Book Title
Book Editors
IOS Press, 2003

Evaluation of OntoLearn, a methodology for automatic learning of domain ontologies

Paola VELARDI [a], Roberto NAVIGLI [a], Alessandro CUCCHIARELLI [b] and Francesca NERI [b]

[a] *Dipartimento di Informatica - Università di Roma "La Sapienza"- Roma, Italy*
[b] *DIIGA - Università Politecnica delle Marche - Ancona, Italy*

Abstract. Ontology evaluation is a critical task, even more so when the ontology is the output of an automatic system, rather than the result of a conceptualization effort produced by a team of domain specialists and knowledge engineers. This paper provides an evaluation of the OntoLearn ontology learning system. The proposed evaluation strategy is twofold: first, we provide a detailed *quantitative* analysis of the ontology learning algorithms. Second, we automatically generate natural language descriptions of formal concept specifications in order to facilitate per-concept *qualitative* analysis by domain specialists.

Keywords. Ontology learning, natural language processing

1. Introduction

Ontologies play an important role in the so-called Semantic Web project [1]. Their aim is to capture domain knowledge in a particular area of interest, favoring interoperability and providing a shared understanding among the involved players of web-based applications (e.g. web services, resource sharing among enterprises, and in general, web information access). In recent years, research related to ontology development produced tangible results concerning the definition of language standards [2] and increasingly powerful ontology editing and management tools [3][4]. Despite the availability of these tools, populating domain ontologies with a sufficiently large number of concepts is a tedious and time-consuming process, preventing wide-scale production and usage of ontologies by industrial institutions. Automatic methods for ontology learning and population have been proposed in recent literature (e.g. ECAI-2002 [5], KCAP-2003 [6] workshops, and [7]), but a co-related issue then becomes the *evaluation* of such automatically generated ontologies, not only to the end of comparing the different approaches, but also to verify whether an automatic process may actually compete with the typically human process of converging on an *agreed* conceptualization of a given domain. Ontology construction, apart from the technical aspects of a knowledge representation task (i.e. choice of representation languages, consistency and correctness with respect to axioms, etc.), is a *consensus building* process, one that implies long and often tedious discussions among the specialists of any one given domain. Can an automatic method simulate this process?

Can we provide domain specialists with a means to measure the *adequacy* of a specific set of concepts as a model of a given domain (by defining a domain as a set of documents related to a certain topic)? Often, specialists are unable to evaluate the formal content [8] of a computational ontology (e.g. the denotational theory, the formal notation, the knowledge representation system capabilities such as property inheritance, consistency, etc.). Evaluation of the *formal content* is mainly tackled by computational scientists, or by automatic verification systems. The role of the specialists is instead to compare their intuition of a domain with the description of this domain, as provided by the ontology concepts.

To facilitate per-concept evaluation, we have devised a method for automatic gloss generation as an extension of the OntoLearn ontology learning system, described in [7][9]. Glosses provide a description, in natural language, of the formal specification automatically assigned to the learned concepts. A domain specialist can easily compare his intuition with this natural language description of the system's choices. The objective of the gloss-based evaluation is to obtain a judgement, by domain specialists, concerning the adequacy of an automatically derived domain conceptualisation.

On the other hand, automatic ontology learning is based on software programs aimed at extracting and formalising domain knowledge, usually starting from unstructured data. It is therefore equally important to evaluate these programs on a *quantitative* ground, in order to gain insight on the internal and external contingencies that may affect the result of an ontology learning process.

In the following, we firstly provide a quantitative evaluation of the OntoLearn ontology learning system, under different learning circumstances. Secondly, we describe the gloss-based per-concept evaluation method. The evaluation on OntoLearn is conducted on several domains analysed in the context of past and on-going national and European projects[1]: Finance, Tourism, Enterprise Interoperability, Computer Networks and Art. Whenever appropriate, we used also available generic test sets.

This paper is organized as follows: section 2 provides a brief overview of the OntoLearn system, section 3 provides a quantitative evaluation of the OntoLearn algorithms and finally section 4 describes the gloss generation algorithm and presents an evaluation experiment, conducted with the help of two domain specialists.

2. The OntoLearn System

Figure 1 provides a snapshot of the OntoLearn ontology learning methodology. The following steps are performed by the system[2]:

[1] E.g. Harmonise IST-2000-29329, a now concluded EC project on *tourism* interoperability, the INTEROP network of excellence on *enterprise interoperability*, started on December 2003, http://www.interop-noe.org/, a national project on web-learning, http://www.web-learning.org, and a bilateral project on *cultural heritage* with ENEA, the national agency for new technologies, energy and environment http://www.enea.it/com/ingl/default.htm.

[2] Limited details on the algorithms are provided here, for obvious sake of space, and because they have been described in detail in other papers. The interested reader can access the OntoLearn bibliography referred to throughout the paper.

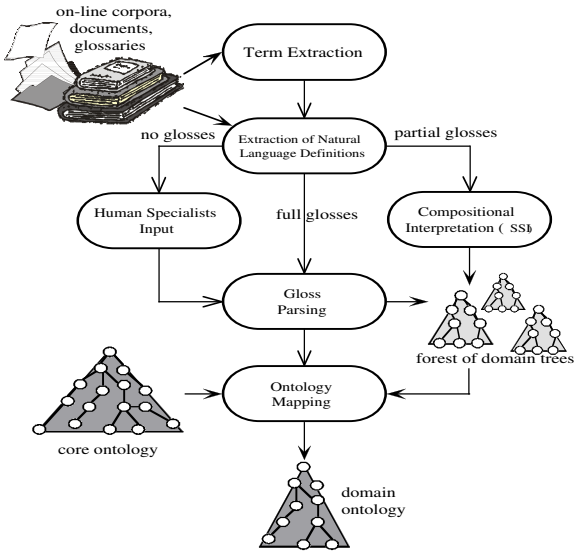

Figure 1. An outline of the ontology learning phases in the OntoLearn system.

1. Extract pertinent domain terminology

Simple and multi-word expressions are automatically extracted from domain-related corpora, like enterprise interoperability (e.g. *collaborative work*), hotel descriptions (e.g. *room reservation*), computer network (e.g. *packet switching network*), art techniques (e.g. *chiaroscuro*). Statistical and natural language processing (NLP) tools are used for automatic extraction of terms [7].

Statistical techniques are specifically aimed at simulating human consensus in accepting new domain terms. Only terms uniquely and consistently found in domain-related documents, and not found in other domains used for contrast, are selected as candidates for the domain terminology. Contrastive domains are in part generic texts (e.g. novels, or balanced corpora like the Brown corpus) and in part texts selected according to the task at hand[3].

2. Search on the web available natural language definitions from glossaries or domain-related documents

Available natural language definitions are searched for on the web using on-line glossaries or extracting definitions in available documents. Regular expressions and a syntactic parser are used to extract and parse definition sentences, as detailed later (step 3.2). The method is tuned for high precision, possibly low recall. In fact, certain expressions (e.g. "*X* is an *Y*") are too generic and produce mostly noise when used for sentence extraction.

[3]For example, in the INTEROP enterprise interoperability domain, we used a generic computer science glossary for contrast, in order to exclude from the list of candidate terminological entries of interoperability generic computer science terms.

3. IF definitions are found:

3.1. Filter out non relevant definitions

Multiple definitions may be found on the Internet, some of which may not be pertinent to the selected domain (e.g. inside the interoperability domain, for the term "federation" we can filter out the definition "the forming of a nation"). A similarity-based filtering algorithm is used to prune out "noisy" definitions, with reference to a domain. Furthermore, an extension of the regular expressions of step 2 is used to select[4], where possible, "well formed" definitions, according to lexicographic criteria.

For example, definitions expressed in terms of genus (*kind-of*) and differentia (*modifier*), are preferred to definitions by example, like *Bon a Tirer* "when the artist is satisfied with the graphic from the finished plate, he works with his printer to pull one perfect graphic and it is marked 'Bon a Tirer', meaning 'good to pull'". These definitions can be pruned out since they usually do not match any of the regular expressions.

3.2. Parse definitions to extract kind-of information

Regular expressions are again used to extract *kind-of* relations from natural language definitions. At first, sentence chunks (NP, PP, etc.) are identified by using the chunker module of a free available POS tagger and chunker, the TreeTagger[5]. Then the regular expression: *r = "^(PP)?(NP)+"* is used to identify the main noun phrase (NP) of a sentence including a term of interest. The expression *r1 = "^(A|D)?((V|C|,|J|N|R)*) (N)"* analyses the main NP. Symbols in *r1* are part of speech tags (POS), e.g. article (A) verb (V) adjective (J) etc. For example, given the sentence:
domain-model: "In the traditional software engineering perspective, a precise representation of specification and implementation concepts that define a class of existing systems".
we obtain:

Syntactic Chunks: (PP **NP** PP CNP RVP NP PP)
POS: (PAJNNN AJ**N** PNCNNWVANPJN)
hyperonym: representation

the bold POS represents the fragment selected as the hyperonym.

$$domain - model \overset{kind-of}{\longrightarrow} representation$$

4. ELSE IF definitions are not found

4.1. IF definitions are available for term components (e.g. no definition is found for the compound integration strategy but integration and strategy have individual definitions)

4.1.1. Solve ambiguity problems

In technical domains, specific unambiguous definitions are available for the component terms, e.g.: *strategy*: "a series of planned and sequenced tasks to achieve a goal" and *integration*: "the ability of applications to share information or to process indepen-

[4]The grammar used for analysing definitions is a superset of the grammar used to extract definitions from texts. The analysed sentences are extracted both from texts and glossaries, therefore expressions like *X* is an *Y* must now be considered.

[5]TreeTagger is available at http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html.

dently by requesting services and satisfying service requests" (interoperability domain). In other domains, like tourism, the definitions of component terms are often extracted from generic lexicons (e.g. for *housing list*, no definitions for *list* are found in tourism glossaries, but in general purpose dictionaries the word *list* is highly ambiguous).

If a definition for a component terms does is not found in domain glossaries or documents, then it is extracted from the WordNet semantic lexicon [10]. But since WordNet is highly ambiguous, a word sense disambiguation algorithm, called SSI[6] [7] is used to select the appropriate *concepts* (senses) for the component terms. Then, a machine-learning algorithm [18] is used to identify the conceptual relation holding between the component concepts. For example, the compositional interpretation of the term *integration strategy* eventually leads to:

$$integration - strategy\#1 \stackrel{kind-of}{\longrightarrow} strategy\#1 \stackrel{purpose}{\longrightarrow} integration\#2$$

where sense numbers are those in WordNet.

The SSI algorithm is a knowledge-based disambiguation algorithm. It uses a lexical knowledge base (LKB), created through the integration of WordNet with several other resources [11], and a pattern matching strategy to identify semantic interconnection patterns between alternative senses, given an initial words context *T*. For example, if *T*=[*estate, stock, ...*] the senses #2 and #1 of *estate* and *stock* are suggested by the existence of the following interconnection pattern in the LKB:

$$estate\#2 \stackrel{relatedness}{\longrightarrow} assets\#1 \stackrel{has-kind}{\longrightarrow} capital\#1 \stackrel{has-kind}{\longrightarrow} stock\#1 \qquad (1)$$

Details are found in the referred papers.

4.1.2. Create a definition compositionally

Once the appropriate meaning components and semantic relations have been identified for a multi-word expression, a generative grammar is used to produce natural language definitions.

The grammar is based on the presumption (often, but not always, verified) that the meaning of a multi-word expression can be generated compositionally from its parts. According to this compositional view, the syntactic head of a multi-word expression represents the *genus* (kind-of), and the other words the *differentia* (modifier). For example, *integration strategy* is a "strategy for the integration".

We use natural language generation also as a means to support human evaluation of the concept hierarchy created by OntoLearn, therefore this step of the ontology learning methodology will be discussed in more detail in Section 4.

4.2. ELSE ask experts

If it is impossible to find even partial definitions for a multi-word expression, the term is submitted to human specialists, who are in charge of producing an appropriate and agreed definition.

[6]The SSI algorithm (Structural Semantic Interconnections) is one of the novel and peculiar aspects of the OntoLearn system.

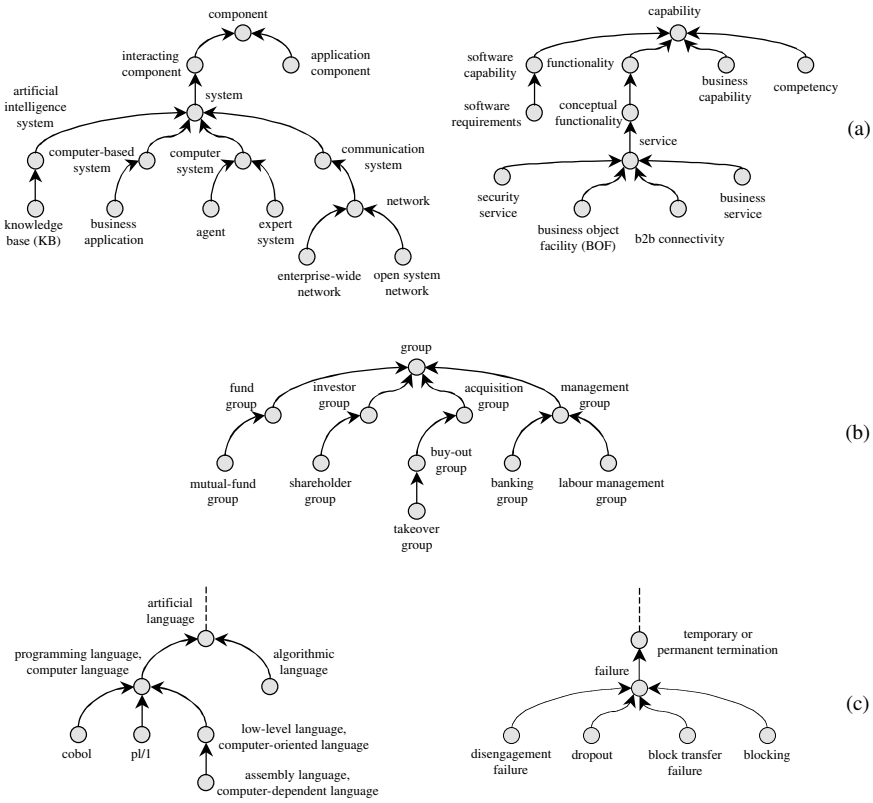

Figure 2. Concept trees generated from a) an enterprise interoperability domain b) a financial domain and c) a computer network domain.

5. Arrange terms in hierarchical trees

Terms are arranged in forests of concept trees, according to the hyperonymy information extracted in steps 3.2 and 4.1.1. Figure 2 shows examples of trees generated from an enterprise interoperability domain, a financial domain and a computer network domain.

6. Link sub-hierarchies to the concepts of a Core Ontology.

The semantic disambiguation algorithm SSI (mentioned in step 4.1.1) is used to append sub-trees under the appropriate node of a Core Ontology. Currently, we use the WordNet semantic lexicon. This is motivated by the fact that sufficiently rich domain ontologies are presently available only in few domains (e.g. medicine).

With reference to the right hand tree of Figure 2c, the root *artificial language* has a monosemic correspondent in WordNet, but *temporary* or *permanent termination* has no direct correspondent. The node is then linked to *termination*, but first, a disambiguation problem must be solved, since *termination* in WordNet has two senses: *"end of a time span"*, and *"expiration of a contract"*. Similarly, the roots *component capability* and *group* in the upper trees have 3 senses each. The SSI algorithm is therefore used to identify the correct attachment for each tree. As better clarified in [7], the context for disambiguation is provided by the other term components in a concept tree. For example,

the context *T* to disambiguate *group* in the middle tree of Figure 2 is *T*=[*fund, investor, management, shareholder, mutual, buy-out, banking, labor*].

3. Evaluation of the OntoLearn Ontology Learning System

The evaluation of an ontology learning method can be split in two problems: the first is to evaluate the effectiveness of the ontology learning *algorithm(s)*, the second is to evaluate the generated ontology, as an artifact. This latter issue is discussed in Section 4. This Section is a summary of the "algorithmic" evaluation of OntoLearn.

OntoLearn is based on four main algorithms:

1. extraction of terms
2. extraction of natural language definitions
3. parsing of natural language definitions
4. semantic disambiguation (to identify the correct sense of term components, and to attach sub-trees under the appropriate node of a generic ontology)
5. identification of semantic relations between term components

The quality of the generated ontology depends on each of these steps, therefore it is important to estimate the performance of individual tasks. A first consideration is that the initial phases of the ontology learning methodology (extraction of terms and of term definitions) critically depends upon the availability of a sufficiently large and representative documentation, which must be made available by domain specialists. In certain domains, e.g. tourism or art, such documents can be easily collected: for example, in the Harmonize EC project on tourism, descriptions of hotel accommodation were provided both by the tourism organizations participating in the project, and collected from the Internet by the authors of this paper. Instead, the domain of INTEROP NoE is enterprise interoperability, a relatively new technical domain, for which a relevant bibliography was still to be collected at the beginning of the project[7] .

The evaluation of tasks 1, 2 and 3 can be conducted by domain specialists, since it only requires a solid domain expertise, but others tasks (especially 4) are more complex, in that they need the evaluators to be aware of the adopted semantic model. For example, if semantic disambiguation is performed with respect to the WordNet lexical ontology (or any other available core ontology), evaluators need to be familiar with the ontology sense inventory and taxonomic organization. Similarly for task 5, evaluators must be aware of the formal specifications of the adopted conceptual relation inventory. This motivates the definition of a gloss generation algorithm (Section 4), whose main purpose is precisely to facilitate per-concept evaluation by domain specialists.

Still, a quantitative evaluation of these tasks is indeed necessary, to estimate the reliability of the ontology learning algorithms.

In what follows we briefly summarize the outcomes of several evaluations conducted on the five tasks. Since the OntoLearn system has been continuously enhanced in these years, the evaluation outcomes are not entirely homogeneous on the various application domains provided by the projects in which OntoLearn has been used in the last few years.

[7] An initial set of documents were collected from partners for the purpose of testing the terminology extraction procedure, but only at the end of the first year of the project (November 2004) a rich bibliography and state of art has been made available, therefore the entire ontology learning process will be repeated.

1) The terminology extraction algorithm has been successfully evaluated in the context of the European project Harmonise on Tourism interoperability, leading to around 80% precision and 55% recall. The recall was estimated by manually identifying truly relevant terms from a list of syntactically plausible multi-word expressions, automatically extracted from a fragment of available documents. In the more recent INTEROP project, we did not estimate the recall, but the precision was quite higher (over 85%), due to the technicality of the domain. In this domain, contrastive analysis wrt other domains (see Section 2, step 1) was more effective at eliminating non domain-pertinent multi word expressions.

In any case, manual analysis of the extracted terminology is advisable *before* proceeding with the subsequent steps. This task lasts about 0.5 minutes per term, and can be easily accomplished in few hours by domain specialists.

2) Extraction of definitions is a task for which we have only a rough estimate of precision, since it is a new feature of the OntoLearn system, still to be enhanced. When using very specific regular expressions (e.g. "*X* is defined as *Y*") the precision is close to 100%, but such expressions allow only a fragment of available definitions embedded in documents to be captured. On the contrary, generic expressions like "*X* is a *Y*" produce excessive noise, therefore we avoided using them. In the first INTEROP terminology extraction experiment, to increase the number of detected definitions, we used a restricted set of reliable expression to query the web. This allowed us to retrieve definitions for about 80% of the terms (often with multiple definitions for a term) but a better strategy would have been to inspect only the domain corpus[8]. Using the web in an unrestricted way caused the extraction of several non-domain-pertinent definitions, a problem only in part solved with automatic classification technique (see Section 2, step 3.1).

Extraction of definitions is a task for which achieving a particularly high precision is not so critical. The objective here is to speed up the work of domain experts, who can easily examine definitions, produce corrections, or delete inappropriate definitions. In the INTEROP experiment, the only one for which we conducted a systematic evaluation of this specific task, six domain experts were asked to review and refine 358 automatically extracted definitions (including multiple definitions for some term). Each expert could review (*rev*), reject (*rej*), accept (*ok*) or ignore (*blank*) a definition, acting on a shared database. The experts added new definitions for brand-new terms, but they also added new definitions for terms which may have more than one sense in the domain. There have been a total of 67 added definitions, 33 substantial reviews, and 26 small reviews (only few words changed or added). Some terms (especially the more generic ones, e.g. *business domain*, *agent*, *data model*) were reviewed by more than one expert who proposed different judgments (e.g. *ok* and *rev*) or different revised definitions. A second pass was therefore necessary for adjustment, but overall, we estimated a total time of 7.5 minutes per definition, a figure that favorably compares with the 16 minutes declared by an expert lexicographer[9] consulted to evaluate the speeding up of our procedure.

3) Parsing of definitions. As briefly reported in Section 2, definitions are analysed using a natural language POS tagger and chunker (the TreeTagger) and regular expres-

[8] However, as already remarked, during the first INTEROP experiment the domain corpus was not entirely available.

[9] We thank Orin Hargraves for his very valuable comments.

Table 1. Precision and recall of the hyperonymy extraction task in three domains.

	Art	Interoperability	Computer Networks
Precision	0.973	0.947	0.955
Recall	0.957	0.914	0.932

sions, in order to extract the word or multi-word expression representing the hyperonym. The evaluation of this task has been conducted on three domains: the already mentioned INTEROP project on interoperability, the ENEA project on cultural heritage, and the web-learning project on computer network courseware[10]. The results are summarized in Table 1.

As shown in Table 1, this task is performed with very high precision and recall. The table measures the ability of the analyser to identify the main NP of the definition, and then to extract the hyperonym from that sentence. Remember that only definitions that obey the regular expression *r* in 3.2 are analysed, i.e. definitions provided in terms of *genus* and *differentia*. Extracting hyperonymy relations from natural language definitions (manually or automatically) may have inherent problems, e.g. attachments too high in the hierarchy, unclear choices for more general terms, or-conjoined heads, absence of hyperonym, circularity, etc., as discussed in [12]. Our purpose here is not to overcome problems inherent with the task of building a domain concept hierarchy: rather, OntoLearn's mission is to speed up the task of ontology building and population, extracting and formalizing domain knowledge expressed by human specialists in an unstructured way. Discrepancies and inconsistencies can be corrected later by the ontology engineers, who will verify and refine the system output.

4) Semantic disambiguation. As already remarked, the semantic disambiguation algorithm SSI is the core algorithm of the OntoLearn methodology. It is used to interpret multi-word *terms* as complex *concepts*, by associating the component words of a complex term to the appropriate *concepts* (word senses) in a reference lexicalised ontology (i.e. WordNet). Furthermore, SSI is used to attach the root nodes of a domain forest under the appropriate node of WordNet.

The authors of this article evaluated SSI in several domains related to both past and present projects (see [7] and [9]), leading to an average precision figure between 82% and 86%, and a recall of between roughly 60% and 70%. These evaluations have been conducted with the aim of relating SSI performances with the specific aspects that may influence the results, e.g. the dimension of the word context *T* to be disambiguated, the technicality of the domain, some variants of the basic algorithm, etc. In all these domains, the test sets have been prepared by the authors of this article, since expert lexicographers were not available to the project team.

On the other hand, SSI is a generic word sense disambiguation algorithm, therefore it can be more objectively evaluated on standard WSD datasets. Such datasets are provided within the SensEval[11] competitions, comparative evaluations of WSD systems that are periodically organized. Even though sense disambiguation contexts provided by SensEval organizers are far more complex that those occurring in ontology learning applications (i.e. words are extracted from generic sentences, and are often weakly semanti-

[10] See footnote 1 for project references.

[11] http://www.senseval.org/

Table 2. Results of gloss disambiguation task in Senseval-3.

System	Prec.	Recall	Attempted
SSI	0.685	0.684	99.9
TALP Research Center	0.702	0.698	99.9
LanguageComputerCorp	0.721	0.516	71.6

Table 3. Results of the English all-words task in Senseval-3.

System	Prec.	Recall
GAMBL-AW-S	0.651	0.651
Sense Learner-S	0.65	0.642
IRST-DDD-00-U	0.583	0.582
SSI	0.604	0.604

cally related), the same difficulty still applies to all participating systems, thus providing a more objective evaluation testbed.

We applied SSI to two Senseval-3 tasks: *gloss disambiguation* and *English all-words*. In the gloss disambiguation task, participant were asked to disambiguate the natural language definitions (glosses) of a subset of WordNet senses. In the all-words task, the participants were asked to disambiguate (almost) all the words in a test set of generic English sentences. Tables 2 and 3 report the results of the best participating systems.

In the gloss disambiguation task, SSI was the second best performing system, close to the first, and well over the third (see [13] for details). Table 3 shows the performance of SSI as compared with the first two supervised WSD systems, and with the best unsupervised system (IRST-DDD). We did not actually participate in the all-words task, but we ran SSI using the standard test set and evaluation program made available by the organizers. Table 3 shows that SSI performs better than the best unsupervised system (SSI is an unsupervised algorithm, close to the best performing supervised systems). We may conclude that SSI favorably compares with the best available WSD algorithms, with two significant advantages: first, SSI is unsupervised, contrary to most existing methods, second, it provides a justification of its disambiguation choices, in the form of semantic patterns (e.g. graph (1) in step 4.1 of Section 2). This proved to be particularly helpful in the gloss disambiguation task, where we have been able to detect certain inconsistencies in the training set provided by the organizers (as discussed in [13] and [14]), but is also useful as a means to help the evaluation, by expert lexicographers, of a semantic annotation task.

5) Annotation with semantic relations. In order to complete the interpretation process, OntoLearn attempts to determine the semantic relations between the components of a complex concept. In order to do this, it was first necessary to select an inventory of semantic relations. We examined several proposals, like EuroWordnet [15], DOLCE [16], FrameNet [17], and others. As also remarked in [8], no systematic methods are available in literature to compare the different sets of relations. Since our objective was to define an automatic method for semantic relation extraction, our final choice was to use a reduced set of FrameNet relations, which seemed general enough to cover our application domains (tourism, economy, computer networks). The choice of FrameNet is motivated

Table 4. a)Performance on Tourism b)Performance on Economy.

	d<=10%	d<=30%	d<=100%
Precision	0.958	0.875	0.847
Recall	0.283	0.636	0.793

(a)

	d<=10%	d<=30%	d<=100%
Precision	1.000	0.804	0.651
Recall	0.015	0.403	0.455

(b)

by the availability of a sufficiently large set of annotated examples of conceptual relations, which we used to train an available machine learning algorithm, TiMBL [18]. The relations used are: *Material, Purpose, Use, Topic, Product, Constituent Parts, Attribute*. The description of these relations can be found in [17], except for *Attribute*, which is not a FrameNet relation. Unfortunately, these relations are not particularly suited for more technical domains, like enterprise interoperability and computer networks. For the art domain, we are currently trying to use the semantic relations of the CRM-CIDOC[12] core ontology, a very accurate domain core ontology.

An evaluation of the semantic tagging with TiMBL was then conducted in the Economy and Tourism domain, as shown in Table 4. We represented training instances as pairs of concepts annotated with the appropriate conceptual relation, e.g.: (computer#1,maker#2),Product]. Each concept is in turn represented by a feature-vector where attributes are the concept's hyperonyms in WordNet.

The parameter **d** in the above tables is a confidence factor defined in the TiMBL algorithm. This parameter can be used to increase the system's robustness in the following way: whenever the confidence associated by TiMBL to the classification of a new instance is lower than a given threshold, we output a "generic" conceptual relation, named *Relatedness*. We experimentally set the threshold for **d** at around 30% (central column of Table 4). In the more technical domains we analyzed, this relation is generated rather more often (in about 50% of the cases).

4. Generating Definitions to Support Per-concept Evaluation

In Section 3, we provide a quantitative evaluation of the SSI algorithm, however, manual evaluation by domain specialists is indeed advisable. OntoLearn is in fact a system to support and speed-up the ontology learning process, but it is not meant to fully replace human annotators. In order to help human evaluation on a per-concept basis, we decided to enhance OntoLearn with a gloss generation algorithm. Often specialists are not computer experts, and in any case a natural language expression can be evaluated far more easily than a conceptualization in some formal language[13].

There are two cases in which gloss generation is necessary: when a definition of a multi-word expression is not found in glossaries or documents, and when attaching a

[12] http://cidoc.ics.forth.gr/

[13] Specifically, OntoLearn generates an ontology in the OWL ontology web language http//www.w3.org /2004/OWL

root node of a domain concept tree to the appropriate WordNet node. In both cases, a semantic disambiguation step is performed, but the result of this disambiguation (a set of WordNet sense numbers and conceptual relations) is evaluated with difficulty by domain specialists. On the other hand, it is rarely the case that expert lexicographers are available in an ontology building team. The idea is to generate glosses in a way that closely reflects the key aspects of the OntoLearn concept learning process, i.e. semantic disambiguation and annotation with a conceptual relation. The gloss generation algorithm is based on the definition of a grammar with distinct generation rules for each type of semantic relation. Let $s_j^h \stackrel{sem-rel}{\longrightarrow} s_j^k$ be the complex concept associated to a complex term $w_h w_k$ (e.g. *jazz festival*, or *long-term debt*), and let:

<H> be the syntactic head of $w_h w_k$ (e.g. *festival, debt*)
<M> be the syntactic modifier of $w_h w_k$ (e.g. *jazz, long-term*)
<GNC> be the gloss of the new complex concept S^{hk}
<HYP> be the selected senses of <H> (e.g. respectively, *festival#1* and *debt#1*)
<MSGHYP> be the main sentence[14] of the gloss of <HYP>
<MSGM> be the main sentence of the gloss of the selected sense for <M>

Two examples of rules for generating GNCs are:

- If sem-rel=*Topic*, <GNC>::= a kind of <HYP>, <MSGHYP>, relating to the <M>, <MSGM>.
  e.g.: GNC(*jazz festival*)=*"a kind of festival, a day or period of time set aside for feasting and celebration, relating to the jazz, a style of dance music popular in the 1920s*".
- If sem-rel=*Attribute*, <GNC>:= a kind of <HYP>, <MSGHYP>, <MSGM>.
  e.g.: GNC(*long term debt*)=*"a kind of debt, the state of owing something (especially money), relating to or extending over a relatively long time*".

Notice that, in the grammar above, the "gloss" for the term components can be either that of a disambiguated word sense in WordNet, or a domain-specific definition found during the definition extraction phase of the OntoLearn methodology. For example, consider the term *knowledge management practice*, extracted in the interoperability experiment. No definition was found for this term, but the definition of *knowledge management* (approved by the INTEROP partners) is: *"The process of capturing value, knowledge and understanding of corporate information, using IT systems, in order to maintain, re-use and re-deploy that knowledge"*. *Practice* is not included in the interoperability terminology, but WordNet has 5 senses for this word. The gloss parsing method selects the term *process* as the hyperonym of *knowledge management*, and the SSI algorithm selects sense 5 of WordNet for *practice* (*"knowledge of how something is customarily done"*), and sense 3 for *process*, a choice supported among the others by the following interconnection pattern[15]:

$$process\#3 \stackrel{kind-of}{\longrightarrow} cognition - knowledge\#1 \stackrel{gloss}{\longleftarrow} practice\#5$$

[14] The main sentence is the gloss pruned of subordinates, examples, etc.
[15] The arrow tagged with gloss is a relation between a word sense and a word sense appearing in its definition.

Table 5. Evaluation of glosses by domain specialists.

	vote=1	vote=2	vote=3	uncertain average	
Tourism total (97)	33 (34.0%)	14 (14.4%)	45 (46.4%)	5 (5.2%)	2.13
Economy total (134)	52 (38.8%)	16 (11.9%)	66 (49.2%)	-	2.10

The generated definition is *"**a kind of** practice, knowledge of how something is customarily done, **relating to the** knowledge management, the process of capturing value, knowledge and understanding of corporate information, using IT systems, in order to maintain, re-use and re-deploy that knowledge"*.

The generated definitions are quite verbose, but have the advantage of explicitly showing the sense choices operated by the sense disambiguation algorithm. A human supervisor can easily verify sense choices and reformulate the definitions in a more compact way.

To verify this, the automatically generated glosses were submitted for evaluation by two human experts, a tourism specialist from ECCA[16], and an economist from the Università Politecnica delle Marche. The specialists were not made aware of the method used to generate glosses; they were simply presented with a list of concept-gloss pairs and asked to fill in an evaluation form (see Appendix) as follows: vote 1 means "unsatisfactory definition", vote 2 means "the definition is helpful", vote 3 means "the definition is fully acceptable". Whenever the evaluator was not fully happy with a definition (vote 2 or 1), he was asked to provide a brief explanation as to why. Table 5 provides a summary of the evaluation.

The following conclusions can be drawn from this experiment:

1. Overall, the two domain specialists fully accepted the system's choices in 46-49% of the cases, and were reasonably satisfied in 12-14% of the cases. The average vote is above 2 in both cases.
2. There are two other main causes of "bad" definitions. One is when the multiword expression cannot be interpreted compositionally, or some of the term components have an idiosyncratic sense not available in the glossary or in WordNet. The other is an OntoLearn error in disambiguation. Examples of OntoLearn errors and idiosyncratic senses (see the Appendix) are the definitions 14_T (wrong sense of *form*) and 19_E (no good sense for *bilateral* in WordNet), respectively.
3. Another cause of unsatisfaction is the verbosity of definitions. One of the specialists is particularly involved in ontology building projects, therefore we report his valuable comment: *"some of the descriptions would not be appropriate to take them over in a tourism ontology just as they are. But most of them are quite helpful as basis for building the ontology. The most important problem from my point of view is the too detailed descriptions of the components itself instead of the meaning of the overall term in this context. Best example is the term 'bed tax'. Nobody would expect a definition of a bed or a tax"*. In other terms, he found disturbing the fact that a definition extensively reports the definitions of its components. On the other hand, our objective is not only to produce concept definitions, but also to organize concepts in hierarchies. Showing the definitions of individual components is a "natural" means to verify that the correct senses have been

[16] ECCA - eTourism Competence Center Austria.

selected (e.g. the correct senses of bed and tax). This is clearly the case, since, for example in definition 14_T (*booking form*) in the Appendix, the specialist was immediately able to diagnose a sense disambiguation error for *form*, though he was unaware of the OntoLearn methodology.

Acknowledgements

Our thanks go to Dr. Wolfram Höpken, from ECCA - eTourism Competence Center Austria and Dr. Donato Iacobucci, from the Università Politecnica delle Marche, who gave up their precious time to evaluate our glosses.

This work has been in part supported by the INTEROP Network of Excellence IST-2003- 508011.

References

[1] T. Berners-Lee *Weaving the Web* Harper, San Francisco, 1999.
[2] *Web Ontology Language (OWL) Reference Version 1.0* http://www.w3.org/TR/2002/WD-owl-ref-20021112/.
[3] *The Protègè Ontology Editor and Knowledge Acquisition System* http://protege.stanford.edu/.
[4] Y. Sure, M. Erdmann, J. Angele, S. Staab, R. Studer, D. Wenke *OntoEdit: Collaborative Ontology Development for the Semantic Web* Proceedings of the first International Semantic Web Conference 2002 (ISWC 2002), June 9-12 2002, Italy.
[5] *ECAI-02 Ontology Learning Tools Workshop* http://www-sop.inria.fr/acacia/WORKSHOPS/ECAI2002-OLT/accepted-papers.html.
[6] *KCAP-2003 Knowledge mark-up and Semantic Annotation workshop* http://km.aifb.uni-karlsruhe.de/ws/semannot2003/papers.html.
[7] R. Navigli and P. Velardi *Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites* Computational Linguistics, MIT press, June 2004.
[8] E. Hovy *Comparing Sets of Semantic relations in Ontologies* in R. Geen, C.A. Bean and S. Myaeng Semantic of relations, Kluwer, 2001.
[9] R. Navigli, P. Velardi, A. Gangemi *Corpus Driven Ontology Learning: a Method and its Application to Automated Terminology Translation* IEEE Intelligent Systems, special issue on NLP, January 2003.
[10] G. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K.J. Miller *Wordnet: An on-line lexical database* International Journal of Lexicography, 3(4):235–312, 1990.
[11] R. Navigli *Semi-Automatic Extension of Large-Scale Linguistic Knowledge Bases* to appear in Proc. of 18th FLAIRS International Conference, Clearwater Beach, Florida, May 16-18th, 2005.
[12] N. Ide and J. Vèronis *Refining Taxonomies extracted from machine readable Dictionaries* http://www.up.univ-mrs.fr/~veronis/pdf/1994rhc2.pdf, 1992
[13] R. Navigli and P. Velardi *Structural semantic interconnections: a knowledge-based approach to Word Sense Disambiguation* in Proc. of Senseval-3 Third international workshop on the Evaluation of Systems for Semantic Analysis of Texts, 25-26 July 2004, Barcelona, Spain, pp13-16.
[14] K. Litkowski *Senseval-3 task: Word Sense Disambiguation of WordNet Glosses* in Proc. of Senseval-3 Third international workshop on the Evaluation of Systems for Semantic Analysis of Texts, 25-26 July 2004, Barcelona, Spain, pp13-16.
[15] P. Vossen *EuroWordNet: General Document version 3 Final* July 1999 http://www.hum.uva.nl/~ewn.

[16] C.Masolo, S.Borgo, A.Gangemi, N.Guarino, A.Oltramari, L.Schneider *Sweetening Ontologies with DOLCE* http://citeseer.nj.nec.com/gangemi02sweetening.html.
[17] J.Ruppenhofer, C.J.Fillmore, C.F.Baker *Collocational Information in the FrameNet Database* http://framenet.icsi.berkeley.edu/~framenet/Papers.html.
[18] W. Daelemans, J. Zavrel, K. van del Sloot, and A. van den Bosch, 1999. *TiMBL: Tilburg Memory Based Learner Version 2.0 Reference Guide* Tilburg University.

Appendix: Excerpt of the Gloss Evaluation Form (Economy and Tourism)

Concept #: 25_E **Term:** *business_plan* **Synt:** N-N **Rel<w_1,w_2>:** Topic
Gloss: a kind of plan, a series of steps to be carried out or goals to be accomplished, relating to the business, the activity of providing goods and services involving financial and commercial and industrial aspects.
Specialist vote: 3
Comment by Specialist: none.
Diagnose: none.

Concept #: 2_T **Term:** affiliated_hotel **Synt:** Agg-N **Rel<w1,w2>:** Attribute
Gloss: a kind of hotel, a building where travelers can pay for lodging and meals and other services, being joined in close association.
Specialist vote: 3
Comment by Specialist: none.
Diagnose: none.

Concept #: 14_T **Term:** *booking_form* **Synt:** N-N **Rel<w_1,w_2>:** Purpose
Gloss: a kind of form, alternative names for the body of a human being, for booking, the act of reserving (a place or passage) or engaging the services of (a person or group).
Specialist vote: 1
Comment by Specialist: definition of 'form' wrong in this context.
Diagnose: OntoLearn disambiguation error for 'form'.

Concept #: 19_E **Term:** *bilateral_aid* **Synt:** Agg-N **Rel<w_1,w_2>:** Attribute
Gloss: a kind of aid, the activity of contributing to the fulfillment of a need or furtherance of an effort or purpose, having identical parts on each side of an axis.
Specialist vote: 1
Comment by Specialist: fully wrong definition.
Diagnose: WordNet gloss of 'bilateral' is not adequate to domain (no better definition is available in WordNet).

Concept #: 76_E **Term:** *foreign_aid* **Synt:** Agg-N **Rel<w_1,w_2>:** Attribute
Gloss: a kind of aid, the activity of contributing to the fulfillment of a need or furtherance of an effort or purpose, of concern to or concerning the affairs of other nations.
Specialist vote: 3
Comment by Specialist: none.
Diagnose: none.