Modelling Semantic Grid knowledge embedded in documents

Maria Teresa Pazienza, Marco Pennacchiotti, Fabio Massimo Zanzotto
University of Rome ”Tor Vergata”,
Department of Computer Science, Systems and Production,
00133 Roma (Italy)
{pazienza, pennacchiotti, zanzotto}@info.uniroma2.it

Abstract

The growing success of Grid technologies inside scientific communities has produced an increasing need for the development of tools and methodologies able to support knowledge sharing and handling among people, built upon the Grid. This ”semantic” infrastructure is becoming to be referred as Semantic Grid*. In this paper we propose an original approach to the development of a system for the creation of the* Knowledge Layer *of the Semantic Grid, that is, the layer which carries the informative content that the community shares. Using well-assessed Natural Language Processing and Machine Learning methodologies and techniques, our goal is to acquire and organize the information stored in the Grid, where this information is supposed to be represented in unstructured documents. Our intent is to extract and shape knowledge in* syntactic patterns *and organize them into a hierarchy of* relational concepts*, whose goal is to improve the process of knowledge retrieval and maintenance.*

1. Introduction

The growing interest on developing Grid [4] technologies has produced a fairly large number of applications and tools, enabling the creation of well defined computing infrastructures. Recently, more attention has turned to the possibility of implementing systems able to exploit the Grid networks in order to allow the diffusion and sharing of knowledge among different people and groups . This *semantic* infrastructure, called Semantic Grid [2], built over the Grid computational layer, has gained more and more interest in the scientific community, where an efficient and widespread knowledge and data sharing is a primary goal. As defined in [2], the information carried by the Semantic Grid can be thus intended as ”data equipped with meaning” and much more.

In this framework, the development of open systems able to acquire knowledge from different sources while supporting its sharing inside a large community is a needed task. Such infrastructures, as defined in [3], should consist of three conceptual layers: *data layer, information layer* and *knowledge layer*. Specifically, the third of these layers concerns the task of knowledge acquisition, retrieval, use, publishing and maintenance.

Textual data are pervasive in collaborative work as natural language is one of the preferred media for communicating knowledge. The success of Semantic Grid technologies then depends on the possibility of designing systems that may help in ”absorbing” such a knowledge existing in the grid.

In this perspective, within the *knowledge lifecycle* (cf. [3]), natural language processing (NLP) techniques play a crucial role. The amount of textual data can be very huge making the manual inspection very cumbersome. Moreover, NLP may help in making viable the interplay between the two phases of *knowledge acquisition* and *knowledge modelling*. During the *knowledge acquisition* phase relevant information stored in the domain text collections should emerge and used to justify the definition of the *knowledge model*. This latter can successively be used to expand the knowledge that can be acquired from domain texts.

Extracting factual knowledge out of domain texts is a process that may be organised in the following steps:

1. *Corpus processing*: linguistic forms denoting very specific domain concepts and domain relational concepts are detected, normalised, and ranked according to their *domain relevance*, e.g. out of a document collection regarding the space domain, normalised forms such as ”`Spacecraft` *launch* `Satellite` *from* `Location`” or ”`Spacecraft` *boost into orbit* `Satellite` *from* `Location`” should be emerge as relevant.
2. *Concept formation*: the most important normalised forms are selected and they provide the set of general domain conceptual relationships. In the above example, the inspection of the forms should induce the

definition of more general relational concepts such as `carry(Spacecraft,Satellite,Location)`.

3. *Form classification*: the normalised linguistic forms are classified according to the general domain concepts.

4. *Instance Extraction*: the classified and normalised linguistic forms are used to extract spedific factual knowledge out from texts, as in Information Extraction (IE) perspective [8]. Linguistic assertions as ”*An Ariane 5 successfully launched Atlantic Bird 1 from Kourou*” and ”*Zarya was boosted into orbit by a Russian Proton rocket from the Baikonur Cosmodrome in Kazakstan*” have to be interpreted with respect to the formal language, i.e. formal assertions like `carry(Ariane_5, Atlantic_Bird_1, Kourou)` and `carry(Proton, Zarya, Baikonur_Cosmodrome)` have to be produced.

This sequence of activities mix the *knoweldge acquistion* phase (steps 1, 3, and 4) with the *knowlede modelling* phase (step 2).

In this paper we will therefore propose a method supporting the *knowledge acquisition* phase that takes advantage of well-assessed NLP techniques and well-assessed Machine Learning algorithms. The main idea is to carry the *corpus processing phase* in a ”terminology extraction perspective” that forces the definition of prototypical admissible forms and the notion of domain relevance (as described in Sec. 2). This could support *knowledge modelling* phases during the *concept formation phase* as the domain expert activity can be focussed on relevant bits of knowledge coming out from domain texts. Moreover, we will propose two feature-value vector models in order to investigate the usability of machine learning algorithms in the *form classification* phase (Sec. 3). Finally, we will empirically investigate our method over a financial domain (Sec. 4).

2. Knowledge acquisition through a terminologial approach

In order to acquire domain knowledge relying only on text collections, we shall process the corpus to extract relevant linguistic *forms*. We expect that the linguistic forms of *relevant* relational concepts could emerge from a possibly domain independent corpus analysis process. For what concerns this analysis, we assume that a relational concept is represented in verb phrases $r = (rv, (ra_1, ra_2, ..., ra_n))$ as ($boost$, (($subj$, `Spacecraft`), (obj,`Satellite`), ($pp(into)$, orbit)). Therefore, we here present an algorithm that, after the detection of *admissible surface forms* (i.e. linguistic ”prototypes” written at a syntactic interpretation level), produces a ranking according to their domain relevance (i.e. their frequency).

In the following sections, we will first define the equivalence among admissible surface forms while estimating the size of the search space of the ranking algorithm. Secondly, an efficient algorithm for the estimation of the importance function based on the frequency of the relations in the target corpus is presented in Sec. 2.2

2.1. Admissible surface forms: size of the problem

A relational concept may appear in a number of different contexts where verbs have some additional arguments. If the corpus C may be seen as a collection of verb contexts $c = (v, (a_1, a_2, ...a_n))$ where v is the governing verb and each argument a_i is a couple (g_i, c_i) representing its grammatical role g_i (e.g. subject, object, pp(for), pp(to), etc.) and the concept c_i semantically governing it, the problem is reduced to understand which are the more stable relationships established by each verb. Note that a context $c \in C$ is a positive example of the target relation $r \in R$ if $rv = v$ and r partially cover c, i.e. the arguments of r should then appear in any order in the context c.

An algorithm evaluating the relevance of all the possible relations $(rv, (ra_1, ra_2, ..., ra_n))$ works on huge search space. The number of different relations are obtained by partitioning the corpus C according to the verb governing the contexts. For each verb v, a subset of the corpus is then defined as $C(v) = \{(a_1, ..., a_n)|(v, (a_1, ..., a_n)) \in C\}$.

Defining $A_\Lambda(v)$ and $A_\Sigma(v)$ respectively as the possible lexicalised arguments and the possible syntactic arguments of a relation $r(v) \in R(v)$:

$$A_\Lambda(v) = \{a|\exists(a_1, ..., a_n) \in C(v) \wedge \exists i.a_i = a\} \qquad (1)$$

$$A_\Sigma(v) = \{ \quad \begin{array}{l}(s, object)|\exists i.g_i = s\wedge \\ \exists((g_1, c_1), ..., (g_n, c_n)) \in C(v)\}\end{array} \qquad (2)$$

the set $R(v)$ of the possible relation for the named v is the following $R(v) = \bigcup_{i=1...MC(v)} R_i(v)$ where $R_i(v)$ is the collection of all the possible combination without repetition of i objects extracted from the set $A(v) = A_\Lambda(v) \cup A_\Sigma(v)$. The distinction between lexicalised and syntactic arguments is useful to take into account the fact that some relations may have a recurrent argument whose surface concept is not recurrent. In these cases, a generalisation of the argument concept, i.e. $object$, is retained.

If $R(v)$ is the set of all the relations for the investigated verb v, the domain importance of each $r(v) \in R(v)$ should be assessed. Therefore, at least the evaluation of the frequency of the relation $r(v)$ over the corpus $C(v)$ has to be used.

Given the defined sets, the size of the $R(v)$ set is, in the worst case, the following:

$$|R(v)| = \sum_{i=1...MC(v)} \left(\begin{array}{c} |A(v)| + i - 1 \\ i \end{array} \right) \qquad (3)$$

where $MC(v)$ is the maximum context size for the verb v in $C(v)$. It is worth noticing that $|R(v)|$ values lie in a very large range, due to the size of $A(v)$. In the next section we will focus on a measure of relevance (for the target domain) that allows to systematically reduce the size of the space where pattern selection is applied for each verb v.

2.2. Estimating relational concept relavance

In order to tackle the inherent complexity due to the argument order freedom neglected in [14], we defined an informed exploration strategy relying on these observations: (1) the target of the analysis is to emphasize the more important relations arising from the domain corpus; (2) the frequency of a specific relation strictly depends on the frequency of a more general relation. A very simple but effective domain relevance estimator is the frequency of the relation over the corpus. Therefore, the above considerations may reduce the complexity of the search algorithm if only promising relation are explored, i.e. patterns whose generalisations are over a frequency threshold.

The idea is then to drive the analysis using the pattern generalisation that may be obtained projecting the patterns on their ”syntactic” counterpart. The projection $\widehat{\Sigma}(r)$ of the relation r over the syntactic space Σ is defined as follows:

$$\widehat{\Sigma}(r) = (\widehat{\Sigma}(ra_1), ..., \widehat{\Sigma}(ra_m))$$

where $\widehat{\Sigma}(ra_i) = ra_i$ if ra_i is a ”syntactic” argument ($ra_i \in A_\Sigma(v)$) or $\widehat{\Sigma}(ra_i) = (s_i, object)$ if $ra_i = (g_i, c_i)$ is a lexicalised argument ($ra_i \in A_\Lambda(v)$). The resulting search space $R_\Sigma(v) = \{\widehat{\Sigma}(r)|r \in R(v)\}$ is greatly smaller than $R_\Sigma(v)$ since $|A_\Lambda(v)| >> |A_\Sigma(v)| = \#preposition + 2$. This search space can be used for the extraction of the more promising generalised relations. This subset $\overline{R_\Sigma}$ can be used for narrowing the search space of the following step. In fact, when the acceptance threshold is settled, the resultant admissible relations are confined in the following set:

$$\overline{R}(v) = \{r|\widehat{\Sigma}(r) \in \overline{R_\Sigma}(v)\} \tag{4}$$

The overall domain importance estimation procedure may take also advantage from considering that the order of the relation arguments may be fixed after the analysis of the promising syntactic patterns. The final counting activity can be thus performed with a simple sorting algorithm of the $O(nlog(n))$ complexity.

3. Machine learning techniques for classifying linguistic forms

Retrieved relational concepts (*forms*) has to be organized in the ontological model, in order to allow an efficient retrieval procedure. Once the hierarchy of relational concepts is in place after the *concept formation phase*, the task of positioning the forms in the hierarchy may be seen as a classification process.

We will explore the possibility of a classification process carried out using ML techniques, applied to lexical and semantic information. The *feature-value vector* model underlying many ML algorithms suggests an observation space in which dimensions represent features of the object to be classified and dimension values are the values of the features as observed in the object. Each instance object is then a point in the feature space, i.e. if the feature space is $(F_1, ..., F_n)$ an instance I is:

$$I = (f_1, ..., f_n) \tag{5}$$

where each f_i is the value of the feature F_i for I.

Classifying linguistic forms with ML algorithms implies their translation in observable object. As we want to investigate the use of general purpose lexical semantic information such as WordNet [6], we propose here the notion of *semantic fingerprint* to introduce a conceptual hierarchy in a feature-value model. Hierarchies in the feature values are somehow in contrast with their expected *flatness*. To use this information, these hierarchies should be somehow reduced to a flat set SF where the problem of the inherent structure is simply forgot.

A word w (a verb or a noun) will leave its fingerprint $SF(w)$ on the set SF that represents all the active senses with respect to the chosen semantic interpretation catalogue SF. The semantic fingerprint of word w is:

$$SF(w) = \{s \in SF|s \text{ generalises } s' \text{ and } s' \in senses(w)\} \tag{6}$$

where $senses(w)$ are all the senses activated by the word w in the considered semantic resource. It will be the task of the machine learning algorithm the selection of the sense (or the senses) more promising for representing the investigated relationship. The algorithm will therefore also work as verb/noun sense disambiguator if the semantic information and the way we use it demonstrates to be useful.

Integrating the semantic fingerprint in the feature vector model is straightforward. Given an $S_i = \{true, false\}$ for each element in SF, the subpart of the feature space related to the semantic fingerprint is $S_1 \times ... \times S_n$ where n is the cardinality of SF. Each instance i containing the word w will have the feature value $s_j = true$ if $s_j \in SF(w)$ and $s_j = false$ otherwise.

With the semantic fingerprint abstraction we investigated two ”semantic” models against a ”bag-of-word” model. These are originated from the assumption that verbs play a relevant role in the problem under analysis. Then, the proposed models are:

$$\text{verb-gen: } V \times W_1 \times ... \times W_n \times VS_1 \times ... \times VS_k \tag{7}$$

$$\text{noun-gen: } V \times W_1 \times ... \times W_n \times NS_1 \times ... \times NS_m \quad (8)$$

where V ranges over all the possible verbs, $W_1 \times ... \times W_n$ represents the ”bag-of-word” approach collecting all the verb arguments, $VS_1 \times ... \times VS_k$ is the semantic fingerprint for the verbs, and, finally, $NS_1 \times ... \times NS_m$ is the semantic fingerprint for the nouns. The baseline model, that it is in itself a good model, is called *plain* and it collects verbs and the bag-of-word of the arguments (i.e. $V \times W_1 \times ... \times W_n$).

4. Experimental analysis

For both clarification and evaluation purposes we adopt a specific domain scenario (financial news) over which to analyse the performance of the knowledge modelling as well as th e retrieval task. We firstly prepared a relevant test set in order to clarify the final classification task. The manual tagging procedure and the results are presented in Sec. 4.1. Then, we have experimented our semantic-fingerprint-based models using well-assessed machine learning algorithms gathered in Weka [13]. It is worth noticing that the *cross-algorithm validation* can give hints on the relevance and the stability of the chosen feature spaces and on the correctness of the proposed model. The results of this investigation are reported in Sec. 4.2.

4.1. Test set preparation

In the test set preparation, our aim has been to have two different sources of information in order to cross check the results of the experiment. Given a catalogue C of relational concepts, we have produced:

- *classified forms*: a set of one-to-many associations between the concepts in C and the linguistic normalised forms
- *classified sentences*: a set of one-to-many associations between the concepts in C and sentences in the analysed corpus somehow related to the analysed linguistic forms

For the experiments, we used a corpus consisting of financial news, a text collection of around 12,000 news items published from the Financial Times in the period Oct./Dec. 2000. As described in Sec. 2, we, firstly, run a *corpus processing phase* selecting around 44,000 forms appearing more that 5 times. Secondly, in the *concept formation phase* a domain expert inspecting the top ranked forms defined 12 target relational concepts (see Tab. 1).

The *classification phase* has been performed by 2 human experts. They were given two separate set of normalised linguistic forms, two separate set of sentences extracted automatically from the corpus and a non-ambiguous definition of each class. The two experts were given respectively 3500 and 2200 forms to classify, taken from the first 6500 forms produced in the corpus processing phase. To evaluate the consistency between the classifications produced by the two experts, 300 of the given forms were in common, and over those forms the rater agreement was evaluated.

		Class	Forms	Sentences
1		RELATIONSHIPS AMONGS COMPANIES		
	1-1	Acquisition/Selling	157	619
	1-2	Cooperation/Splitting	96	471
2		INDUSTRIAL ACTIVITIES		
	2-1	Funding/Capital	12	86
	2-2	Company Assets (Financial Performances , Balances, Sheet Analysis)	166	1335
	2-3	Staff Movement (e.g. Management Succession)	70	355
3		GOVERNMENT ACTIVITIES	12	283
	3-1	Tax Reduction/Increase	3	40
	3-2	Anti-Trust Control		19
4		JOB MARKET - MASS EMPLOYMENT/UNEMPLOYMENT	7	50
5		COMPANY POSITIONING	4	
	5-1	Position vs Competitors	10	174
	5-2	Market Sector	10	369
	5-3	Market Strategies and plans	149	1512
6		STOCK MARKET	2	3
	6-1	Share Trends	319	1197
	6-2	Currency Trends	2	30

Table 1. The class (relational concepts) hierarchy of the financial domain, and corresponding forms and sentences distributions.

In case of doubt during the classification the expert could ask the system to show one or more sentences instance of the form, in order to gain enough information to classify the form itself. Annotators were also asked to classify all the shown sentences.

At the end of the phase, out of the normalised forms considered, 787 were retained as useful by the first expert, 298 by the second, i.e. the information carried in the words or in the named entity classes survived in the form has been considered sufficient to draw a conclusion on the classification. Moreover, the first expert classified 6609 sentences and the second 3550.

The two data sets, *classified forms* and *classified sentences*, have then been prepared. The first one consists of the 1091 forms obtained merging the two experts forms retained sets (for the 300 common forms, in case of disagreement the first expert class has been chosen). The second data set comprise the 6609 sentences classified by the first expert. The overall distribution of forms and sentences, for both the domain experts, is reported in Tab. 1.

Finally, the inter-annotation agreement has been computed to check the consistency of the data set. The model chosen to compute the agreement is the well known raw index $p_0 = \frac{1}{N} \sum_i n_i$ where N is the number of instances, and where n_i is 1 if the two experts classified the i-th instance in the same category and 0 otherwise. The agreement on the normalised forms is 90%, while the agreement on the sentences is 74%. These results show us a sufficient consistency over the data set, that can be thus considered a well defined gold standard for the experiments.

Method		*plain*	*verb-gen*	% inc/dec	*noun-gen*	% inc/dec
Trees	j48.J48	63,91%	63,68%	-0,23%	64,37%	+0,46%
	ID3	59,31%	59,31%	0	59,54%	+0,23%
	DecStump	26,44%	31,95%	+5,52%	26,44%	0%
Lazy	IB1	58,39%	63,22%	+4,83%	57,70%	-0,69%
	IBk	62,53%	65,98%	+3,45%	60,69%	-1,84%
Rules	j48.PART	59,77%	60,00%	+0,23%	63,22%	+3,45%
Bayes	NaiveBayes	53,33%	58,85%	+5,52%	40,23%	-13,10%
Misc	VFI	59,31%	57,24%	-2,07%	58,39%	-0,92%
	HyperPipes	60,92%	62,76%	+1,84%	62,07%	+1,15%

Table 2. Results on the set of *classified forms*, using a 5-fold cross-validation (baseline is 27%)

Method		*plain*	*verb-gen*	% inc/dec	*noun-gen*	% inc/dec
Trees	j48.J48	59,19%	64,80%	+5,61%	64,98%	+5,78%
Lazy	IBk	59,19%	54,72%	-4,47%	53,99%	-5,34%
Bayes	NaiveBayes	47,25%	54,03%	+6,78%	42,48%	-4,77%
Misc	VFI	43,81%	52,08%	+8,27%	51,84%	+8,03%
	HyperPipes	31,21%	42,56%	+11,35%	42,48%	+11,27%

Table 3. Results on *classified sentences*, using a 5-fold cross-validation (baseline is 40%)

4.2. Analysis of the results

The classification problem over the two different proposed data set has been therefore analysed with a pool of algorithms. We firstly analyse the results on the *classified forms* and then we check our intuitions on the *classified sentences*.

For the first set, the *classified forms*, results are reported in tab. 2. The baseline of the classification is around 27%, corresponding to a naive classification of all the instances in the more probable class (i.e. 6-1). All the algorithms report both in the lexical and the two lexical-semantic spaces better results with respect to the baseline, showing that the chosen features convey the right information for our classification problem. Moreover, the use of the semantic information seems to be relevant, as it emerges in the performance improvement obtained with the majority of the investigated algorithms using the semantic prints on both verbs and nouns.

In particular, the verb semantic generalization features seem to be particularly useful: the best performance for the vast majority of the tested algorithms is in fact achieved using the lexical-semantic verb space. Furthermore, the experiment overall best performance is obtained by the IBk algorithm working on this space.

In order to verify how the verb semantic information drives the classification, it can be interesting to examine the rules produced by a rule based algorithm, such j48.PART. This algorithm derives its rules from a pruned partial decision tree built using the C4.5 implementation [9]. One of these rules that involves semantic information, is the following:

$$\left.\begin{array}{l} price = no \wedge job = no \wedge \\ hire = no \wedge succeed = yes \wedge \\ entityNE = yes \end{array}\right\} \Longrightarrow \text{2-3} \tag{9}$$

That rule indicates that every sentence containing a verb of *succession* (i.e., a troponym, in the Wordnet sense, of the verb *succeed*) together with an *entityNE* (that is, a company or a person) has to be classified in class 2-3 (*staff movement* events). This semantic generalised rule, according to the Wordnet hierarchy, therefore classifies verbs of *succession* like *enter, supplant, replace, substitute*. Such a general rule can not be captured in a simple lexical space.

Analysing the results of tab. 2, the noun semantic generalization seems to be slightly less effective than the one on verbs. It is interesting to notice how in the tree obtained by j48 the noun semantic information is used. For instance, the presence in a form of a noun whose *base concept* (i.e. noun semantic generalization in EuroWordNet [12]) is *financial_obligation* is used to capture *”government activities: tax-reduction/increase”* events (class 3-1). In this way forms that contain nouns like *debt, rate, tax* are all classified in class 3-1. This simple rule has been very effective on our data set, classifying positive instance with 100% precision.

For the experiment on the *classified sentences* (tab. 3) we used a reduced pool of algorithm, representative of the different classification methodologies. In this case the baseline is around 40%. Similarly to the previous experiment, the results show a performance improvement using the verb and noun semantic information. In that case the improvement is even more sensible, thanks to the larger data set which emphasize the beneficial effect of the information carried by the used features. Looking at the decision trees produced by the j48 algorithm, it can be noticed that in the lexical space the verb lemmas are the most selective information, while in the lexical-semantic space the semantic verb generalisations and the noun generalizations and lemmas tend to discriminate over the data set more than the verb lemmas. Since the introduction of the semantic spaces improves the algorithm performance, it can be stressed again that this kind of information has an important discrimination power.

5. Conclusions

In this paper we introduced a knowledge based approach to improve development of the Semantic Grid, based on NLP and ML techniques and methodologies. Our approach is strongly based on the idea that an ontological organization of the knowledge and the use of terminological and semantic information automatically extracted from a domain corpus can support the development of a coherent and consistent Semantic Grid infrastructure. The explicit use we make of many-to-one mappings between linguistic forms and their corresponding meaning (i.e. relational concepts) is strengthened by its diffusion in other linguistic applications. Many researches are in fact devoted to propose methods for automatically building equivalence classes of patterns in fields such as Information Extraction [14, 11], Question Answering [10], Terminology Structuring [7], or Paraphrasing [1, 5]. As for all the methods, the use of some previous specific knowledge (not always available) seems mandatory, i.e. focused and structured templates plus examples in [14, 11], definitions and examples of the target relationships in [7, 10], and parallel corpora for [1], we tried to attack the problem from a different perspective.

Many issues are still open, firstly those related to the knowledge publishing (as described in [3]) and the development of a related usable tool. We will also address the problem of an automatic generation of relational concept classes from the corpus itself, using advanced clustering techniques.

In any case, we got a few indications that the proposed way to use semantic hierarchies and IE techniques may be helpful in the creation of an organized domain knowledge repository sharable among an heterogeneous community, as the experiment results show.

References

[1] R. Barzilay and K. McKeown. Extracting paraphrases from a parallel corpus. In *Proceedings of the 39th ACL Meeting*, Toulouse, France, 2001.

[2] B. M. J. N. R. de Roure, D. and N. Shadbolt. The evolution of the grid. In F. G. Berman, F. and A. J. G. Hey, editors, *Grid Computing - Making the Global Infrastructure a Reality*, pages 65–100. John Wiley and Sons Ltd., 2003.

[3] J. N. R. de Roure, D. and N. Shadbolt. The semantic grid: A future e-science infrastructure. In F. G. Berman, F. and A. J. G. Hey, editors, *Grid Computing - Making the Global Infrastructure a Reality*, pages 437–470. John Wiley and Sons Ltd., 2003.

[4] K. C. e. Foster, I. *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, 1998.

[5] N. Kaji, D. Kawahara, S. Kurohashi, and S. Sato. Verb paraphrase based on case frame alignment. In *Proceedings of the 40th ACL Meeting*, Philadelphia, Pennsilvania, 2002.

[6] G. A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, Nov. 1995.

[7] E. Morin. *Extraction de liens sémantiques entre termes à partir de corpus de textes techniques*. PhD thesis, Univesité de Nantes, Faculté des Sciences et de Techniques, 1999.

[8] M. T. Pazienza. *Information Extraction. A Multidisciplinary Approach to an Emerging Information Technology*. Number 1299 in LNAI. Springer-Verlag, Heidelberg, Germany.

[9] J. Quinlan. *C4:5:programs for Machine Learning*. Morgan Kaufmann, San Mateo, 1993.

[10] D. Ravichandran and E. Hovy. Learning surface text patterns for a question answering system. In *Proceedings of the 40th ACL Meeting*, Philadelphia, Pennsilvania, 2002.

[11] E. Riloff. Automatically generating extraction patterns from untagged text. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI-96)*, Portland, Oregon, 1996.

[12] P. Vossen. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers, Dordrecht, 1998.

[13] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, Chicago, IL, 1999.

[14] R. Yangarber. *Scenario Customization for Information Extraction*. PhD thesis, Courant Institute of Mathematical Sciences, New York University, 2001.