential improvement of this parallel approach is higher as more tasks (concepts) to execute are available. In a complete learning process (involving hundreds of multi-level taxonomic and non-taxonomic analyses) the percentage of fully parallel execution is much higher in relation to the sequential parts and the throughput improvement will tend to be similar to the hardware resources provided. As shown in the first test of this section, the overhead introduced of the agent and parallelism management are negligible in relation to the size of the tasks to execute.

Again, the runtime of each task depends linearly on the number of queries performed (see Figure 36).

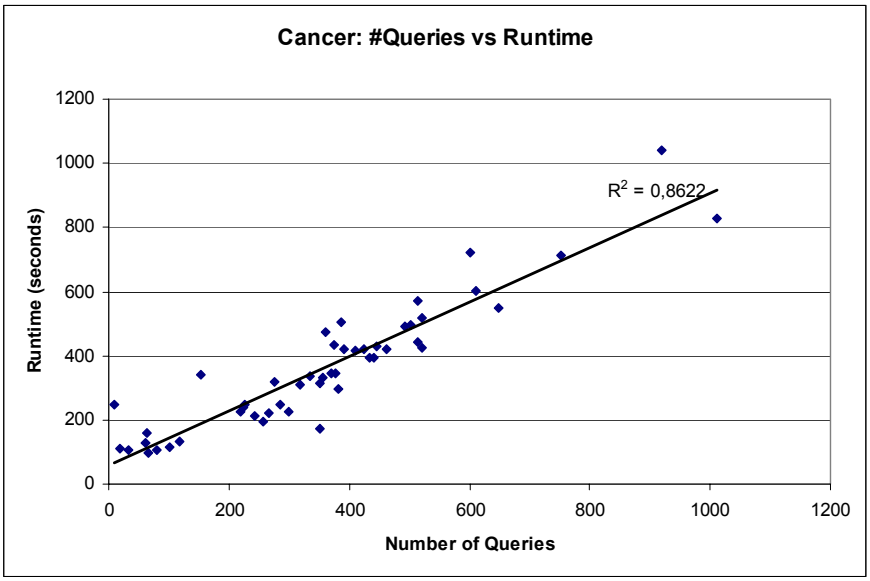

Figure 38. Number of queries *vs* runtime for each learning task (subclass) of the Sensor domain. A linear dependence can be inferred.

Considering this new execution environment, we can compare it to the sequential approach in relation to performance. As presented in §7.1.1, the computational cost in a sequential approach is a polynomic function of the number of concepts retrieved at each iteration, multiplied an amount of times defined by the maximum depth of the relationships. Without considering the limitations introduced by the available hardware or the Web search engine, in the distributed approach we can parallelise the full set of concepts retrieved, reducing the runtime of one iteration to T (the runtime required to evaluate one concept, depending linearly on the number of web queries). At the end, we are able to obtain a runtime of T^{max_depth}, where the exponent is maximum depth of the taxonomic and non-taxonomic relationships (among 2 and 4). In consequence, we can reduce the runtime from to $T(taxo\_concepts+notaxo\_concepts)^{max\_depth}$ to T^{max_depth} using a $(taxo\_concepts+notaxo\_concepts)$ degree of parallelism.

In the real world, however, it is very unlikely to have available such an amount of hardware and, in consequence, the real runtime will depend on the maximum degree of parallelism that we are able to achieve.

In this sense, other interesting questions about the parallelisation of learning tasks that we have observed during the development are the following:

- Ideally, each learning task (modelled by the corresponding IA) will be executed exclusively in one computer. However, due to the limitation of computer nodes, in our tests, we have determined that one computer with enough hardware resources (*i.e.* 2 Gigs of Ram, Pentium4 CPU or later) is able to execute among 6 to 8 tasks (and IAs) before the performance is degraded due to the concurrence overhead.
- When executing several learning tasks in parallel, the Web search engine employed may receive a considerable amount of queries at the same time. MSNSearch scales quite well under those heavy load conditions but, for other search engines, the performance is degraded. This is motivated because, in our case, several computer nodes of the network share the same external IP and they are identified as the same machine by the search engine. In those cases, access control policies are applied, decreasing the query priority and, in consequence, increasing the response time. In order to minimize this problem, each computer executing IAs should have a different IP.

7.1.5 Formal representation of the results

There exist several standard ontology languages such as RDF[17], DAML+OIL[18] or OWL[19]. This last one, the *Web Ontology Language* is the newest one. It is a semantic mark-up language specially designed for publishing and sharing ontologies on the World Wide Web. It is developed by the WebOnt group as a vocabulary extension of RDF and is derived from DAML+OIL. It is designed to be used by applications that need to process the content of information and facilitates greater machine interpretability by providing additional vocabulary along with a formal semantics [Fensel *et al.*, 2001]. OWL is supported by many ontology visualizers and editors. There exist three different OWL specifications in which a particular ontology can be defined:

- *OWL full* is meant for users who want maximum expressiveness and the syntactic freedom of RDF with no computational guarantees. Inference is undecidable.
- *OWL DL* supports those users who want the maximum expressiveness while retaining computational completeness. It is based on Description Logic, provides a well defined semantic and allows inferences (there are available reasoners such as FACT++[20], Pellet[21] or F-OWL[22]).
- *OWL Lite* supports those users primarily needing a classification hierarchy and simple restrictions.

[17] Resource Description Framework: http://www.w3.org/RDF
[18] DAML+OIL WebOntology Language: http://www.w3.org/TR/daml+oilreference
[19] Web Ontology Language: http://www.w3c.org/TR/owl-features/
[20] http://owl.man.ac.uk/factplusplus/
[21] http://www.mindswap.org/2003/pellet/
[22] http://fowl.sourceforge.net/

As our main purpose is the representation of domain knowledge with full expressiveness but allowing inference, we use OWL DL. From the full set of ontological components supported by the OWL specification, we have used the following ones:

- *RDF Schemas Features*: they define basic ontological components.
  - *Classes*: are sets of individuals with common characteristics. In our case they correspond to domain concepts.
  - *Subclasses*: define class specializations by constraining their coverage. Class hierarchies can be specified by making one or more statements that a class is a subclass of another class. They are retrieved through several iterations of the taxonomy learning procedure.
  - *Individuals*: Individuals are the objects in the domain. In our case they are limited to named entities found during the ontology learning.
  - *Properties*: can be used to state relationships between individuals or from individuals to data values. Relationships in OWL are binary. There exist three types of properties:
    - *Object Property*: it establishes relationships between pair of individuals. We have used them to define verb-labelled non-taxonomic relationships.
    - *Datatype Property*: relates an individual to a data value (*int, string, float*, *etc.*). Can be considered "attributes". We have used them to define the class "features" extracted during the post-processing stage.
    - *Annotation Property*: used to attach metadata (*e.g. version*, *author* or *comment*) to classes, individuals or properties. We have used them to add meta-information about the learning process and the web content.
- *Equality* and *Inequality*: allows expressing equalities and inequalities between ontological components:
  - *Equivalent Classes*: it states that the set of individuals belonging to a particular class is the same as the set corresponding to another class. It may be used to create synonymous classes. We have used it to define the alternative class names that are referred to the same concept (during the linguistic analysis and ontology post-processing stage).
- *Property Characteristics*: they define the semantics of properties:
  - *Inverse Property*: one property may be stated to be the inverse of another property. We have used it to define inverse semantic relationships between the passive and active voice of a non-taxonomic relationship.
- *Property Restrictions*: they define the "meaning" of classes by specifying a statement between a pair of entities (classes or datatypes) and a property with specific semantics:
  - *SomeValuesFrom*: a particular class may have a restriction on a property that at least one value for that property is of a certain type. We have used this type of restriction to state non-taxonomic relationships between a pair of classes using a previously defined verb-labelled property.
  - *HasValue*: for a particular class, a default value for a datatype property is stated. We have used this type of restriction to define the appropriate Boolean values of the automatically discovered domain features (previously defined as datatype properties) in the corresponding taxonomic level.

See Figure 39, Figure 40, Figure 42 and Figure 41 for examples of the concrete OWL notation used in some of the mentioned ontological components.

```
<owl:Class rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:breast_cancer">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty
rdf:resource="http://grusma.etse.urv.es/ontologies/cancer/#D:Is_OPERABLE" />
      <owl:someValuesFrom>
        <rdfs:Datatype rdf:about="http://www.w3.org/2001/XMLSchema#boolean"/>
      </owl:someValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty
rdf:resource="http://grusma.etse.urv.es/ontologies/cancer/#D:Is_RECURRENT" />
      <owl:allValuesFrom>
        <rdfs:Datatype rdf:about="http://www.w3.org/2001/XMLSchema#boolean"/>
      </owl:allValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Class rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:cancer">
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
```

Figure 39. *breast_cancer* is subclass of *cancer* and has two features: *Is_OPERABLE* and *Is_RECURRENT* .

```
<rdf:Description
rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#I:American_breast_cancer">
  <rdf:type>
    <owl:Class rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:breast_cancer">
    </owl:Class>
  </rdf:type>
</rdf:Description>
<rdf:Description
rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#I:NCCN_breast_cancer">
  <rdf:type>
    <owl:Class rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:breast_cancer">
    </owl:Class>
  </rdf:type>
</rdf:Description>
```

Figure 40. *American_breast_cancer* and *NCCN_breast_cancer* are instances of *breast_cancer.*

```
<owl:Class rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:intestinal_cancer">
  <owl:equivalentClass>
    <owl:Class
rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:intestine_cancer">
    </owl:Class>
  </owl:equivalentClass>
  <rdfs:subClassOf>
    <owl:Class rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:cancer">
    </owl:Class>
  </rdfs:subClassOf>
</owl:Class>
```

Figure 41. *intestinal_cancer* and *intestine_cancer* are stated to be equivalent.

```
<owl:Class rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:chemotherapy">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty
rdf:resource="http://grusma.etse.urv.es/ontologies/cancer/#P:reduces" />
      <owl:someValuesFrom>
        <owl:Class
rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:breast_cancer">
        </owl:Class>
      </owl:someValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty
rdf:resource="http://grusma.etse.urv.es/ontologies/cancer/#P:is_used_in" />
      <owl:someValuesFrom>
        <owl:Class
rdf:about="http://grusma.etse.urv.es/ontologies/cancer/#C:liver_cancer">
        </owl:Class>
      </owl:someValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Figure 42. *chemotherapy* has the following non-taxonomic relationships: "*chemotherapy reduces breast_cancer*" and "*chemotherapy is_used_in liver_cancer*".

7.1.6 Prototype components

The implemented application is fully written in Java in order to achieve good interoperability with the freely available tools for Web and NL processing. Concretely, the main tools and libraries used in the development of the prototype:

- JADE 3.3[23]: the *Java Agent Development Framework* is the tool used to implement the presented Multi-Agent system. It provides a set of programming libraries for implementing agents and an execution environment in which to perform the deployment. It follows the FIPA[24] standards about how agents should be defined in order to guarantee the interoperability between applications. This version includes features about agent mobility that have been extensively used in our implementation in order to provide a fully distributed solution.
- English Stemmer 1.0[25]: it provides a stemming algorithm to find the morphological root of a word in the English language. This has been extensively used in order to detect equivalent forms of expressing the same ontological concept.
- Text processing tools from OpenNLP Tools 1.1[26]: is a mature Java package that hosts a variety of *Natural Language Processing* tools which perform sentence detection, tokenization, pos-tagging, chunking and parsing, allowing morphological and syntactical analysis of texts. It is based on maximum entropy models [Borthwick, 1999] and, in consequence it requires annotation samples. Models[27] of annotation for each task exhaustively trained for the English Language are used (pro-

[23] http://jade.tilab.com/
[24] http://www.fipa.org/specs/fipa00001/
[25] http://sourceforge.net/projects/stemmers/
[26] http://opennlp.sourceforge.net/
[27] http://cvs.sourceforge.net/viewcvs.py/opennlp/models/

vided "officially" by the developers of the library). It has been used to analyse interesting pieces of web content (*i.e.* a pattern matching found within a particular web site). Even though the computational cost of this analysis can be high when evaluating large texts, only the particular sentence in which the keyword has been found is considered. Concretely, we have used the sentence detector and the morphological and syntactical –parts of speech- analyser.

- Named-entity tool from OpenNLP Tools 1.1[28]: it is able to detect some word patterns like *organization*, *person*, and *location* names using, again, maximum entropy models. Previously trained model files[29] with annotation examples for those categories are used in this task. We have used it for evaluation purposes, comparing its tagged terms with our extracted named entities over the same sources.
- Html Parser 1.6[30]: this is a powerful HTML parser that allows processing web content. It has been used to extract automatically clear text contained in a web resource.
- Web search engine APIs: one of the most important parts of the implemented system, as they provide access to the Web search engine services. We have extensively used them to retrieve ranked lists of web resources, statistics, snippets and html caches. In order to avoid an abusive use of a particular engine, several alternatives have been implemented.
  - Google Web API[31]: this is the library of functions that the Google search engine provides to programmers to allow them to make queries and retrieve search results. However, the maximum amount of daily search queries per account is restricted to 1000.
  - Yahoo Search 1.1.0[32]: in the same way as Google, Yahoo recently provided an API for accessing Yahoo Search services. Similarly, it is also limited to a maximum of 5000 queries per day, account and IP.
  - For the other search engines that have also been considered (Altavista, AlltheWeb, MSNSearch), ad-hoc libraries for performing web queries and parsing the page of results have been implemented. They are based on analysing the query language used by each search engine and studying the format in which result pages are presented. In consequence, this is not a flexible solution as any change in both the query language and/or the result page format will require modifications of the implemented modules. In addition, many search engines impose IP limitations (MSNSearch is the only one offering an unlimited access).

- OWL API 1.4[33]: it is one of the first libraries providing functions to construct and manage OWL files. As we have selected OWL as the formal language for representing our learned domain ontologies, this library is used to write them in the corresponding format.

[28] http://opennlp.sourceforge.net/
[29] http://cvs.sourceforge.net/viewcvs.py/opennlp/models/
[30] http://sourceforge.net/projects/htmlparser
[31] http://www.google.com/apis/
[32] http://developer.yahoo.com/search/
[33] http://sourceforge.net/projects/owlapi

- WordNet 2.0[34] (more details in §6.2): one of the latest versions of the WordNet semantic electronic repository. As described in chapter 6, it has been extensively used for evaluation purposes.
- JWNL WordNet API 1.3[35]: offers an interface for accessing WordNet 2.0 from Java programs. It allows querying words specifying a morphological category and retrieving corresponding synsets and glosses. Moreover it also allows exploring the semantic network that links WordNet's entities.
- WordNet::Similarity 1.03[36]: offers an implementation of some WordNet-based similarity and relatedness measures between terms (more details in §6.2). Concretely, it works in conjunction with a WordNet 2.0 instance to provide the following measures: *Path length, Leacock & Chodorow, Wu & Palmer, Resnik, Hirst & St-Onge, Jiang & Conrath, Extended Gloss Overlaps, Gloss Vector, Gloss Vector (pairwise)* and *Random*. For evaluation purposes we have compared our Web-based relatedness scores with *Gloss Vector* which seems to offer the best quality measures [Patwardhan and Pedersen, 2006]. As a similarity measure used to evaluate the designed methods for dealing with semantic ambiguity, we have employed a simple *path length* derived measure as, in those cases, we are only interested in the WordNet's *is-a* hierarchies. However, as the package is implemented in Python, a wrapper module has been implemented to allow a transparent communication with our Java-based prototype.
- VerbNet 1.5 & API[37]: it is an XML-based electronic repository which contains semantic information about verbs. As introduced in §5.4.1.3, it includes refinements of Levin's classification of verbs, WordNet synsets and additional information such as thematic roles or syntactic frames. A Java-based API is provided. We have used to classify and to add semantic content to our verb labelled non-taxonomic relationships.

Moreover, we have used Protégé 3.1[38] as a visualization and edition tool. Protégé represents the latest in a series of interactive tools for knowledge-system development. It facilitates the construction of knowledge bases in a principled fashion from reusable components. It allows a variety of plug-ins to facilitate customization in various dimensions. From April 2003, an OWL extension of Protégé has been developed, featuring access to description logics reasoners and graphical editors. Concretely, we have used the OWLviz and Jambalaya plug-ins to create visual representations of an OWL file (see Figure 43 and Figure 44).

[34] http://wordnet.princeton.edu/
[35] http://jwordnet.sourceforge.net/
[36] http://www.d.umn.edu/~tpederse/similarity.html
[37] http://verbs.colorado.edu/~mpalmer/projects/verbnet/downloads.html
[38] http://protege.stanford.edu/download/download.html

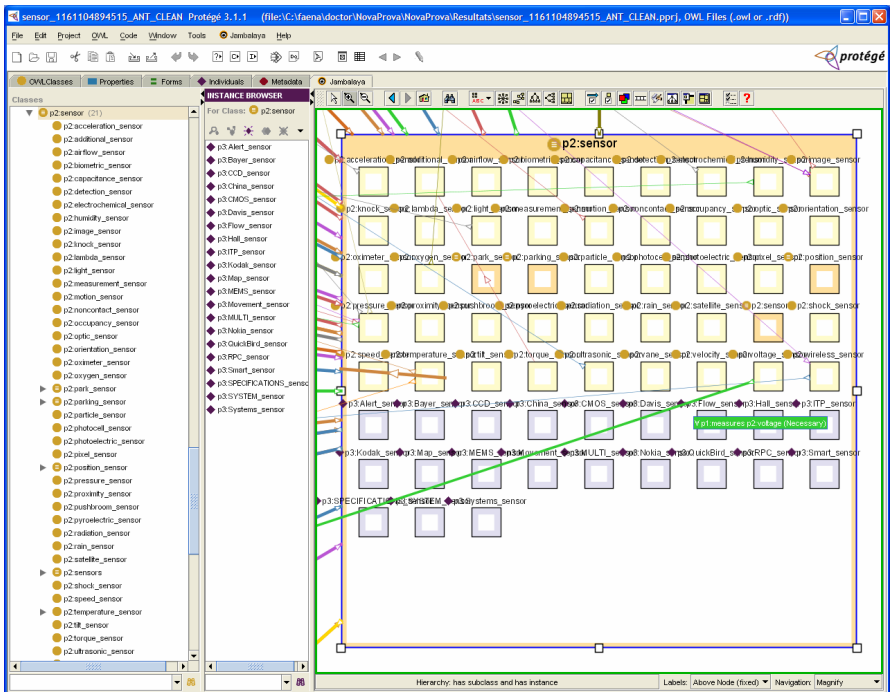

Figure 43. Taxonomic and non-taxonomic graphical visualization of the *Sensor* domain in Protégé with Jambalaya plug-in.

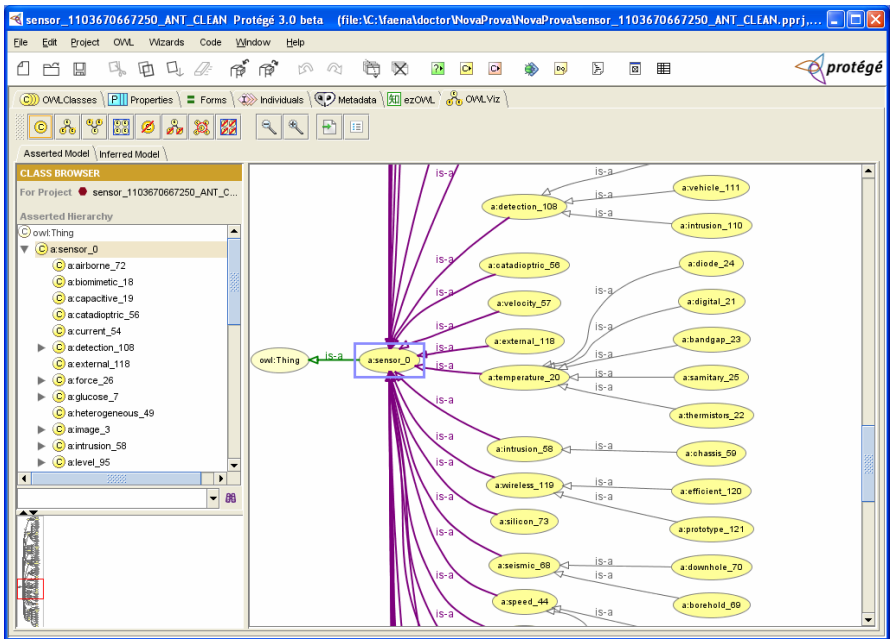

Figure 44. Taxonomic visualization for the *Sensor* domain in Protégé with OWLviz plug-in.

7.1.7 Domain ontology visualizer

Even though Protégé is certainly useful for managing ontologies, it does not scale well with big ontologies (with thousands of concepts). In consequence, due to the size of our domain ontologies, the program's performance is easily degraded and the visualization becomes confusing and overwhelming. Moreover, additional meta-information included in our domain ontologies (mainly statistics, learning traces and web resources) cannot be visualized.

For those reasons, we have developed an especially designed tool for visualizing our domain ontologies with the following features (see an example in Figure 45):

- Thanks to the efficient *ad-hoc* programming that includes a complete loading of the ontology's content on memory over especially designed data structures, it scales well with huge ontologies, maintaining a good visualization response time.
- It provides an incremental visualization centred on the domain's initial concept. In this manner, the user can recursively explore domain branches and nodes showing those parts in which he is interested. Expansion/collapse and *drag&drop* of graphical nodes are fully supported.
- It provides a two dimensional representation of taxonomic and non-taxonomic verb labelled relations.
- It used ontology meta-information to enrich the visualisation, using colours, shapes and sizes as additional visualisation dimensions to represent statistical relatedness measures for concepts and relationships.
- This quantitative meta-information also allows the implementation of visualization filters. In this manner, the user can, for example, specify a minimum relatedness value for the visualized classes and relationships, obtaining a partial visualization of the domain ontology containing only the most related entities.
- It offers direct access to the categorised list of associated Web resources (also contained in the domain ontology). In this manner the user can consult, at every moment, corresponding Web resources related to the visualized concepts.
- It has been implemented as a Web applet, offering complete integration with web-based interfaces.

The only limitations are that only an OWL subset (the part used for constructing our domain ontologies) is supported and it does not support editing ontologies.

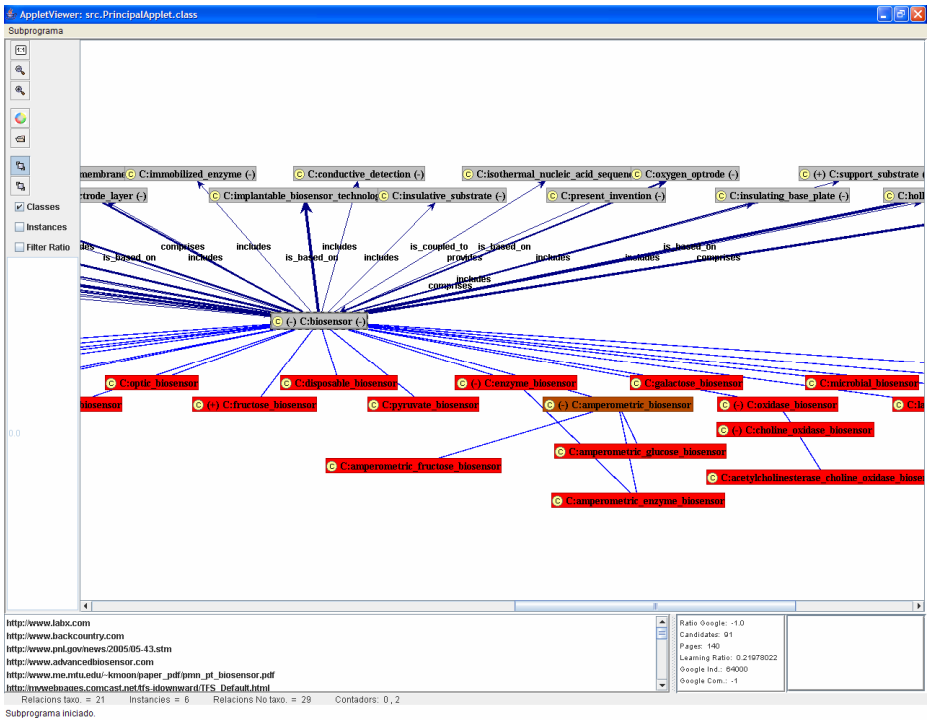

Figure 45. Especially designed and implemented domain ontology visualization applet.

7.2 Applications

Ontologies have many interesting applications. The fact of providing machine readable semantic content to a computer program dealing with a certain domain of knowledge makes themselves an essential component to many knowledge-intensives services like:

- Information Extraction: [Buitelaar *et al*., 2006], [Stevenson *et al*., 05], [Maedche *et al*., 2003].
- Information Retrieval (Semantic Search): WebKB [Martin and Eklund, 2000], SHOE [Helflin and Hendler, 2000], OntoSeek [Guarino *et al*., 1999].
- Question Answering: [Sinha and Narayanan, 2005], [Schlobach *et al*., 2004], Aqualog [Lopez and Motta, 2004], [Pasca and Harabagiu, 2001].
- Machine Translation: [Nirenburg *et al*., 2004].
- Business Process Modeling: [Uschold *et al*., 1998].
- Information Integration: [Kashyap, 1999].
- Knowledge Management (including the Semantic Web): [Fensel, 2001], [Mulholland *et al*., 2001], [Staab and Schnurr, 2000], [Sure *et al*., 2000].
- Software Agents: [Gluschko *et al*., 1999], [Smith and Poulter, 1999].

In addition to the several mentioned benefits of using ontologies in knowledge related tasks, in this section we present some practical applications of our learning methodologies and obtained results for solving some real world problems. Concretely, first, we present a way to structure domain related web resources in a meaningful taxonomic way that is fully integrated with our learning methodology. Following the same principle, secondly, we introduce a way for automatically structuring web-based digital repositories using our taxonomy learning methodology. Next, we provide an example of application of our potential results to improve Web information retrieval using a knowledge-based searching platform.

7.2.1 Structuring web sites

One important application of term taxonomies in the Web environment is the meaningful organization of available web resources in order to ease the way in which the user finds and access the desired information. Hierarchical classifications are quite useful for document classification and retrieval. Users browse hierarchies of concepts and quickly access the documents associated with the different concepts.

As shown in §3.4.1, taxonomic search engines perform in that way, using a manually created (as Yahoo directory) or automatically obtained (as Clusty) structure of terms that are relevant for a domain, classifying web sites according to the available categories. That way of representing information is an improvement over classical ranked lists of webs [Magnini *et al.*, 2003], especially when the amount of returned results is overwhelming. However, as introduced in §3.4.1, the current state of both manual and automatic classification engines has serious drawbacks that impact in the quality of the results.

As our proposal performs a wide analysis over the Web in order to extract a rich repository of concepts and semantic relationships, we can take advantage of the ontology learning process. Concretely, we can classify the returned and analysed sets of web resources obtained from the search engine into that meaningful organisation. So, at the end of the process, the user will not only be able to explore relevant knowledge regarding a domain in an ontological fashion, but also to obtain the web resources that cover each concept as a topic hierarchy of web resources [Lawrie and Croft, 2003].

In our approach, each class and instance, stores the set of web sites from where it was selected (*e.g.* the *skin cancer* contains the set of web sites returned by the search engine when setting the keyword *cancer* that contain the candidate concept *skin cancer*).

Named entities are particularly interesting as, if the name is restrictive enough, it is typical than the first(s) web site(s) proposed into the hierarchical classification of web resources corresponds to the homepage for that entity.

In addition to the conceptual classification of web resources according to the discovered categories, the extra information obtained through the analysis of the web content may be useful. In some cases, we can categorize each individual web site with the context in which the covered concept is applied. Concretely, in the noun phrase-based analysis, the immediate *posterior* word for the initial keyword may bring new

information about the *context of application* [Grefenstette, 1997]. In this case, those concepts are used to categorize the set of web sites associated to each class. For example, if we find that for a web site associated to the class *breast cancer* this keyword is followed by the word *research,* the web site will be categorized with this word that represents a *context of application*. This provides the user with richer information and allows him a higher level of understanding of the available resources, minimizing the selection time of the suitable ones according to his preferences.

Some examples of the proposed topic classification of web resources obtained for the *Lung Cancer* domain are shown in Figure 46.

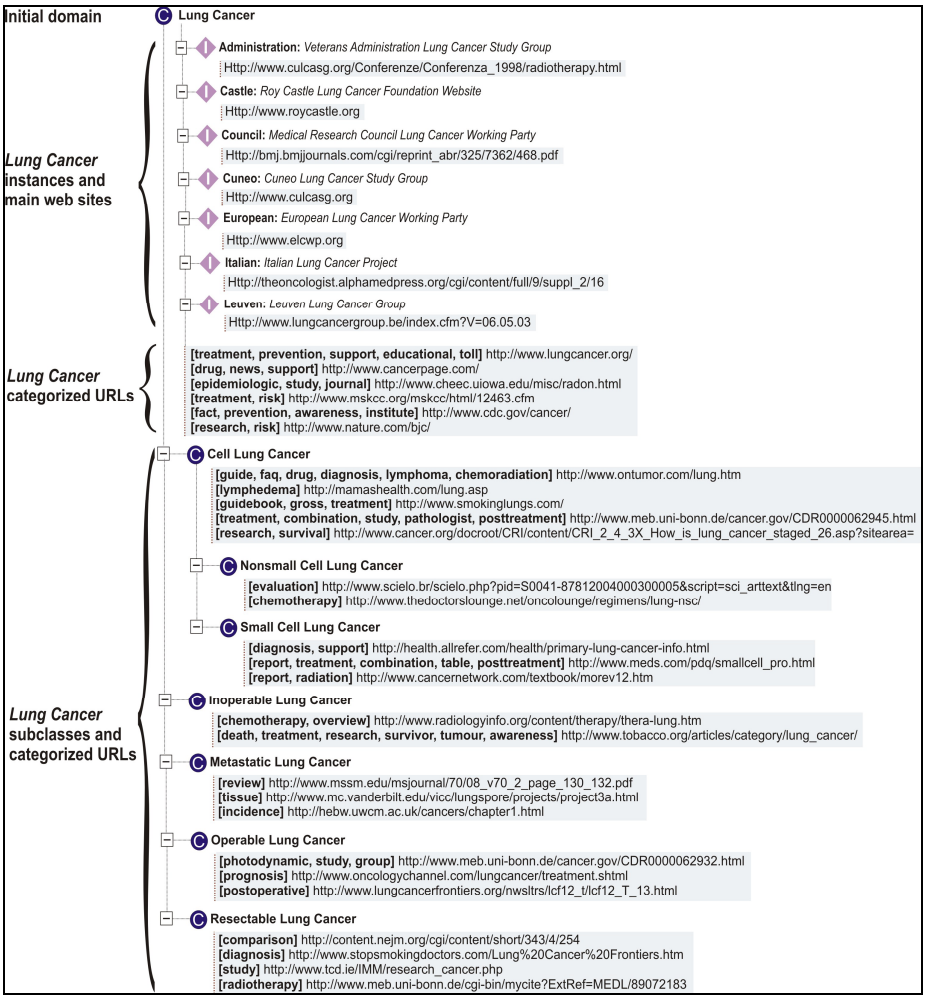

Figure 46. Example topic hierarchy of web resources in the *Lung Cancer* domain according to the discovered knowledge (instances and subclasses).

7.2.1.1 Evaluation

As a measure of comparison of the results' quality against similar available systems, we have evaluated precision and recall of our taxonomies against hand-made web directory services and taxonomic search engines (a comparison of those approaches was presented in §3.4.1). On the one hand, we have used Yahoo directory service, as it can be considered one of the most popular human-made directories. On the other hand, we have selected the taxonomic search engine Clusty that automatically presents concept hierarchies using clustering techniques. In both cases, we query the search engine and collect the returned topic categorization of web sites, considering it as a domain taxonomy. Those taxonomies are then concept-per-concept evaluated against a gold standard and/or a domain expert in the same way as described in §6.3. As a result, we can compute precision and recall for the different approaches. Local recall is only computable for our approach because rejected candidates are not available for the compared search engines.

As an example of evaluation, we present the results obtained for two well distinguished domains: a medical one (*Cancer*) and a technological one (*Biosensor*). The first one has been presented in §6.3 and evaluated against the MESH *neoplasm* classification using the same evaluation criteria. The second one is a very specific technological concept that is not found in typical semantic repositories (like WordNet). Even though the domain is highly structured, there does not exist a global consensus about the specific classification. Only the IUPAC (*International Union of Pure and Applied Chemistry*) defines some general classes and different forms of classification of biosensors according to their specific properties. Concretely, according to the specific measured entity, at least 100 different classes can be defined. This last measure has been considered when computing the global *recall*. The particular domain evaluation has been carried by a domain expert. The specific evaluation criteria is very similar to the *sensor* evaluation presented in §6.3, considering physical magnitudes and measuring principles and technologies as valid specialisations.

The evaluation of the results presented by the three approaches is presented in Figure 47 and Figure 48. Our results have been obtained with the same execution conditions presented for the taxonomic evaluation in §6.3.

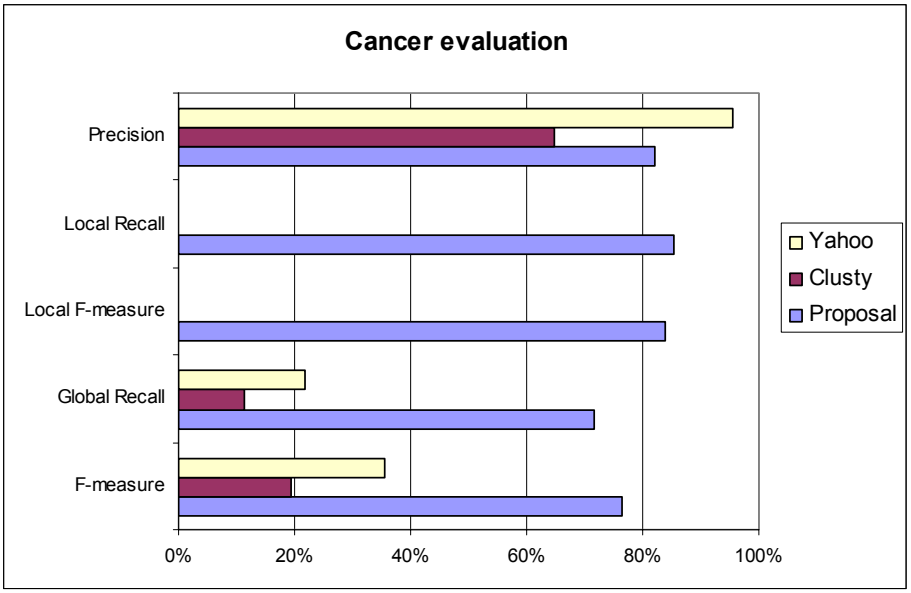

Figure 47. Evaluation results for the *Cancer* taxonomy for the proposed methodology against several taxonomic Web search engines considering the MESH standard classification.

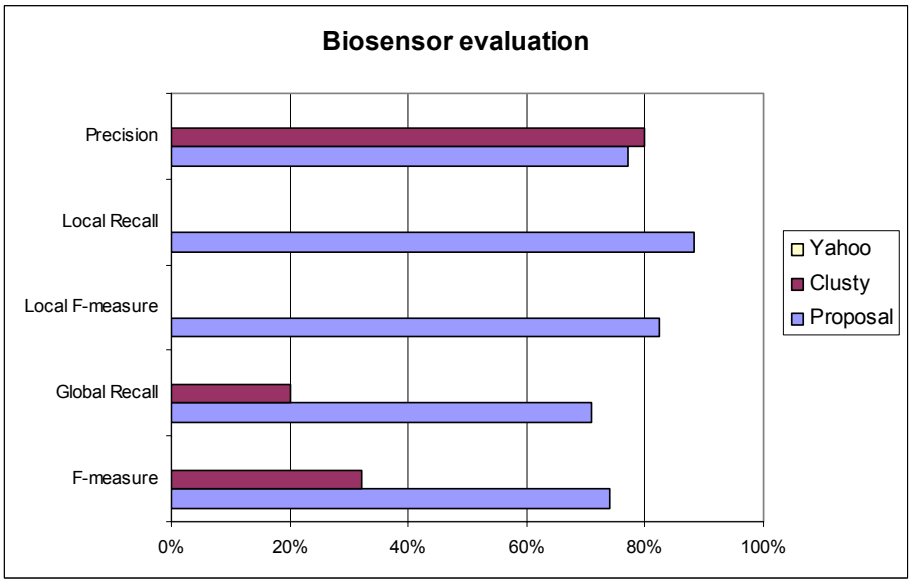

Figure 48. Evaluation results for the *Biosensor* taxonomy for the proposed methodology against a taxonomic Web search engine considering a domain expert's opinion.

Observing the figures, we can conclude that, as stated in §6.3, the correctness of our candidate selection procedure is high as the number of mistakes (incorrectly selected and rejected concepts from the candidate list), represented by the precision and local recall measures, is maintained around a 15-20% in both cases.

Compared with other web-based systems from the topic categorization point of view, our proposal surpasses easily their structuring capabilities. Comparing to the Yahoo directory, we can see that, although its *precision* is the highest, as it has been manually composed, the number of results presented (*recall*) is quite limited. Concretely, for the *cancer* domain (see Figure 47), we achieve an *F-Measure* of 76.39% that easily doubles the 35.49% presented by Yahoo. In addition, for the much more concrete technological domain, *biosensor* (see Figure 48), Yahoo is not able to provide any classification, showing the limited coverage of manual attempts of structuring information (WordNet does not contain that concept either). Compared with the automatic taxonomic search engine, Clusty, its *precision* is similar to the one presented by our proposal (both present similar mistakes due to their automatic and unsupervised nature) but its recall is very limited (it is able to return very few subclasses for the *biosensor* domain). In consequence, we achieve an *F-Measure* of 76.39% and 73.9% for the *cancer* and *biosensor* domains respectively, in comparison to the 19.45% and 32% presented by Clusty.

This comparison can give us an idea of the potential improvement that our domain structuring may bring to the topic categorization of web resources.

7.2.2 Automatic structuring of digital libraries

Digital libraries are an invaluable repository of information. Web-based digital libraries (*e.g*. Citeseer, PubMed, *etc*.) provide an environment in which the scientific production for a particular domain is stored, offering a trusted, updated and immediate repository of information. However, due to the success of these initiatives, the amount of available resources is beginning to be, so huge that the difficulty of searching and obtaining the desired information has become a serious problem in a similar way as with the whole Web but in a lower scale [Kobayashi and Takeda, 2000]. That is why the need of tools for information retrieval that ease the way in which those resources are accessed and analyzed has been growing in pair with the information itself.

Similarly to the Web, the most common way for accessing the resources is by means of the keyword-based search engines that many of these libraries incorporate. This type of search usually suffers from two problems derived from the nature of textual queries and the lack of structure in the documents: a) the difficulty to set the most appropriate and restrictive search query, and b) the tedious evaluation of the huge amount of potential resources obtained.

Taking all those points into consideration, we have designed a solution for automatic construction of structured representations (in a taxonomic fashion) of a library's content, according to the main topics discovered for a particular domain. These results are used as concrete queries for retrieving resources from the library's search engine, providing an access similar to a directory service but composed in a

completely automatic and unsupervised way. The premises about the working environment and the learning bases are the same that those presented in chapters 3, 4 and 5 for learning taxonomies.

However, the difference in this case is that we consider the response time as a goal. Certainly, instead of building a domain ontology (with independence of the required time and computational resources), our purpose is to provide a usable and immediate tool for structuring digital libraries with a reasonable response time. In consequence, an especially optimised and adapted learning procedure –omitting some aspects considered in the full ontology learning- has been designed.

7.2.2.1 Constructing topic hierarchies

The base of our proposal is the analysis of the resources available for a specific domain in an electronic repository to detect the main topics covered in it. In order to perform this process automatically and unsupervisedly, two main tasks are performed: *i)* extraction of candidates that represent different topics for the domain and *ii)* evaluation of their relevance in order to select the most representative ones for constructing a taxonomy. In the same way as for the ontological case, our bases are the pattern-based linguistic analysis for extracting candidates and the statistical analyses for computing relevance measures.

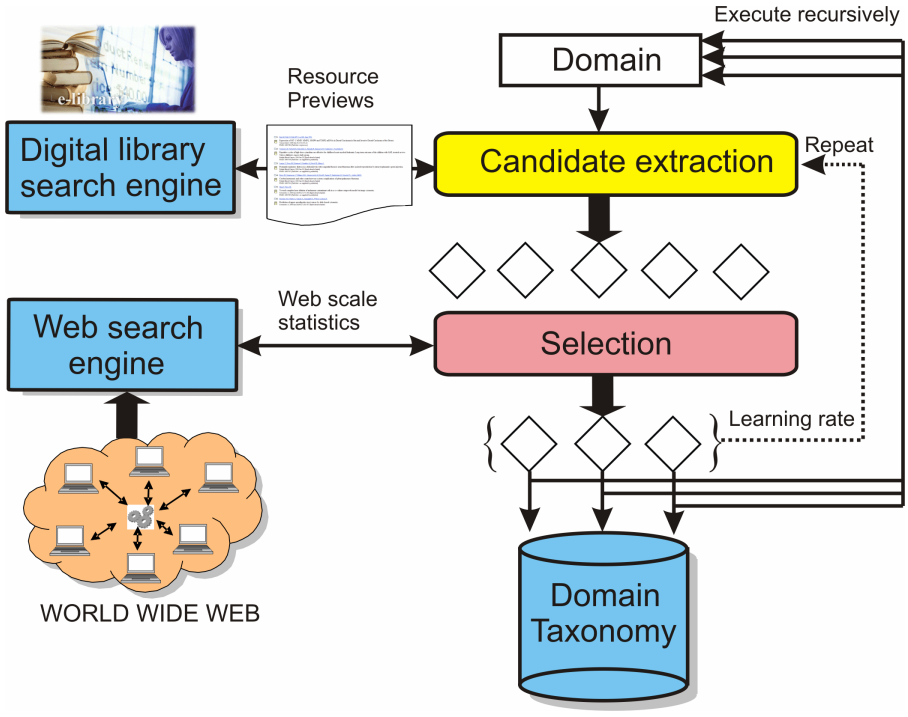

Figure 49. General schema for constructing topic taxonomies from large digital libraries.

As shown in Figure 49, the process is very similar to the taxonomic learning explained in §5.2.2. The differences introduced to improve the response time are:

- The local digital library search engine is used to retrieve resources. However, statistics are extracted by querying a general Web search engine (such as Google) in order to obtain more robust measures (based on a much wider corpus).
- Only noun phrase-based linguistic patterns are considered. The throughput of those patterns in terms of runtime *vs*. extracted knowledge is better than Hearst's ones. This is caused by the higher degree of complexity and the reduced amount of pattern instances retrieved by the later. The way of extracting candidates and computing relatedness measures is the same as described in §5.2.2.1.
- Due to the nature of the resources contained in those digital repositories (typically scientific publications), in many cases it is possible to obtain previews, abstracts or summaries of the particular resources. As we only want to extract the main subtopics for the specified domain, those pieces of text (in conjunction to the title) are typically enough for detecting them. It is usually possible to specify to the repository's search engine to show that information for each item in the results page. Only considering that page (containing dozens of resources) we can extract valuable knowledge without having to analyze large amounts of redundant information and to perform additional access to the web to download each resource.
- Even though considering the mentioned optimization, there can be thousands of potential result pages that should be accessed and processed. However, for many domains, the main interesting topics are a reduced set that can be mostly detected at the beginning of the analysis. For that reason only a reduced set of resource summaries are analyzed. As described in §5.6.2, the system automatically and dynamically decides the number of analyzed resources according to the domain's generality and the potential amount of available subtopics using learning rates as feedback measures.

At the end, we obtain a one level taxonomy that includes the main subtopics available for the particular electronic repository for the specific domain (*i.e.* a topic hierarchy of web resources [Lawrie and Croft, 2003]). Each subtopic represents a specialisation of the initial term. Querying those terms into the repository's search engine, we are able to retrieve resources corresponding to that specialisation. Considering each topic as a new query to the search engine, the user is able to browse the available resources in the same way as a directory service. In this manner, we complement the functionality of the keyword-based search engine but overcoming its main limitations (mentioned in the introduction), which derive from its lack of semantics.

In addition, for each new subtopic of the hierarchy (that at the same time, represents a new more specific domain of knowledge), the same process can be repeated recursively, obtaining a more detailed multi level taxonomy. Through this mechanism, the user can request further details (finer grained hierarchies) in the topics in which he is particularly interested.

As a final note, the characteristics that a particular electronic repository should fulfil in order to be able to apply our methodology are:

- It must have an internal search engine that allows standard query formulations. This is mandatory as it is a crucial part of the proposed methodology.
- It should be possible to present the result in a summarized form, in terms of abstract, previews, *etc*.
- It must allow external access to perform queries and retrieve result sets via a computer program.

7.2.2.2 Prototype

The proposed methodology has been implemented as a web interface that is placed on top of a particular digital library and provides a portal for accessing its resources in a taxonomic directory service fashion. The system controls the access to the library's search engine to retrieve resources according to the extracted topics transparently.

The interface (as shown in Figure 50) provides the main functionalities to manage searches, allowing to refine a particular subtopic or to specify different predefined settings for the mentioned selection and learning thresholds, controlling the behaviour of the system. Concretely, "*Search width*" controls several predefined learning thresholds (from 80% to 50% learning rates), resulting in *simple, medium* and *complex* searches (with better domain's coverage at the cost of increasing the processing time). On the other hand, "*Search precision*" controls the selection threshold (between 0.001 and 0.00001), allowing *high, medium* and *low* precision (with increasing recall). Results are presented as a hierarchy (on the left) in which each item represents an hyperlink to the results of the search associated to that automatically extracted subtopic into the electronic repository.

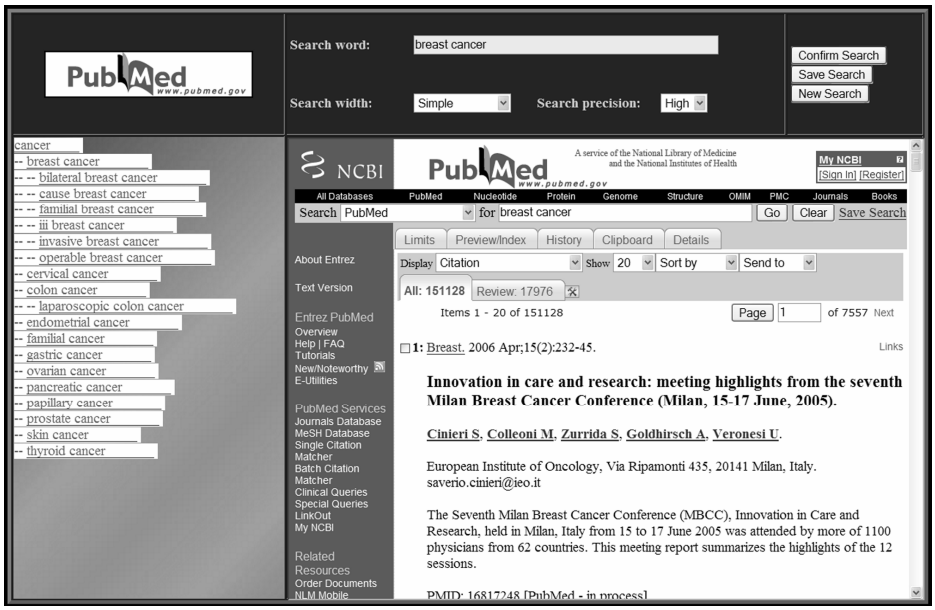

Figure 50. Web interface provided for the PubMed electronic library.

We have adapted the system to the following digital libraries (that fulfil the requisites exposed above):

- The Association for Computing Machinery[39] (ACM): ACM provides the computing field's premier Digital Library and serves its members and the computing profession with leading-edge publications, conferences, and career resources.
- PubMed[40]: PubMed is a service of the U.S. National Library of Medicine that includes over 16 million citations from MEDLINE and other life science journals for biomedical articles back to the 1950s.
- IEEE Computer Society[41]: With nearly 100,000 members, the IEEE Computer Society is the world's leading organization of computer professionals. Founded in 1946, it is the largest of the 39 societies of the IEEE.
- NASA Astrophysics Data System[42]: is a NASA-funded project which maintains three bibliographic databases containing more than 4.7 million records: Astronomy and Astrophysics, Physics, and ArXiv e-prints.

From the user's point of view, the process starts by specifying through the web interface a particular digital library from a list of supported ones. Then, a particular query and the search parameters can be specified in the top frame. Once the search is confirmed, the system executes the described taxonomy learning methodology. When the process is finished the resulting one level taxonomy is presented. By clicking over each topic the system automatically retrieves (by querying the library's search engine) the associated available resources, which are presented in the main frame. At this point the user has also the opportunity of refining a specific subtopic by selecting it and defining a new search (with the desired parameters), in order to obtain a multilevel hierarchy as shown in Figure 50. It is also possible to save and store in HTML format the taxonomies obtained through several recursive searches.

This results in a system that is able to return automatically, depending on the specific library and searching parameters, a hierarchy of topics for every possible domain from less than one minute (for small general searches useful for casual users) to half an hour (for enormously detailed searches useful for researchers or web managers).

7.2.2.3 Evaluation

The evaluation is performed in the same way as for the taxonomic case (described in §6.3). The list of subtopic candidates of the initial concept, which are finally selected or rejected, is manually evaluated. Checking the presence or absence of the extracted concepts in a domain's standard classification and comparing it to the decision of the selection procedure, we can compute the amount of correctly and incorrectly classified terms and measure the performance of the proposed algorithm.

---

[39] http://www.acm.org/

[40] http://www.ncbi.nlm.nih.gov/entrez/query .fcgi? DB=pubmed

[41] http://www.computer.org/ portal/site/ieeecs/index.jsp

[42] http://adswww.harvard.edu/

sensor
-- accelerate sensor
-- acoustic sensor
-- active sensor
-- adaptive sensor
-- air sensor
-- airborne sensor
-- amperometric sensor
-- angle sensor
-- aperture sensor
-- array sensor
-- bio sensor
-- biological sensor
-- boom sensor
-- cantilever sensor
-- capacitive sensor
-- ccd sensor
-- chemical sensor
-- chemochromic sensor
-- classification sensor
-- cmos sensor
-- common sensor
-- compact sensor
-- control sensor
-- cophasing sensor
-- current sensor
-- curvature sensor
-- detection sensor
-- disparate sensor
-- displacement sensor
-- disposable sensor
-- dual sensor
-- edge sensor
-- electrode sensor
-- elevation sensor
-- eo sensor
-- fiber sensor
-- fibre sensor
-- field sensor
-- flexible sensor
-- fluidic sensor
-- force sensor
-- gas sensor
-- glucose sensor
-- gmr sensor
-- gmti sensor
-- gpr sensor
-- gravitational sensor
-- ground sensor
-- guidance sensor
-- hall sensor
-- hartmann sensor
-- heterogeneous sensor
-- hoc sensor
-- humidity sensor
-- hybrid sensor
-- hyperspectral sensor
-- image sensor
-- inertial sensor
-- instrumentation sensor
-- ir sensor
-- ladar sensor
-- laser sensor
-- lidar sensor
-- lightweight sensor
-- loop sensor
-- lsst sensor
-- magnetic sensor
-- magnetostrictive sensor
-- mass sensor
-- measurement sensor
-- mechanical sensor
-- mems sensor
-- mesh sensor
-- metrology sensor
-- microcantilever sensor
-- micron sensor
-- microwave sensor
-- modal sensor
-- monitoring sensor
-- motion sensor
-- multifunctional sensor
-- multimodal sensor
-- multiple sensor
-- multispectral sensor
-- nanoribbon sensor
-- netted sensor
-- novel sensor
-- optical sensor
-- optoelectronic sensor
-- passive sensor
-- piezoresistive sensor
-- pixel sensor
-- pmmw sensor
-- position sensor
-- power sensor
-- predictive sensor
-- pressure sensor
-- prototype sensor
-- pulse sensor
-- radar sensor
-- radiometric sensor
-- remote sensor
-- resonator sensor
-- rf sensor
-- robotic sensor
-- satellite sensor
-- scale sensor
-- seismic sensor
-- silicon sensor
-- simulated sensor
-- single sensor
-- slope sensor
-- smart sensor
-- spectral sensor
-- spectroscopic sensor
-- static sensor
-- steerable sensor
-- tactile sensor
-- temperature sensor
-- tilt sensor
-- triangulation sensor
-- type sensor
-- typhimurium sensor
-- unattended sensor
-- undersea sensor
-- vacuum sensor
-- vector sensor
-- volumetric sensor
-- wavefront sensor
-- weather sensor
-- wireless sensor
-- zoom sensor

Figure 51. One level taxonomy of *Sensor* subtopics discovered in the NASA library with *Medium precision* and *Medium search*.

Figure 52. One level taxonomy of *Bacteria* subtopic discovered in the PudMed library with *High precision* and *Complex search*.

As an example, we present the results obtained in two well distinguished domains over their more adequate repositories: a technological one (*Sensor*) for the NASA repository, included in Figure 51, and a medical one (*Bacteria*) for the PubMed library, shown in Figure 52. Following the same concept per concept expert-based evaluation guidelines presented in §6.3, we obtain measures about *precision* and *local recall* shown in Table 48 and Table 49. The evaluation is performed for different search sizes, including other statistics such as the number of extracted topics or the runtime.

Table 48. Evaluation results and statistics for several search sizes for the *Bacteria* domain in the PudMed digital library with *High* search precision and one level search.

Bacteria **Search size**	**Precision**	**Local Recall**	**Local F-measure**	**#Correct topics**	**#Analyzed resources**	**Run time**
Simple	83 %	100 %	90.7%	10	20	12 sec.
Medium	87.5 %	87.5 %	87.5%	14	60	45 sec.
Complex	91.4 %	82.3 %	86.6%	107	1260	6 min.

Table 49. Evaluation results and statistics for several search sizes for the *Sensor* domain in the NASA Astrophysics digital library with *Medium* search precision and one level search.

Sensor Search size	Precision	Local Recall	Local F-measure	#Correct topics	#Analyzed resources	Run time
Simple	90 %	96.6%	93.18%	29	40	1 min
Medium	87.7%	93%	90.27%	93	240	4.5 min
Complex	77.4%	88.8%	82.7%	429	3700	33 min

Observing the results, we can see that, following the same tendency observed in the previous taxonomic evaluations (see §6.3 and §7.2.1.1), the correctness of the candidate selection procedure is high as the number of mistakes (incorrectly selected and rejected concepts from the candidate list), is maintained around a 10-20%. In this case, it is curious to see that for the *Bacteria* domain, the precision grows up in relation to the search size. However, observing the number of topics that we are able to extract for *simple* and *medium* search sizes, one can see that the number is too low (in comparison to the *Sensor* domain) to obtain trustworthy measures.

Concerning the number of correct extracted topics, as expected, it grows in relation to the number of explored resources that, at the same time, requires more runtime. Here we can see how the system adapts its behaviour to the domain generality, analysing more or less resources according to the search parameters and the feedback provided by learning rates.

As a final test, we compare these results to the ones obtained by our general taxonomy learning methodology from the whole Web (introduced in §6.3.2) using the same domain of knowledge (*Sensor,* which is characterized by the proliferation of noun phrase-based hyponyms). We have set the search precision and search size to Medium as those thresholds are the same used as default for the Web taxonomy learning process. The results of this evaluation are presented in Table 50.

Table 50. Comparison of the result quality (*precision* and *local recall*) and learning performance (*correct topics vs*. *runtime*) for the first level of the *Sensor* taxonomy using a NASA Astrophysics digital library search (with *Medium* search precision and *Medium* search size) against the full Web search using the default thresholds.

Sensor Search	Precision	Local Recall	Local F-measure	#Correct topics	Run-time	Topics per min.
NASA *Medium*	87.7 %	93 %	90.27%	93	4.5min.	20.6
Web *Default*	80.6%	88.2%	82.7%	106	17 min.	6.2

One can see that, as expected, using a high quality source such as a digital library against the full Web using similar executing conditions, brings better quality results (90,27% against 82,7% *local F-measures*). In addition, the especially designed analytical procedure results in a higher learning performance (20,6 *vs*. 6,2 *correct topics extracted per minute*). Even though, the general Web learning approach is not that far in terms of result quality and represents a more general approach (due to the heterogeneity of the Web) with potentially higher domain coverage (thanks to the use of

Hearst patterns and the amount of available resources). The conclusion is that both approaches seem valid enough for achieving their respective goals (efficient structuring of digital libraries *vs*. learning domain ontologies).

As a summary, the presented methodology for structuring digital libraries can bring benefits for the users of a particular electronic repository. On the one hand, it allows normal users to browse and access the library's electronic resources in a directory fashion in a very immediate way (performing short searches). On the other hand, it can also represent a valuable tool for web masters or domain experts that can automatically generate indexes for structuring large digital libraries (executing exhaustive searches).

7.2.3 Ontology-based web search

In the last years it has been argued that the performance of a web search engine can be improved by using ontologies [Fensel, 2001]. They provide a semantic ground that can help to sort out web pages with relevant information about a concept from those containing data with just syntactic similarities to the concept.

In order to demonstrate the suitability of our domain ontologies in guiding semantic web searches, we have designed an integrated approach for web information retrieval and filtering. The domain ontologies needed for this process are the hierarchical tree structure containing classes (concepts) and main features (attributes) that we are able to obtain.

The system uses two previously developed tools for knowledge acquisition and information retrieval. The first one is the domain ontology learning prototype system presented in §7.1. Its results can be used as input for the system described in [Moreno *et al*., 2004], which implements methods and techniques that allow the use of the information contained in the domain ontology in order to move from a purely syntactic keyword-based web search to a semantically grounded search. The final result is a set of filtered, ranked and classified web resources according to the concepts contained in the domain ontology. As the processing required to treat with a huge repository like the Web is a very time consuming task, the full system is presented in a distributed approach. Again, in order to provide a scalable solution, we have used the agent paradigm [Wooldridge, 2002] as the implementation approach.

Following the same philosophy that characterizes our research, the full system (described ontology learning and ontology-based web retrieval) operates in a fully unsupervised, automatic and domain independent way.

7.2.3.1 Ontology-driven web information retrieval

In this section, the ontology-based Web information retrieval system [Moreno *et al*., 2004; Bocio *et al*., 2005] is introduced. Its aim is to find the web pages which are relevant to a given domain of interest using a *domain ontology* as input (manually or automatically composed). It is required that the ontology contains concepts of the search domain and features of each one. It should represent concepts as classes in a hierarchical class-subclass structure, and the features as slots of the classes. A class

and all its ancestors define a *class path*. Each class contains its own slots and it inherits all those which are defined in the ancestors.

For instance, Figure 53 depicts a manually composed domain ontology about a subset of machine learning technologies where the classes are labelled as C and the slots as S. The names of the classes are used to find the web pages which are related to the search domain, and the names of the slots in a class are used to evaluate to what extent the retrieved pages have interesting information. The main idea is that the retrieved web pages are textual instances of the concepts, but conditioned to the meaning of the concept in the whole ontology. That means that the same concept in a different ontology would produce different results because it is in a different context.

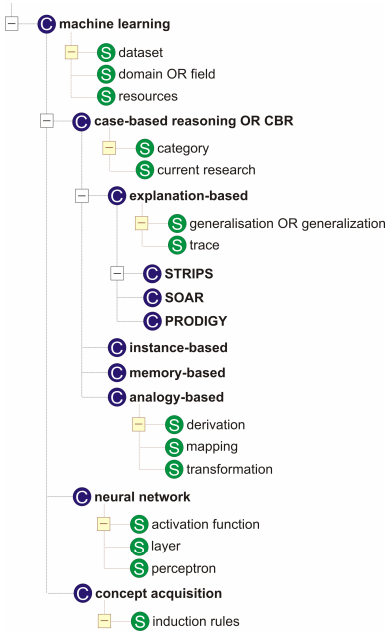

Figure 53. *Machine Learning* domain ontology.

The ontology-based web retriever is designed as an autonomous multi-agent system that can be deployed as a complement of the ontology learning multi-agent system described in §7.1. In that case, the first one receives the output of the second one in the form of a domain ontology that fulfils the requirements described above. According to the available knowledge, different types of agents are created and managed dynamically (created, configured and finalized) in function of the execution requirements at each moment.

In more detail, the search process is composed by several stages: *splitting* the domain ontology, *retrieving* the web pages, *rating* the retrieved pages, and *joining* the results. Those tasks are performed co-ordinately, as shown in Figure 54, by three types of agents: a *Coordinator Agent* (CA), a *Weight Agent* (WA) and some *Internet Agents* (IA).

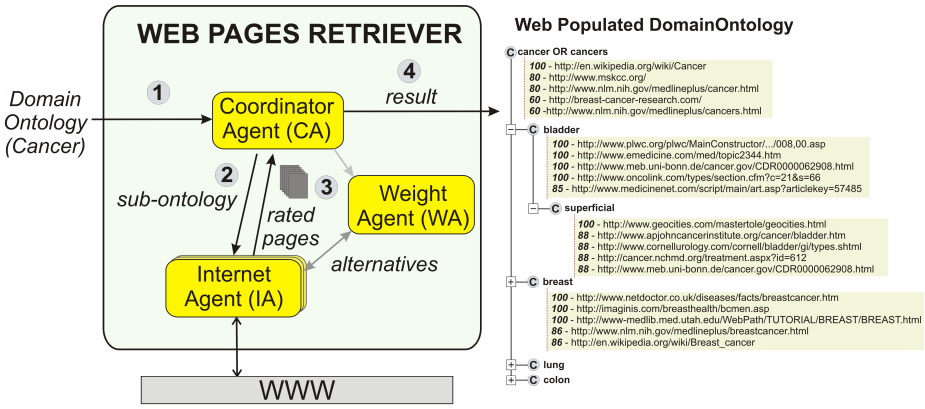

Figure 54. Agent-based ontology driven web retriever platform. The example results correspond to the *Cancer* domain.

The process starts when the automatically acquired domain ontology is received by CA (step 1). Then, it performs the *splitting* stage, in which the ontology is divided in smaller parts. Concretely, each class of the domain ontology defines a smaller ontology which contains not only the class itself but also its class path; this sort of ontology is called *query ontology*. The CA distributes those sub-ontologies among the available IAs (step 2).

Each IA uses the names of the classes in the query ontology as keywords to define a *query* into a standard keyword-based search engine. For each of these queries, a set of web pages is *retrieved*.

If the number of web pages does not reach an expected value (if the particular query is excessively restrictive), the system raises an additional process to increase the number of pages. In this case, IAs can request the help of the WA. This agent is able to find less constrained sets of keywords that can be used by IAs to find more pages. This process is based in a weighted expansion tree that is built up from the initial query, as Figure 55 depicts for the class STRIPS. The building process is as it follows: each node of the tree is expanded with sub-nodes representing queries where one of the keywords in the parent query has been removed, except the keyword that represents the name of the current class.

For instance, the right bottom side of Figure 55 shows the list of the keywords that are in the query related to the class STRIPS. Observe that only the initial letters of the keywords are displayed in the figure. When one of the antecedents of STRIPS (i.e. "*machine learning*", "*case-based reasoning*", or "*explanation-based*") is removed from the initial query, the nodes A, B, C are respectively expanded. The figure also shows how the keyword "*STRIPS*", represented by the letter S, is in all three sub-nodes. Finally, the numbers in the nodes indicate the amount of web pages that the search system can find using all the keywords in the node.

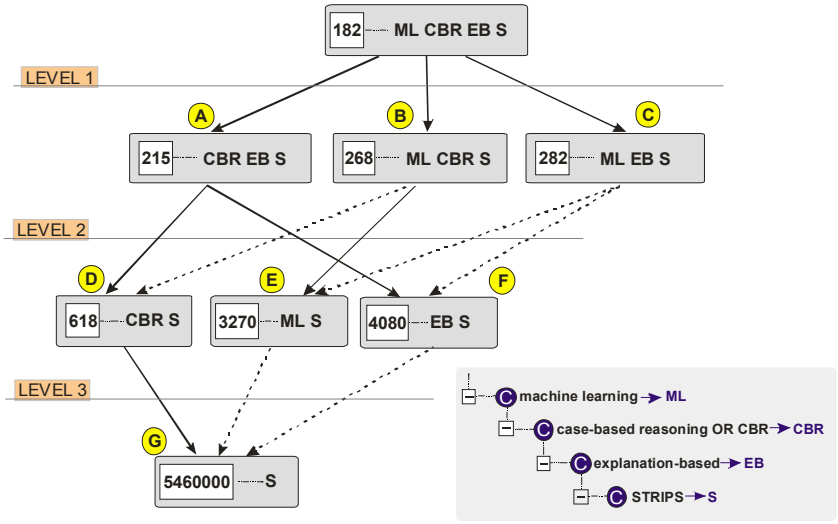

Figure 55. Best First search implemented by the Weight Agent to retrieve additional web sites.

Once the required amount of web resources have been obtained, the IA uses the semantic information of its subontology (class attributes) to *rank* these pages. Concretely, all the obtained web sites are rated according to their relevance within the query ontology (20).

$$R_c(p,A) = \frac{number\,of\,attributes\,found\,(p,A)*100}{total\,number\,of\,attributes\,(A)} \qquad \textbf{(20)}$$

If p stands for the recovered web page for a class C whose rate is being calculated, and A is the set of attributes (inherited or not) of C, the *attributes found* are the ones in A that appear in the page p. $R_C(p,A)$ defines the relevance of the web page p with respect to the class C and, after normalising it in the range [0,1], it is used to rank the retrieved pages.

Once the process is finished, the IA sends the rated and ranked list of web pages to the CA (step 3). Then, the CA incorporates them into the domain ontology. When all the IAs have returned their partial results, all the pages obtained for all the classes in the domain ontology are *joined* in a single structure. It contains each single page as an instance of the class in the ontology. This is presented to the user as the final result (step 4). Concretely, for each automatically acquired concept, a set of 2-tuple formed by an URL and a rate is presented. This last value indicates the degree of relevance of the particular URL and its associated concept according to the ranking measure employed during the retrieval and ranking process. Note that due to a specificity policy implemented, no redundant results between classes and subclasses are presented.

7.2.3.2 Evaluation

As the present proposal is an integration of two previously developed tools, the quality of the final results depends on the performance of each methodology. Regarding the evaluation of the taxonomies obtained by the first module, a discussion is offered in §6.3. With respect to the second module, in [Moreno *et al*., 2004] several evaluations are presented in different technological domains, starting from ontologies composed manually by experts.

The full platform has been tested in technical domains such as medicine, biotechnology and computer science. The evaluation has been performed by comparing the results against the web search engine used during the analysis (Google). More concretely, for the list of web sites retrieved for each automatically discovered concept, two users were requested to rate each web site according to their degree of interest for the particular domain with a value between 0 and 100. The same process was repeated for the first web sites returned by Google when manually querying the same acquired concept. These ratings indicate which approach returns, in average, the most interesting set of web resources for the particular domain.

As an example, in Figure 57 and Figure 56, expert's rating for our results against Google for a pair of concepts of the cancer domain is presented.

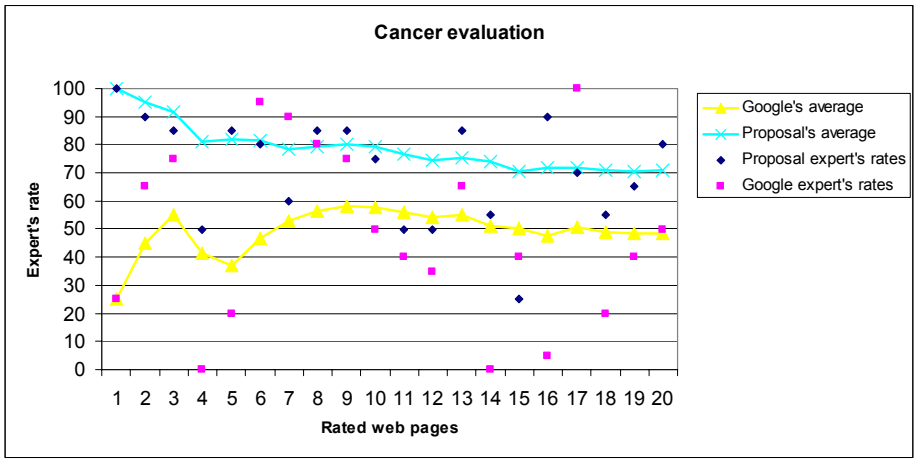

Figure 56. User's ratings for the first 20 web pages returned by our approach against the ones retrieved by Google for the *Cancer* concepts.

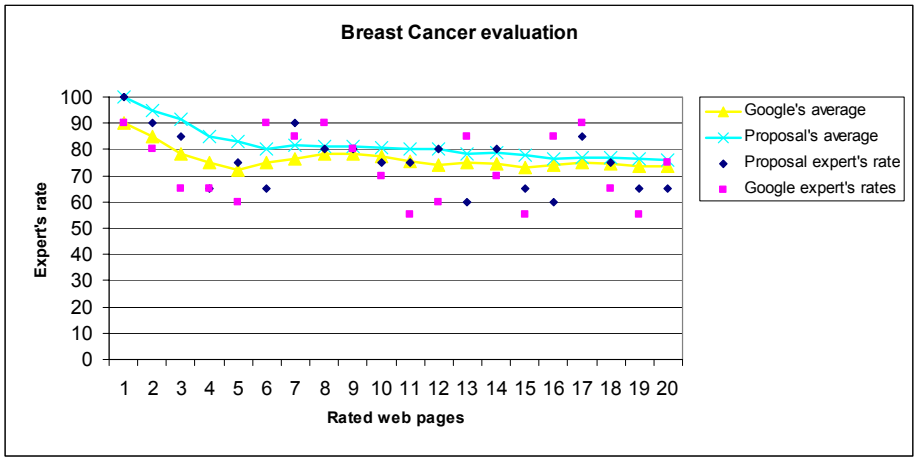

Figure 57. User's ratings for the first 20 web pages returned by our approach against the ones retrieved by Google for the *Breast Cancer* concept.

One can see that, for the most general concept (*cancer*), the quality of our results overpasses significantly, in average, the ones presented by Google. This behaviour has been observed for several tested domains and it is caused by the higher contextualization that the presented approach can apply to the web sites analysis thanks to the automatically acquired knowledge for the domain. Observing the average rating for a more concrete concept (*breast cancer*), we can see that the quality of the returned web sites by each system is very similar. In this case, the search is, in both cases, contextualized enough to retrieve high quality resources.

As a conclusion, as other authors has previously argued [Fensel, 2001], the use of knowledge (domain ontologies) can improve the classical web search, especially for general queries. In addition, the nature of our ontologies makes them adequate for those purposes as they have been directly extracted from the Web content.

7.3 Summary

In this chapter we have offered a detailed overview on how our different knowledge acquisition methodologies have been implemented in a distributed fashion. The employed programming paradigm based on agents is suitable to define the highly flexible and scalable system that our approach requires. Studying the computation complexity, one can see how a parallel approach is very suitable –and necessary- to obtain a good performance in wide domains. We have shown how our system is able to scale well when enough computational resources (in function on the number of tasks to execute) are provided.

Moreover, several applications of the proposed methodologies and their potential results have been presented. In addition to the concrete applications introduced up to this moment, other interesting aspects can be mentioned.

On the one hand, the domain ontology provides a structured representation of the knowledge associated to a certain domain. In this sense it can be used in several

knowledge-demanding tasks that require interoperability such us electronic commerce, distributed information systems such us multi-agent systems, Web Services and the mentioned Semantic Web [Berners-lee *et al*., 2001]. Moreover, intelligent knowledge guided methodologies for searching information from unstructured sources [Magnin *et al*., 2002; Alani *et al*., 2003; Sheth, 2003] can also use the results as the knowledge base for performing semantic searches.

On the other hand, topic hierarchies of the web resources considered according to the extracted knowledge represent an improvement over the classical searching for web resources [Lawrie and Croft, 2003]. This allows the user to access the desired information in a much direct way even if he is not an expert on the concrete domain. In this sense, it can be used as a tool for e-learning tasks where a student without specific knowledge in a certain domain can explore it in an interactive way, selecting new concepts, discovering important terms and how they are related and, finally, accessing concrete websites that contain specific information.

Aside from improving the access to web resources, the semantic structure extracted from the Web can help to improve the classical searching process by allowing query refinements according to the discovered concept hierarchy [Pasca, 2005].

Chapter 8

Conclusions and future work

Up to this moment, we have described in detail all the developed learning methodologies, the evaluation tests performed, the implementation and the possible applications of the methods and results. In this final chapter, we provide a final summary of the work and present the conclusions from the general knowledge acquisition point of view and from the concrete perspective of each ontology learning method. In the last section, we suggest several lines of future research about different open issues presented in previous chapters and give some ideas on how they can be tackled.

8.1 Summary

The main aim of the present work has been to develop methods for acquiring knowledge from the Web in order to compose a domain ontology. The most important and novel point is the complete integration with this environment, offering an especially adapted, automatic, unsupervised and domain independent approach that covers the main aspects of the ontology learning process (concepts, taxonomies, instances and labelled non-taxonomic relationships).

Many learning methodologies from different information repositories have been developed in the past, but it is not until now that authors are starting to focus their efforts on the Web. This environment adds new troubles to the information processing, derived from the *untrustworhiness*, *size*, *noise* and *lack of structure* of web resources. However, other characteristics as the *redundancy* and the existence of *web search engines* may help to tackle this environment. Regarding the first point, redundancy can allow us to infer information relevance, manage untrustworthiness and develop lightweight analytical approaches that can be adequate and scalable for the size of the Web. In relation to the second point, web searchers classically conceived as a final user interface for accessing web resources hide lots of potential regarding the inference of information distribution. Valuable web scale statistics can be extracted efficiently if adequate queries are performed. This can save us from analysing large amounts of resources and help us to obtain scalable learning methodologies. Moreover, their lack of any semantic content makes them suitable for any domain of knowledge. This is especially interesting in dynamic technological domains.

In addition, as we want to obtain results for these specific and concrete domains, in many situations, we will not be able to start from any predefined knowledge that many methodologies employ. This is why we have developed a completely unsupervised and automatic methodology that makes the minimum assumptions about previous knowledge or information structure. In order to achieve good results and learning performance following those premises, we have opted for an incremental learning methodology: several learning procedures are performed iteratively and potentially concurrently, using the knowledge acquired up to a moment as a bootstrap. Introduced feedback mechanisms allow a certain degree of self-control, including a dynamic adaptation of the size of the analysed corpus in function of the domain's productiveness and a management of the finalisation of the learning process.

Finally, manual and automatic evaluation procedures for each learning step have been designed. They provide encouraging results on the suitability of our approach for learning ontological entities in several well distinguished domains.

Taking all of these characteristics into consideration, we believe that our proposal can represent a new and interesting addition over the current state of the art of the technology in the ontology learning area.

8.2 Conclusions

Considering the performed research, the developed methodologies and the obtained and evaluated results, we have extracted the following general conclusions:

- As other authors have enounced in the past [Brill, 2003; Cilibrasi and Vitanyi, 2004; Etzioni *et al*., 2005], we expect to have contributed in considering the Web as a valid repository for performing knowledge acquisition tasks. In fact, we have developed learning methods covering the main steps of the ontology construction process especially adapted to this environment, obtaining reliable results.
- Available web IR tools (web search engines) can be extensively exploited to aid the ontology learning process. Through the development, we have presented ideas and methods for constructing suitable queries in function of the knowledge already acquired and the specific learning stage. As a result, we can dynamically obtain a corpus of resources to analyse at each moment and very robust web scale statistics about Web information distribution.
- Several knowledge acquisition techniques can be adapted to the Web. Considering the characteristics introduced in chapter 3 and our goals, the employed techniques should be simple and lightweight. Concretely, the use of linguistic patterns fits very well with the unsupervised nature of our learning approach and can be adapted to the limited query expressiveness offered by web search engines (our massive and automatic IR tools). Statistical analyses used to infer semantics (such as taxonomic relationships or concepts' relatedness) are very suitable as we have an enormous and heterogeneous repository of information and a way to obtain robust measures in a very immediate way. Finally, lightweight natural language analytical procedures are needed in order to *i)* maintain the domain independence of our learning approach (even limited to English written texts) and *ii)* scale well when dealing with huge amounts of noisy information resources.

- When developing automatic and unsupervised approaches, self-control mechanisms are required. We have included feedback about how the learning process evolves and bootstrapping techniques applied over fine grained learning steps. Both can improve the learning performance.

Other conclusions related with our developed methodologies are:

- In relation to the taxonomy learning, widely used Hearst's and noun phrase-based patterns can be combined to improve the final results. Concretely, on the one hand, Hearst's based extractions *precision* can be improved with noun phrase-based extractions by minimizing the semantic ambiguity. On the other hand, noun phrase-based extractions *recall* can be improved by incorporating the more general Hearst's extractions.
- The classical approach for taxonomy learning using linguistic patterns and statistical analyses can be also applied to the much less studied non-taxonomic learning. In this case, verb phrases can be considered as domain related patterns used to compose IR queries and compute statistical measures. The semantics of the relationships between concepts are expressed by the particular verb phrase.
- Almost any stage of the knowledge acquisition process (taxonomic and non-taxonomic learning and semantic disambiguation) requiring an estimation of the information distribution, can be addressed in an unsupervised way with a carefully designed and tuned statistical score, computed directly by querying a web search engine, as those presented in chapter 5.
- Regarding the evaluation, the use of WordNet as the base from which to develop automatic procedures can be a valid approach. However, it has been observed during the evaluation that its coverage for certain domains (in relation to glosses, synsets, semantic links, *etc*.) can be too limited to extract reliable conclusions.
- When developing highly distributed systems with requirements of flexibility and dynamicity, the use of multi-agent systems can be a suitable high level implementation paradigm. They certainly offer some advantages over other approaches such as the dynamic management of working threads or the highly elaborated execution framework, including mobility and communication capabilities that ease the development of complex distributed systems.

8.3 Future work

In this section, we describe several future lines of research and present some preliminary ideas on how they can be tackled. Regarding the learning process, some issues can be addressed in order to improve the final results:

- The recall of the taxonomy learning process may be improved if additional linguistic patterns for hyponymy detection are applied. Concretely, some authors [Agichtein and Gravano, 2000; Iwanska *et al*., 2000; Pasca, 2004; Snow *et al*., 2004] have been working in refining Heart's patterns. However, many of the new regular expressions define very subtle variations or specific forms rarely used. In consequence, it should be studied if including additional concrete patterns to the taxonomic learning results in a final improvement or it only overheads the learn-

ing process. In our opinion, the basic but general pattern set used until this moment (introduced in §4.1) is enough for obtaining good coverage (in function of the established learning thresholds) thanks to the size and redundancy of information in the Web (as observed during the learning rates analysis in §5.6.2).

- If one is particularly interested in the retrieval of instances used for ontology population, the proposed method for detecting named entities based on linguistic patterns and capitalization heuristics can be more widely developed. Concretely, it can be executed a posteriori of the ontology learning process over the different taxonomically and non-taxonomically related classes and in conjunction with a search engine that properly distinguishes capitalized terms in order to retrieve additional named entities. Moreover, at this moment, the identification of the full entity name (in those cases in which it is composed by several terms) is left to the named entity detection package used during the evaluation (see §6.4). Contributions to solve this last issue can be developed with novel algorithms [Downey *et al*., 2007]. In any case, the particular instance semantic should be taken into consideration by, for example, analysing named-entity's context for the particular ontology, in order to make a contribution to the ontology population field.
- The non-taxonomic learning can be improved if verb phrases (used as domain dependent pattern) are further processed. Concretely, due to the diversity of ways of expressing a particular verb phrase (in function of the verbal tense or subject number), some valid candidate extractions may be omitted due to the too restrictive matching policy implemented by keyword-based search engines. In this case, a procedure to properly conjugate verb phrases in common forms, applying each of them to the retrieval of candidates by constructing different queries, may aid to increase the quantity of extracted knowledge.
- The information extracted from VerbNet and associated to the verb phrases during the non-taxonomic learning can be used to infer the semantics of the relations [Gómez and Segami, 2007]. We may detect verbs that have a similar "meaning" or express the same "kind" of relationship. In addition, thematic roles may be exploited to interpret in which way the subject or the object of the relationships is affected. All that information can then be modeled in the ontology using more advanced ontological formalisms such as property characteristics or class restrictions. However this is certainly an intricate task and may require the use of more complex analyses to achieve the proper natural language understanding.
- As has been commented in previous sections, the semantic disambiguation methods can be integrated in the full learning process. On the one hand, synonym sets can be used to expand the search to other web resources that were not potentially retrieved by the keyword-based search engine and a specific domain keyword. This could improve the *recall* of the final results when dealing with narrow domains where a limited amount of resources is retrieved (see in Figure 58 an example of taxonomies retrieved for synonyms discovered for the *cancer* domain). On the other hand, polysemy disambiguation may aid to improve the *precision* of the final taxonomy in polysemic domains by presenting a more structured hierarchy with clustered classes according to superclass senses. However the final impact on the results of those extra processing stages should be considered carefully. On the one hand, for the first case, even though the final recall can be higher, the

noise introduced by not truly equivalent terms can affect negatively to the precision. On the other hand, for the second case, the precision improvement may be questionable when dealing with non well-differentiated senses.

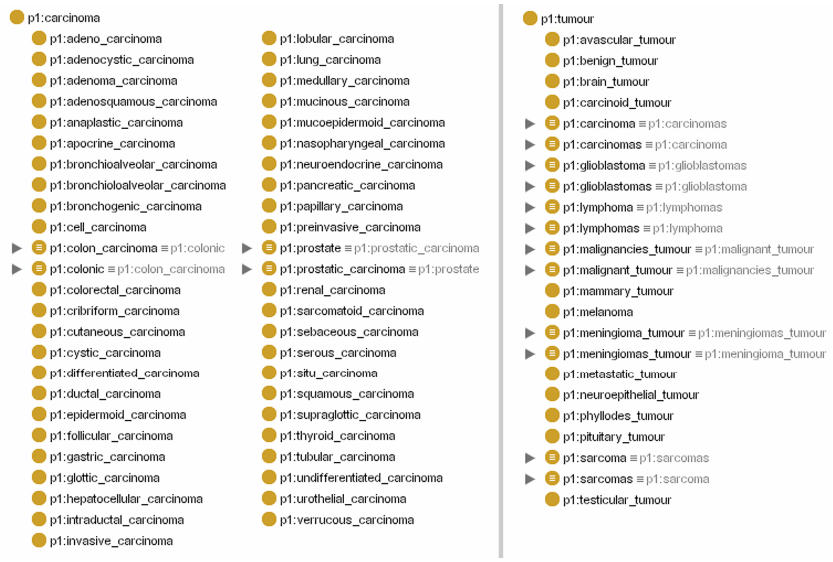

Figure 58. Results obtained for the first level of the taxonomy of the *Cancer* domain using two of its automatically discovered synonyms (*carcinoma* on the left, *tumour* on the right) with the same execution conditions**.**

- Even though we are able to extract some domain features from the redundancies observed in the taxonomy of classes, this is only a first approach for detecting class attributes. Class attributes are rarely considered in ontology learning techniques due to their potential complexity [Gómez-Pérez *et al*., 2004] even though they can certainly enrich the semantics of the corresponding class. Attribute detection can be addressed by developing especially adapted Web methodologies in order to exploit semi-structured data associated to domain classes such as itemizes, lists, tables or indexes [Popescu and Etzioni, 2005].
- More efforts can be put in the future regarding the evaluation of results. More expert opinions can be requested to further evaluate the results, including other domains or verb labels. Other tests over standard reduced repositories can be performed in order to compare the learning performance with other approaches.
- The implemented multi-agent system can be improved. On the one hand, more fine grained tasks may be defined (*e.g.* one for each phase of the learning process) and modeled over different agents, improving the parallelism. On the other hand, we can exploit agent communication capabilities. In this last case, in addition to the task coordination, they may exchange partial results or retrieved web resources in order to avoid redundant analyses or repeat web requests already performed. Even though this may represent an improvement in relation to the runtime spent in accessing, retrieving and analyzing web resources, the overhead introduced by the inter-agent communication should be considered.

Regarding the applications of the proposed methodologies and results, some interesting cases can be researched:

- One of the most important applications of domain ontologies consists on bringing machine readable semantic content that web documents lack by employing annotation. This will help to achieve the paradigm of semantic web search proposed by the Semantic Web [Berners-Lee *et al*, 2001]. However, even if representative semantic structures can be obtained in an automatic and efficient way, the labor of annotating web documents is usually performed manually [Kahan *et al.*, 2001]. In our case, as domain ontologies are obtained directly from the analysis of web documents, a certain degree of automatic annotation could be performed directly during at the construction stage. For example, if we have discovered instances (as named entities) for a specific class, we can annotate the web resources from which those instances have been extracted according to the specific class to which they belong. Methodologies for annotating automatically web content (the web resources analysed) can be studied. In addition, once the semantic structure is obtained and the annotation methodology developed, it should not be difficult to extend the annotation to other web resources.
- From a more general point of view, results obtained from web annotation can be used to bring a further understanding of the domain by means of reasoning. On the one hand, a domain ontology can be populated using discovered annotated entities. On the other hand, ontology semantics can be used to perform inference over those individuals, resulting in additional knowledge not directly discovered.
- Once the reliability of our learning methodologies has been evaluated in bringing structure to electronic repositories such as digital libraries, and in comparison with other available approaches (such as taxonomic search engines), it could be also interesting to apply them to wider environments such as the automatic composition of web directory services.
- It could be interesting to test from the final user point of view the advantages of structured representation of web resources that we are able to obtain in relation to the classical way of presenting results by a web search engine. For example, we can measure the efficiency of the user's searching for information of a specific domain using those two different approaches.
- In order to test the real performance of our learning approaches and applications, it would be very interesting to have direct access to a web search engine IR database without limitations. This will minimize the delays and overheads introduced by the web queries requested during the learning (that represent a time interval several orders of magnitude higher than the time required to perform the analysis of the web content). Even though we try to minimize active waits using a distributed parallel approach, they still represent an important waste of time.
- From another point of view, detecting named-entities can be a useful tool for performing market studies, retrieving important companies and organisations and their associated web resources, which are related to a certain aspect of the knowledge domain. Moreover, those instances are selected without classical restrictions (*e.g.* "organisations", "persons"…) allowing to detect all kinds of entities and events.

- Several executions in different moments for the same domain using the same parameters, can allow us to study the evolution of the information in that domain, and detect for example that a new concept has potentially appeared. This aspect can support a certain degree of high level question answering of the kind of "which items have appeared in the domain?", "which ones are now more relevant than before?", "which seem to be obsolete?", *etc.*

Bibliography

[Abecker *et al*., 1999] Abecker, A., Bernardi, A., and Sintek, M.: Proactive knowledge delivery for enterprise knowledge management. In Proceedings of the 11th Conference on Software Engineering and Knowledge Engineering. Kaiserslautern, Germany, June 17-19 1999. 103-117.

[Adelberg, 1998] Adelverg, B.: NoDoSE: A tool for semi-automatically extracting semistructured data from text documents. In Proceedings of SIGMOD'98 Conference. 1998. 283-294.

[Agichtein and Gravano, 2000] Agichtein, E. and Gravano, L.: Snowball: Extracting relations from large plaintext collections. In Proceedings of the 5th ACM International Conference on Digital Libraries (DL-00), San Antonio, Texas, 2000. 85-94.

[Agirre and Rigau, 1995] Agirre, E. and Rigau, G.: A Proposal for Word Sense Disambiguation using Conceptual Distance. In: Proceedings of the International Conference on Recent Advances in NLP. RANLP'95. 1995. 16-22.

[Agirre *et al*., 2000] Agirre, E., Ansa, O., Hovy, E., and Martinez, D.: Enriching very large ontologies using the WWW. In Proceedings of the Workshop on Ontology Construction of the European Conference of AI (ECAI-00). Berlin, Germany, 2000.

[Agrawal *et al*., 1993] Agrawal, R., Imielinksi, T. and Swami, A.: Mining association rules between sets of items in large databases. In Proceedings of the ACM SIGMOD Conference on Management of Data. 1993. 207-216.

[Ahmad *et al*., 2003] Ahmad, K., Tariq, M., Vrusias, B. and Handy, C.: Corpus-based thesaurus construction for image retrieval in specialist domains. In Proceedings of the 25th European Conference on Advances in Information Retrieval (ECIR). 2003. 502-510.

[Alani *et al*., 2003] Alani, H., Kim, S., Millard, D., Eal, M., Hall, W., Lewis, H, and Shadbolt, N.: Automatic Ontology-Based Knowledge Extraction from Web Documents. IEEE Intelligent Systems, IEEE Computer Society, 2003. 14-21.

[Alfonseca and Manandhar, 2002] Alfonseca, E. and Manandhar, S.: An unsupervised method for general named entity recognition and automated concept discovery. In Proceedings of the 1st International Conference on General WordNet, Mysore, India. 2002.

[Alfonseca and Rodríguez, 2002] Alfonseca, E., and Rodríguez, P.: Automatically Generating Hypermedia Documents depending on User Goals, Workshop on Document Compression and Synthesis in Adaptive Hypermedia Systems, AH-2002, Málaga, Spain. 2002.

[Arocena and Mendelzon, 1998] Arocena A. and Mendelzon A.O.: Restructuring documents, databases, and Webs. In Proceedings of ICDE'98 Conference. 1998. 24-33.

[Aussenac-Gilles and Seguela, 2000] Aussenac-Gilles, N. and Seguela, P.: Les relations sémantiques: du linguistique au formel. Cahiers de grammaire, N° spécial sur la linguistique de corpus, 25. Toulouse : Presse de l'UTM. 2000. 175-198.

[Aussenac-Gilles *et al*., 2000] Aussenac-Gilles, N., Biébow, B. and Szulman, S.: Corpus Analysis for Conceptual Modelling. Workshop on Ontologies and Text, Knowledge Engineering and Knowledge Management: Methods, Models and Tools, 12th International Conference EKAW'2000, Juan-les-pins, France, Springer-Verlag. 2000. 13-20.

[Bachimont *et al*., 2002] Bachimont, B., Isaac, A. and Troncy, R.: Semantic commitment for designing ontologies: a proposal. In A. Gomez-Perez and V.R. Benjamins (Eds.): EKAW 2002, LNAI 2473. 2002. 114–121.

[Bachimont, 2000] Bachimont, B.: Engagement sémantique et engagement ontologique: conception et réalisation d'ontologies en ingénierie des connaissances. In Ingénierie des Conniassances: Evolutions récentes et nouveaux défis, Eyrolles, 2000.

[Banko *et al*., 2007] Banko, M., Cafarella, M., Soderlan, S., Broadhead, M. and Etzioni, O.: Open Information Extraction from the Web. In proceedings of IJCAI 2007. 2007. 2670-2676.

[Baugartner *et al*., 2001] Baugartner, R., Flesca S. and Gottlob, G.: Visual Web information extraction with Lixto. In Proceedings of VLDB'01 Conference. 2001. 119-128.

[Berners-lee *et al*., 2001] Berners-lee, T., Hendler, J. and Lassila, O.: The semantic web. Scientific American. 2001.

[Berry *et al*., 1995] Berry, M.W., Dumais, S.T. and Letsche, T.A.: Computational Methods for Intelligent Information Access. Proceedings of Supercomputing '95, San Diego, California, 1995.

[Bhat *et al*., 2004] Bhat, V., Oates, T., Shanbhag, V. and Nicholas, C.: Finding aliases on the web using latent semantic analysis. Data Knowledge & Engineering 49. 2004. 129-143.

[Bisson *et al*., 2000] Bisson, G., Nedellec, C. and Cañamero, D.: Designing Clustering Methods for Ontology Building. The Mo'K Workbench. In S. Staab, A. Maedche, C. Nedellec, P. WiemerHasting (eds.), Proceedings of the Workshop on Ontology Learning, 14th European Conference on Artificial Intelligence, ECAI'00, Berlin, Germany. August 20-25, 2000. 13-19.

[Bocio *et al*., 2005] Bocio, J., Isern, D., Moreno, A., Riaño, D.: Semantically Grounded Information Search on the WWW. In: Artificial Intelligence Research and Development, Vol. 100. IOS Press. 2005. 349–356.

[Borst, 1997] Borst, W.N.: Construction of Engineering Ontologies. Centre for Telemática and Information Technology, University of Tweety. Enschede, The Netherlands. 1997.

[Borthwick, 1999] Borthwick, A.: A Maximum Entropy Approach to Named Entity Recognition. Phd. thesis. New York University. 1999.

[Brewster *et al*., 2001] Brewster, C., Ciravegna, F. and Wilks, Y.: Knowledge Acquisition for Knowledge Management: Position Paper. In Proceeding of the IJCAI-2001 Workshop on Ontology Learning held in conjunction with the 17th International Conference on Artificial Intelligence (IJCAI-01), Seattle, August, 2001.

[Brewster, 2002] Brewster, C.: Techniques for Automated Taxonomy Building: Towards Ontologies for Knowledge Management. In Proceedings of the 5th Annual CLUK Research Colloquium. Leeds, 2002.

[Brewster et al., 2004] Brewster, C., Alani, H, Dasmahapatra, S and Wilks, Y.: Data-driven Ontology Evaluation. In Proceedings of the 4th International Conference on Language Resources and Evaluation. 2004.

[Brill *et al*., 2001] Brill, E., Lin, J., Banko, M. and Dumais, S.: Data-intensive Question Answering. In Proceedings of the Tenth Text Retrieval Conference TREC-2001. 2001. 393-400.

[Brill, 2003] Brill, E.: Processing Natural Language without Natural Language Processing. In Proceedings of CICLing 2003, LNCS 2588. 2003. 360–369.

[Budanitsky and Hirst, 2001] Budanitsky, A. and Hirst, G: Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. NAACL01. 2001.

[Buitelaar, et al., 2003] Buitelaar, P., Olejnik, D. And Sintek, M.: A protégé plug-in for ontology extraction from text based on linguistic analysis. In Proceedings of the International Semantic Web Conference (ISWC). 2003.

[Buitelaar *et al*., 2005] Buitelaar, P., Cimiano, P., Magnini, B. (eds): Ontology Learning from Text: Methods, Evaluation and Applications Frontiers in Artificial Intelligence and Applications Series, Vol. 123, IOS Press, July 2005.

[Buitelaar *et al*., 2005b] Buitelaar, P., Cimiano, P., Grobelnik, M. and Sintek, M.: Ontology Learning from Text. Tutorial at ECML/PKDD, Oct. 2005, Porto, Portugal.

[Buitelaar *et al*., 2006] Buitelaar, P., Cimiano, P., Racioppa, S., Siegel, M.: Ontology-based information extraction with SOBA. In Proceedings of LREC 2006. 2006. 2321-2324.

[Bunescu, 2003] Bunescu, R.: Associative Anaphora Resolution: A Web-Based Approach. In Proceedings of the EACL-2003 Workshop on the Computational Treatment of Anaphora, Budapest, Hungary, April, 2003. 47-52.

[Burton-Jones *et al*., 2003] Burton-Jones, A., Storey, V.C., Sugumaran, V. and Purao, S.: A Heuristic-based methodology for Semantic Augmentation of User Queries on the Web. In Proceedings of Conceptual modelling- ER 2003. LNCS 2813. Chicago, USA. 2003. 476-489.

[Byrd and Ravin, 1999] Byrd, R. and Ravin, Y.: Identifying and extracting relations from text. In NLDB'99 - 4th International Conference on Applications of Natural Language to Information Systems. 1999.

[CACM, 2002] CACM, Special issue on ontology. Communications of ACM, 45(2). 2002.

[Califf and Mooney, 1999] Califf M.E. and Mooney, R.J.: Relational learning of pattern-match rules for information extraction. In Proceedings of AAAI'99. 1999. 328-334.

[Calvo and Gelbukh, 2003] Calvo, H. and Gelbukh., A.: Improving Prepositional Phrase Attachment Disambiguation Using the Web as Corpus. LNCS 2905. 2003. 604–610.

[Cameron, 2002] Cameron, I.: Web-based cape systems -now and the future-. In CAPE Forum, Tarragona, Spain, 2002.

[Caraballo, 1999] Caraballo, S.A.: Automatic construction of a hypernym-labeled noun hierarchy from text. In Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics. 1999. 120-126.

[Chaelandar and Grau, 2000] Chaelandar, G. and Grau, B.: SVETLAN' - A System to Classify Words in Context. In S. Staab, A. Maedche, C. Nedellec, P. Wiemer-Hastings (eds.) Proceedings of the Workshop on Ontology Learning, 14th European Conference on Artificial Intelligence ECAI'00, Berlin, Germany, August 20-25. 2000. 19-24.

[Charniak, 1999] Charniak, E. and Berland, M.: Finding parts in very large corpora. In Proceedings of the 37th Annual Meeting of the ACL. 1999. 57-64.

[Church *et al*, 1991] Church, K.W., Gale, W., Hanks, P. and Hindle, D.: Using Statistics in Lexical Analysis. In: Uri Zernik (ed.), Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon. New Jersey: Lawrence Erlbaum. 1991. 115-164.

[Cilibrasi and Vitanyi, 2004] Cilibrasi, R. and Vitanyi, P.M.B.: Automatic meaning discovery using Google. Available at: http://xxx.lanl.gov/abs/cs.CL/0412098. 2004.

[Cilibrasi and Vitanyi, 2006] Cilibrasi, R. and Vitanyi, P.M.B.: The Google Similarity Distance. IEEE Transaction on Knowledge and Data Engineering. 19(3). 2006. 370-383.

[Cimiano *et al*., 2004] Cimiano, P., Pick, A., Schmidt, L. and Staab, S.: Learning Taxonomic Relations from Heterogeneous Sources of Evidence. In Procedings of the ECAI 2004 Ontology Learning Workshop. 2004.

[Cimiano and Staab, 2004] Cimiano, P. and Staab, S.: Learning by Googling. SIGKDD Explorations, 6(2). 2004. 24-33.

[Cimiano and Wenderoth, 2005] Cimiano, P. and Wenderoth, J.: Automatically Learning Qualia Structure from the Web. In Proceedings of the ACL Workshop on Deep Lexical Acquisition. 2005. 28-37.

[Cimiano, 2006] Cimiano, P.: Text Analysis and Ontologies. Summer School on Multimedia Semantics. 2006.

[Ciravegna 2001] Ciravegna, F.: Adaptive Information Extraction from Text by Rule Induction and Generalisation. In Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI), 2001.

[Ciravegna *et al*., 2003] Ciravegna, F., Dingli, A., Guthrie, D. and Wilks, Y: Integrating Information to Bootstrap Information Extraction from Web Sites. In Proceedings of the IJCAI Workshop on Information Integration on the Web. 2003. 9-14.

[Collins and Singer, 1999] Collins, M. and Singer, Y.: Unsupervised models for named entity classification. In Proceedings of the 1999 Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99), College Park, Maryland, 1999. 189–196.

[Corcho *et al*., 2006] Corcho, O., Fernández-López, M. and Gómez-Pérez, A.: Ontologies for Sofware Engineering and Software Technology. Calero, C., Ruiz, F. and Piattini, M. (eds). 2006.

[Crescenzi and Mecca, 1998] Crescenzi, V. and Mecca, G.: Grammars have exception. Information Systems 23(8). 1998. 539-565.

[Crescenzi *et al*., 2001] Crescenzi, V., Mecca G. and Merialdo P.: RoadRunner: towards automatic data extraction from large Web sites. In Proceedings of VLDB'01 Conference, 2001. 109-118.

[Cucerzan and Yarowsky, 1999] Cucerzan, S. and Yarowsky D.: Language independent named entity recognition combining morphological and contextual evidence. In Proceedings of the Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99). College Park, Maryland, 1999. 90–99.

[Cui *et al*., 2004] Cui, H., Kan, M. and Chua, T.: Unsupervised learning of soft patterns for generating definitions from online news. In Proceedings of the 13th World Wide Web Conference. 2004. 90-99.

[Cutting, 1992] Cutting, D., Karger, D., Pedersen, J. and Tukey, J.W.: Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections. In Proceedings of the 15th Annual International ACM/SIGIR Conference, Copenhagen. 1992. 318-329.

[Daudé *et al*., 2003] Daudé J., Padró L. and Rigau G.: Validation and Tuning of WordNet Mapping Techniques. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP'03). Borovets, Bulgaria, 2003.

[de Lima, 1999] de Lima, E.F. and Pedersen, J.O.: Phrase Recognition and Expansion for Short, Precision biased Queries based on a Query Log. In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 1999. 145-152.

[Decker *et al*., 1999] Decker, S., Erdmann, M., Fensel, D. and Suder, R.: Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In Proceedings of Semantic Issues in Multimedia Systems (DS-8). Rotorua, New Zealand. 1999. 351-369.

[Deerwester *et al*., 1990] Deerwester, S., Dumais, S., Landauer, T., Furnas, G. and Harshman, R.: Indexing by latent semantic analysis. Journal of the American Society of Information Science 41(6). 1990. 391-407.

[Dellschaft and Staab, 2006] Dellschaft, K, and Staab, S.: On How to Perform a Gold Standard Based Evaluation of Ontology Learning. In proceedings of The Semantic Web - ISWC 2006 LNAI 4273. 2006. 228-241.

[Ding *et al*., 2004] Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R.S., Peng, Y., Reddivari, P., Doshi, V.C., Sachs, J.: Swoogle: A Search and Metadata Engine for the Semantic Web. In Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management, ACM Press. 2004. 652-659.

[Doorenbos *et al*., 1997] Doorenbos, R., Etzioni, O. and Weld, D.S.: A Scalable Comparison-Shopping Agent for the World-Wide Web. In Proceeding of the AGENTS '97 Conference. 1997. 39-48.

[Downey *et al*., 2007] Downey, D., Broadhead, M. and Etzioni, O.: Locating Complex Named Entities in Web text. In Proceedings of ICJAI 2007. 2007.

[Dujmovic and Bai, 2006] Dujmovic, J. and Bai, H.: Evaluation and Comparison of Search Engines Using the LSP Method. ComSIS 3(2). 2006. 711-722.

[Economist, 2005] Economist: Corpus collosal: How well does the world wide web represent human language? The Economist, January 20, 2005. Available at: http://www.economist.com/science/displayStory.cfm?story id=3576374. 2005.

[Eikvil, 1999] Eikvil, L.: Information extraction from World Wide Web – a survey, Technical report 945. Norweigan computing Center, 1999.

[Embley *et al*., 1999] Embley, D.W, Campbell, D.M., Jiang, Y.S., Liddle, S.W., Lonsdale, D.W., Ng, Y. and Smith R.D.: Conceptual-model-based data extraction from multiple-record Web Pages, Data Knowledge Engineering 31(3). 1999. 226-251.

[Engels, 2001] Engels, R.: CORPORUM-OntoExtract. Ontology Extraction Tool. Deliverable 6. Ontoknowledge. http://www.ontonowledge.org/del.shtml. 2001.

[Etzioni *et al*., 2004] Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A., Shaked, T., Soderland, S. and Weld, D.S.: WebScale Information Extraction in KnowItAll. In Proceedings of WWW2004, New York, USA. 2004.

[Etzioni *et al.*, 2005] Etzioni, O., Cafarella, M., Downey, D., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S. and Yates, A.: Unsupervised named-entity extraction form the Web: An experimental study. Artificial Intelligence 165. 2005. 91-134.

[Faatz and Steinmetz, 2002] Faatz, A. and Steinmetz, R.: Ontology enrichment with texts from the WWW. In Proceedings of Semantic Web Mining 2nd Workshop at ECML/PKDD-2002. Helsinki, Finland. 20th August 2002.

[Farreres *et al*., 2004] Farreres, J., Gibert, K. and Rodríguez, H.: Towards Binding Spanish Senses to WordNet Senses through Taxonomy Alignment. In Proceedings of GWC 2004. Masaryk University. 2004. 259-264.

[Faure and Nedellec, 1998] Faure, D. and Nedellec, C.: A corpus-based conceptual clustering method for verb frames and ontology acquisition. In Proceedings of LREC-98 Workshop on Adapting Lexical and Corpus Resources to Sublanguages and Applications, Granada, Spain. 1998. 1-8.

[Faure and Poibeau, 2000] Faure, D. and Poibeau, T.: First experiments of using semantic knowledge learned by ASIUM for information extraction task using INTEX. In: S. Staab, A. Maedche, C. Nedellec, P. Wiemer- Hastings (eds.), Proceedings of the Workshop on Ontology Learning, 14th European Conference on Artificial Intelligence ECAI'00, Berlin, Germany. 2000. 7-12.

[Fellbaum, 1998] Fellbaum, C.: WordNet: An Electronic Lexical Database. Cambridge, Massachusetts: MIT Press. More information: http://www.cogsci.princeton.edu/~wn/. 1998.

[Fensel *et al*., 2001] Fensel, D., van Hermelen, F., Horrocks, I., McGuiness, D.L. and Patel-Schneider, P.F.: OIL: An Ontology Infrastructure for the Semantic Web. IEEE Intelligent Systems (16). 2001. 38-44.

[Fensel, 2001] Fensel, D.: Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce. Springer Verlag. 2001.

[Fernández-López *et al*., 1997] Fernández-López, M., Gómez-Pérez, A. and Juristo, N.: METHONTOLOGY: From Ontological Art Towards Ontological Engineering. In Proceedings of the Spring Symposium on Ontological Engineering of AAAI. Stanford University. USA, 1997. 33-40.

[Finkelstein-Landau and Morin, 1999] Finkelstein-Landau, M. and Morin, E.: Extracting Semantic Relationships between Terms: Supervised vs. Unsupervised Methods. In Proceedings of the International Workshop on Ontological Engineering on the Global Information Infrastructure, Dagstuhl. 1999. 71-80.

[Fleischman and Hovy, 2002] Fleischman, M. and Hovy, E.: Fine grained classification of named entities. In Proceedings of the 19th Conference on Computational Linguistics (COLING), 2002. 1-7.

[Flesca *et al*., 2004] Flesca, S., Manco, G., Masciari, E., Rande, E. and Tagarelli, A.: Web wrapper induction: a brief survey. AI Communications 17. IOS Press. 2004. 57-61.

[Fortuna *et al*., 2005] Fortuna, B., Grobelnik, M., Mladenic, D.: Visualization of Text Document Corpus. Informatica, 29. 2005. 497-502.

[Fox, 1992] Fox, M.S.: The TOVE Project: A Common-sense Model of the Enterprise. In Proceedings of the Industrial and Engineering Applications of Artificial Intelligence and Expert Systems. LNAI 604. 1992. 25-34.

[Freitag and MacCallun, 1999] Freitag, D. and McCallum, A.: Information extraction with HMMs and shrinkage. In Proceedings of the AAAI-99 Workshop on Machine Learning for Information Extraction, 1999. 31-36.

[Freitag, 2000] Freitag, D.: Machine learning for information extraction in informal domains, Machine Learning 39(2-3). 2000. 233-272.

[Gal *et al*., 2004] Gal, A., Modica, G. and Jamil, H.: OntoBuilder: Fully Automatic Extraction and Consolidation of Ontologies from Web Sources. In Proceedings of the 20th International Conference on Data Engineering (ICDE'04). 2004. 853-860.

[Gibbins *et al*., 2003] Gibbins, N., Harris, S. and Shadbolt, N.: Agent-based semantic web services. In Proceedings of the Twelfth International World Wide Conference (WWW2003), ACM Press. Hungary, 2003.

[Girju and Moldovan, 2002] Girju, R. and Moldovan, D.: Text Mining for Causal Relations. In Proceedings of the FLAIRS Conference. 2002. 360-364.

[Gluschko *et al*., 1999] Gluschko, R., Tenenebaum, J., and Meltzer, B.: An XML Framework for Agent-based E-Commerce. Communications of the ACM 42(3). 1999. 106-114.

[Gómez and Semagi, 2007] Gómez, F. and Segami, C.: Semantic interpretation and knowledge extraction. Knowledge-Based systems, 20. 2007. 51-60.

[Gómez-Perez and Manzano-Macho, 2003] Gómez-Pérez, A., Manzano-Macho, D.: A survey of ontology learning methods and techniques. Deliverable 1.5. OntoWeb. 2003.

[Gómez-Pérez et al., 2004] Gómez-Pérez, A., Fernández-López, M. and Corcho, O.: Ontological Engineering, 2nd printing. Springer Verlag.. ISBN: 1-85233-551-3. 2004

[Grefenstette, 1992] Grefenstette, G.: Finding Semantic Similarity in Raw Text: The Deese Antonyms. In: R. Goldman, P. Norvig, E. Charniak and B. Gale (eds.), Working Notes of the AAAI Fall Symposium on Probabilistic Approaches to Natural Language. AAAI Press. 1992. 61-65.

[Grefenstette, 1997] Grefenstette, G.: SQLET: Short Query Linguistic Expansion Techniques: Palliating One-Word Queries by Providing Intermediate Structure to Text. In Proceedings of Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology. LNAI 1299. International Summer School, SCIE-97. Italy, 1997. 97-114.

[Greffenstette, 1999] Grefenstette, G.: The World Wide Web as a resource for example-based Machine Translation Tasks. In Proceedings of Aslib Conference on Translating and the Computer. London. 1999.

[Gruber and Olsen, 1994] Gruber, T.R. and Olsen, F.: An ontology for Engineering Mathematics. In Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning. Bonn, Germany. 1994. 258-269.

[Gruber, 1993] Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing. In Guarino, N., Poli, R. (eds) International Workshop on Formal Ontology in Conceptual Analysis and Knowledge Representation. Padova, Italy. 1993. 907-928.

[Guarino *et al*., 1999] Guarino, N., Masolo, C., Vetere, G.: OntoSeek: Content-Based Access to the Web. IEEE Intelligent Systems, 14(3). 1999. 70-80.

[Guarino, 1998] Guarino, N.: Formal Ontology in Information Systems. In Guarino N. (ed) 1st International Conference on Formal Ontology in Information Systems (FOIS'98). IOS Press. Trento, Italy. 1998. 3-15.

[Gupta *et al*., 2002] Gupta, K.M., Aha, D.W., Marsh, E. and Maney, T.: An architecture for engineering sublanguage WordNets. In Proceedings of the First International Conference On Global WordNet. Mysore, India. 2002. 207-215.

[Haase, 2000] Haase, K.: Interlingual BRICO. IBM Systems Journal, 39. 2000. 589-596.

[Hahn and Schnattinger, 1998] Hahn, U. and Schnattinger, K.: Towards text knowledge engineering. In Proceedings of the 15th National Conference on Artificial Intelligence & 10th Conference on Innovative Applications of Artificial Intelligence. AAAI Press / MIT Press. Madison, Wisconsin, Menlo Park, CA; Cambridge, MA. 1998. 524-531.

[Hahn and Schulz, 2000] Hahn, U. and Schulz, S.: Towards Very Large Terminological Knowledge Bases: A Case Study from Medicine. In Proceedings of the Canadian Conference on AI 2000. 2000. 176-186.

[Hammer *et al*., 1997] Hammer, J. McHugh, J. and Garcia-Molina, H.: Semistructured data: The TSIMMIS experience. In Proceedings of the 1st East-European Symposium on Advances in Databases and Information Systems. 1997. 1-8.

[Hearst, 1992] Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In Proceedings of the 14th International Conference on Computational Linguistics (COLING-92). Nantes, France, 1992. 539–545.

[Hearst, 1996] Hearst, M.A.: Improving Full-Text Precision on Short Queries using Simple Constraints. In Proceedings of the Symposium on Document Analysis and Information Retrieval. Las Vegas, NV. 1996.

[Hearst, 1998] Hearst, M.A.: Automated Discovery of WordNet Relations. In Christiane Fellbaum (Ed.) WordNet: An Electronic Lexical Database, MIT Press. 1998. 132-152.

[Helflin and Hendler, 2000] Heflin, J. and Hendler, J.: Searching the Web with SHOE, In: Papers from the AAAI Workshop on Artificial Intelligence for Web Search, pp. 35-40, 2000.

[Hirst and St-Onge, 1998] Hirst, G. and St-Onge D.: Lexical chains as representations of context for the detection and correction of malapropisms. In Fellbaum C, editor. WordNet: An electronic lexical database. Cambridge, MA: MIT Press. 1998. 305–321.

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

[Hsu and Dung, 1998] Hsu, C.N. and Dung, M.T.: Wrapping semistructured Web pages with finite-state transducers. In Proceedings of the Conference on Automatic Learning and Discovery, 1998. 66-73.

[Hwang, 1999] Hwang, C.H.: 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. July 29-30, 1999. 14-20.

[Ide and Veronis, 1998] Ide, N. and Veronis, J.: Introduction to the Special Issue on Word Sense Disambiguation: The State of the Art. Computational Linguistics. 24(1). 1998. 1–40.

[IEEE, 2001] IEEE, Special issue on the Semantic Web. IEEE Intelligent Systems. 2001.

[Iria, 2006] Iria, J., Brewster, C., Ciravegna, F. and Wilks, Y.: An Incremental Tri-Partite Approach to Ontology Learning. In Proceedings of the International Conference on Language Resources and Evaluation. Genoa, 22-28 May, 2006.

[Iwanska *et al*., 2000] Iwanska, L.M., Mata, N. and Kruger, K.: Fully automatic acquisition of taxonomic knowledge from large corpora of texts. In L.M. Iwanksa and S.C. Shapiro, editors, Natural Language Processing and Knowledge Processing. MIT/AAAI Press. 2000. 335-345.

[Jans, 2000] Jans, T.B.: The effect of query complexity on Web searching results. Information Research, 6(1). October 2000.

[Jennings, 2000] Jennings, N.: On agent-based software engineering. Artificial Intelligence 117. 2000. 277-296

[Jiang and Conrath, 1997] Jiang, J. and Conrath, D.: Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In Proceedings of the 10th International Conference on Research on Computational Linguistics. Taiwan. 1997.

[Jones and Paynter, 2002] Jones, S. and Paynter, G.W.: Automatic extraction of document keyphrases for use in digital libraries: evaluation and applications. Journal of the American Society for Information Science and Technology 53(8). 2002. 653-677.

[Kahan *et al*., 2001] Kahan, J., Koivunen, M., Hommeaux, E. and Swick, R.: Annotea: an open rdf infrastructure for shared web annotations. In Proceedings of the WWW10 Conference. Hong Kong. May 1-5, 2001. 623-632.

[Kashyap, 1999] Kashyap, V.: Design and Creation of Ontologies for Environmental Information Retrieval. In Proceedings of the 11th European Workshop on Knowledge Acquisition, Modelling and Management (EKAW). 1999.

[Kavalec and Svatek, 2005] Kavalec, M. and Svatek, V.: A Study on Automated Relation Labelling in Ontology Learning. In P.Buitelaar, P. Cimiano, B. Magnini (eds.), Ontology Learning and Population from Text: Methods, Evaluation and Applications, IOS Press, 2005. 44-58.

[Kavalec *et al*., 2004] Kavalec, M., Maedche, A. and Skátek, V.: Discovery of Lexical Entries for Non-taxonomic Relations in Ontology Learning. In Proceedings of SOFSEM 2004, LNCS 2932. 2004. 249-256.

[Keller *et al*., 2002] Keller, F., Lapata, M. and Ourioupina, O.: Using the web to overcome data sparseness. In Proceedings of EMNLP-02. 2002. 230-237.

[Kesseler 1996] Kesseler, M.: A Schema Based Approach to HTML Authoring. World Wide Web Journal 96(1). O'Reilly, 1996.

[Khan and Luo, 2002] Khan, L. and Luo, F.: Ontology Construction for Information Selection In Proceedings of 14th IEEE International Conference on Tools with Artificial Intelligence. Washington DC. November 2002. 122-127.

[Kietz *et al*., 2000] Kietz, J.U., Maedche, A. and Volz, R.: A Method for Semi-Automatic Ontology Acquisition from a Corporate Intranet. In Aussenac-Gilles N, Biébow B, Szulman S (eds) EKAW'00 Workshop on Ontologies and Texts. Juan-Les-Pins, France. October, 2000. 37-50.

[Kipper *et al*., 2000] Kipper, K., Dang, H.T. and Palmer, M.: Class-based construction of a verb lexicon. In Proceedings of the 7th National Conference on Artificial Intelligence AAAI 2000. Austin, USA. 2000. 691-696.

[Kobayashi and Takeda, 2000] Kobayashi, M. and Takeda, K.: Information Retrieval on the Web. ACM Computing Surveys, 32(2). 2000. 144–173.

[Krupka and Hausman, 1998] Krupka, G. and Hausman, K.: IsoQuest, Inc.: Description of the NetOwl extractor system as used for MUC-7. In Proceedings of the 7th Message Understanding Conference (MUC-7), Fairfax, Virginia. 1998.

[Kusmerick 2000] Kusmerick, N.: Wrapper induction: efficiency and expressiveness. Artificial Intelligence Journal 118(1-2). 2000. 15-68.

[Kwok *et al*., 2001] Kwok, C.T., Etzioni, O. and Weld, D.S.: Scaling question answering to the web. ACM Transactions on Information Systems. 2001. 150-161.

[Laender *et al*., 2002] Laender, A.H.F., Ribeiro-Neto, B.A., da Silva, A.S. and Teixeira, J.S.: A brief survey of Web data extraction Tools. SIGMOD Records 31(2). 2002. 84-93.

[Lamparter *et al*., 2004] Lamparter, S., Ehrig, M. and Tempich, C.: Knowledge Extraction from Classification Schemas. In Proceedings of the CoopIS/DOA/ODBASE 2004. LNCS 3290. 2004. 618-636.

[Landauer and Dumais, 1997] Landauer, T.K. and Dumais, S.T.: A Solution to Plato's Problem: The Latent Semantic Analysis Theory of the Acquisition, Induction, and Representation of Knowledge. Psychological Review, 104. 1997. 211-240.

[Lassila and McGuinness, 2001] Lassila, O., McGuinness, D.: The Role of Frame-Based Representation on the Semantic Web. Technical Report KSL-01-02. Knowledge Systems Laboratory. Stanford University. Standford, California. 2001.

[Lawrence, 2000] Lawrence, S.: Context in Web Search. IEEE Data Engineering Bulletin 23(3). 2000. 25–32.

[Lawrie and Croft, 2003] Lawrie, D.J. and Croft, B.: Generating Hierarchical Summaries for Web Searches. In Proceedings of SIGIR'03. Toronto Canada. 2003. 457-458.

[Leacock and Chodorow, 1998] Leacock, C. and Chodorow, M.: Combining local context and WordNet similarity for word sense identification. In C. Fellbaum, editor, WordNet: An electronic lexical database. MIT Press.1998. 265–283.

[Lee *et al*., 1993] Lee, J.H., Kim, M.H. and Lee, Y.J.: Information Retrieval Based on Conceptual Distance in ISA Hierarchies. Journal of Documentation, 49. 1993. 188-207.

[Lee *et al*., 2003] Lee, C., Na, J. and Khoo C.: Ontology Learning for Medical Digital Libraries. In Proceedings of ICADL 2003. LNCS 2911. 2003. 302-305.

[Lenat and Guha, 1990] Lenat, D.B. and Guha, R.V.: Building Large Knowledge-based Systems: Representation and Inference in the Cyc Project. Addision-Wesley, Boston, Massachusetts. 1990.

[Levin, 1993] Levin, B.: English verb classes and alternations. PhD Thesis. Chicago University Press. 1993.

[Lin, 1998] Lin, D.: Automatic Retrieval and Clustering of Similar Words. In Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics. Montreal. 1998. 768-773.

[Lonsdale *et al*., 2002] Lonsdale, D., Ding, Y., Embley, D.W. and Melby A.: Peppering Knowledge Sources with SALT; Boosting Conceptual Content for Ontology Generation. Proceedings of the AAAI Workshop on Semantic Web Meets Language Resources, Edmonton, Alberta, Canada. July 2002. 30-36.

[Lopez and Motta 2004] Lopez, V. and Motta, E.: Ontology-Driven Question Answering in AquaLog. In Proceedings of NLDB. 2004. 89-102.

[Maarek *et al*., 2000] Maarek, Y.S., Fagin, R., Ben-Shaul I.Z. and Pelleg, D.: Ephemeral document clustering for web applications. Technical Report RJ 10186. IBM Research. 2000.

[Maedche and Staab, 2000] Maedche, A. and Staab S.: Discovering Conceptual Relations from Text. In Proceedings of the 14th European Conference on Artificial Intelligence. IOS Press, Amsterdam. 2000. 321-325.

[Maedche and Staab, 2001] Maedche, A. and Staab, S.: Ontology Learning for the Semantic Web. IEEE Intelligent Systems, Special Issue on the Semantic Web, 16(2). 2001.

[Maedche and Staab, 2003] Maedche, A. and Staab, S.: Ontology Learning. In S. Staab & R. Studer (eds.) Handbook on Ontologies in Information Systems. Springer. 2003.

[Maedche *et al*., 2003] Maedche, A., Neumann, G. and Staab, S.: Bootstrapping an Ontology-Based Information Extraction System, Studies in Fuzziness and Soft Computing. Intelligent Exploration of the Web, Springer, 2003. 245-259.

[Maedche, 2002] Maedche, A.: Ontology Learning for the Semantic Web. Kluwer Academic Publishers. 2002.

[Magnin *et al*., 2002] Magnin, L., Snoussi, H., Nie, J.: Toward an Ontology–based Web Extraction. In Proceedings of the Fifteenth Canadian Conference on Artificial Intelligence, 2002.

[Magnini *et al*., 2003] Magnini, B., Serafini. L. and Speranza, M.: Making explicit the hidden Semantics of hierarchical classifications. ITC-first Technical report 0306-09. June 2003.

[Manning and Schütze, 1999] Manning, C.D. and Schütze, H.: Foundations of Statistical Natural Language Processing. Cambridge, Massachusetts: MIT Press. 1999.

[Markert *et al*., 2003] Markert, K., Modjeska, N. and Nissim, M.: Using the web for nominal anaphora resolution. In Proceedings of the EACL Workshop on the Computational Treatment of Anaphora. 2003.

[Martin and Eklund, 2000] Martin, P. and Eklund, P: Knowledge Indexation and Retrieval and the Word Wide Web. IEEE Intelligent Systems, Special Issue "Knowledge Management and Knowledge Distribution over the Internet". 2000.

[McCallum, 2003] McCallum, A.: Efficiently inducing features or conditional random fields. In Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence, 2003.

[Meaning, 2005] Meaning Project: Developing Multilinguial Web Scale Technologies. IST-2001-34460. http://nipadio.lsi.upc.edu/wei4/doc/mcr/meaning.html. 2005.

[Mihalcea and Edmonds, 2004] Mihalcea, R. and Edmonds, P.: Proceedings of Senseval-3: The 3rd Int. Workshop on the Evaluation of Systems for the Semantic Analysis of Text. Barcelona, Spain. 2004.

[Mikheev and Finch, 1997] Mikheev, A. and Finch, S.: A Workbench for Finding Structure in Texts. In Proceedings of ANLP-97. Washington D.C. 1997. 8-16.

[Mikheev *et al*., 1999] Mikheev, A., Moens, M. and Grover, C.: Named entity recognition without gazetteers. In Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL-99). Bergen, Norway. 1999. 1–8.

[Miller, 1996] Miller, G.A.: Contextuality, in Mental Models in Cognitive Science. J. Oakhill and A. Garnham, Editors. Psychology Press: East Sussex, UK. 1996. 1–18.

[Missikoff *et al*., 2002] Missikoff, M., Navigli, R. and Velardi, P.: The Usable Ontology: An Environment for Building and Assessing a Domain Ontology. In Proceedings of the International Semantic Web Conference (ISWC) 2002. Sardinia, Italia. June 9-12th, 2002. 39-53.

[Moldovan and Girju, 2001] Moldovan, D.I. and Girju, R.C.: An interactive tool for the rapid development of knowledge Bases. International Journal on Artificial Intelligence Tools 10(1-2). March, 2001. 65-86.

[Montoyo, 2000] Montoyo, A: Método basado en Marcas de Especificidad para WSD. Procesamiento del Lenguaje Natural, 26. Septiembre, 2000.

[Moreno *et al*., 2004] Moreno, A., Riaño, D., Isern, D., Bocio, J., Sánchez, D., Jiménez, L.: Knowledge Exploitation from the Web. In Proceedings of the 5th International Conference on Practical Aspects of Knowledge Management (PAKM 2004). LNAI, 3336. Vienna. 2004. 175-185.

[Moreno *et al*., 2005] Moreno, A., Valls, A., Sánchez, D. and Isern, D.: Construcción automática de ontologies para la Web Semántica. In Proceedings of the Workshop in Ontologías y la Web Semántica, CAEPIA 2005. Santiago de Compostela. 2005.

[Moreno *et al*., 2006] Moreno, A., Valls, A., Isern, D. and Sánchez, D.: Applying Agent Technology to Healthcare: The GruSMA Experience. IEEE Intelligent Systems 21(6). 2006. 63-67.

[Morin, 1999] Morin E.: Automatic acquisition of semantic relations between terms from technical corpora. In Proceedings of the fifth international congress on terminology and knowledge engineering (TKE-99). Vienna. 1999. 268-278,

[Morita *et al*., 2004] Morita, T., Shigeta, Y., Sugiura, N., Fukuta, N., Izumi, N., Yamaguchi, T.: DODDLE-OWL: OWL-based Semi-Automatic Ontology Development Environment. In Proceedings of EON2004. 2004.

[Mulholland *et al*., 2001] Mulholland, P., Zdrahal, Z., Domingue, J., Hatala, M., Bernardi, A.: A Methodological Approach to Supporting Organizational Learning. International Journal of Human-Computer Studies, 55(3), 2001. 337-367.

[Muslea *et al*., 2001] Muslea, I., Minton, S. and Knoblock, C.: Hierarchical wrapper induction for semistructured information sources, Autonomous Agents and Multi-Agent Systems 4. 2001. 93-114.

[Navigli and Velardi, 2004] Navigli, R. and Velardi, P.: Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites. In Computational Linguistics 30(2). June, 2004. 151-179.

[Neches *et al*., 1991] Neches, R., Fickes, R.E., Finin, T. Gruber, T.R., Senator, T. and Swartout W.R.: Enabling technology for knowledge sharing. AI Magazine 12(3). 1991. 36-56.

[Nirenburg and Raskin, 2004] Nirenburg, S. and Raskin, V.: Ontological Semantics SERIES: Language, Speech, and Communication, MIT Press, 2004.

[Nobécourt, 2000] Nobécourt, J.: A method to build formal ontologies from text. In Proceedings of the EKAW-2000 Workshop on ontologies and text. Juan-Les-Pins, France. 2000.

[Oliveira *et al*., 2001] Oliveira, A., Pereira, F.C. and Cardoso, A.: Automatic Reading and Learning from Text. In Proceedings of the International Symposium on Artificial Intelligence, ISAI'2001. December, 2001.

[OntoWeb, 2002] OntoWeb D.1.3: "Whitepaper: ontology evaluation tools". Available at: http://www.aifb.unikarlsruhe.de/WBX/ysu/publications/eon2002_whitepaper.pdf. 2002.

[Palmer *et al*., 1998] Palmer, M., Rosenzweig, J. and Schuler, W.: Capturing Motion Verb Generalizations with Synchronous TAGs. Predicative Forms in NLP. Kluwer Press. December, 1998.

[Pantel and Ravichandran, 2004] Pantel, P. and Ravichandran, D.: Automatically labelling semantic classes. In Proceedings of the 2004 Human Language Technology Conference (HLT-NAACL-04). Boston, Massachusetts. 2004. 321–328.

[Pasca and Harabagiu 2001] Pasca, M. and Harabagiu, S.: The Informative Role of WordNet in Open-Domain Question Answering. In Proceedings of the NAACL Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations. 2001. 138-143.

[Pasca, 2004] Pasca, M.: Acquisition of Categorized Named Entities for Web Search. In Proceedings of the thirteenth ACM International Conference on Information and Knowledge Management. USA. 2004. 137-145.

[Pasca, 2005] Pasca, M.: Finding Instance Names and Alternative Glosses on the Web: WordNet Reloaded. In Proceedings of CICLing 2005. LNCS 3406. 2005. 280-292.

[Patwardhan and Pedersen, 2006] Patwardhan S, Pedersen T.: Using WordNet-based context vectors to estimate the semantic relatedness of concepts. In: Proceedings of the EACL 2006 workshop, making sense of sense: Bringing computational linguistics and psycholinguistics together. Trento, Italy. 2006. 1–8.

[Patwardhan, 2003] Patwardhan, S.: Incorporating dictionary and corpus information into a context vector measure of semantic relatedness. Master of Science Thesis, Department of Computer Science, University of Utah. 2003.

[Pease and Niles, 2002] Pease, R.A. and Niles, I.: IEEE Standard Upper Ontology: A Progress Report. The Knowledge Engineering Review 17(1). 2002. 65-70.

[Pedersen *et al*., 2006] Pedersen, T., Serguei, Pakhomov, S., Patwardhan, S., Chute, C.: Measures of semantic similarity and relatedness in the biomedical domain. Journal of Biomedical Informatics. 2006.

[Pedersen, *et al*., 2004] Pedersen, T., Patwardhan, S. and Michelizzi, J.: WordNet::Similarity – Measuring the Relatedness of Concepts. http://search.cpan.org/dist/WordNet-Similarity. American Association for Artificial Intelligence. 2004.

[Petrie, 2001] Petrie, C.: Agent-based software engineering. Agent-Oriented Software Engineering. LNAI. 1957. Springer-Verlag, Berlin. 2001. 58-76.

[Phillips and Riloff, 2002] Phillips, W. and Riloff, E.: Exploiting strong syntactic heuristics and co-training to learn semantic lexicons. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-02). Philadelphia, Pennsylvania. 2002. 125–132.

[Popescu and Etzioni, 2005] Popescu, A. and Etzioni, O.: Extracting Product Features and Opinions from Reviews. In Proceedings of HLT-EMNLP. 2005.

[Pustejovsky *et al*., 2002] Pustejovsky, J., Castano, J., Zhang, J., Cochran, B. and Kotecki, M.: Robust relational parsing over biomedical literature: Extracting inhibit relations. In Proceedings of the Pacific Symposium on Biocomputing. 2002. 362-373.

[Ravichandran and Hovy, 2002] Ravichandran, D. and Hovy, E.: Learning surface text patterns for a question answering system. In Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics (ACL-02), Philadelphia, Pennsylvania. 2002. 41-47.

[Reinberger and Sypns, 2004] Reinberger, M.L., Spyns, P.: Discovering knowledge in texts for the learning of DOGMA inspired ontologies. In Proceedings of the ECAI 2004 Workshop on Ontology Learning and Population. 2004. 19-24.

[Reinberger *et al*., 2004] Reinberger, M.L., Spyns, P., Pretorius, A.J. and Daelemans, W.: Automatic initiation of an ontology. In R. Meersman, Z. Tari *et al*. (eds.), On the Move to Meaningful Internet Systems, LNCS 3290 , Springer. 2004. 600–617.

[Resnik and Smith, 2003] Resnik, P. and Smith, N.: The web as a parallel corpus. Computational Linguistics, 29(3). 2003. 349-380.

[Resnik, 1993] Resnik, P.: Selection and Information: A Class-based Approach to Lexical Relationships. PhD thesis, University of Pennsylvania. 1993.

[Resnik, 1998] Resnik, P.: Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. Journal of Artificial Intelligence Research, 11. 1998. 95-130.

[Richardson *et al*., 1994] Richardson, R., Smeaton, A. and Murphy, J.: Using WordNet as a Knowledge Base for Measuring Semantic Similarity between Words. In Proceedings of the AICS Conference. Trinity College, Dublin. 1994.

[Ridings and Shishigin, 2002] Ridings, C. and Shishigin, M: PageRank Uncovered. Available at: http://www.voelspriet2.nl/PageRank.pdf. 2002.

[Rinaldi *et al*., 2005] Rinaldi, F., Yuste, E., Schneider, G., Hess, M. and Roussel, D.: Exploiting Technical Terminology for Knowledge Management. In P.Buitelaar, P. Cimiano, B. Magnini (eds.), Ontology Learning and Population, IOS Press, 2005.

[Rosso *et al*., 2003] Rosso, P., Masulli, F., Buscaldi, D., Pla, F. and Molina, A.: Automatic Noun Disambiguation. In Proceedings in Computational Linguistics and Intelligent Text Processing (CICLing-2003). LNCS, 2588. Springer-Verlag. 2003. 273–276.

[Rosso *et al*., 2005] Rosso P., Montes-y-Gomez, M., Buscaldi, D., Pancardo-Rodrıguez, A. and Villaseñor, L.: Two Web-Based Approaches for Noun Sense Disambiguation. In Proceedings of CICLing 2005. LNCS 3406. 2005. 267–279.

[Roux *et al*., 2000] Roux, C., Proux, D., Rechermann, F. and Julliard, L.: An ontology enrichment method for a pragmatic information extraction system gathering data on genetic interactions. In Proceedings of the ECAI2000 Workshop on Ontology Learning (OL2000). Berlin, Germany. August, 2000.

[Sabou, 2004] Sabou, M.: Extracting ontologies from software documentation: a semi-automatic method and its evaluation. In Proceedings of the ECAI-2004 Workshop on Ontology Learning and Population (ECAI-OLP). 2004.

[Sabou, 2005] Sabou, M.: Learning Web Service Ontologies: an Automatic Extraction Method and its Evaluation In Ontology Learning. In P.Buitelaar, P. Cimiano, B. Magnini (eds.), Ontology Learning and Population. IOS Press, 2005.

[Sabou, 2006] Sabou, M.: Building Web Service Ontologies. PhD Thesis. SIKS Dissertation Series. 2006.

[Sánchez and Moreno, 2004a] Sánchez, D. and Moreno, A.: Creating ontologies from Web documents. In Proceedings of the Setè Congrés Català d'Intel·ligència Artificial (CCIA'04) IOS Press. Barcelona. October 21-22, 2004. 11-18.

[Sánchez and Moreno, 2004b] Sánchez, D. and Moreno, A.: Automatic generation of taxonomies from the WWW. In Proceedings of the 5th International Conference on Practical Aspects of Knowledge Management (PAKM 2004). LNAI 3336. Vienna, Austria. December 2-3, 2004. 208-219.

[Sánchez and Moreno, 2005a] Sánchez, D. and Moreno A.: Development of new techniques to improve Web Search. In Proceedings of the 9th International Joint Conference on Artificial Intelligence (IJCAI'05) Edinburgh, Scotland. 30 July – 5 August 2005. 1632-1633.

[Sánchez and Moreno 2005b] Sánchez, D. and Moreno, A.: Web Mining Techniques for Automatic Discovery of Medical Knowledge. In Proceeding of the 10th Conference on Artificial Intelligence in Medicine (AIME 05). LNAI 3581. Aberdeen, Scotland. 23 - 27 July 2005. 409-413.

[Sánchez and Moreno 2005c] Sánchez, D. and Moreno, A.: Web-scale taxonomy learning. In Proceedings of the Workshop Learning and Extending Lexical Ontologies by using Machine Learning Methods. ICML 2005. Bonn, Germany, 7 - 11 August 2005.

[Sánchez and Moreno, 2005d] Sánchez, D. and Moreno, A.: Automatic discovery of synonyms and lexicalizations from the Web. In Proceedings of the Vuitè Congrés Català d'Intel·ligència Artificial (CCIA'05). Artificial Intelligence Research and Development 131. IOS Press. L'Alguer, Italy. 26-28 October 2005. 205-212.

[Sánchez and Moreno, 2005e] Sánchez, D. and Moreno, A.: A Multi-agent System for Distributed Ontology Learning. In Proceedings of the Third International Workshop on Multi-Agent systems, EUMAS 2005. Brussels, Belgium. 7-8 December 2005. 504-505.

[Sánchez and Moreno, 2006a] Sánchez, D. and Moreno, A.: A methodology for knowledge acquisition from the web. International Journal of Knowledge-Based and Intelligent Engineering Systems 10(6). 2006. 453-475.

[Sánchez and Moreno, 2006b] Sánchez, D. and Moreno, A.: Discovering Non-taxonomic Relations from the Web. In Proceedings of the 7th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2006). E. Corchado *et al*. (Eds). LNCS 4224. Burgos, Spain. 20-23 September, 2006. 629-636.

[Sánchez and Moreno, 2007a] Sánchez, D. and Moreno, A.: Learning Medical Ontologies from the Web. In Proceedings of the Workshop From Knowledge to Global Care. 11th Conference on Artificial Intelligence in Medicine. 7 July, 2007.

[Sánchez and Moreno, 2007b] Sánchez, D. and Moreno, A.: Bringing taxonomic structure to large digital libraries. To be published in the International Journal on Metadata, Semantics and Ontologies. 2007.

[Sánchez and Moreno, 2007c] Sánchez, D. and Moreno, A.: Semantic disambiguation of taxonomies. In Proceedings of the Desè Congrés Internacional de l'Associació Catalana d'Intel·ligència Artificial (CCIA'07). IOSPress. Andorra. October 25-26, 2007.

[Sánchez and Moreno, 2007d] Sánchez, D. and Moreno, A.: Pattern-based automatic taxonomy learning from the Web. To be published in the European Journal on Artificial Intelligence (AI Communications). IOS Press. 2007.

[Sánchez *et al*., 2005] Sánchez, D., Isern, D. and Moreno, A.: An Agent-Based Knowledge Acquisition Platform. 9th International Workshop on Cooperative Information Agents. In Proceedings of the Third German Conference on Multiagent System Technologies (MATES/CIA 2005). LNAI 3550. Koblenz, Germany, 11-13 September 2005. 118-129.

[Sánchez *et al*., 2006] Sánchez, D., Isern, D. and Moreno, A.: Integrated Agent-Based Approach for Ontology-Driven Web Filtering. In Proceedings of the 10th International Conference on Knowledge-Based & Intelligent Information & Engineering System (KES 2006). LNAI 4253. Bournemouth, UK. 9-11 October, 2006. 758-765.

[Sánchez *et al*., 2007] Sánchez, D., Rodríguez, A. and Moreno, A.: Parallel execution of complex tasks using a distributed, robust and flexible agent-based platform. In Proceedings of the III Taller en Desarrollo de Sistema Multiagente. II Congreso Español de Informática. Setember 11-14, 2007.

[Sanderson and Croft, 1999] Sanderson, M. and Croft, B.: Deriving concept hierarchies from text. In Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Berkeley, USA. 1999. 206-213.

[Schlobach *et al*., 2004] Schlobach, S., Olsthoorn, M. and de Rijke, M.: Type checking in open-domain question answering. In Proceedings of the European Conference on Artificial Intelligence (ECAI). 2004.

[Schurr and Staab, 2000] Schnurr, H.P. and Staab, S.: A proactive inferencing agent for desk support. In Proceedings of the AAAI Symposium on Bringing Knowledge to Business Processes. Stanford, CA, USA. AAAI Technical Report, Menlo Park. 2000.

[Schutz and Buitelaar, 2005] Schutz, A., Buitelaar, P.: RelExt: A Tool for Relation Extraction in Ontology Extension. In Proceedings of the 4th International Semantic Web Conference. 2005. 593-606.

[Schütze, 1993] Schütze, H.: Word Space. In: S.J. Hanson, J.D. Cowan, and C.L. Giles (eds.), Advances in Neural Information Processing Systems 5, San Mateo California: Morgan Kaufmann. 1993. 895-902.

[Senseval, 2004] Sens Eval: Evaluation exercises for Word Sense Disambiguation. http://www.senseval.org/publications/senseval.pdf. 2004.

[Shamsfard and Barforoush, 2002] Shamsfard, M. and Barforoush, A.: An introduction to Hasti: An Ontology Learning System. Artificial Intelligence Soft Computing. 2002.

[Sheth, 2003] Sheth, A.: Ontology-driven information search, integration and analysis. In Proceedings of MATES. 2003.

[Sidorov and Gelbukh, 2001] Sidorov, G. and Gelbukh, A.: Word Sense Disambiguation in a Spanish Explanatory Dictionary. In Proceedings of TALN-2001. France. 2001. 398-402.

[Sinha and Narayanan, 2005] Sinha, S. and Narayanan, S.: Model Based Answer Selection. In: Proceedings of the AAAI Workshop on Textual Inference in Question Answering. 2005.

[Sintek *et al*., 2004] Sintek, M., Buitelaar, P. and Olejnik, D.: A Formalization of Ontology Learning From Text. In Proceedings of EON2004. 2004.

[Skounakis *et al*., 2003] Skounakis, M., Craven, M. and Ray, S.: Hierarchical hidden markov models for information extraction. In Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence. 2003. 427-433.

[Smith and Poulter, 1999] Smith, H. and Poulter, K.: Share the Ontology in XML-based Trading Architectures. Communications of the ACM 42(3). 1999. 110-111.

[Snow *et al*., 2004] Snow, R., Jurafsky, D. and Ng, A.Y.: Learning syntactic patterns for automatic hypernym discovery. Advances in Neural Information Processing Systems 17. 2004. 1297-1304.

[Soderland *et al*., 1995] Soderland, S., Fisher, D., Aseltine, J. and Lehnert, W.: CRYSTAL: Inducing a conceptual dictionary. In Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence. 1995. 1314–1321.

[Soderland, 1999] Soderland, S.: Learning information extraction rules for semistructured and free text. Machine Learning, 34(1-3). 1999. 233-272.

[Solorio *et al*., 2004] Solorio, T., Pérez, M., Montes, M., Villaseñor, L. and López, A.: A Language Independent Method for Question Classification. In Proceedings of the 20th International Conference on Computational Linguistics (COLING-04). Geneva, Switzerland. 2004. 1374-1380.

[Sowa, 1999] Sowa, J.F.: Knowledge Representation: Logical, Philosophical, and Computational Foundations. Brooks Cole Publishing Co., Pacific Grove, California. 1999.

[Spink, 2001] Spink, A., Wolfram, D., Jansen, B.J. and Saracevic, T.: Searching the Web: The Public and Their Queries. Journal of the American Society for Information Science. 52(3). 2001. 226–234.

[Staab and Schnurr 2000] Staab, S. and Schnurr, H.P.: Smart Task Support through Proactive Access to Organizational Memory. Journal of Knowledge-based Systems, Elsevier, 2000. 251-260.

[Stevenson and Gaizauskas, 2000] Stevenson, M. and Gaizauskas, R.: Using corpus-derived name lists for named entity recognition. In Proceedings of the 6th Conference on Applied Natural Language Processing (ANLP-00). Seattle, Washington. 2000. 290-295.

[Stevenson *et al*., 2005] Stevenson, M., and Greenwood, M.: A Semantic Approach to IE Pattern Induction. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics, 2005. 379-386.

[Stokoe *et al*., 2003] Stokoe, C., Oakes, M.P. and Tait, J: Word Sense Disambiguation in Information Retrieval Revisited. In Proceedings of the 26th ACM SIGIR. Canada. 2003. 159-166.

[Studer *et al*., 1998] Studer, R., Benjamins, V.R. and Fensel., D.: Knowledge Engineering: Principles and Methods. IEEE Transactions on Knowledge and Data Engineering 25(1-2). 1998. 161-197.

[Stumme *et al*., 2003] Stumme, G., Ehrig, M., Handschuh, S., Hotho, A., Maedche, A., Motik, B., Oberle, D., Schmitz, C., Staab, S., Stojanovic, L., Stojanovic, N., Studer, R., Sure, Y., Volz, R., and Zacharias, V.: The Karlsruhe View on Ontologies. Technical Report University of Karlsruhe, Institute AIFB, 2003.

[Sure *et al*. 2000] Sure, Y., Maedche, A. and Staab, S.: Leveraging Corporate Skill Knowledge -- From ProPer to OntoProPer, In Proceedings of PAKM. 2000. 1-9.

[Surowiecky, 2004] Surowiecki, J.: The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations. Doubleday Books, 2004.

[Szulman *et al*., 2002] Szulman, S., Biebow, B. and Aussenac-Gilles, N.: Structuration de Terminologies à l'aide d'outils d'analyse de textes avec TERMINAE. Traitement Automatique de la Langue (TAL) 43(1). 2002. 103-128.

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

[Turney, 2001] Turney, P.D.: Mining the Web for synonyms: PMI-IR versus LSA on TOEFL. In Proceedings of the Twelfth European Conference on Machine Learning. Freiburg, Germany 2001. 491-499.

[Uschold and Grunninger, 1996] Uschold, M. and Grunninger, M.: Ontologies. Principles, Methods and Applications. Knowledge Engineering Review 11(2), 1996. 93-155.

[Uschold *et al*. 1998] Uschold, M., King, M., Moralee, S. and Zorgios, Y.: The Enterprise Ontology, Knowledge Engineering Review, 13(1). 1998. 31-89.

[Valarakos *et al*., 2004] Valarakos, A.G., Paliouras G., Karkaletsis V. and Vouros, G.: Enhancing Ontological Knowledge Through Ontology Population and Enrichment. In Proceedings of EKAW 2004. LNAI 3257. 2004. 144-156.

[Van Heijst *et al*., 1997] Van Heijst, F., Schreiber A., and Wielinga, B.J.: Using explicit ontologies in KBS development. International Journal of Human-Computer Studies 45. 1997. 183-292.

[Velardi *et al*., 2002] Velardi, P., Navigli, R. and Missikoff, M.: Integrated approach for Web ontology learning and engineering. IEEE Computer 35. November, 2002. 60-63.

[Velardi *et al*., 2005] Velardi, P., Navigli, R., Cucchiarelli, A. and Neri, F.: Evaluation of OntoLearn, a methodology for automatic learning of domain ontologies. Paul Buitelaar Philipp Cimmiano and Bernardo Magnini Editors. IOS Press. 2005.

[Vintar *et al*., 2003] Vintar, S., Todorovski, L., Sonntag, D. and Buitelaar, P.: Evaluating context features for medical relation mining. In Proceedings of the ECML/PKDD Workshop on Data Mining and Text Mining for Bioinformatics. 2003.

[Volk, 2001] Volk, M.: Exploiting the WWW as a Corpus to Resolve PP Attachment Ambiguities. In Proceedings of Corpus Linguistics. Lancaster. 2001.

[Volk, 2002] Volk, M.: Using the Web as Corpus for Linguistic Research. Catcher of the Meaning. Pajusalu, R., Hennoste, T. (Eds.). Department of General Linguistics 3, University of Tartu, Germany. 2002.

[Voorhees, 1994] Voorhees, E.M.: Query Expansion Using Lexical-Semantic Relations. In Proceedings of the 17th Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland. 1994. 61-69.

[Voorhess, 2001] Voorhees, E.M.: Overview of the TREC 2001 question answering track. In Proceedings of the Text REtrieval Conference, 2001. 42-51.

[Vossen, 1998] Vossen, P.: EuroWordNet: A Multilingual Database with Lexical Semantic Networks. Dordrecht, Netherlands: Kluwer. Available at: http://www.hum.uva.nl/~ewn/. 1998.

[Vossen, 2001] Vossen, P.: Extending, trimming and fusing WordNet for technical documents. In Proceedings of the NAACL Workshop on WordNet and Other Lexical Resources. Pittsburgh. 2001.

[Wagner, 2000] Wagner, A.: Enriching a lexical semantic net with selectional preferences by means of statistical corpus analysis. In Proceedings of the ECAI-2000 Workshop on Ontology Learning, Berlin. August, 2000. 37-42.

[Weiss, 1999] Weiss, G.: Mutliagent systems: a modern approach to distributed artificial intelligence. The MIT Press, Cambridge. 1999.

[Weng *et al*., 2006] Weng, S., Tsai, H., Liu, S. and Hsu, C.: Ontology construction for information classification. Expert Systems with Applications 31. 2006. 1–12.

[Widdows, 2003] Widdows, D.: Unsupervised methods for developing taxonomies by combining syntactic and statistical information. In Proceedings of the Human Language Technology / Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL). Canada, 2003. 276-283.

[Wiemer-Hastings *et al*., 1998] Wiemer-Hastings, P., Graesser, A.: Inferring the meaning of verbs from context. In Proceedings of the Twentieth Annual Conference of the Cognitive Science Society. 1998. 1142-1147.

[William, 2002] William, L.: Measuring Conceptual Distance Using WordNet: The Design of a Metric for Measuring Semantic Similarity. R. Hayes, W. Lewis, E. Obryan, and T. Zamuner (Eds.), The University of Arizona Working Papers in Linguistics. Tucson: University of Arizona. 2002.

[Wooldridge, 2002] Wooldridge, M.: An Introduction to multiagent systems. West Sussex, England: John Wiley and Sons, Ltd. 2002.

[Wu and Hsu, 2002] Wu, S.H and Hsu W.L.: SOAT: A Semi-Automatic Domain Ontology Acquisition Tool from Chinese Corpus. In Proceedings of the 19th International Conference on Computational Linguistics. Taipei, Taiwán. 2002. 1-5.

[Wu and Palmer, 1994] Wu Z. and Palmer M.: Verb semantics and lexical selection. In Proceedings of the 32nd annual meeting of the association for computational linguistics. Las Cruces. 1994. 133–8.

[Xu *et al*., 2002] Xu, F., Kurz, D., Piskorski, J. and Schmeier, S.: A Domain Adaptive Approach to Automatic Acquisition of Domain Relevant Terms and their Relations with Bootstrapping. In Proceedings of LREC 2002, the third international conference on language resources and evaluation. Las Palmas, Canary island, Spain. May 2002.

[Yarowsky, 1995] Yarowsky D.: Unsupervised Word-Sense Disambiguation Rivaling Supervised Methods. In Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics, Cambridge, MA. 1995. 189-196.

[Yeol and Hoffman, 2003] Yeol Yoo, S. and Hoffmann, A.: A New Approach for Concept-Based Web Search. In Proceedings of the Australian Conference on Artificial Intelligence. LNAI 2903. 2003. 65–76.

[Zamir and Etzioni, 1999] Zamir, O. and Etzioni, O.: Grouper: A dynamic clustering interface to web search results. Computer Networks 31. 1999. 1361–1374.

[Zhang and Dong, 2004] Zhang, D. and Dong, Y.: Semantic, Hierarchical, Online Clustering of Web Search Results. In Proceedings of the 6th Asia Pacific Web Conference (APWEB), Hangzhou, China. 2004.