Language Learning and Ontology Engineering: an integrated model for the Semantic Web

R. Basili, M. Pennacchiotti, F.M. Zanzotto
University of Rome Tor Vergata,
Department of Computer Science, Systems and Production,
00133 Roma (Italy),
{basili,pennacchiotti,zanzotto}@info.uniroma2.it

Abstract

Automatic acquisition of semantic knowledge via text mining activities is widely recognized as a support to ontology engineering for intelligent applications ((Maedche and Staab, 2000; Hahn and Schnattinger, 1998; Basili et al., 2003)). However, poor attention has been paid to the integration between the inductive process and the target intelligent task. In this paper a model integrating lexical semantic information as required by a corpus-driven natural language learning perspective and a domain ontology is proposed. An OWL ontology built to represent the domain of a legal application (financial fraud detection as targeted in FF-Poirot (Zhao and Verlinden, 2003)) will be used as a running example of the approach.

1 Motivation

Automatic acquisition of semantic knowledge via text mining activities is widely recognized as a relevant approach to ontology engineering ((Maedche and Staab, 2000; Hahn and Schnattinger, 1998; Basili et al., 2003)). We can emphasize at least three different lexical acquisition tasks useful to ontology design and development: domain specific terminology extraction, acquisition of selectional preferences and subcategorization information ((Jacquemin, 1997)) and machine learning of information extraction rules (Riloff, 1996; Yangarber, 2001; Basili et al., 2000). In all these tasks, important components of the domain ontology are seen as the outcome of an inductive process based on source text collections.

It is nowadays, widely accepted that linguistic information is relevant in ontology engineering at least for two main reasons. First, ontologies are data models whose concepts are identified by names and processing such information under a linguistic perspective is helpful in the understanding of the often implicit semantics imposed by the knowledge engineer. Harmonisation of domain ontologies throughout the use of linguistic resources has been for example proposed in (Magnini et al., 2002).

Second, although ontologies capture aspects of the semantics that are somewhat independent from linguistic information, most text processing applications (e.g. knowledge-based IR or question-answering) require explicit mapping between domain concepts and their textual counterparts: an example is given by the terminology (e.g. multiword expressions) that embody linguistic variants of an often complex concept in the ontology. Other examples are event matching rules in information extraction systems. Although targeted to specific event types, IE systems must be aware of the different ways the events are linguistically expressed: which verbs and which concepts are used to communicate a given event? This form of lexical semantic information is strictly part of the ontological description especially with respect to paradigmatic properties. An event type e is a specific concept and when inheritance is required it may be connected with topological properties in a hierarchy. However, such semantic dimensions are independent from the linguistic properties (e.g.the rules needed for detecting potential realisations of e in textual material). Linguistic rules should include a combination of syntagmatic S and semantic M constraints that a text t must satisfy in order to realise e. In other words when $S(t) \wedge M(t)$ is a true formula then we can state that t realises e, i.e.

$$\forall t. S(t) \wedge M(t) \Rightarrow e(t) \qquad (1)$$

Rules like the above one are needed for IE over data sets of a realistic size, although they are usually not represented ontologically. Examples of properties S or M are for example distributional properties (e.g. mutual information of word collocations in corpora) able to suggest a concept (e.g. a terminology item that represents a concept in the domain) or a relation. Machine learning techniques are widely used to observe these properties and inductively develop the required concepts/relations. However, once learned they are usually mapped into the target KB throughout validation (often manual, as in (Bozsak et al., 2002)) that determine the (new) topological properties and their implicatures. As a consequence the *textual* semantics of a concept or a relation is not preserved in the target ontology. Syntagmatic and semantic properties are used to justify the eligibility of a given lexical (structure) as an ontological concept but are then neglected. The textual properties that were used to justify an inductive decision (e.g. a given fragment is a terminological expression, a given sequence/syntagmatic structure is a prototypical rule for the relation or event type e), are not associated (after the decision is taken) to the resulting concept or relation type. Although such properties in principle depend on the underlying domain ontology, they are never explicitly represented. Rare exceptions are works where integration between world and textual semantics for text understanding is adopted as in (Hahn and Schnattinger, 1998; Hahn and Mark, 2002).

Although standard practice, the tight separation of the ontology from the bundle of linguistic properties emerging from text mining activities is a major drawback. First, all the beneficial ontological properties are not available *during learning*. For example, subsumption relations valid within the ontology IS_A hierarchy are not available for justifying common semantic properties of texts. Instances cannot be clustered according to some of their subsumers so that equivalence of different textual fragments cannot be established according to ontological properties. This weakens the generalisation ability of the learning subsystem.

Second, once concepts (or relations) are added to the ontology, as they have been observed in training texts, no linguistic information (e.g. phrase structure rules or selectional constraints over predicate argument structures) is made available *within the ontology* in order to match them in incoming new texts. It is trivially evident that this option is not possible within any knowledge-based IE application. IE systems need a tight integration between domain and language knowledge. Even non-pure IE systems (e.g. knowledge management system whose target is not mainly IE) suffer from the above limitation. The lack of this information in fact makes the learning phase an *all-in-once* process with no possibility for incrementality. The dynamics of knowledge in most applications scenarios is such that a static modeling is undesirable (or even impossible).

1.1 Induction and deduction in knowledge-based IE

Eq. (1) expresses the typical inference triggered *during* an IE process: an incoming text t is analysed and, by the verification of properties at syntagmatic and semantic level, an event type is deduced and correspondingly assigned to t. In general, this deductive approach requires:

- a system of event E possibly organized into an hierarchy and a corresponding system of entities (a universe of discourse) with its own internal organization (e.g. again an hierarchy)
- a system of syntactic-semantic constraints (i.e. S and M constraints in eq. (1)) as a language over textual phenomena
- a large set of rules like (1) able to define the overall mapping between textual phenomena (i.e. the left hand side of the rule) and the corresponding domain specific events.

If we look at the way several systems learn event matching rules like Eq. (1) we would find a different perspective. In order to derive (semi-automatically) event types and event rules from texts ((Riloff, 1996; Yangarber, 2001)) a cascade of steps in generally undertaken by different authors:

- (*Cluster*) First texts and fragments of them are clustered according to surface or distributional properties[1]
- (*Linguistic Extraction*) Then, some syntactic and semantic properties S and M are observed in clusters. Examples of properties are the occurrence of the same verb argument structure or the same word sense(s) in all the sentences of a cluster.

[1] Example of clusters are all the sentences where variants of a terminological expression appear or paragraphs that include uses of similar (or synonym) verbs. A distinctive factor of the different approaches is given by the level of supervision related to the available source data: sometimes clusters derive from human annotations, e.g., all sentences giving rise to the same template type, while in most cases clusters are built automatically.

- (*Induction*) Then, an inductive process takes place. When, within each cluster, a significant (i.e. large enough) set of sentences share the same properties S and M then it must exist an underlying event e relevant for the domain for which S and M should be (at least) the necessary conditions. More formally, given an underlying ontology $\mathcal{O}$, a system of syntagmatic and semantic constraints $\mathcal{L}$ and a cluster C, then

$$\begin{gathered}\text{If for some properties } S, M \text{ and} \\ |T_e| \triangleq |\{t \in C | (\mathcal{O}, \mathcal{L}) \models S(t) \wedge M(t)\}| > n \\ \text{then an event } e \text{ exists such that} \\ \forall t \in T_e \rightarrow e(t) \quad (2)\end{gathered}$$

  Here, the process work inductively. The existence of a domain specific event e and the ways of its textual realization are stated when enough texts show (at least) the same syntagmatic and semantic properties S and M (i.e. n is a relatively large integer).

- (*Validation*) Human validation is applied to the S and M properties characterizing a rule so that they become sufficient conditions for events as in Eq. (1).

It should be now noticed that almost all the above steps (in particular, clustering and extraction) makes a critical use of semantic information. Clustering is often based on similarity at the syntactic (e.g. same structures/sequences) and lexical (e.g. same verbal heads) level. However, semantic similarity is often employed to establish equivalence among texts through generalization (i.e. using shared hypernims or *highly similar* synsets instead of lexicals). Textual entailment is a recently explored area where semantic similarity among words and structures should be used to proof implications among sentences. Shared lexical, syntactic and semantic properties are then extracted from texts in a cluster as those properties that can be factored out of individual texts.

The above processes can be critically influenced by the availability of large repositories of semantic information as more interesting training phenomena can be generated. The availability of a domain ontology in the early design phases of an IE system (the engineering process of adapting its knowledge to a new task, i.e. template, or domain). For example, more specific systems of semantic classes may result in a better clustering with emphasis on more appropriate textual phenomena.

1.2 Reconciling Ontology based Induction and Deduction

The previous sections outlined the need for specific semantic information either in the standard IE task (Eq. 1) or during automatic learning of some components of the underlying KBs. The real aim here is thus to design a framework where the ontology is made available even in the early phases of the ontology engineering process itself: an incremental process here should be thus adopted where NL learning interleaves with ontology engineering.

In this scenario we need to make available in principle to all the components in the architecture at least the following semantic components:

- a set of domain concepts and relations mainly devoted (as in the traditional view on the ontology) to define properties of individuals, relations and typical task involved in the application process
- a language component including lexical semantic information (e.g. word sense and thematic descriptions for specific classes like verbs) structured according to linguistic methods and principles and modeled independently from the domain knowledge
- a mapping between linguistic and domain concepts usually not captured by concept linguistic labels as naming assumptions may vary hugely across domains and applications
- a systematic mapping between domain relations (e.g. properties of individuals or binary relationships like `professor(s) teach course(s)`) and their linguistic counterparts (e.g. terminological expressions or predicate argument structures of verbs denoting those properties). Notice how the distinctions between the latter linguistic rules and the former relationships reflects the traditional separation in linguistics between grammatical functions and thematic roles.

Although ontology merging methods as well as meaning negotiation are usually devoted to define a bridge between linguistic resources and ontologies, it is important to emphasize here that in no way it is possible to use these two notions interchangeably. Every domain concept is justified by assumptions meaningful to the application task, e.g. a topical category. Every lexical sense exists to denote a linguistic notion and is meaningful within the underlying linguistic community. Although sometimes equivalence between a linguistic item and a domain concept can be found this is in general accidental.

Although in the discussion above we treat separately the domain and the lexicon, a variety of common properties hold for these two semantic components. For instance, subsumption hierarchies are useful devices largely applied not only to ontology but also as lexical models (e.g. Wordnet). For these reasons a common lexical formalism, i.e. DL, is a suitable framework to integrate all the semantic components useful to an intelligent application and to support an incremental approach to language learning and ontology engineering.

2 The Ontological Model

The ontological model must be able to handle and *link* domain knowledge to lexical-syntactic knowledge, as expressed at least by word senses, terms and verb predicates. The former (called hereafter *Domain Ontology*) conceptually models the domain in terms of domain concepts, as classes of individuals, and semantic relations among them. For example, in a commerce domain the concepts *company* and *product* may be linked through a relation like *buy*. On the other end, the lexical-syntactic level (hereafter *Linguistic Knowledge Base*) should at least include lexical representation of domain concepts as terminological entries, word senses and verb predicates. Terms can be given a priori or derived by corpus analysis. Word senses can be given by a lexicon (e.g. Wordnet) or being the outcome of a learning process over a domain independent resource: the status of the sense dictionary in LKB depends on the stage in which the development cycle of the application is. No special difference exist in terms of the formalism used to model it. Finally a specific component of LKB define word usages, their grammatical projections and the corresponding (expected) argument semantics.

Concepts can be naturally linked to word senses while semantic relations are mapped to (verb) predicates. The mapping between domain concepts and their lexical counterparts is straightforward, while the link between semantic relations and verbal predicates envisions a complex mechanism: verbal arguments must be mapped into semantic arguments by preserving selectional restrictions and syntactic constraints on the arguments themselves. The proposed ontology thus includes a specific *Syntactic-semantic Mapping* hierarchy devoted to represent explicitly every link between the two types of relations.

The proposed ontology models the following main types of knowledge:

1. A *Domain Ontology* (*DCH*), that explicitly includes *Concepts*, *Named Entities* and *Semantic Relations* among Concepts

2. A *Lexical Knowledge Base* (*LKB*) including: a sub-hierarchy of *lexical senses* and an hierarchy of *Verbal Predicates*[2]. The first is inspired by the WordNet and makes use of a consistent subset of the hyponymy hierarchy, the Wordnet Base Concepts (*WNBC*) ((Vossen, 1998)). The second encodes linguistic properties that characterize textual phenomena as syntactic-semantic rules. In line with neo-Davidsonian perspectives on verb lexical semantics, we model (classes of) events through conjunctive combinations of constraints over verb syntax and selectional restrictions (similarly to Eq. (1))

3. A linking between the Relations represented in the Domain Ontology and the predicate rules in the *Verbal Predicates* hierarchy in LKB. Events, as special cases of relations, will be thus mapped to the set of verbal rules (e.g. semantically restricted predicate argument structures). This information is modeled into a separate hierarchy called *SyntSemMapping*.

4. Complementary information is also made available in a separate and independent hierarchy. For example the variety of linguistic types are defined formally in an independent hierarchy called *Reference linguistic dictionaries*. Every catalogue of syntagmatic (e.g. classes of verb arguments), grammatical types (e.g. POS tags), as well as their useful properties (like the degree of obliqueness of arguments) are here represented.

In the following subsections a brief description of each sub-hierarchy is presented.

[2] A more precise name would in fact be *Lexical predicates* as most nouns can be modeled here alike the verbs.

2.1 Defining Semantic Information

A specific component of an ontology is the `SemanicObject` hierarchy. It includes the *DCH* (i.e. Concepts and Semantic Relations) and the *LKB* (i.e. WordNet, Terminological networks and Verbal Predicates). Roughly, it represents the semantic core of the ontology.

2.1.1 The DomainOntology: Concepts

The *DCH* hierarchy contains the pure semantic level of the ontology where the application specific Concepts and Semantic Relations are defined. In this hierarchy also Named Entities are defined in a independent subhierarchy.

Each concept is represented by a set of specific properties (i.e. slots and arguments) and their coding is supposed to involve only domain experts. Often this component preexists to an application. An instance here is a term, that is the literal occurrence of a concept (name) in texts. Automatic terminology extraction (Basili et al., 2002) aims to upload this portion of an ontology and to establish a domain specific dictionary.

2.1.2 The DomainOntology: Named_Entities

A flat Named Entities (NE) hierarchy has been implemented to increase modularity. It aims to capture a general semantic property for applications: a specific subset of conceptual entities, e.g. organizations, refers unambiguously to (classes of) individuals in the domain (e.g. companies in the real world). NEs, as well as the other *DCH*s, are useful restrictions for the selectional constraints characterizing Verbal Predicates. Classes currently represented in the ontological model are `Persons`, `Companies`, `Locations`, `Currency`, `Base_Organizations`, `Governmental organizations` and `Numbers`.

2.1.3 The DomainOntology: Relations

Semantic Relations define the useful (typed) relationships required by a given application. Relations usually define what is often expressed linguistically in terms of complex verb predicates. In this view, instances of the Semantic Relations types are ground verbal predicates, i.e. predicates whose main argument slots have been all filled with some textual material. In an *e*-commerce application, for example, a typical Semantic Relation is *Selling* and it involves concepts like *legal entities* (companies and persons), *products*, *money* and so on. Major properties of the domain Relations are *Semantic Roles*, usually employed to characterize the concepts participating (i.e. that act as slot fillers). Semantic Roles are thus role labels for the Concepts involved in a relation. As semantic relations determine the specific concepts allowed as fillers to r, legal (i.e. allowed) values for the Role slot are ontological concepts, i.e. semantic restrictions to the individuals suitable for Role fillers. In this way, Selectional Restrictions are implemented as type restrictions on Role fillers. For example, a typical Semantic Role for the Selling relation is *Buyer*: its slot filler could be the `legal entities` concept i n*DCH*. Also *Good* and *Money* are roles with type restrictions as v`products/shares` and `money` respectively. More formally, using a Description Logic formalism the *Selling* relation can be defined as follows:

$$
\begin{aligned}
&\texttt{Selling} \equiv (\exists\ \texttt{hasBuyer.Legal\_Entity}) \\
&\sqcap\ (\exists\ \texttt{hasDonor.Legal\_Entity}) \\
&\sqcap\ (\exists\ \texttt{hasGood.(Share} \sqcup \texttt{Product))} \\
&\sqcap\ (\exists\ \texttt{hasMoney.Money})
\end{aligned}
$$

The above example is not accidental. A Semantic Relations in our ontological model has a frame-like semantics. The resemblance with the notion of *Frame*, as used within the FrameNet project (Baker et al., 1998) is strong. Semantic Role here corresponds to *Frame Elements*. Notice that, like in Framenet, Relations do not postulate syntagmatic constraints and are relatively independent from grammatical arguments. Not accidentally, in our ontology, they are intended as domain facts (i.e. they are defined within the *DCH*). Verbal Predicates (in the LKB hierarchy) are in charge of defining grammatical constraints, that are in fact linguistic properties.

The current Semantic Role dictionary has been partially derived from FrameNet II project. Semantic Relations adhere to the FrameNet schema that is, Semantic Roles in our ontology have a one-to-one correspondence with Frame Elements. The semantic roles in the *Selling* Semantic Relation (i.e. *Buyer*, *Money*, *Good*, *Recipient* and *Donor*) correspond to the Frame Elements of the *Commerce.buy* frame.

2.2 Modeling the linguistic knowledge: LKB

The Linguistic Knowledge Base (*LKB*) refers to the lexical-syntactic level of the ontology. It thus contains the counterpart of concepts and semantic relations.

2.2.1 LKB: the Noun hierarchy

EuroWordNet (Vossen, 1998) has been used as the underlying lexical framework. EuroWordNet is a multilingual lexical knowledge (KB) base with wordnets for the different languages. The wordnets are structured in the same way as the English WordNet, developed at Princeton. In the LKB component the about 1,000 synsets that correspond to EuroWordNet Base Concepts (WNBC) (Vossen, 1998) are represented together with their hyponymy/hyperonimy relationships. As each wordnet represents a unique language-internal system of lexicalisations and they are linked to an Inter-Lingual-Index (ILI), based on the Princeton WordNet (WN), the LKB supports also lexical inferences for several languages.

The noun hierarchy in LKB is intended as the lexical component that is not possible to consider as ontological as (1) it may be not relevant for the domain or (2) there is not much agreement about its members. In other words, linguistic information is used as a back-up for every inference where the consensus about semantics of terms is weak or absent. This implies that the link between synsets and domain concepts cannot fully preserve the ontology semantics (e.g. being isomorphic wrt subsumption). However, a form of reference (e.g. a many-to-many mapping) is still useful to support weaker inferences (e.g. make hypothesis about textual content even when lack of information in the ontology prevent the full understanding). A similarity metrics based on Wordnet ((Basili et al., 2004)) has been used to map automatically domain concepts to LKB nouns and its evaluation over a medical domain hierarchy is discussed in (Basili et al., 2003).

The result is that for each concept of the Domain Ontology a specific (multivalued) role expresses all its possible lexicalisations, in form of Wordnet synsets. For example the `Product` is mapped to synsets *food_product, artefact* and *commodity* in the LKB.

2.2.2 LKB: the VerbalPredicates hierarchy

Verbal Predicate Patterns define relevant verb relations among domain concepts. As lexicalisations are also represented in the ontology (i.e. in LKB), relations can also be established among synsets.

A VerbalPredicate is the generalisation of relational linguistic properties (e.g. sentence schemata). For example an event matching rule can be seen as the generalization of syntactic and semantic properties common to a cluster of text fragments. It always refers to a unique Semantic Relation. On the contrary, Semantic Relations may be linked to more than one Verb Predicate. For example the above mentioned Semantic Relation *Selling* has the following as corresponding Verbal Predicates:

purchase_22:
`(SUBJ(legal_entity)` *`purchase`* `OBJ(share))`
and
sell_11:
`(SUBJ(legal_entity)` *`sell`* `OBJ(product))`

In the perspective of Eq. 1, the above rules establish the properties S and M needed to infer the *Selling* Relation.

VerbalPredicates are automatically extracted from a domain corpus (Basili et al., 2000) uploaded in the hierarchy via a semi-automatic process and then linked by the knowledge engineer to a suitable Semantic Relation. Because they are usually derived via linguistic induction they use to fix semantic properties in terms of synset based selectional constraints. The properties of a Verb Predicate are thus the heading verb, a Weight expressing domain relevance of the rule, and the linguistic constraints: verb arguments and legal filler as synsets in LKB, NEs or domain concepts. In Description Logic a generic Verbal Predicate can be defined as follows:

$$\texttt{VerbalPredicate} \equiv$$
$$(\exists\ \texttt{hasVerb.Verb}) \sqcap (\exists\ \texttt{hasWeight.Integer})$$

The hierarchy is then basically organized on the basis of the syntactic properties of verbs so that classes contains verbs with the same syntactic properties, expressed by the appropriate roles. Each role indicates a specific Syntactic Role (`Subject`, `Object`, etc.) that can be filled consistently only by some domain concepts or Named Entities (in DCH) or eventually by WordNet synsets (in LKB). For example the class *TransitivePredicates* has the properties *Subject* and *Object*. In Description Logic:

$$\texttt{TransitiveVerb} \equiv \texttt{VerbalPredicate}$$
$$\sqcap\ (\exists\ \texttt{SemSubj.(Concepts} \sqcup \texttt{Named\_Entities} \sqcup \texttt{WordNet))}$$
$$\sqcap\ (\exists\ \texttt{SemObj.(Concepts} \sqcup \texttt{Named\_Entities} \sqcup \texttt{WordNet))}$$

Increasing level of depth indicate thus more complex syntactic verbal constructions.

Deeper levels in the VerbalPredicates hierarchy account for verb semantics. Selectional Restrictions are applied to each (valid) Syntactic Role and fillers are semantically restricted (by using OWL assertions) to a certain class of concepts.

In synthesis each VerbalPredicates class represents those verbs with shared syntactic-semantic properties, i.e. VerbalPredicate patterns with the same arguments filled by instances of the same concept type. As an example, the *purchase_22* class (under the transitive verb subhierarchy) represents those patterns where the Object exists and is restricted to be a `share` concept as well as the Subject as a `legal_entity` in *DCH*. In Description Logic notation the above propoerties is stated as:

$$\texttt{Purchase\_22} \equiv \texttt{TransitiveVerb} \sqcap (\exists\ \texttt{SemSubj.Legal\_Entity}) \sqcap (\exists\ \texttt{SemObj.Share})$$

2.3 The SyntSemMapping Hierarchy

Although they express independent information aiming to support linguistic and conceptual inferences during *inductive development* (Eq. 2) and *use* (Eq. 1) of the ontology, they must be linked. In fact, each Verbal Predicate can be intended as a semantically-restricted syntactic pattern conveying sufficient information to infer the relational information of a unique Semantic Relation. The aim of the sub-hierarchy is to model this mapping.

Instances in this hierarchy represent individual links between a Verbal Pattern and a Semantic Relation (*SemanticStructure* and *SyntacticStructure* properties). Each mapping define for every syntactic role of the involved Verbal Patterns the corresponding Role of the Semantic Relation.

For example, the mapping between the *Selling* Relation and the Verbal Predicate *purchase_22* (Figure 1) foresees a Semantic Role of *buyer_role* for the Subject of the Verbal Predicate. The Object has the correct assignment into the *good_role.* When a different verb predicate is mapped, e.g. *sell_11*, its Subject is mapped in the *donor_role* Role and its Object in the *good_role.* Notice that Semantic Roles are mapped into Syntactic Roles with the compatible (identical in these cases) Selectional Restrictions.

Figure 1: An example of mapping between Semantic Relation and Verbal Pattern

The mapping implementation is declarative version of coindexing rules.

2.4 Implementing the ontological model

The ontology has been written in the OWL language, the variant of Description Logic widely acknowledged as a *de-facto standard* for modelling ontologies. As a developing and managing graphical interface has been used the Java based tool Protege (Grosso et al., 1999). This improves interoperability, intuitiveness and support to a wide range of representation languages of our proposal. Moreover, Protege supports the integration of dedicated inference engines, such as *Racer* (Haarslev and Moller, 2001), that support manteinance of high quality and coherent ontologies, by powerful deductive reasoning and automatic consistency checking. Dedicated APIs make Protege also ideal for its support to development of applications and extensions.

3 Conclusive Remarks

In this paper we present an integrated model for linguistic and ontological knowledge that supports (semi-)automatic ontology engineering as well as its applications. The major advantages of the integrated ontological model are:

- (*Modularity*): learning to extract is inherently decoupled from learning domain knowledge as different sub hierarchies are targeted
- (*Logical harmonisation*): the learning (sub)system and the application use the same reasoning device (i.e. the logical formalism underlying the ontology)
- (*Scalability*): support to the design of large scale resources as a weakly supervised process of text mining (validation is applied instead of development *from scratch*)

Although in use for large scale projects and applied to different domains, the presented model is preliminary especially for what concern evaluation. Two forms of evaluation will be undertaken. First, we will check the metalogical consistency of our approach by trying to switch from one lexical resource to another (e.g. from Wordnet to Longman Dictionary) for what concern the Linguistic component. The amount of reusable domain knowledge that can be still expressed over the new lexical resource (e.g. role restrictions for semantic relations) will be a measure of the plausibility of our modelling. If plausibility keeps high for many different domains then the proposed model is characterized by a general validity. Second, we need to test the quality of the modelled knowledge under a task specific perspective. Poirot (Zhao and Verlinden, 2003) will be thus used as a test bed: the accuracy of the fraud detection process will be measured both in absence (baseline system) and presence of the underlying ontology model. The accuracy rate will provide a direct evidence of the usefulness of the above ontological model.

References

Collin F. Baker, Charles J. Fillmore, and John B.Lowe. 1998. The berkeley framenet project. In *In Proceedings of the COLING-ACL*, Montreal, Canada.

R. Basili, A. Moschitti, and M.T. Pazienza. 2000. Language sensitive text classification. In *In proceeding of 6th RIAO Conference (RIAO 2000), Content-Based Multimedia Information Access, Coll ge de France*, Paris, France.

R. Basili, M.T. Pazienza, and F. Zanzotto. 2002. Acquisition of domain conceptual dictionaries via decision tree learning. In *Proceedings of the "15th European Conference on Artificial Intelligence (ECAI 2002)*, Lyon, France.

Roberto Basili, Michele Vindigni, and Fabio Massimo Zanzotto. 2003. Integrating ontological and linguistic knowledge for conceptual information extraction. In *Web Intelligence*, Halifax, Canada.

Roberto Basili, Marco Cammisa, and Fabio Massimo Zanzotto. 2004. A semantic similarity measure for unsupervised semantic tagging. In *Proceedings of 4th International Conference on Language Resources and Evaluation (LREC2004)*, Lisbon, Portugal.

E. Bozsak, M. Ehrig, and S. Handschub. 2002. Kaon – towards a large scale semantic web. In A.; Quirchmayr G. Bauknecht, K.; Min Tjoa, editor, *Proc. of the 3rd Intl. Conf. on E-Commerce and Web Technologies (EC-Web 2002)*.

W. E. Grosso, H. Eriksson, R. W. Fergerson, J. H. Gennari, S. W. Tu, and M. A. Musen. 1999. Knowledge modeling at the millennium (the design and evolution of protege-2000.

V. Haarslev and R. Moller. 2001. Description of the racer system and its applications. In *DL2001 Workshop on Description Logics*, Stanford, CA.

Udo Hahn and K. G. Mark. 2002. An integrated, dual learner for grammars and ontologies. *Data & Knowledge Engineering*, 42(3):273–291.

Udo Hahn and K. Schnattinger. 1998. A text understander that learns. In *COLING-ACL '98 – Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics & 17th International Conference on Computational Linguistics*, Montreal, CANADA. Morgan Kaufmann.

Christian Jacquemin. 1997. Variation terminologique : reconnaissance te acquisition automatique de termes et de leurs variantes en corpus. habilitation 'a diriger des recherches. IRIN, Univ. de Nantes.

A. Maedche and S. Staab. 2000. Discovering conceptual relations from text. In *Proceedings of ECAI-00*, Amsterdam. IOS Press.

B. Magnini, Serafini L., and Speranza M. 2002. Linguistic based matching of local ontologies. In *Proceedings of Workshop on Meaning Negotiation*, Edmonton, Canada.

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

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

Roman Yangarber. 2001. Scenario customization for information extraction (phd thesis). Courant Institute of Mathematical Sciences, New York University.

G. Zhao and R. Verlinden. 2003. Ff poirot ontology development portal. In *FF Poirot Deliverable D6.1*, Brussel. STAR Lab.