A Taxonomy Learning Method and its Application to Characterize a Scientific Web Community

Paola Velardi, Alessandro Cucchiarelli, and Michaël Pétit

Abstract— The need to extract and manage domain-specific taxonomies has become increasingly relevant in recent years. A taxonomy is a form of business intelligence, used to integrate information, reduce semantic heterogeneity, describe emergent communities and interest groups, facilitate the communication between information systems. We present a semi-automated strategy to extract domain-specific taxonomies from web documents and its application to model a Network of Excellence in the emerging research field of enterprise interoperability.

Index Terms— interoperability, knowledge publishing, web text analysis, ontology design.

1 Introduction

Taxonomies (from the Greek words *taxis*, meaning arrangement, and *onoma*, meaning name) have recently emerged from specific fields like biology, book indexing and library science into the corporate limelight. Taxonomies are considered the backbone of an organization's information architecture. Whether the focus is on document indexing or knowledge management, a domain taxonomy is the first step towards effective classification and retrieval, concept sharing, interoperability among web communities, corporate enterprises, and interest groups. Furthermore, building a taxonomy is considered the first step towards creating a *formal ontology* of a domain, as suggested by the METHONTOLOGY framework [1]. The NASA agency has recently published a white paper [2] in which a taxonomy development roadmap is outlined. Taxonomy development is seen as a two-phase process: in phase one the scope of the taxonomy is defined through interviews with stakeholders and subject matter experts, and the analysis of exemplary documents. In phase two, the taxonomy is actually developed by specialized personnel (a company which specializes in taxonomy) through a close cooperation with the stakeholder community in order to get continuous feedback. The outlined process is time consuming and costly, and can hardly be followed by non-profit organizations such as web communities and interest groups.

In this paper we present a general-purpose semi-automated methodology to speed up and facilitate the design of a domain taxonomy. We then show an application of the taxonomy for improving accessibility of knowledge and data repositories in a web community.

The methodology has several novelties in comparison with the state-of-art literature, and furthermore, it has been submitted to large-scale evaluation by the members of a web community, the INTEROP NoE[1], whose main mission is to support scientific advancements and dissemination actions in the field of enterprise and software interoperability. One of the objectives of INTEROP was to build a so-called "*Knowledge Map*" (KMap) of partner competences, to perform a periodic diagnostic of the extent of research collaboration and coordination among the NoE members (a similar application is described in [3]).

The main benefits of the KMap for its users are: i) to be able to diagnose current interoperability research within INTEROP and in Europe; ii) to receive an overview of all European research activities on interoperability and subordinated topics; iii) to receive an overview of organisations and experts as well as research results; iv) to find potential partners for collaborating in research activities.

The target groups of the INTEROP KMap system are not only project members but also the **scientific community** in the field of interoperability, including universities, research institutes, researchers, companies, etc.

These objectives and targets may be considered relevant for any scientific web community in any research field.

The INTEROP KMap is a Knowledge Management application, exploiting recent research results in the area of Semantic Web, Text Mining, and Information Retrieval. These techniques have been set up to create a *semantically indexed* information repository, storing data on active collaborations, projects, research results and organizations. The KMap interface allows users to retrieve information about partner collaborations, research results and available (or missing) competences, as well as to obtain summarized information (presented in a graphical or tabular format) on

- *P. Velardi is with the Department of Computer Science at the University of Roma "La Sapienza", Via Salaria 113, 00198 Roma, Italy. E-mail: velardi@di.uniroma1.it.*
- *A. Cucchiarelli is with the Department of Computer Science, Management and Automation (DIIGA), Polytechnic University of Marche, Via Brecce Bianche, 60131 Ancona, Italy. E-mail: a.cucchiarelli@diiga.univpm.it*
- *M. Pétit is with the Computer Science Department of the University of Namur, rue Grangagnage 21, B-5000 Namur, Belgium. E-mail: mpe@info.fundp.ac.be.*

Manuscript received (insert date of submission if desired).

[1] http://www.interop-noe.org (2003-2007), Noe-IST 508011.

the overall degree of collaboration and overlapping competence, based on a measure of *semantic similarity* between pieces of information.

In order to index and navigate the KMap, it was decided to build a taxonomy of concepts on enterprise interoperability research. The taxonomy is evolving towards the full power of an ontology during the last year of the INTEROP project. With reference to the roadmap in [2], the development of an INTEROP taxonomy appeared to be particularly complex for several reasons:

1. The project is based on the intersection of three research areas: *ontology*, *enterprise modeling* and *architecture and platforms*. A better understanding of the interrelations among the three INTEROP domains of interest and the identification of *common concepts* was one of the objectives of the NoE.
2. As a consequence of (1), not one of the over 180 researchers participating in the project could identify, by himself, the relevant terms that should appear in the taxonomy: in other terms, creating a taxonomy of enterprise interoperability was considered the first step to typify a research domain whose boundaries were vaguely specified.
3. A further problem was the absence, in the network, of lexicographic skills, to ensure professional quality of definitions.

Given the above, the use of automated techniques seemed particularly appropriate, thus providing an ideal test-bed for our methodology.

The paper is organized as follows: Section 2 presents the state-of-art on automated taxonomy learning. Section 3 presents our method in some detail. Section 4 describes the validation process and briefly presents the web applications used to support validation. Section 5 describes the application of the taxonomy to the semantic indexing of the KMap.

2 The State-of-Art in Taxonomy Learning

Taxonomy learning is a three stage process: *terminology extraction*, *glossary extraction*, and finally, *extraction of taxonomic relations between terms*. Terminology extraction is widely analyzed in literature: the majority of methods propose a combination of statistical and natural language processing techniques. Among the most recent and best performing systems, in [4] a method is presented to extract a domain terminology based on a pipeline architecture and the use of a statistical measure called *lexical cohesion*, also used in our work.

Automatic glossary extraction is instead a relatively new field, with very few contributions. In [5] the DEFINDER system is described, a text mining method to extract embedded definitions in on-line texts. The system is based on pattern matching at the lexical level, guided by cue phrases like "*is called*" "*is defined as*" etc. As shown later in this paper, the application of simple lexical patterns on unrestricted documents (e.g. the web) may produce poor results in terms of performance. For example, most definitions have the pattern "*x is a y*" but the choice of using or not using this pattern has a negative impact on precision and recall, respectively. A problem similar to glossary extraction is that of answering "*what is x?*" questions in the field of open domain question answering (QA). Many approaches presented in QA literature, especially those evaluated in TREC conferences[2], require the availability of training data, e.g. large collections of sentences tagged as "definitions" or "non-definitions". Supervised approaches like [6], [7] would be rather time consuming if applied to glossary building in emerging domains, above all due to the need of manually tagged data.

The third stage, automated learning of taxonomic relations (i.e. *kind_of* relations), is a relatively mature field, pioneered by seminal work at IBM [8] on dictionary parsing, and recently surveyed in [9]. Research on taxonomy learning can be grouped into three areas: i) methods based on the use of manually or automatically created regular expressions applied to web documents, e.g. [10], [11], [12]; ii) methods based on statistical methods and vectors of word features extracted from web documents, like [13], [9], [14], [15], [16], [17]; iii) methods based on dictionary parsing, like the already mentioned [8] and the relatively more recent ArtRank system [18].

Each of these methods has some drawbacks: we have already mentioned possible problems when using regular expressions at the lexical level, either manually or automatically learned. Statistical and machine learning methods are mostly based on the analysis and comparison of contextual features of terms, extracted from their occurrences in texts (see [17] for a comparison of different vector-based hierarchical clustering algorithms). Taxonomies obtained through this approach are very hard to evaluate by a human judge, since kind-of relations are learned on the basis of statistical measures, prone to noise and idiosyncratic data. For example, when the method is used to replicate the structure of an already existing taxonomy, the error rate (for noun classification) is over 50-60% [13].

Finally, dictionary parsing has well known disadvantages [19], like circularity of definitions, over-generality, etc.

To sum up, while a variety of methods are available in literature to address the single phases of taxonomy learning, no method addresses the complete *knowledge extraction value chain*, from lexicon to glossary and taxonomy. Another issue is the predominant use of trained methods: the availability of training sets cannot be assumed in general, and furthermore, preparing a training set requires professional annotators, like e.g. in TREC contests. Finally, performance is often measured with reference to the judgment of two/three human evaluators, usually the authors themselves, again with the exception of TREC contests, where systems are more objectively compared against the same, professionally developed, test set. In the rest of this paper we show how to overcome (at least in part) these limitations.

3 Automatic Learning of the Interoperability Taxonomy

The knowledge acquisition value chain adopted in this

[2] The Question Answering track page of TREC is http://trec.nist.gov/data/qa.html.

work is based on a sequence of automatic and manual steps. Progressively more formal data structures (lexicon, glossary, taxonomy, and currently, ontology) are first, automatically (A) acquired, and then, manually (M) validated and enhanced by the community members, using suitable web applications and collaborative work tools. The steps are the following:

1. (M) Collect a document archive *D* of domain-pertinent documents;
2. (A) Extract a domain lexicon *L* from *D*, where *L* is a list of relevant terms in the subject domain;
3. (A) For *t*∈*L*, search in *D* and on the web for the sentences that are *candidate definitions* for that term;
4. (A) Filter candidates to reduce noise (sentences that are not domain-pertinent, or non-definitions); let D'^t be the filtered set definitions ∀*t*∈ *L*, and *G* the glossary including all D'^t;
5. (M) Manually validate the obtained lexicon and glossary through a web-based collaborative application;
6. (A) Parse glosses to extract the *hypernym* (*kind of*) information;
7. (A) Use extracted hypernyms as well as other available taxonomies (e.g. on-line general-purpose linguistic ontologies) to arrange terms in a forest of taxonomically ordered sub-trees *F*;
8. (M) Use a web application to: i) define the *core* concepts *C* of the taxonomy, ii) validate *F* (step 7) and iii) enrich *C* with *F*.

The outlined procedure is domain independent, even though the examples throughout the paper are mostly chosen from the INTEROP domain. The idea behind this "*learn-and-validate*" approach is that, despite great progress in the area of taxonomy-ontology building and knowledge acquisition, automated techniques cannot fully replace the human experts and the stakeholders of a semantic web application. In our view, automated procedures are useful to achieve a significant speed-up factor in the development of semantic resources, but human validation and refinement are unavoidable when resources are to be used in real environments and applications. In the rest of this paper we will further motivate this strategy.

3.1 Collect Relevant Natural Language Communications

The first step of the procedure is to collect a large number of documents in written form, which should represent at best *what is communicated and exchanged* among the members of a community. Starting with available documents exchanged among or published by the members of a web community, the task is to incrementally identify a larger set of related web pages (or web accessible documents) including not only those of the community members, but also those of authoritative sources on the subject topics. This is a partly manual, partly automated step, and its complexity and involved effort greatly depends on the community under consideration. For the sake of space, and because this process is essentially manual, we will not go into further detail.

3.2 Extraction of a Domain Lexicon

A *domain lexicon L* is a list of terms *t* (single or multi-word expressions) commonly used within a given community of interest. The purpose of this phase is to automatically extract *L* from the documentation collected in phase 1. Terminological *candidates* are multi-word strings with a precise syntactic structure (e.g.: compounds, adjective+compound, etc) and certain distributional properties across the domain documents. Examples in various fields are the following: in enterprise interoperability: *enterprise intra organizational integration*, in tourism: *gourmet restaurant*, in computer networks: *packet switching protocol*, in art techniques: *chiaro scuro*. Statistical and syntactic processing tools are used for automatic extraction of terms (details and performance evaluation of this phase have already been published in [19] with reference to INTEROP and many other domains). The main novelty of our terminology learning method (e.g. wrt [4]) is the definition of an entropy-based measure called *domain consensus*: only terms with an even probability distribution across the documents of the domain are selected, simulating the *consensus* that an emerging concept must gain in a community before being adopted.

3.3 Extraction of Definitions

Once an initial lexicon has been extracted, the subsequent phase is to obtain a list of (one or more) definitions for each term. Again, the number and quality of the documents available for this task may vary depending upon the domain. In more "stable" domains, such as art, tourism and computer networks, glossaries are often available on the Internet. In emergent domains, such as enterprise interoperability, and for new terms gaining popularity in well established domains, the definitions must be extracted from community-provided and web documents.

Extraction of definitions relies on an incremental filtering process. Definitions are initially looked up in existing web glossaries. If they are not found, simple patterns at the lexical level (e.g. *t is a Y*, *t is defined as Y*, etc.) are used to extensively search for an initial set of candidate definitions among web documents. On the set of candidates, a first statistical filtering is applied to verify *domain pertinence*. A subsequent *stylistic* filtering is applied, based on fine-grained regular expressions at the lexical, part of speech and syntactic level. The objective is to select "*well-formed*" definitions, i.e. definitions expressed in term of *genus* (the *kind* a concept belongs to) and *differentia* (what specializes the concept wrt its kind).

Stylistic filtering is a novel criterion wrt related literature on definition extraction, and has several advantages: i) to prefer definitions adhering to a uniform style, commonly adopted by professional lexicographers; ii) to distinguish definitions from non-definitions (especially when candidate definitions are extracted from free texts, rather than glossaries); iii) to be able to extract from definitions a *kind-of* information, used to arrange terms taxonomically.

For each term t in the extracted lexicon we first search for a definition, in on-line glossaries and then, we extract definition sentences from web documents. We have defined the following algorithm to extract definitions:

1. From the set of word *components* forming the ex-

tracted lexicon L of a domain D, learn a *probabilistic model of the domain*, i.e. assign a probability of occurrence to each word component. More precisely, let L be the lexicon of extracted terms, LT the set of singleton word components appearing in L, and let:

$$(E(P(w)) = \frac{freq(w)}{\sum\limits_{w_i \in LT} freq(w_i)}$$

be the estimated probability of w in D, where $w \in LT$ and the frequencies are computed in L. For example, if L=[*distributed system integration, integration method*] then LT=[*distributed, system, integration, method*] and *E(P(integration))*=2/5

2. For each term t in L, do the following:
   a. Search definitions for t in on-line glossaries[3]. Let D_g^t be the set of extracted candidate definitions;
   b. If $D_g^t \neq \varnothing$ then apply a *domain relevance* and a *style* filter, as described later. Let D'^t_g be the filtered set.
   c. If either D_g^t or D'^t_g are empty, then extract a set of sentences including t from the community-provided documents first, and after from the web. A first rough filtering is applied to the usually large set of extracted documents, searching for simple patterns like "*t is*" "*t defines*" "*t refers*" etc. Let D_{web}^t be the set of candidates after this step.
   d. On D_{web}^t apply a *domain relevance* and a *style filter*, as described later. Let D'^t_{web} be the filtered set.
3. Let:

$$G = \bigcup_{t \in L} D'^t_s, \ \ s \in \{g, web\}$$

be the extracted domain glossary.

We now describe the domain pertinence and style filters. The domain pertinence is used to filter out from D_g^t or D_{web}^t candidates which appear to be not relevant for a given domain (e.g. enterprise interoperability). For example, one of the retrieved glossary definitions for the word model is: "*A model is a person who acts as a human prop for purposes of art, fashion, advertising, etc.*" while a domain-appropriate definition is "*A representation of a set of components of a process, system, or subject area, generally developed for understanding, analysis, improvement, and/or replacement of the process*".

Let *def(t)* be a candidate definition for t, W_t be the set of content words in *def(t)* and $W_t^{'} \subseteq W_t$ be the subset of words in *def(t)* belonging to LT (see step 1 above). The domain pertinence of *def(t)* is defined as:

$$weight(def(t)) = \sum_{w \in W'_t} E(P(w)) \log(N_t / n_t^w) + \alpha \sum_{w \in LT} E(P(w))$$

where N_t is the number of candidate definitions extracted for the term t (either the set D_g^t or D_{web}^t), and n_t is the number of such candidates including the word w. The log factor, called *inverse document frequency* in information re trieval literature, reduces the weight of words that have a very high probability of occurrence in any candidate sentence, regardless of the domain (e.g. words like "*system*"). For these very common words, the log value is close to zero. The additional sum in this formula assigns a higher weight to those sentences including some of the components of the term t to be defined (this heuristic is supported by the analysis of professional on-line glossaries), e.g. "*Schema integration* is [the process by which *schemata* from heterogeneous databases are conceptually *integrated* into a single cohesive *schema*]".

The second applied filter is a stylistic filter, called well-formedness.

Usually, well-formed definitions are provided in terms of genus (the kind, or hypernym, to which an entity belongs) and differentia (what differentiates the entity wrt the more general class), e.g. "*enterprise information integration is* ***the process*** *of integrating structured data from any relevant source for the purpose of presenting an intelligent, real-time view of the business to a business analyst or an operational application.*" In this definition, the noun phrase that identifies the genus is marked in bold. Not all definitions are well-formed in the above mentioned sense, e.g. "*component integration is obtained by composing the component's refinement structures together, resulting in (larger) refinement structures which can be further used as components*", and many sentences being not well-formed are non-definitions, e.g. "*component integration has been recently proposed to provide a solution for those issues*".

To verify well-formedness, we write regular expressions [21] that impose constraints on a sentence structure at the *lexical, part of speech* and *syntactic level*. Part of speech and syntactic elements are identified using an available parser, the TreeTagger[4]. The well-formedness criterion is verified by increasingly refined expressions. First, *the main noun phrase* (NP) of the definition is identified, e.g: r="^(PP)?(NP)+". This regular expression prescribes a sentence structure with syntactic constraints. It reads: "a definition is formed by a facultative prepositional phrase (^(PP)?) followed by the main noun phrase (NP), followed by anything else (+)". An example of sentence matching r is: "*domain model: [in the traditional software engineering perspective]$_{PP}$, [a precise representation]$_{NP}$ of specification and implementation concepts that define a class of existing systems*". Additional regular expressions are then applied on the main NP, e.g:

p_1="^(Refers|Referring)\\sto\\s(((a|the)\\s)?(type|kind)\\sof\\s)? (.*)"

The regular expression p_1 applies only lexical constraints (much in the same way as [4] and related work on extracting relations from corpora, like [10], [11], [14]) and detects the presence of "cue" words like *is a, refers, is a type of, is the process of,* etc. in the main NP of the definition. Note that the potentially noisy "*is a*" pattern is included, since the pres-

[3] We use the Google's "*define*:" feature.

[4] TreeTagger is available at http://www.ims.unistuttgarrt.de/projekte/corplex/TreeTagger/Decision/TreeTagger.html.

ence of the other filters significantly reduce the noise. Finally, additional regular expressions are used to detect the presence of the kind_of (hypernym) information in the main NP.

For example, consider the regular expression:

r_1="^(A|D)?((V|C|,|J|N|R)*)(N)".

r_1 imposes part of speech constraints on a sentence fragment. Symbols in r_1 are part of speech tags (POS), e.g. article (A), verb (V), adjective (J), noun (N) etc. The previous definition of *domain model* matches both r and r_1. When parsing this sentence with the TreeTagger we obtain:

Syntactic Analysis: (PP **NP** PP CNP RVP NP PP)

POS: (PAJNNN AJ**N** PNCNNWVANPJN)

The bold POS (N) represents the fragment selected as the hypernym. The application of r_1 returns: hypernym: *representation*

Or-conjoined hypernyms are also handled: "The systematic format and technical structure that..." is a sentence that returns two hypernyms, *format* and *structure*. Definitions for which none of the "hypernymy-seeker" regular expressions apply, are rejected.

We underline that all the regular expressions used are general purpose, i.e. not tailored for a specific domain. To write these expressions, we have analyzed several professional glossaries on the web. Domain dependence is instead verified through the automatically learned probabilistic model (step 1).

Candidates in D_g^t or D_{web}^t, purged from sentences not adhering to the style criterion, are ordered according to their domain pertinence weight. The first k candidates are selected, according to a term-dependent automatically computed threshold: $weight(def(t)) \geq \vartheta_t$. In short, the computation is based on ordering the $weight(def(t))$ values for each *t*, and computing the average difference between consecutive values.

The performance of the definition extraction algorithm described in this section is analyzed in Section 4, however we remark that definitions of new terms in an emerging community are not found in glossaries, simply because of their novelty. In the INTEROP experiment only 20% of the terms in *L* were found in on-line glossaries. It is often the case that the inventors of new terms, or their initial users, provide a definition in their communications to the reference community. For example, the term "*federated ontology*" appeared in scientific literature only in 2001 [22], but the first explicit definition is in a paper[5] dated 2004, that rephrases the proposed concept of federated ontology in a less explicit: "*Federated ontologies are distributed, connected ontologies, somewhat analogous to federated databases*".

Even though identifying definitions in texts is far more complicated than separating domain-pertinent and non-pertinent definitions in glossaries, we have applied more or less the same algorithm to filter candidate definitions extracted either from glossaries or from the web. The major difference is the *impact* that the various steps have on the performance of gloss extraction. For example, the performance of glossary filtering (the set D_g^t) is mostly determined by the *domain pertinence* computation, while the performance of gloss extraction from free text (the set D_{web}^t) is considerably improved by the stylistic filter.

3.4 Creating a Domain Taxonomy

The application of the well formedness criterion, discussed in the previous section, allows it to extract the kind of information from definitions, as defined by the author of a definition. This information may help the structuring of the terms in *L* in taxonomic order. However, ordering terms according to the hypernyms extracted from definitions has well-known drawbacks [18]. Typical problems found when attempting to extract (manually or automatically) hypernymy relations from natural language definitions are: over-generality of the provided hypernym (e.g. "*Constraint checking is one of many techniques…*"), unclear choices for more general terms, or-conjoined hypernyms (e.g. "*Non-functional aspects define the overall qualities or attributes of a system*"), absence of hypernym[6] (e.g. "*Ontological analysis is accomplished by examining the vocabulary that…*"), circularity of definitions, etc. These problems are more or less evident – especially over-generality – when analyzing the term trees forest generated on the basis of glossary parsing. To reduce these problems, we have defined the following algorithm:

1. First, terms in the lexicon *L* are arranged taxonomically according to simple *string inclusion*. String inclusion is a very reliable indicator of a taxonomic relation, given the *compositional* nature of meaning in most nominal multi-word terms[7]. This step produces a forest of sub-trees. Let ST_i be one of such trees, shown in Table 1a.
2. Then, we use our word sense disambiguation algorithm SSI[8] [23] to detect hypernymy relations, according to the WordNet[9] taxonomy, among term components at the same level of generality. WordNet is used because of its high coverage and because many singleton terms in the domain (e.g. *knowledge, data, integration*) have at least one correspondent sense in WordNet. For example, when fed with the sequence of words: *representation, model, schema, ontology, knowledge, data, information*, SSI produces a graph showing semantic relations between disambiguated senses, as in Fig. 1. By selecting only hypernymy relations from the graph, the taxonomic structure shown in Table 1b is obtained.
3. Hypernymy information extracted from definitions is then used to capture additional relations. We know that: $ontology \xrightarrow{kind\_of} specification$, $application \xrightarrow{kind\_of} program$, $service \xrightarrow{kind\_of} program$. If we again apply SSI on: *knowledge, specification, program, model* etc., we further learn the path: $data \xrightarrow{kind\_of} program \xrightarrow{kind\_of} specification$, and thus we learn the structure shown in Table 1c.

[5] Http://www.meteck.org/AspectsOntologyIntegration.pdf.

[6] This type of error is avoided since we use the well-formedness criterion.

[7] On multi-word expressions, see http://mwe.stanford.edu/index.html.

[8] SSI is on-line at http://lcl.di.uniroma1.it.

[9] Http://www.wordnet.princeton. edu.

TABLE 1
TREES GENERATED BY THE HYPERNYMY RELATION EXTRACTION

Step 1 (a)	Step 2 (b)	Step 3 (c)
integration	knowledge integration	knowledge integration
representation integration	representation integration	representation integration
model integration	schema integration	schema integration
enterprise model integration	model integration	model integration
schema integration	enterprise model integration	enterprise model integration
ontology integration	information integration	information integration
knowledge integration	data integration	data integration
data integration	ontology integration	program integration
information integration	application integration	application integration
service integration	service integration	service integration
		specification integration
		ontology integration

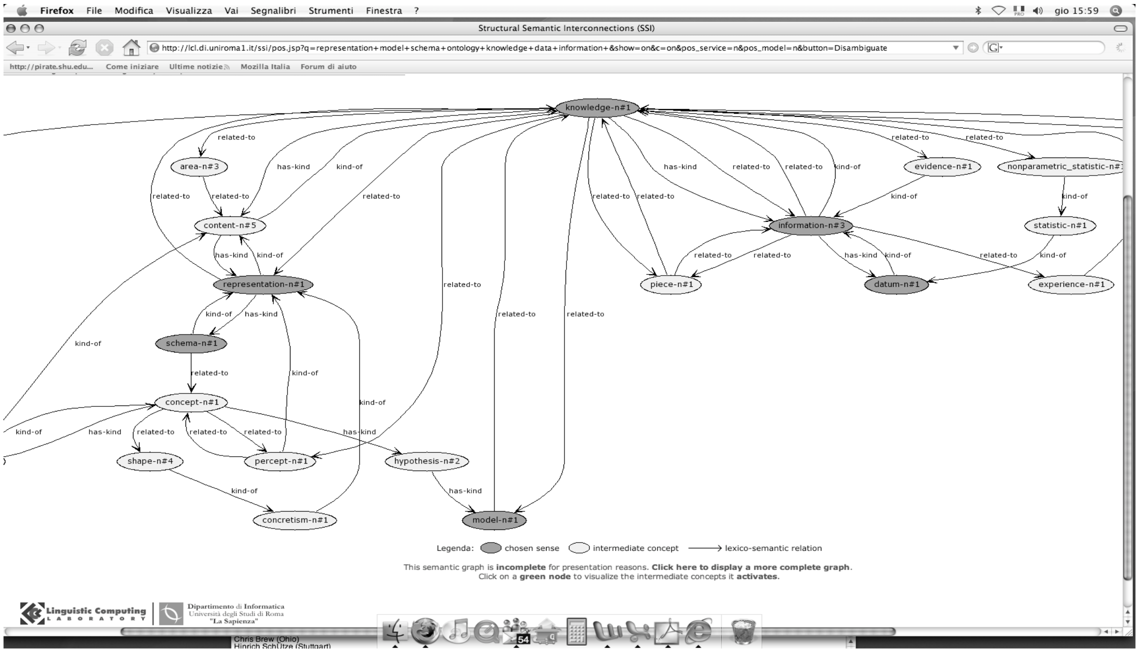
In general, some errors are introduced by this automated process, caused both by disambiguation errors and by the absence, in WordNet, of domain-specific senses. The performance of SSI under variable conditions is discussed in detail in [23]. Notice however that SSI chooses a sense only when enough evidence is provided. If not disambiguated, a node remains at the same level as before the application of steps 2 and 3.

The algorithm outlined above produces a forest *F* of subtrees ST_i. In the INTEROP domain, we obtained a forest of 621 trees of rather variable dimension and depth. String inclusion is the main mechanism used to identify *kind-of* relations, however, several additional taxonomic relations are identified by the algorithm described in this section. Overall, 243 (24%) of the total automatically detected taxonomic relations in $\mathrm{ST_i}$ have been obtained through the application of steps 2-3 above. An advantage of this method wrt hierarchical clustering methods (surveyed in [17]) is that the principles used to create the term ordering (string inclusion, WordNet hypernymy relations selected by SSI) are clear, and easy to evaluate by a human expert. Furthermore, the ordering criteria are applied consistently over the taxonomy, rather than depending upon the specific contexts in which a term appears in texts.

The automatic method described above produces a forest *F* of unconnected trees. To obtain a unique tree, a *core* taxonomy (a taxonomically ordered set of “basic” domain concepts) was created. In INTEROP, the root nodes of the sub-trees were linked to a core taxonomy, manually created by a restricted team of partners. The choice of topmost nodes was inspired by the Enterprise Ontology (EO) [24], while intermediate nodes are mostly WordNet concepts, used to fill the gap between the core EO concepts and the 621 roots. Justifying the principles followed for creating a good set of core concepts is outside the scope of this paper. Any interested readers may access the full taxonomy from: http://lcl.di.uniroma1/tav. See also [25] for a discussion on the re-use of foundational ontologies in specific domains.

4 EVALUATION

As commented in Section 2, one of the drawbacks of most taxonomy learning algorithms is subjective evaluation by a limited number of judges. During the INTEROP project it was instead possible, and even explicitly required, to


Firefox File Modifica Visualizza Vai Segnalibri Strumenti Finestra ?
gio 15:59
Structural Semantic Interconnections (SSI)
http://pirate.shu.edu... Come iniziare Ultime notizie Mozilla Italia Forum di aiuto
knowledge-n#1
area-n#3
content-n#5
representation-n#1
schema-n#1
concept-n#1
shape-n#4
percept-n#1
hypothesis-n#2
concretism-n#1
model-n#1
evidence-n#1
nonparametric_statistic-n#1
statistic-n#1
information-n#3
piece-n#1
datum-n#1
experience-n#1
related-to
has-kind
kind-of
Legenda: chosen sense intermediate concept lexico-semantic relation
Linguistic Computing LABORATORY
Dipartimento di Informatica Università degli Studi di Roma "La Sapienza"
Chris Brew (Ohio)
Hinrich Schütze (Stuttgart)

http://lcl.di.uniroma1.it/ssi/pos.jsp?q=representation+model+schema+ontology+knowledge+data+information+&show=on&c=on&pos_service=n&pos_model=n&button=Disambiguate

This semantic graph is **incomplete** for presentation reasons. **Click here to display a more complete graph.**
Click on a **green node** to visualize the intermediate concepts it **activates.**

Fig. 1. Detection of taxonomic relations with the SSI WSD algorithm

TABLE 2.
ERROR RATE IN GLOSSARY EXTRACTION

	Statistical filtering+ syntactic regexps	+hypernymy-seeker regexps
Total number of terms in L searched on web	1660	1660
Total number of extracted sentences (S) matching simple lexical patterns	31061	31061
Number of definitions over the threshold ϑ_t	6136	7926
Number of definitions selected by evaluators (from the 31061)	1566	1566
True Positive (TP)	902	1058
True negative (TN)	27592	22571
Accuracy (TP+TN)/S	81.01%	76.21%
Recall on positive TP/P	57.60%	67.60%
Final number of terms with at least one definition (T1)	865	865
Final number of terms over the threshold with at least one definition (T2)	607	692
Coverage T2/T1	70.01%	80.00%

submit the acquired lexico-semantic resources (lexicon, glossary and taxonomy) to the validation of the entire community. This section describes the validation process and results.

4.1 Lexicon and Glossary Validation

The objective of the evaluation by the NoE members was i) to reach an agreement on the set of terms to be included in the final glossary and ii) to perform a fine grained evaluation of the definitions quality. To avoid bothering partners with blatantly wrong data (e.g. non-definitions), a preliminary analysis of the glossary learning procedure was conducted by three project partners, with the aim of pruning coarse errors (non-definitions or clearly non-pertinent definitions). A second objective of this analysis was to measure the *accuracy* of the glossary extraction methodology described in 3.3. Table 2 summarizes the results, for the case of glossary extraction from texts[10].

What initially comes out is that the mere application of lexical patterns produces a considerable amount of noise (31061 candidates), in particular the "*x is a y*" pattern. However, subsequent filtering (using domain pertinence and well-formedness) is particularly effective in eliminating about 80% of the initial candidates (only 6-7000 candidates survive). The evaluators, however, reviewed all the extracted candidates to compute precision and recall of the method. Notice that no fine-grained evaluation of glosses was performed here, but only error pruning: the objective was to tune at best the threshold parameters and the regular expressions, with an emphasis on recall and coverage. Indeed, as we have already noticed, emerging and yet relevant concepts might have few, even only one definition, and what matters is to capture the majority of them.

A fine-grained evaluation of the glossary and lexicon was performed on the INTEROP platform, using a glossary validation interface developed in Zope-Plone[11]. Lexicon extraction was initially performed on a collection of documents belonging to at least one of the three research domains of INTEROP, and only through a collective and wide-scale judgment was it possible to decide whether or not a given concept contributes to the understanding of enterprise interoperability.

The evaluation was conducted in two phases: 1) Lexicon pruning: Therefore, the partners were requested to inspect all terms and simply reject those terms they believed irrelevant. Only rejections were allowed. 2) Glossary evaluation-extension: in this phase, partners were solicited to express a graded vote for each definition, ranging from -3 to +1. A -1 vote is given to "not fully convincing" definitions, -3 are unacceptable definitions. Partners were also encouraged to add a new definition, if they felt that none of the available definitions were adequate, or if no definition was available for a term.

Table 3 summarizes the results of the voting task. Over 4500 votes were expressed by 45 partners from 24 institutions. The partners seemed to be somehow reluctant to suggest new definitions for terms without any definition, but they were more willing to evaluate available definitions, thus confirming the fact that adopting an automated glossary learning procedure was indeed a good strategy. After voting, definitions cumulating a vote <-1 were deleted, and partners were requested to manually modify a subset of 260 "-1" definitions, belonging to terms with no non-negative definitions available. The final result is a large glossary with 963 (703+260) definitions for 804 terms, of which, 595 fully automatically learned, 108 manually inserted, and 260 automatically learned and manually re fined.

4.2 Taxonomy Validation

The validation of the taxonomy was conducted by 17 selected partners with some experience in knowledge management and ontology. A dedicated web interface[12] allowed partners to move terms and sub-trees, create new core nodes and browse the taxonomy. A policy was established among partners, to avoid inconsistent changes and support collaborative decision-making, based on the use of polls. Before making any change, partners were requested to i) clearly illustrate the proposed action as a binary or multiple choice (e.g.: "*do you agree to move concept x under concept y...?*") ii) activate a poll and iii) execute the action which receives the majority of votes. Changes on the taxonomy were recorded by the interface, and visible to subsequent evaluators, to discourage

[10] Glossary extraction from on-line glossaries is considerably more reliable. The interesting case for accuracy evaluation is glossary extraction from texts.

[11] http://interop-noe.org/backoffice/workspaces/wpg/ps_interop/.

[12] Available on: http://lcl.di.uniroma1.it/tav (where readers can also access the final INTEROP taxonomy).

TABLE 3
SUMMARY OF LEXICON AND GLOSSARY VOTING TASK

Lexicon validation	
n. of terms in L	1902
Total expressed votes in phase 1 (on 1903 terms)	2453
Total different terms in L with a negative vote	783 (41%)
Survived terms	1120 (59%)
Of which with at least one definition	868 (77.5%)
Glossary validation	
Total n. of available definitions (covering 868 terms)	1450
Total expressed votes in phase 2	2164
Total new definitions added	108
Total definitions (including new definitions)	1558
Definitions with a negative vote (including new defs.)	855
Definitions with a non-negative vote (incl. new defs.)	703
Terms with at least one definition after deleting all definitions with vote < -1	804 (72%)
Terms with a definition -1 that had to be manually enhanced	260

TABLE 4
RESULTS OF COLLABORATIVE TAXONOMY VALIDATION

Number of partners who voted the taxonomy		11
Total number of activated polls		21
Total number of performed actions		133
Of which:	*Movement of single terms or term sub-trees*	*25*
	Deleted core nodes	*3*
	Created core nodes	*6*
New or modified definitions (for core concepts)		99

multiple changes on the same concepts. The validation was concluded in three weeks, and globally, 133 changes (movement of single nodes or sub-trees) have been recorded.

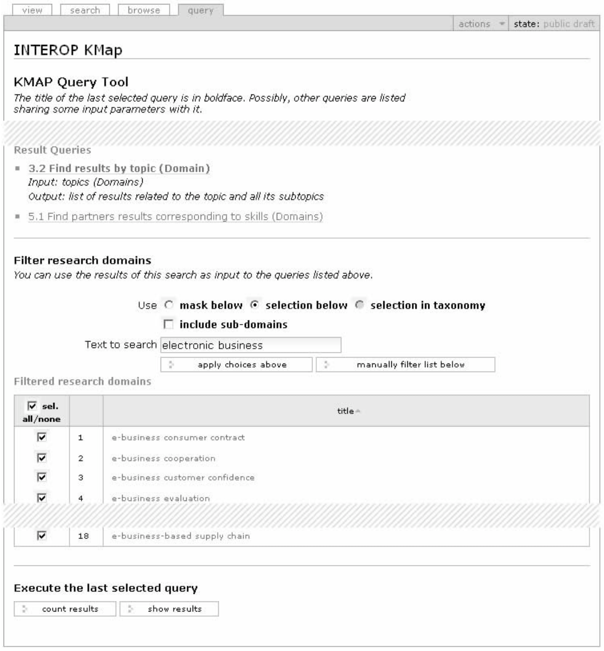
The agreement was overall quite high, since the taxonomy includes 1337 "kind_of" relations, of which only about 10% (133) have been modified by partners. A comparison between the number of actions performed by partners in Table 3 and Table 4 suggests that domain specialists can easily perform certain tasks (i.e. lexicon pruning) but are less confident when asked to contribute to the creation of progressively more "abstract" representations of their domain of expertise (from glossary to taxonomy and, eventually, to ontology). This seems to further support the use of automated techniques, followed by a validation phase.

5 USING THE TAXONOMY TO INDEX A COMPETENCE MAP

The taxonomy created through the procedure illustrated so far has been used to semantically index the INTEROP KMap. Fig. 2 shows the screen-dump of a possible query type *("find all the results – papers and projects – dealing with a subset of concepts in the taxonomy*"). The user can select concepts (referred to as *knowledge domains*, or simply *domains*, in the query interface) by "string search" in the taxonomy (as in the example of Fig. 2), they can arrange concepts in boolean expressions, and perform query expansion (including in the query all or some of the concept's hyponyms). "Global" information can also be obtained, e.g. a map of a member's competence *similarity*, or an analysis of research results similarity. Being able to express the competences of each INTEROP partner as a set of concepts indicating its fields of research, the similarity between partners' competences can be obtained through the computation of a cumulative semantic similarity function between sets.

In literature, a variety of methods for measuring the similarity has been proposed, and they are all based on concepts semantic. The main approaches are founded on concepts distance within an ontological structure [26], [27], on concept information content [28], on concept feature matching [29] and on a combination of the above ones [30], [31]. Recently, with the diffusion of fuzzy ontologies for concepts representation in the information retrieval field, more specific measures of similarity have been defined [32]. These approaches can roughly be divided into two categories. The first includes the methods which define the similarity as the length of the path between two concepts in the ontology [26], [27]. The second is composed by the other cited methods, which refine the concept distance also taking into account the weights of the links which compose the path, as well as the variability of 'conceptual distance' covered by a single link inside the ontology. The methods belonging to this second category require a representation of concepts richer than the one developed at the current stage of taxonomy implementation. Moreover, the learning process described in section 3 reduces the variability of conceptual distance associated to each taxonomy link (especially for mid/low abstraction level concepts, the most used by INTEROP partners to describe their domains of interest). Stemming from these considerations, we have defined a similarity function based on the distance of concepts in line with [26] and [27].

Finally, in order to display the results of the similarity measure, we have built a graphic tool showing the obtained similarity network as a graph structure.



INTEROP KMap

KMAP Query Tool

The title of the last selected query is in boldface. Possibly, other queries are listed sharing some input parameters with it.

Result Queries

- **3.2 Find results by topic (Domain)**
  Input: topics (Domains)
  Output: list of results related to the topic and all its subtopics
- 5.1 Find partners results corresponding to skills (Domains)

Filter research domains

You can use the results of this search as input to the queries listed above.

Use ○ **mask below** ● **selection below** ○ **selection in taxonomy**

□ **include sub-domains**

Text to search: electronic business

Filtered research domains

☑ sel. all/none		title
☑	1	e-business consumer contract
☑	2	e-business cooperation
☑	3	e-business customer confidence
☑	4	e-business evaluation
☑	18	e-business-based supply chain

Execute the last selected query

Fig. 2. Taxonomy-based search of INTEROP research results

5.1 Experimental Set-Up

The following information has been extracted for each partner P from the KMap. a) Domains of interest: partners feeding the KMap were requested to choose one or more terms from a subset *S* of the INTEROP taxonomy *T*. b) Partner-defined domains of interest: each partner was allowed to freely enter his additional terms in the form of a list. c) Description of partner defined domains of interest: partners were requested to enter a free text description of their interests d) Title and description of each publication cited by the partner e) Name (not considering the acronyms) and description of each project in which the partner participates, f) Short description of each software product used by or pro duced by a partner.

All information related to points *c-f*, collected in the KMap as English sentences, was parsed to extract additional terms $t \in T$ representing partners' competences, not included in *a* and *b*. At the end of this phase, the following data was available: 1) a set of *terms* $C \subset T$ representing the partners' domains of interest (as remarked, these are a subset of WPG taxonomy T), resulting from the merge of terms entered in point (a) and (b) of the previous list and terms extracted by the linguistic processor; 2) a set of *generalized concepts* $C' \subset T$, i.e. the elements of the kind-of taxonomy *T* reachable through a <u>single generalization step</u> starting from terms in set *C*. Topmost concepts are filtered out to avoid over generality. Table 5 summarizes the data.

For each partner P_i a binary vector V_i is defined, with $dim(V_i)=|C|+|C'|$. This vector can be seen as composed by two sub-vectors: V_{iT} ($dim(V_{iT})=|C|$) representing terms indicated by the partner in an explicit way or extracted by indirect information, and V_{iC} ($dim(V_{iC})=|C'|$) representing the generalized concepts related to them. Considering *C* a lexicographically ordered set of terms $\{c_1,c_2,\ldots,c_n\}$, the element *k* of V_{iT} (V_{iT}^{K}) is set to 1 if the term c_k in *C* is a domain of interest of P_i, to 0 otherwise. V_{iC} is defined in a similar way, with respect to the ordered set *C′* and the generalized concepts associated to P_i.

The similarity measure between each pair of partners *A* and *B*, $Sim(P_A,P_B)$, has been defined by considering:

a) *Direct matches:* the matches between elements in V_{AT} and V_{BT}

b) *Indirect matches, first type*: the matches between terms represented in V_{AT} not considered in (a) and generalized concepts represented in V_{BC}, and vice versa (i.e. the matches between two elements, one in V_{AT} and the other in V_{BT}, which represent two terms re lated by a *kind_of* link in *T*, e.g.: *application integration* $\xleftarrow{kind\_of}$ *service integration*)

c) *Indirect matches, second type*: the matches between generalized concepts related to terms represented in V_{AT} and V_{BT} not considered in (a) and (b) (i.e. the matches between elements in V_{AC} and V_{BC})

For each case, the contribution to $Sim(P_A,P_B)$ has been defined by using the *cosine similarity* measure (well known in the information retrieval field, see [33]), applied to the pairs of vectors used in (a), (b) and (c). Given two numeric vectors A and B with the same number of elements, the cosine similarity between them is defined as:

$$\cos(A,B)=\frac{\sum_i a_i b_i}{\sqrt{\sum_i a_i^2}\sqrt{\sum_i b_i^2}}$$

where a_i (b_i) is the i-th element of *A* (*B*).

To satisfy the condition *dim(A)=dim(B)* needed by the cosine similarity measure, the implemented algorithm maps each vector used in (a), (b) and (c) into a binary vector *V′*, being $dim(V')=|C \cup C'|$, with element *k* related to the *k-th* term of the ordered set $C \cup C'$.

By calling *SIM1* the result of this measure for case (a) (i.e. $SIM1=cos(V_{AT},V_{BT})$), *SIM2* and *SIM3* for case (b) and *SIM4* for case (c), we define the measure of $Sim(P_A,P_B)$ as:

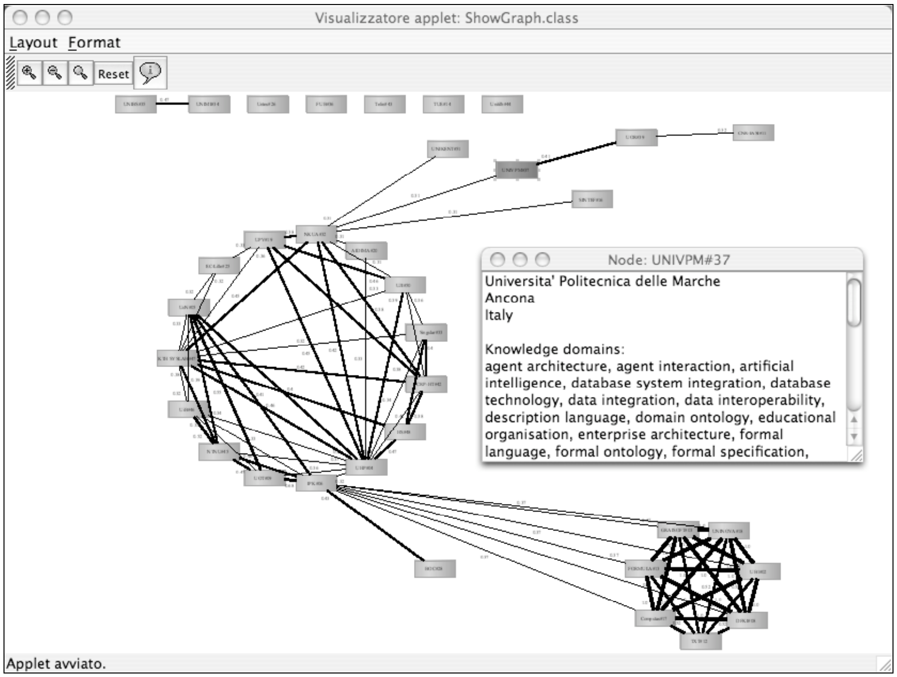
$$Sim(P_A,P_B)=\alpha SIM1+\beta\left(\frac{SIM2+SIM3}{2}\right)+\chi SIM4$$

where: $\alpha > \beta + \chi$

In our experiments, the parameters α, β and χ have been experimentally tuned to 1.0, 0.2 and 0.2 respectively.

TABLE 5.
SUBSET OF THE INTEROPERABILITY LEXICON EXTRACTED FROM KMAP DATA

Terms				**Generalized Concepts**	**Grand Total**
Selected from Domains of Interest	*User Added*	*Extracted from textual data*	*Total*		
315	23	99	437	80	517



Universita' Politecnica delle Marche
Ancona
Italy

Knowledge domains:
agent architecture, agent interaction, artificial intelligence, database system integration, database technology, data integration, data interoperability, description language, domain ontology, educational organisation, enterprise architecture, formal language, formal ontology, formal specification,

Fig. 3. Competence Map of INTEROP partners

5.2 Visualization of Partner Competences

As the result of the similarity computation between each pair of INTEROP partners, a graph data structure was obtained, with nodes representing partners, and the generic edge between nodes P_A and P_B expressing the competence similarities $Sim(P_A,P_B)$ between each pair of partners. A graphic interface has been developed to visualize this information in a comprehensible way, and it is based on yFiles (http://www.yWorks.com.), a commercial Java class library. Basic information, as partner code identification and similarity value between partners competences are shown as node and edge labels respectively. The interface uses the thickness of edges to reflect the value of the semantic similarity $Sim(P_A,P_B)$: the thicker the line, the higher the similarity (see Fig. 3).

The user can select the visualization layout to apply for graph display within a set of options provided by the Layout menu. The circular layout has been especially useful in detecting clusters of partners sharing the same domains of competence. This is clearly shown in Fig. 3, where the selection of such a layout results in a node arrangement of two circles, each formed by partners with high domain competences similarity. The figure also displays the information related to a node, in terms of partner's name and domains of competence.

This interface provides some initial diagnostics of the INTEROP K-Map that will be extended in the near future. For example, it is possible to identify partners with similar competences, and use this information to organize at best the cooperation between members. It is also possible to identify the concepts that are most "popular", i.e. those appearing on the highest number of edges, and similarly, those for which there is limited competence in the network. Finally, several concepts have no coverage in the INTEROP community, at least from what appears in the K-Map data. Table 6 provides an excerpt of the most covered competences and the least covered.

The graphic interface is also able to show detailed information related to partners and shared domain competencies, simply by clicking on a node or on an edge. Both actions display a window showing either the partner's full name, address and competences, or direct and indirect (i.e. based on taxonomic relations) matches between terms.

TABLE 6
AN EXCERPT OF COMPETENCE DOMAINS AND THEIR COVERAGE

Highly Covered Competences		**Poorly Covered Competences**	
Competence	*Number of Partners*	*Competence*	*Number of Partners*
interoperability	31	text mining	1
enterprise model	25	trust software component	1
ontology	15	usability evaluation	1
process model	14	virtual enterprise	1
data integration	13	virtual reality	1
domain ontology	13	visual language	1
conceptual model	12	web mining	1
web service technology	12	wireless system	1

TABLE 7
SUMMARY OF PARTNERS' SIMILARITY LINKS

Number of Interconnections	34	33	30	28	27	20
Number of Partners	19	6	6	1	2	1

(a)

Similarity Threshold ϑ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Number of Partners	35	35	31	21	14	8	6	6	6	6

(b)

Finally, Table 7 provides a summary of similarity links among partners. Table 7a clusters partners by the number of interconnections, showing that 19 partners over 35 share their domain competences with all the others, and that the lowest number interconnections is 20. Table 7b lists the number of partners who have at least one link with a similarity value equal to or greater than a threshold ϑ.

Overall, 5545 matches have been identified, of which 3354 (65%) direct matches and 1791 (35%) indirect matches (first and second type), thus proving the relevance of the last ones.

6 CONCLUSION

This paper illustrated (in a forcefully sketchy way) a complete application of Semantic Web techniques [34] to the task of modeling the competences of a web-based research community, INTEROP. We are not aware of any example of fully implemented knowledge acquisition value chain, where the acquired knowledge is first, extensively validated through the cooperative effort of an entire web community, and then, put into operation to improve accessibility of web resources. The adopted techniques are fully general[13] and the tools and interfaces developed within INTEROP can be applied to any other domain.

The proposed approach allows us to speed-up the development of the taxonomy and requires less effort than in traditional manual approaches. With the help of figures provided by an experienced lexicographer, who developed several glossaries for publishing companies, we estimated a speed-up factor of more than 50%[14]. In comparison with machine learning taxonomy building techniques [17], the main advantage of our method is that the principles that justify the resulting term ordering are *clear, consistently applied, easier to evaluate and modify*. Furthermore, its application to a real web community, the INTEROP NoE, has allowed us to validate in the large the entire process, and to develop adequate software support.

The obtained taxonomy was used in the INTEROP pro ject to index a partner's competence map, the KMap, and will be used for various other purposes such as document indexing and semantic searching.

ACKNOWLEDGMENT

This work has been supported by the INTEROP Network of Excellence under contract n. IST-2003- 508011. Many thanks to Giovanni Toffoli, Roberto Navigli, Pierluigi D'Amadio and Francesca Neri for software developments.

REFERENCES

[1] M. Fernández, A. Gómez-Pérez, N. Juristo, "METHONTOLOGY: From Ontological Art Towards Ontological Engineering" Spring Symposium Series. Stanford. pp. 33-40, 1997.

[2] J. Dutra and J. Busch, "Enabling Knowledge Discovery Taxonomy Development for NASA" NASA technical whitepaper webserices.gov/NASA%20Taxonomy%20White%20Paper_final_rev.doc, 2003.

[3] T. Hädrich, T. Priebe T, "A Context-Based Approach for Supporting Knowledge Work with Semantic Portals", in International Journal of Semantic Web and Information Systems (IJSWIS), Vol. 1, Issue 3, pp: 64-88, 2005

[4] Y. Park, R. J. Byrd, and B. K. Boguraev, "Automatic Glossary Extraction: Beyond Terminology Identification" Proceedings of the Nineteenth International Conference on Computational Linguistics, pp. 772–778, 2002.

[5] J. Klavans and S. Muresan, "Text Mining Techniques for fully automatic Glossary Construction" Proceedings of the HTL2001 Conference, San Diego (CA), March, 2001.

[6] S. Miliaraki and I. Androutsopoulos, "Learning to identify single-snippet answers to definition questions" In Proceedings of COLING-2004, pages 1360–1366, 2004.

[7] H.T. Ng, J.L.P. Kwan, and Y. Xia, "Question answering using a large text database: A machine learning approach" In Proceedings of EMNLP-2001, pp. 67–73, Pittsburgh, PA., USA, 2001.

[8] R. Byrd, N. Calzolari, M. Chodorow, M. Neff and O. Risk, "Tools and Methods for Computational Lexicography" Computational Linguistics, vol. 13, n. 3-4, 1987.

[9] A. Maedche, V. Pekar and S. Staab, "Ontology learning part One: On Discovering Taxonomic Relations from the Web" in In Web Intelligence. Springer, Chapter 1, 2002.

[10] M. Hearst, "Automatic Acquisition of Hyponyms from large Text Corpora", Proc. of 14th COLING, Nantes, France, July 1992.

[11] M. P. Oakes, "Using Hearst's Rules for the Automatic Acquisition of Hyponyms for Mining a Pharmaceutical Corpus", RANLP Text Mining Workshop, 2005.

[12] R. Snow, D. Jurafsky, A. Y. Ng, "Learning syntactic patters for automatic hypernym discovery", NIPS 17, 2005.

[13] D. Widdows, "Unsupervised methods for developing taxonomies by combining syntactic and statistical information", HLT-NAACL 2003, Edmonton, USA, 2003.

[14] M. Thelen, E. Riloff, "A Bootstrapping Method for Learning Semantic Lexicons using Extraction Pattern Contexts", proceedings of the Conference on Empirical Methods in Natural Language Processing, 2002.

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

[16] D. Zhang and W. S. Lee, "Web Taxonomy Integration through Co-Bootstrapping", SIGIR-04, July 25-29, Sheffield UK, 2004.

[17] P. Cimiano, A. Hotho and S. Staab, "Comparing Conceptual, Divisive and Agglomerative Clustering for learning Taxonomies from Text", in 16th ECAI 2004, Valencia, Spain, 2004.

[13] Except, of course, for the identification of a set of core domain concepts: however, we suggested to re-use available resources whenever possible.

[14] We thank Orin Hargraves for providing us this information.

[18] J. Jannink, “Thesaurus Entry Extraction from an On-line Dictionary”, in Proceedings of the Second International Conference on Information Fusion, July 1999.

[19] N. Ide and J. Véronis, “Refining Taxonomies extracted from machine readable Dictionaries”, Research in Humanities Computing 2, Oxford University Press, pp. 145-59, 1994

[20] R. Navigli, P. Velardi, “Learning Domain Ontologies from Document Warehouses and Dedicated Websites”, Computational Linguistics 30(2), MIT Press, 2004.

[21] J.E.F. Friedl, “Mastering Regular Expressions”, O’Reilly eds., ISBN: 1-56592-257-3, First edition January 1997.

[22] G. Stumme, and A. Maedche, “Ontology Merging for Federated Ontologies on the Semantic Web”, Workshop on Ontologies and Information Sharing, IJCAI, 2001.

[23] R. Navigli, P. Velardi, “Structural Semantic Interconnections: a Knowledge-Based Approach to Word Sense Disambiguation” IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 27(7), 2005.

[24] M. Uschold, M. King, S. Moralee, Y. Zorgios, “The Enterprise Ontology”, http://citeseer.ist.psu.edu/255311.html

[25] S. Borgo, P. Leitão, “The Role of Foundational Ontologies in Manufacturing Domain Applications”, in P. R. Meersman, Z. Tari et al. (eds.) OTM Confederated International Conferences, ODBASE 2004, Ayia Napa, Cyprus, LNCS 3290, Proceedings Part 10, Springer Verlag, pp. 670-688, 2004.

[26] J. Lee, M. Kim and Y. Lee, “Information Retrieval Based on Conceptual Distance in IS-A Hierarchies”, Journal of Documentation, Vol.49(2), pp. 188-207, 1993.

[27] N. Foo, B. Garner, A. Rao and E. Tsui, “Semantic Distance in Conceptual Graphs”, in P. Eklund, T. Nagle, J. Nagle and L. Gerhotz (eds.) “Current Directions in Conceptual Structure Research”, Hellis Horwood, pp. 149-154, 1992.

[28] P. Resnik, “Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language”, Journal of Artificial Intelligence Research", Vol.11, pp. 95-130, 1999.

[29] A. Tversky, “Features of Similarity”, Psychological Review, Vol.84(4), pp. 327-352, 1977.

[30] J.J. Jiang and D.W. Conrath, “Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy”, in Proceedings of the 10th International Conference on Research in Computational Linguistics, Taipei, Taiwan, 1997.

[31] M. Rodriguez, M. Egenhofer, “Determining Semantic Similarity Among Entity Classes From Different Ontologies”, IEEE Trans. on Knowledge and Data Eng., Vol.15(2), pp. 442-456, 2003.

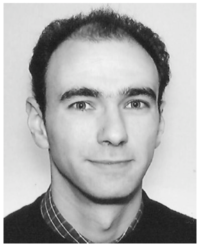
[32] V. Cross, “Fuzzy Semantic Distance Measures Between Ontological Concepts”, in Proceedings of IEEE-NAFIPS’04, pp. 236-240, 2004.

[33] G. Salton and M. McGill, “An Introduction to Modern Information Retrieval”, McGraw-Hill, New York, NY, 1983.

[34] A. Naeve, “The Human Semantic Web: Shifting from Knowledge Push to Knowledge Pull”, International Journal of Semantic Web and Information Systems (IJSWIS), Vol. 1, Issue 3, pp: 1-30, 2005.

Paola Velardi is a full professor in the Department of Computer Science at the University of Roma "La Sapienza". Her research interests include natural language processing, machine learning, and the semantic web. She received a Laurea degree in electrical engineering from the University of Roma "La Sapienza". Contact her at University of Roma "La Sapienza", Dipartimento di Informatica, via Salaria 113, 00198 Roma, Italy; velardi@di.uniroma1.it

Alessandro Cucchiarelli received a master degree in electronic engineering in 1985. In 1991 he became a researcher and joined the Department of Computer Science, Management and Automation (DIIGA) of the Polytechnic University of Marche. Since 2005 he has been an associate professor at the same Department. His research interests are focused on e-Learning technologies, and on the application of Natural Language Processing techniques to the automatic extraction of information from the Web and to domain ontology definition. He has been a member of many research teams working on projects funded by the EU and the Italian ministry of research (MURST/MIUR). He is author or co-author of about 70 papers on refereed journals, conference proceedings and book chapters. Contact him at a.cucchiarelli@diiga.univpm.it



Michaël Pétit is an associate professor at the the Computer Science (CS) Department of the University of Namur. His interests lie in software requirements engineering method and applications, enterprise modeling (UEML), eBusiness modelling and Business models. He is author of many papers and is active in conferences and workshops on these topics. He is responsible for the Knowledge map workpackage in the INTEROP project. He obtained a PhD in Computer Science in 1999. Contact him at mpe@info.fundp.ac.be