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 in an ontology 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 [5] 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 [3], 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 [3], 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 [4], 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.

The aim of our work is to fulfill some of the requirements needed for the development of the *knowledge layer* of the aforementioned system, assuming that the data inside the grid consist in a collection of unstructured natural language documents (*corpus*) concerning a domain of interest (for instance science, finance, news). Using well assessed Natural Language Processing (NLP) and Machine Learning (ML) techniques and methodologies, our subsystem will focus on the following specific processes, in the implementation perspective of a complete Semantic Grid tool:

Knowledge acquisition. The first task that the system could carry out is recognizing the information inside the data grid, and structure this information into a sharable knowledge, through a careful step of corpus analysis. As we assume unstructured documents, the goal is thus to process the corpus looking for domain specific ontological resources. We try to address this problem from an Information Extraction (IE) perspective ([9]), but, at the same time, trying to overcome the specific limitations that emerge from the application of a generic IE system in a knowledge acquisition task. IE tools, in fact, often assume that the type of information to be extracted from the corpus is narrow and known in advance, shaped by a user *template*. In our case, by contrast, we need to extract all the domain useful information held in the domain corpus (i.e. extraction is not driven by a specific information need). In such a perspective, the acquisition phase must totally and solely rely on the corpus itself, and the common IE techniques must therefore be enhanced with other methodologies: using a terminological approach to IE, our expectation is that relevant information for the specific do-

main could emerge from the corpus processing. In our approach this information takes the form of *syntactic patterns* (or *forms*), that is, functional relations among important concepts for the specific domain (i.e., verb phrases, that is verbs with all their arguments). For instance, in am astronautical domain, a relevant relation could be `launch(organizationNE,entityNE,dateNE)`, which indicates the launch by an organization of an entity in a certain date. A *form* is thus, for the sake of clarity, a normalized form of knowledge, which emerges form the generalization of different *linguistic forms* met in the corpus and that share the same syntactic structure (e.g. *"NASA will launch the Discovery in July 2005"* and *"Tuesday 2nd of March 2004 ESA has launched Ariane 5G"*). At the end of the acquisition phase information could be extracted and made accessible to other components of the Grid, turning therefore into knowledge.

Knowledge modelling and retrieval. Once the knowledge has been acquired from the data grid, it must be organized and modelled, in order to be used efficiently. The previous acquisition step, in facts, returns a collection of semantically unordered forms representing the domain knowledge. The modelling step is needed to engineer that knowledge in a semantically coherent hierarchical structure, that is, an *ontology*. For instance, in an astronautical domain it could be useful to gather under the same category all the forms concerning *satellites positioning* (forms like `launch(organizationNE,entityNE,dateNE)` and `orbit(entityNE,placeNE)`. From this point of view the ontology enriches the semantic content of the extracted information, and, more significantly, normalise the knowledge in a sharable form agreed by the community that accesses to the Grid. In this way the knowledge can be accessed consistently and unambiguously by different agents (humans, systems, tools) that share the common model. In our approach the ontology is built around a number of categories (*relational concepts*), in which each form extracted through the acquisition phase is classified, using ML techniques. At the end of this phase the corpus knowledge results organized and modelled in a clear semantic framework. The aim of this ontological model is to allow an efficient procedure of knowledge retrieval. One of the major problems in knowledge retrieval (as underlined in [4]) is, in fact, to select a specific needed knowledge inside the vast repository created during acquisition. Suppose, for instance, that a Grid user in an astronautical domain is looking for information about the position of all the satellites: he could obtain his desired bit of knowledge simply accessing the ontological model at the *satellites positioning* category, or questioning an automatic tool that translates his question as a relational concept request that gets identified in a particular ontological class.

normalised form	*relational conc.*
(subj,entity_ne) own (dirobj,percent_ne)	1-2
(subj,entity_ne) join (dirobj,entity_ne)	1-2
(subj,entity_ne) lose (dirobj,percent_ne)	6-1
(subj,share) fall (dirobj,percent_ne)	6-1

Table 1. An excerpt of classified admissible forms for a financial domain (relational concepts refers to the hierarchy in Tab. 2)

Knowledge maintenance. The proposed ontological model can be easily updated with novel information, supporting system scalability. In our framework, in facts, the insertion of new knowledge, in the form of new documents, can be easily performed repeating the abovementioned acquisition and modelling phase. Moreover, since the ontological classes don't change over the updating process, the model remains coherent and congruent to the original knowledge organization.

2. Knowledge acquisition through a terminologial approach

In order to acquire domain knowledge relying solely on the source of this information (i.e. the corpus), one of the more convenient ways is to analyse the corpus and to extract syntactic patterns (linguistic *forms*). We expect that the linguistic forms of *relevant* relational concepts could regularly emerge from a possibly domain independent corpus analysis process.

For this purpose, it is necessary to implement an algorithm able to detect *admissible surface forms* (i.e. linguistic "prototypes" written at a syntactic interpretation level) , to normalize them and to produce a ranking according to their domain relevance (i.e. their frequency). Each *normalised form* obtained in the ranking constitutes a syntactic pattern (see for an example Tab. 1).

In the following section (Sec. 2.1), the admissible surface forms and their equivalence are stated and the size of the problem is estimated. On the other hand, 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 relation $r = (rv, (ra_1, ra_2, ..., ra_n))$ (as those seen in the aforesaid examples) may be represented in a number of different surface forms. Due to the fact that the corpus should suggest the important relations, we will only

consider the realisation of r in verbal phrases. The corpus C s then 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. 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.

Given the domain corpus C represented as a collection of verb contexts, the objective is to evaluate the relevance of each possible relation $(r, (ra_1, ra_2, ..., ra_n))$. The first problem is to estimate how many different relations have to be analysed. This may be obtained after partitioning the corpus C according to the verb governing the contexts. For each verb v, a subset of the corpus is then defined as:

$$C(v) = \{(a_1, ..., a_n) | (v, (a_1, ..., a_n)) \in C\} \quad (1)$$

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

$$A_\Lambda(v) = \{a | \exists (a_1, ..., a_n) \in C(v) \wedge \exists i.a_i = a\} \quad (2)$$

$$A_\Sigma(v) = \{ \begin{array}{l} (s, object) | \exists i.g_i = s \wedge \\ \exists ((g_1, c_1), ..., (g_n, c_n)) \in C(v)\} \end{array} \quad (3)$$

the set $R(v)$ of the possible relation for the named v is the following:

$$R(v) = \bigcup_{i=1...MC(v)} R_i(v) \quad (4)$$

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:

$$|R(v)| = \sum_{i=1...MC(v)} \left(\begin{array}{c} |A(v)| + i - 1 \\ i \end{array} \right)^1 \quad (5)$$

1 Notice that, in syntactically meaningful contexts, arguments may appear with multiplicity higher than 1, so that the factorial expression is a useful approximation.

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 concentrate 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 the relations importance

Given the corpus C, the space of the possible relations is huge. This inherent complexity is the result of tackling the argument order freedom that is neglected in [16]. In order to tackle with the problem, an informed exploration strategy may be settled. This strategy can not take advantage on the biasing given by the awareness of the final information need that is typical of the IE pattern extraction algorithm. However, some observations may be useful for the purpose:

- the target of the analysis is to emphasize the more important relations arising from the domain corpus
- 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 represented by the frequency of the relation in 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:

$$\overline{R}(v) = \{r | \widehat{\Sigma}(r) \in \overline{R_\Sigma}(v)\} \quad (6)$$

The overall domain importance estimation procedure may take also advantage from the fact 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 with the $O(nlog(n))$ complexity. In this case n is directly related to the number of context samples in the corpus $C(v)$. The procedure is sketched in the following:

procedure SelectAndRankRelations($R(v)$,$C(v)$)
begin
Select $\overline{R_\Sigma}(v) = \{r \in R_\Sigma(v) | hits(r, C(v)) \geq K\}$;
Set $L = \emptyset$;
for each $r \in \overline{R_\Sigma}(v)$
$L := L \cup prj(C(v), r)$;
$RankedR(v)$:= CountEquals(L);
return $RankedR(v)$;
end

where $hits(r, C(v))$ is the number of instances of the relation r in $C(v)$ e $prj(C(v), r)$ is the projection of the contexts in $C(v)$ on the syntactic relation r. The procedure $CountEquals(L)$ using a standard sorting algorithm counts the repetition of each element in L. Finally, $RankedR(v)$ is the set of couples (f, r) where f the frequency of the relation $r \in \overline{R}(v)$ on the corpus.

3. Knowledge modelling and retieval using machine learning techniques

Once relational concepts (*forms*) have been retrieved from the corpus, this knowledge has to be organized in the ontological model, in order to allow an efficient retrieval procedure. We define this task as a classification process of the forms extracted in a structured framework of domain classes. The definition of these classes (*concept formation*) is left to domain experts and to a preceding phase of ontology definition performed by the Grid users, agreeing upon a common structure of the domain knowledge content. A possible class hierarchy for a financial domain is reported in Tab. 2.

The forms classification process is carried out using ML techniques, applied to lexical and semantic information related to the forms themselves. As ML model we use the *feature-value vector*. This model suggests an observation space in which dimensions represent features of the object we want to classify 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:

$$I = (f_1, ..., f_n) \tag{7}$$

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

In our framework, the classification information must be thus translated into features (*lexical-semantic model*).

For sake of clarity, before describing the lexical-semantic model adopted (Sec. 3.2), we will examine the limitations of the general feature-value vector model when used over models for natural language (Sec. 3.1).

3.1. Limitations of Feature-Value Vector in NLP

Exploiting the feature-value vector model and the related learning algorithms in NLP may be a very cumbersome problem mainly when the successful bag-of-word abstraction required [13] is abandoned for deeper language interpretation models. The *a-priori independence* among features, the *flatness* of the possible values, and the certainty of the observations are not very well suited for syntactical and semantic models. In fact, syntactical models would require the definition of relations among features in order to represent either constituents or dependencies among words. A semantic interpretation of the words (intended as their mapping in an is-a hierarchy such as WordNet [7]) would require the possibility of managing hierarchical value sets in which the substitution of a more specific node with a more general one can be undertaken as generalisation step. Finally, the ambiguity of the linguistic interpretations (either genuine or induced by the interpretation model) limits the basic assumption of the *certainty of the observations*; due to ambiguity, a given instance of a concept may be seen in the syntactic or semantic space as set of alternative observations. The limits of the linguistic interpretation models in selecting the best interpretation requires specific solutions to integrate the *uncertainty* in the feature-value vector.

3.2. Lexical-semantic model for forms classification

The lexical-semantic model proposed for the classification task includes as lexical feature all the verb arguments of the forms appearing in the corpus.

The definition on semantic features is a more cumbersome problem. We propose here the notion of *semantic fingerprint*: we try here to investigate the possibility of integrating some sort of ”semantic” generalisation for verbs and nouns in the normalised forms. Verbs semantically govern the verbal phrases taken as forms admissible for the relationships and may give an important input to cluster prototype forms in classes. For instance, let us take the patterns in Tab. 1 and suppose that the first three lines have already been met, i.e. these can be considered training examples. The new instance could be assigned both to class 6-1 and to the class 1-2 as it has one common feature with all the considered known instances. The only possibility of classifying the new instance in one of the two classes relies on some sort of generalisation: the verb seems to be a very good candidate. According to WordNet *lose* and *fall* have

two common ancestors *change* and *move-displace*, while it does not happen for *fall* and *join* or *fall* and *own*. The use of such kind of knowledge seems therefore to be helpful for the classification task as in [2, 6], where noun conceptual hierarchies have been exploited using the definition of distance measures among nodes.

The introduction of a conceptual hierarchy is somehow in contrast with what has been above called the *flatness* of the feature values. If we want 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. One possibility is choosing one level of generalisation and reducing each element to this level. This is the one we adopt in our model for the exploitation of conceptual hierarchies in the problem of detecting equivalent surface forms. As we will see, this choice helps to solve the issue of the ambiguity of the sense assignment: verb and noun senses should be determined in the domain defined by the text collection. The ambiguity should then be modelled in the images of the pattern prototypes in the feature space. It is as if we model uncertainty in the observations of concept instances. However, features can not have multiple values. 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:

$$SF(w) = \{s \in SF | s \text{ generalises } s' \text{ and } s' \in senses(w)\} \quad (8)$$

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:

$$\text{verb-gen: } V \times W_1 \times ... \times W_n \times VS_1 \times ... \times VS_k \quad (9)$$

$$\text{noun-gen: } V \times W_1 \times ... \times W_n \times NS_1 \times ... \times NS_m \quad (10)$$

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

To clarify the methodology proposed in the previous sections and to evaluate the performance of our system, we analise in this section the performance of the knowledge modelling and retrieval task over a specific domain scenario (financial news). We then 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 [15]. 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. 2).

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

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

the two experts, 300 of the given forms were in common, and over those forms the rater agreement was calculated.

In case of indecision 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. 2.

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.

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. 3. 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 [10]. One of these rules that involves semantic information, is the following:

$$\left.\begin{array}{l} price = no \wedge job = no \wedge \\ hire = no \wedge succeed = yes \wedge \\ entityNE = yes \end{array}\right\} \Longrightarrow \text{2-3} \qquad (11)$$

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. 3, 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 [14]) 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.

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

For the experiment on the *classified sentences* (tab. 4) 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 discriminational 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 ontologial 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 [16, 12], Question Answering [11], Terminology Structuring [8], or Paraphrasing [1, 6]. 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 [16, 12], definitions and examples of the target relationships in [8, 11], 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 [4] 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] R. Basili, M. T. Pazienza, and M. Vindigni. Corpus-driven learning of event recognition rules. In *Proceedings of Workshop on Machine Learning for Information Extraction, held in conjunction with the 14th European Conference on Artificial Intelligence (ECAI)*, Berlin, Germany, 2000.

[3] 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.

[4] 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.

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

[6] 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.

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

[8] 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.

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

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

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

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

[13] G. Salton. *Automatic text processing: the transformation, analysis and retrieval of information by computer*. Addison-Wesley, 1989.

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

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

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