Automatic semantics extraction in law documents

Conference Paper · January 2005

DOI: 10.1145/1165485.1165506 · Source: DBLP

5 authors, including:

Carlo Biagioli
Italian National Research Council
21 PUBLICATIONS **245** CITATIONS
SEE PROFILE

Enrico Francesconi
Italian National Research Council
86 PUBLICATIONS **654** CITATIONS
SEE PROFILE

Simonetta Montemagni
Italian National Research Council
122 PUBLICATIONS **1,078** CITATIONS
SEE PROFILE

Claudia Soria
Italian National Research Council
71 PUBLICATIONS **746** CITATIONS
SEE PROFILE

Automatic extraction of semantics in law documents

Claudia Soria*, Roberto Bartolini*, Alessandro Lenci°, Simonetta Montemagni*, Vito Pirrelli*
*Istituto di Linguistica Computazionale - CNR, Via Moruzzi 1, 56124 Pisa, Italy
°University of Pisa, Department of Linguistics, Via S. Maria 36, 56100 Pisa, Italy

Abstract. In this paper we address the problem of automatically enriching legal texts with semantic annotation, an essential pre–requisite to effective indexing and retrieval of legal documents. This is done through illustration of a computational system developed for automated semantic annotation of (Italian) law texts. This tool is an incremental system using Natural Language Processing techniques to perform two tasks: i) classify law paragraphs according to their regulatory content, and ii) extract relevant text fragments corresponding to specific semantic roles that are relevant for the different types of regulatory content. The paper sketches the overall architecture of the tool and reports results of a preliminary case study on a sample of Italian law texts.

Keywords: semantic analysis of law documents, content enrichment

1. Introduction

The huge amount of documents available in the legal domain calls for computational tools supporting efficient and intelligent search and filtering of information. Over the last several years, machine-learning oriented research in information retrieval and document classification has spawned a number of systems capable of handling structural content management, helping users to automatically or semi-automatically identify relevant structured portions of legal texts, such as paragraphs, chapters or intertextual references. However, while knowledge management systems can certainly profit from automated detection of structural properties of regulatory texts, advanced document indexing and retrieval functions are bound to require more granular and rich semantically oriented representations of text content. Suppose you are interested in finding all regulations applying to a particular type of individual, say an employer. Searching legal texts by using "employer" as a keyword is likely to return many irrelevant text excerpts, as plain keyword matching is blind to the particular semantic role played by the concept encoded by the term in context. Alternatively, one might be interested in tracking down all the penalties that a citizen or a member of a particular category of people is subjected to in connection with a particular behaviour. Simply searching for "citizen" and "penalty" would not be enough, since there are many other possible ways to express the same concepts, whose recognition requires advanced text understanding capabilities.

To successfully address all these issues, then, we expect regulatory texts to explicitly contain the sort of implicit information that human readers are naturally able to track down through reading, *e.g.* that a certain text expresses an obligation for an employer to provide a safe environment for his employees, or that another text encodes a penalty for doing or not doing something. The process of augmenting a text with labels expressing its semantic content is what we shall hereafter refer to as *semantic annotation.* In the past, indexing of textual documents with semantic tags has been a manual chore, and methods to automate this process are highly desirable. Semantic annotation of unstructured natural language texts can significantly increase the value of text collections by promoting deployment of advanced services, well beyond traditional full-text search functionalities. In this paper we intend to address the problem of automatically enriching legal texts with semantic tags through illustration of SALEM (Semantic Annotation for LEgal Management), an NLP system currently used as an advanced module of the NIR[1] legal editor (see (Biagioli et al., 2003)) to automatically tag the semantic structure of Italian law paragraphs through an integration of NLP and information extraction-inspired technology.

1.1. Previous work

Functionalities for retrieving relevant documents on the basis of text queries are provided by most current legal knowledge management tools, as witnessed by the well–known LexisNexis© and WestLaw© systems. Surely, systems differ as to the types of information search queries are sensitive to. In most cases only low-level text structures can be searched for. For instance, the tool described in (Bolioli et al., 2002) is used for the automatic recognition of structural elements of a law text, and allows for intra- and inter-textual browsing of documents. Fully automatic or semi-automatic systems that carry out semantic text analysis, thus providing content-based representations, are far less common. Notable exceptions are the DIAsDEM system (Graubitz et al., 2001) and the approach proposed by De Busser *et al.* (Busser et al., 2002), which is however not specialized for the legal domain. In DIAsDEM, unstructured texts are iteratively processed to yield a semantic representation of their content. Although developed for a different domain, for different purposes and with different techniques, the output of this system is

[1] NIR ("Norme in Rete", *Laws on the web*) is a national project sponsored by the Ministry of Justice for the free access by citizens to Italian jurisdiction.

in line with the one described in this paper: a text augmented with domain–specific tags explicitly representing portions of its content.

A technique more similar to the one presented here is adopted by Saias and Quaresma (Saias and Quaresma, 2003), who exploit NLP techniques to yield a syntactic annotation of law texts and populate a legal ontology. Their output representation allows users to retrieve a document on the basis of sophisticated queries such as, for instance, the presence of a certain action X in a document, the occurrence of individual Y as a subject of an unspecified action, or as the performer of X. SALEM identifies similar types of information in texts, but its semantic representation also contains the particular type of regulation expressed by a law paragraph, in addition to the entities and actions involved. In our view, this extra information represents an added value for effective indexing and retrieval of documents. Finally, the area of research in automatic construction and population of legal ontologies, albeit not specifically intended to address the task of semantic annotation as such, also shares many of the issues we are interested in here (see for instance the work of Lame (Lame, 2003) and Mommers (Mommers, 2001), among the others).

2. Methodology and motivations

2.1. The legal text

As textual units, (Italian) laws are typically organized into hierarchically structured sections, the smallest one being the so-called *law paragraph*. Law paragraphs are usually numbered sections of an article, as in Figure 1 below[2]. As to its content, a law paragraph is associated with a particular *legislative provision*, which could be seen as the illocutionary point of a law section. For instance, a paragraph expresses a

Article 6.
1. The Commission shall be assisted by the committee set up by Article 5 of Directive 98/34/EC.
2. The representative of the Commission shall submit to the committee a draft of the measures to be taken.

Figure 1. A typical article.

[2] The examples provided in the paper are taken from EC laws. Every time the purpose of the example is to illustrate semantic, language-independent phenomena, we use the English text for the sake of clarity. We remind the reader, however, that SALEM works on Italian law texts only.

permission or an obligation for some actor to perform or not to perform a certain course of action, as in Figures 2 and 3.

Directive. A Member State may provide that a legal body the head office of which is not in the Community may participate in the formation of an SCE provided that legal body is formed under the law of a Member State, has its registered office in that Member State and has a real and continuous link with a Member State's economy.

Figure 2. A Permission.

Licence applications shall be accompanied by proof of payment of the fee for the period of the licence's validity.

Figure 3. An Obligation.

Law paragraphs may also have an inter–textual content, *i.e.* they can contain some sort of amendments to existing laws. In this case they are said to be *modifications*. For instance, a paragraph may contain an insertion with respect to another law, or a replacement, or a repeal, as the Figures 4 and 5 illustrate.

The following point shall be inserted after point 2g (Council Directive 96/61/EC) in Annex XX to the Agreement: "2h. 399 D 0391: Commission Decision 1999/391/EC of 31 May 1999 concerning the questionnaire relating to Council Directive 96/61/EC concerning integrated pollution prevention and control (IPPC) (implementation of Council Directive 91/692/EEC) (OJ L 148, 15.6.1999, p. 39)."

Figure 4. An Insertion.

The text of point 2eg (Commission Decision 95/365/EC) in Annex XX to the Agreement shall be replaced by the following: "399 D 0568: Commission Decision 1999/568/EC of 27 July 1999 establishing the ecological criteria for the award of the Community eco–label to light bulbs (OJ L 216, 14.8.1999, p. 18)."

Figure 5. A Replacement.

2.2. SALEM Framework

SALEM has a twofold task: a) to assign each law paragraph to a given *legislative provision type*; b) to automatically tag the parts of the paragraph with domain-specific semantic roles identifying the *legal entities* (i.e. actors, actions and properties) referred to in the legislative provision.

The type of semantic annotation output by SALEM is closely related to the task of Information Extraction, defined as “the extraction of information from a text in the form of text strings and processed text strings which are placed into slots labelled to indicate the kind of information that can fill them” (MUC, Message Understanding Conference). Law text analysis in SALEM is driven by an *ontology* of legislative provision types (e.g. obligation, insertion, etc.). Classes in the ontology are formally defined as *frames* with a fixed number of (possibly optional) *slots* corresponding to the semantic roles played by the legal entities specified by a given provision type. For instance, in Figure 1 above, which expresses an obligation, the relevant roles in the first sentence of paragraph 2 are the addressee of the obligation (i.e. *The representative of the Commission*), the action (i.e. what the addressee is obliged to do, in this case to *submit to the committee a draft of the measures to be taken*) and a *third_party* (i.e. the action recipient, here *the committee*). In a similar way, a modification such as an insertion can have up to four relevant roles: (1) the reference text being modified, or *rule* (in Figure 4 above, the text *(Council Directive 96/61/EC) in Annex XX to the Agreement)*, (2) the *position* where the new text is going to be inserted (here, *after point 2g*); (3) the new text or *novella* (here, the captioned text); (4) the verbatim text to be replaced by the novella (*novellato*, not occurring in the example above). The following example illustrates the frame for an obligation:

FRAME : obligation
ADDRESSEE: the member State
ACTION: pay the advance within 30 calendar days of submission of the application for advance payment
THIRD PARTY: –

while the following is an example of the frame for a replacement.

FRAME : replacement
RULE: (Commission Decision 95/365/EC) in Annex XX to the Agreement
POSITION: –
NOVELLATO: point 2eg
NOVELLA: 399 D 0568: Commission Decision 1999/568/EC of 27 July 1999 establishing the ecological criteria for the award of the Community eco-label to light bulbs (OJ L 216, 14.8.1999, p. 18).

Automatic identification of the provision type expressed by a law paragraph is important for effective management of law texts. Law databases could be queried through fine–grained “semantic” searches according to the type of legal provision reported by a law paragraph. Furthermore, automatic identification and extraction of text portions of law that are subject to modifications could enable (semi)automatic updating of law texts, or make it possible for the history of a law to be traced throughout all its modifications; the original referenced text could be imported and modified, etc. Finally, automatic assignment of the relevant paragraph parts to semantic slots is bound to have an impact on effective legal content management and search, allowing for fine–grained semantic indexing and query of legal texts, and paving the way to real–time analysis of legal corpora in terms of logical components or actors at the level of individual provisions. In the near future, it will be possible to search an on-line legislative corpus for all types of obligations concerning a specific subject, or to highlight all possible legislative provisions a given action or actor happens to be affected by.

3. SALEM architecture

3.1. General overview

Although legal language is considerably more constrained than ordinary language, its specific syntactic and lexical structures still pose a considerable challenge for state-of-the-art NLP tools. Nonetheless, if our goal is not a fully-fledged representation of their content, but only identification of specific information portions, legal texts are relatively predictable and hence tractable through NLP–based techniques.

SALEM is a suite of NLP tools for the analysis of Italian texts (Bartolini et al., 2002a), specialized to cope with the specific stylistic conventions of the legal parlance, with the aim to automatically classify and semantically annotate law paragraphs. A first prototype of SALEM has just been brought to completion and its performance evaluated. The NLP technology used for SALEM is relatively simple, but powerful, also thanks to the comparative predictability of law texts. SALEM takes in input single law paragraphs in raw text and outputs a semantic tagging of the text, where its classification together with the semantic roles corresponding to different frame slots are rendered as XML tags. An output example (translated into English for the reader's convenience) is given in Figure 6, where the input paragraph is classified as an *obl(igation)* and portions of the text are identified as respectively denoting the *addressee* and the *action.*

```
<obligation><obl:addressee>The Member State</obl:addressee>
shall <obl:action>pay the advance within 30 calendar
days of submission of the application for advance payment
</obl:action>.</obligation>
```

Figure 6. SALEM output example.

SALEM approach to classification and semantic annotation of legal texts follows a two stage strategy. In the first step, a general purpose parsing system, hand-tuned to handle some idiosyncracies of Italian legal texts, pre–processes each law paragraph to provide a shallow syntactic analysis. In the second step, the syntactically pre–processed text is fed into the semantic annotation component proper, making explicit the information content implicitly conveyed by the provisions.

3.2. Syntactic pre–processing

Syntactic pre–processing produces the data structures to which semantic annotation applies. At this stage, the input text is first tokenized and normalized for dates, abbreviations and multi–word expressions; the normalized text is then morphologically analyzed and lemmatized, using an Italian lexicon specialized for the analysis of legal language; finally, the text is POS-tagged and shallow parsed into non–recursive constituents called “chunks”.

A sample chunked output is given in Figure 7. A chunk is a textual unit of adjacent word tokens sharing the property of being related through dependency relations (es. pre–modifier, auxiliary, determiner, etc.). Each chunk contains information about its type (e.g. a noun chunk (N_C), a verb chunk (FV_C), a prepositional chunk (P_C), an adjectival chunk (ADJ_C), etc.), its lexical head (identified by the label *potgov*) and any intervening modifier, causative or auxiliary verb, and preposition. A chunked sentence, however, does not give information about the nature and scope of inter–chunk dependencies. These dependencies, whenever relevant for semantic annotation, are identified at the ensuing processing stage (see section 3.3 below).

Although full text parsing may be suggested as an obvious candidate for adequate content processing, we contend that shallow syntactic parsing provides a useful intermediate representation for content analysis. First, at this stage information about low level textual features (e.g. punctuation) is still available and profitably usable, whereas it is typically lost at further stages of analysis. In this connection, it should be appreciated that correct analysis of modifications crucially depends on punctuation marks, and in particular on quotes and colons, which are used to identify the text of the amendment (*novella*) and the amending text (*novellato*). Secondly, chunked sentences naturally lend themselves to incrementally being used as the starting point for partial functional analysis, whereby the range of dependency relations that are instrumental for semantic annotation is detected. In particular, dependency information is heavily used for the mark–up of both modifications and obligations, which requires knowledge of the underlying syntactic structure. Finally, a third

All'articolo 6 della legge 24_gennaio_1986 , n._17 , le parole: "entro centottanta giorni dalla pubblicazione della presente legge nella Gazzetta Ufficiale" sono soppresse.

1. `[ [ CC: P_C] [ PREP: A#E] [ DET: LO#RD@MS] [ AGR: @MS] [ POTGOV: ARTICOLO#S@MS]]` All' articolo
2. `[ [ CC:ADJ_C] [ POTGOV: 6#N]]` 6
3. `[ [ CC: di_C] [ DET: LO#RD@FS] [ AGR: @FS] [ POTGOV: LEGGE#S@FS]]` della legge
4. `[ [ CC: ADJ_C] [ POTGOV: 24_gennaio_1986#N]]` 24_gennaio_1986
5. : `[ [ CC: PUNC_C] [ PUNCTYPE: ,#@]]` ,
6. `[ [ CC: ADJ_C] [ POTGOV: n._17#N]]` n._17
7. `[ [ CC: PUNC_C] [ PUNCTYPE: ,#@]]` ,
8. `[ [ CC: N_C] [ DET: LO#RD@FP] [ AGR: @FP-@FP] [ POTGOV: PAROLA#S@FP PAROLE#S@FP]]` le parole
9. ...

Figure 7. A fragment of chunked text.

practical reason is that chunking yields a local level of syntactic annotation. As a result, it does not "balk" at domain–specific constructions that violate general grammar patterns; rather, parsing is carried on to detect the immediately following allowed structure, while ill–formed chunks are left behind, unspecified for their category.

3.3. Semantic annotation

As mentioned above, the SALEM approach is closely inspired by mainstream techniques in Information Extraction. In particular, semantic annotation consists in the identification of all the instances of particular provision types in text. The frame defining a provision type can then be regarded as an *extraction template* whose slots are filled with the textual material matching the corresponding conceptual roles.

The semantic annotation component takes in input a chunked representation of each law paragraph and identifies semantically–relevant structures by applying domain–dependent, finite state techniques locating relevant patterns of chunks. Semantic mark–up is performed through a two–step process:

1. each paragraph is assigned to a frame (corresponding to the legislative provision expressed in the text);
2. slots of the frame identified at step (1) are turned into an extraction template and instantiated through the structural components (i.e. sentences, clauses, phrases) of the law paragraph.

The current version of the semantic annotation component is a specialized version of the ILC finite–state compiler of grammars for dependency syntactic analysis (Bartolini et al., 2002b). The SALEM version of the grammar compiler uses a specialized grammar including (i) a core group of syntactic rules for the identification of basic syntactic dependencies (e.g. subject and object), and (ii) a battery of specialized rules for the semantic annotation of the text.

All rules in the grammar are written according to the following template:

> <chunk-based regular expression> WITH <battery of tests> => <actions>

The recognition of provision types and the subsequent extraction of information from the text are based on structural patterns which are combined with lexical conditions and other tests aimed at detecting low level textual features (such as punctuation) as well as specific syntactic structures (e.g. the specification of a given dependency relation). Structural patterns are expressed in terms of regular expressions over sequences of chunks, whereas all other conditions (*e.g.* lexical, syntactic, etc.) are checked through a battery of tests. The action type ranges from the identification of basic dependency relations (in the case of syntactic rules) to the semantic mark–up of the text (in the case of semantic annotation rules).

The assignment of a paragraph to a provision class is based on a combination of both syntactic and lexical criteria. As already mentioned above, a preliminary study of the linguistic features of law paragraphs revealed a strong association between classes of verbs and provision types: an obligation is typically expressed with the modal verb “dovere” (*shall/must*) or the verbal construction “essere obbligato/tenuto a” (*to be obliged to*); similarly, lexical cues for the identification of an insertion include verbs like “aggiungere” or “inserire” (*to add, to insert*). We will refer to these verbs as trigger verbs. The presence of a trigger verb in the text is only the first step towards the detection of a specific provision class which, in order to be completed, needs to be complemented with structural tests which include the role of the trigger verb as the sentence head and the types of dependency relations governed by it.

Reliable identification of dependency relations is also important for assigning structural elements to semantic roles, since the latter tend to be associated with specific syntactic functions (e.g. subject, object). To give the reader only but one example, the addressee of an obligation typically corresponds to the syntactic subject of the sentence, while the action (s)he is obliged to carry out is usually expressed as an infinitival clause, as in the example reported below:

> `[[`Il comitato misto`]subj]addressee]` è tenuto a raccomandare modifiche degli allegati secondo le modalità previste dal presente accordo
> `[[`*The Joint Committee*`]subj]addressee]` *shall be responsible for recommending amendments to the Annex as foreseen in this Agreement.*

Note, however, that this holds only when the verbal head of the infinitival clause is used in the active voice. By contrast, the syntactic subject can express the third–party if the action verb is used in the passive voice and is governed by specific lexical heads.

4. Case study

We report the results of a case study carried out with SALEM on the basis of a small ontology covering 8 provision types. This section presents the ontology of provisions and illustrates the system's performance against a corpus of law paragraphs previously annotated by law experts.

4.1. The frame–based legal ontology

In this case study, we have used a small ontology designed by experts in the legal domain at ITTIG–CNR.[3] The ontology distinguishes three major categories of provisions: *obligations*, *definitions* and *modifications.* A main distinction can be made between obligations, addressing human actors, and modifications, which are rather aimed at modifying the textual content of pre–existing laws. Obligations in turn divide into the following classes: *obligation*, *prohibition*, *permission*, and *penalty.* In their turn, modifications are further subdivided into *replacement*, *insertion* and *repeal.* The taxonomical structure of the SALEM ontology is illustrated in Figure 8.

As mentioned above, law paragraphs are analyzed in SALEM not only according to the particular type of legislative provision they express, but also with respect to the main legal entities involved by the

[3] Istituto di Teoria e Tecniche dell'Informazione Giuridica, CNR, Florence, Italy.

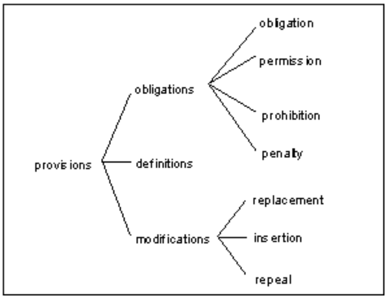

Figure 8. Taxonomical structure of the ontology of provisions

Table I. Frame-based description of the different provision types

Provision class	Slots
Obligation	Addressee, Action, Third-party
Permission	Addressee, Action, Third-party
Prohibition	Action, Third-party
Penalty	Addressee, Action, Object, Rule
Definition	Definiendum, Definiens
Repeal	Rule, Position, *Novellato*
Replacement	Rule, Position, *Novellato*, *Novella*
Insertion	Rule, Position, *Novella*

law. Consistently, each ontology class is formally defined as a *frame* with a fixed number of (possibly optional) *slots*. The slot types required for the description of the 8 bottom classes in the SALEM taxonomy are illustrated in Table I.

4.2. Evaluation results

SALEM preliminary results are very promising. The system was evaluated on a sample of 473 law paragraphs, covering seven classes of the ontology of Table I. The test corpus was built by law experts at ITTIG-CNR, who also provided a hand–annotated version used as gold standard for evaluation. The aim of the evaluation was to assess the system's performance on two tasks: classification of paragraphs according to the ontology (henceforth referred to as "classification task") and mark–up of semantic roles (henceforth "information extraction task").

Table II. SALEM results for the classification task.

Class	Total	SALEM answers	OK	Precision	Recall
Obligation	19	19	18	0.95	0.95
Permission	15	18	15	0.83	1
Prohibition	15	15	14	0.93	0.93
Penalty	122	117	109	0.93	0.89
Repeal	70	69	69	1	0.99
Replacement	111	111	111	1	1
Insertion	121	119	119	1	0.98
Tot.	**473**	**468**	**455**	**0.97**	**0.96**

Table II summarizes the results achieved for the paragraph classification task with reference to the seven bottom classes of provisions, where Precision is defined as the ratio of correctly classified provisions over all SALEM answers, and Recall refers to the ratio of correctly classified provisions over all provisions in the test corpus. Note that here a classification is valued as correct if the automatically assigned class and the manually assigned one are identical. The classification performance is even better if it is related to the corresponding first level ontology classes (i.e. obligations and modifications). In fact, in some cases, mostly penalties and permissions, multiple answers are given due to the fact that obligations bottom classes share a great deal of lexical and morpho-syntactic properties; yet, these answers are to be considered correct if classification is evaluated with respect to the first level ontology classes (i.e. obligations and modifications). On the other hand, when unambiguous linguistic patterns are used, the system easily reaches 1 for both Precision and Recall, as with the class of Modifications.

Table III illustrates the performance of the system in the information extraction task. The aim of the evaluation here was to assess the system's reliability in identifying, for each provision type or frame, all the semantic roles that are relevant for that frame and are instantiated in the text. For each class of provisions in the test corpus we thus counted the total number of semantic roles to be identified; this value was then compared with the number of semantic roles correctly identified by the system and the total number of answers given by the system. Here, Precision is scored as the number of correct answers returned by system

Table III. SALEM results for the information extraction task.

Class	OK	Expected answers	SALEM answers	Precision	Recall
Obligation	38	38	38	1	1
Permission	20	25	24	0.83	0.8
Prohibition	21	21	27	0.78	1
Penalty	303	388	330	0.92	0.78
Repeal	104	108	106	0.98	0.96
Replacement	303	309	306	0.99	0.98
Insertion	373	376	375	0.99	0.99
Tot.	**1162**	**1265**	**1206**	**0.96**	**0.92**

over the total number of answers returned, while Recall is the ratio of correct answers returned by system over the number of expected answers.

5. Conclusions and future work

In this paper we presented SALEM, an NLP–based system for classification and semantic annotation of law paragraphs. Although we follow the mainstream trend in state-of-the-art NLP architectures towards use of shallow parsing techniques, the novelty of our approach rests in the incremental composition of shallow parsing with higher levels of syntactic and semantic analysis, leading to simultaneous, effective combination of low- and high-level text features for fine-grained content analysis. Besides, the cascaded effect of incremental composition led to a considerable simplification of the individual parsing modules, all of which are implemented as finite state automata. The effectiveness and performance of the system was tested with promising results on the basis of a small ontology of provision types, against a test-bed of text material hand-annotated by human experts.

We are presently working along several lines of development. On the one hand, we intend to make the system more robust by testing it on a larger sample of law texts. Although laws are quite stable as a language genre, they can also be stylistically variable depending on the personal inclinations of the author, the particular domain they apply to, not to mention variations determined by historical changes. Collection of a wider variety of text material is bound to have an impact on

SALEM flexibility and coverage. On the other hand, we also aim at expanding the ontology of provisions for semantic annotation to cover new provision types.

References

Bartolini, R., Lenci, A., Montemagni, S., Pirrelli, V. The lexicon-grammar balance in robust parsing of Italian. *Proc. of 3rd International Conference on Language Resources and Evaluation.* (2002).

Bartolini R., Lenci A., Montemagni S., Pirrelli V. Grammar and lexicon in the robust parsing of Italian: Towards a non-naïve interplay. *Proc. of Coling 2002 Workshop on Grammar Engineering and Evaluation.* Academia Sinica, Nankang, Taipei, Taiwan, 1st September (2002).

Biagioli, C., Francesconi, E., Spinosa, P., Taddei, M. The NIR project. Standards and tools for legislative drafting and legal document Web publication. *Proc. of the International Conference of Artificial Intelligence and Law.* Edinburgh, June 24 (2003).

Bolioli, A., Dini, L., Mercatali, P., Romano, F. For the automated mark-up of Italian legislative texts in XML. *Proc. of JURIX 2002*, London, UK, 16-17 December (2002).

De Busser, R., Angheluta, R., Moens, M.-F. Semantic Case Role Detection for Information Extraction. *Proc. of COLING 2002.* New Brunswick (2002) 1198–1202.

Graubitz, H., Winkler, K., Spiliopoulou, M. Semantic Tagging of Domain-Specific Text Documents with DIAsDEM. *Proc. of the 1st International Workshop on Databases, Documents, and Information Fusion (DBFusion 2001).* Gommern, Germany (2001) 61–72.

Saias, J., Quaresma, P. Using NLP techniques to create legal ontologies in a logic programming based web information retrieval system. *Proc. of ICAIL 2003 Workshop on Legal Ontologies and Web Based Legal Information Management.* Edinburgh, UK, June 24-28 (2003).

Lame, G. Using text analysis techniques to identify legal ontologies' components. *Proc. of ICAIL 2003 Workshop on Legal Ontologies and Web Based Legal Information Management.* Edinburgh, UK, June 24-28 (2003).

Mommers, L. A knowledge-based ontology for the legal domain. *Proc. of the Second International Workshop on Legal Ontologies.* December 13 (2001).