Chapter 13
A Learning Approach for Knowledge Acquisition in the Legal Domain

Enrico Francesconi

13.1 Introduction

Knowledge modelling represents a structural pre-condition for implementing the Semantic Web concept as well as intelligent systems dealing with legal information. In the last few years many efforts have been made for implementing legal ontologies and a vast literature exists in this domain (see (Breuker et al. 2009) for a state-of-the-art review). One of the main problem in this field, addressed in literature, is represented by a trade-off existing between *consensus* and *authoritativeness* in legal knowledge representation.

Consensus is an issue faced in knowledge representation in general, as underlined by several authors (Gangemi et al. 2002, Guarino 1997), since ontological conceptualization has to be shared between stakeholders (Studer et al. 1998). Several approaches have been undertaken to reach consensus in legal knowledge representation: for example the *common-sense terms* approach (Hoekstra et al. 2009) based on common sense understanding of the terminology identifying concepts, as well as the *folksonomy* approach[1] based on social and collaborative activities of concepts selection and categorization (Gruber 2006).

Knowledge representation in the legal domain, however, shows peculiarities due to the importance of having authoritative systems based on legal rules for legal assessment and reasoning, concerning the problem of determining whether a case is allowed or disallowed given a body of legal norms (Breuker et al. 2008; Capon and Visser 1997). Authoritative issues are also important for advanced search engines, based on semantic annotation of legal documents, able to retrieve not just documents but also the contained norms (Biagioli and Turchi 2005). In these systems users are interested in *rules* on a specific domain (relations between norms, support

E. Francesconi (✉)
ITTIG-CNR, Institute of Legal Information Theory and Techniques, Italian National Research Council, Florence, Italy
e-mail: francesconi@ittig.cnr.it

[1] Folksonomies (or social tagging mechanisms) have been widely implemented in knowledge sharing environments; the idea was first adopted by the social bookmarking site del.icio.us (2004) http://delicious.com

G. Sartor et al. (eds.), *Approaches to Legal Ontologies*, Law, Governance and Technology Series 1, DOI 10.1007/978-94-007-0120-5_13,

to legal reasoning), as a consequence they look for *authoritativeness* in knowledge representation.

Both common-sense terms and folksonomy approaches are well suited to reach *consensus* on domain concepts, however, when applied to the description of *legal rules*, the gap between consensus and authoritativeness is usually emphasized. For example, by the common-sense terms approach, social and communicative words typical of the legal domain can be provided (Breuker and Hoekstra 2004a): in this approach experts may provide description of rules on entities as well as translating them into technical terminology (Hoekstra et al. 2009), but this activity might reduce *consensus*. Similarly, in the folksonomy approach stakeholders may provide description of rules regulating entities, which might be lacking in *authoritativeness*.

Another way, discussed in literature (Euzenat and Shvaiko 2007; Francesconi et al. 2008), to provide consensus in knowledge modelling is to find conceptual equivalences through ontology mapping techniques; however, such techniques do not provide any additional contribution to the authoritativeness in knowledge representation.

Nowadays a very active research area in ontology development is represented by knowledge acquisition from texts (Buitelaar et al. 2005,Cimiano 2006), since electronic texts still represent the most widely used communication medium on the Web. This approach can play an important role in legal knowledge modelling, in particular in *legal rules* modelling, since written text is the most widely used way of communicating legal matters (Lame 2005; Saias and Quaresma 2005; Walter and Pinkal 2006). Knowledge acquisition techniques, usually supported by machine learning and natural language processing, can be used for implementing taxonomies or suggesting concepts for upper level ontologies, mainly hand-crafted by domain experts, as well as for identifying and representing *legal rules*. Such techniques represent a more neutral approach for identifying relevant concepts for knowledge modelling, thus contributing to reach consensus.

In this paper a learning approach supporting the acquisition of legal rules contained in legislative documents is presented: it is based on a semantic model for legislation and implemented by using knowledge extraction techniques over legislative texts. This methodology is targeted to provide a contribution to bridge the gap between consensus and authoritativeness in implementing systems based on legal rules: *consensus* can be better reached by limiting human intervention in legal rules description, which are extracted from *authoritative* texts as the legislative ones.

This paper is organised as follows: in Section 13.2 an approach to legal rules modelling and acquisition is presented, in Section 13.3 a semantic model for legislative texts is introduced, in Section 13.4 a knowledge acquisition methodology is shown and tested, finally in Section 13.5 some conclusions, discussing the benefits of the described learning approach, are reported.

13.2 A Learning Approach for the Acquisition of Legal Rules

The proposed approach for legal knowledge acquisition is based on learning techniques targeted to extract *legal rules* from text corpora. Legal rules are essentially

"speech acts" (Searle 1969) expressed in legislative texts regulating *entities* of a domain: their nature therefore justifies an approach targeted to the analysis of such texts.

Therefore, the proposed knowledge acquisition framework is based on a twofold approach:

1. Knowledge modelling: definition of a semantic model for legislative texts able to describe legal rules;
2. Knowledge acquisition: instantiation of legal rules through the analysis of legislative texts, being driven by the defined semantic model.

This approach traces a framework which combines a top–down and a bottom–up strategy: a top–down strategy provides a model for legal rules, while a bottom-up strategy identifies rules instances as expressed in legislative texts.

The bottom-up knowledge acquisition strategy in particular can be carried out manually or automatically. The manual bottom–up strategy consists, basically, in an analytic effort in which all the possible semantic distinctions among the textual components of a legislative text are identified. On the other hand the automatic (or semi-automatic) bottom–up strategy consists in carrying out the previous activities being supported by tools able to classify Rules, according to a defined model, and to identify the involved Entities. In this paper this second strategy is presented.

13.3 Knowledge Modelling

The proposed approach is based on knowledge modelling, which is oriented to interopreability and reusability. It is conceived according to two main principles: (1) Knowledge representation by Semantic Web standards; (2) Separation betweeen types of knowledge.

The first aspect is aimed at reaching interoperability among knowledge-based applications, exploiting the expressiveness and reusability of the RDF/OWL semantic Web standards. The second aspect, on the other hand, aims to guarantee reusability of the knowledge resources.

The need of identifying and separating different types of knowledge has been widely addressed in literature (Casellas 2008). For example (Breuker and Hoekstra 2004b) criticised a common tendency to indiscriminately mix domain knowledge and knowledge on the process for which it is used, addressing it as *epistemological promiscuity*. Similarly (Bylander and Chandrasekaran 1987), (Chandrasekaran 1986) and (van Heijst 1995) pointed out that usually knowledge representation is affected by the nature of the problem and by the applied inference strategy; this key-point is also referred by (Bylander and Chandrasekaran 1987) as *interaction problem*: it is related to a discussion regarding whether knowledge about the domain and knowledge about reasoning on the domain should be represented independently. In this respect (Clancey 1981) pointed out that the separation of both types of knowledge is a desirable feature, since it paves the way to knowledge sharing and reuse.

The knowledge model proposed in this work reflects these orientations and it is organized into the following components:

1. Domain Independent Legal Knowledge (DILK)
2. Domain Knowledge (DK)

DILK is a semantic model able to provide classification of Rules expressed in legislative texts, while DK is any terminological or conceptual knowledge base (thesaurus, ontology, semantic network) able to provide information and relationships among the Entities of a regulated domain. The combination of a DILK model with one or more DKs is able to describe, from a semantic point of view, Rules instances and related domain Entities expressed in legislative texts. For this reason we call the proposed methodology to legal knowledge modelling the *DILK-DK* approach.

13.3.1 DILK

DILK is conceived as a model for legal Rules, independently from the domain they apply to. In literature several models (classification) of legal rules have been proposed, from the traditional Hohfeldian theory of legal concepts (Hohfeld 1913), (Hohfeld 1917), until more recent legal philosophy theories due to Rawls (1955), Hart (1961), Ross (1968), Bentham and Hart (1970, 1st ed. 1872), Kelsen (1991).

In this respect a particular attention is worth to be given to the work of Biagioli (Biagioli 1991), (Biagioli 1997). In the 1990s Biagioli tried to combine the work of legal philosophers on rules classification with the Searlian theory of rules preceived as "speech acts". Following the Raz's lesson (Raz 1977), which considers the entire body of laws and regulations as a set of *provisions* carried by speech acts, namely sentences endowed with meaning, Biagioli underlined two views or *profiles* according to which a legislative text can be perceived:

- a structural or *formal profile*, representing the traditional legislator habit of organizing legal texts in chapters, articles, paragraphs, etc.;
- a semantic or *functional profile*, considering legislative texts as composed by *provisions*, namely fragments of regulation (Biagioli 1997) expressed by speech acts.

Therefore a specific classification of legislative provisions was carried out by analyzing legislative texts from a semantic point of view, and grouping provisions into two main families: *Rules* (introducing and defining entities or expressing deontic concepts) and *Rules on Rules* (different kinds of amendments).

Rules are provisions which aim at regulating the reality considered by the including act. Adopting a typical law theory distinction, well expressed by Rawls, they consist in:

- *constitutive rules*: mainly rules on entities of the regulated reality. They consist in rules introducing entities ("rules of the game" (Ricciardi 1997)) and rules which assign a juridical profile to the entities ("empowering norms" or "rules in the game" (Ricciardi 1997));
- *regulative rules*: they discipline actions ("rules on actions") or the substantial and procedural defaults ("remedies").

On the other hand, Rules on Rules are provision types in which we can distinguish:

- *content amendments*: they modify literally the content of a norm, or their meaning without literal changes;
- *temporal amendments*: they modify the times of a norm (come-into-force and efficacy time);
- *extension amendments*: they extend or reduce the cases on which the norm operates.

In Biagioli's model each provision type has specific arguments describing the roles of the entities which a provision type applies to (for example *Bearer* is argument of a *Duty* provision).

Provision types and related *Arguments* represent a semantic model for legislative texts (Biagioli 1997). They can be considered as a sort of metadata scheme able to describe analytically the content of a legislative text.

For example, the following fragment of the Italian privacy law:

> A controller intending to process personal data falling within the scope of application of this act shall have to notify the "Garante" thereof, . . .

besides being considered as a part of the physical structure of a legislative text (a *paragraph*), can also be viewed as a component of the logical structure of it (a *provision*). In particular, it can be qualified as a *provision* of type *Duty*, whose arguments are:

Bearer: "Controller"
Object: "Process personal data"
Action: "Notification"
Counterpart: "Garante".

The specific textual anchorage of the Biagioli's model represents, in our point of view, its main strenght. Since the DILK-DK approach aims at representing Rules instances as expressed in legislative texts, we consider the Biagioli's model, limited to the group of Rules, as a possible implementation of the *Domain Independent Legal Knowledge* (DILK). Rules on Rules affect indirectly the way how the reality is regulated, since they amend Rules in different respects (literally, temporarily, extensionally): therefore such provisions type should be taken into account as far as they change Rules.

13.3.2 DK

In legislative texts *Entities* regulated by provisions are expressed by lexical units, however no additional information on such entities are provided. This information can be provided by a *Domain Knowledge* (DK) giving conceptualization of entities expressed by language-dependent lexical units.[2] Information on such entities at language-independent level, as well as their lexical manifestations in different languages can be described by a DK.

A possible DK architecture has been proposed within the DALOS project[3] according to two layers of abstraction:

- an *Ontological layer*: conceptual modelling at language-independent level;
- a *Lexical layer*: language-dependent lexical manifestations of the concepts at the Ontological layer.

More details on the DALOS DK architecture, as well as a possible implementation of it for the domain of consumer protection, can be found in (Agnoloni et al. 2009).

13.4 Knowledge Acquisition

Knowledge acquisition within the DILK-DK framework consists of two main steps:

1. DILK instantiation
2. DK construction

13.4.1 DILK Instantiation

The DILK instantiation phase is a bottom-up strategy of legislative text paragraphs classification into *provision types*, as well as specific lexical units identification, assigning them roles in terms of *provision arguments*. The automatic bottom–up strategy, here proposed, consists in using tools able to support the human activity of classifying provisions and extracting their arguments.

Three main steps can be foreseen:

- Collection of legislative texts and conversion into an XML format (Bacci et al. 2009)

[2] "Typically regulations are not given in an empty environment; instead they make use of terminology and concepts which are relevant to the organisation and/or the aspect they seek to regulate. Thus, to be able to capture the meaning of regulations, one needs to encode not only the regulations themselves, but also the underlying ontological knowledge. This knowledge usually includes the terminology used, its basic structure, and integrity constraints that need to be satisfied." Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher, "On the modeling and analysis of regulations", in *Proceedings of the Australian Conference Information Systems*, pages 20–29, 1999.

[3] http://www.dalosproject.eu

- Automatic classification of legislative text paragraphs into provisions (Biagioli et al. 2005; Francesconi and Passerini 2007)
- Automatic argument extraction (Biagioli et al. 2005).

Legislative documents are firstly collected and transformed into a jurisdiction-dependent XML standard (NormeInRete in Italy, Metalex in the Netherlands, etc.). For the Italian legislation a module called xmLegesMarker of the xmLeges[4] software family, has been developed (Bacci et al. 2009): it is able to transform legacy contents into XML so to identify the formal structure of legislative documents.

13.4.1.1 Automatic Classification of Provisions

For the automatic classification of legislative text paragraphs into provision types, a tool called xmLegesClassifier, of the xmLeges family, has been developed. xmLegesClassifier has been implemented using a Multiclass Support Vector Machine (MSVM) approach, as the one reporting the best results in preliminary experiments with respect to other machine learning techniques (Francesconi and Passerini 2007). With respect to (Francesconi and Passerini 2007), in this work MSVM is specifically used to classify Rules. Documents are represented by vectors of weighted terms and some preprocessing operations are performed on pure words to increase their statistical qualities:

- Stemming on words in order to reduce them to their morphological root
- Stopwords elimination
- Digit and non alphanumeric characters represented using a special character

Moreover *feature selection* techniques are applied to reduce the number of terms to be considered, thus actually restricting the vocabulary to be employed (see e.g. (Sebastiani 2002,Yang and Pedersen 1997)). We tried two simple methods:

- An unsupervised *min frequency* threshold over the number of times a term has been found in the entire training set, targeted to eliminate terms with poor statistics.
- A supervised threshold over the Information Gain (Quinlan 1986) of terms, which measures how much a term discriminates between documents belonging to different classes. Being D the set of training documents, the Information Gain of term w is computed as:

$$ig(w) = H(D) - \frac{|D_w|}{|D|}H(D_w) - \frac{|D_{\bar{w}}|}{|D|}H(D_{\bar{w}})$$

where H is a function computing the entropy of a labelled set ($H(D) = \sum_{i=1}^{|C|} -p_i \, log_2(p_i)$, being p_i the portion of D belonging to the *ith* class, D_w is the set of training documents containing the term w, and $D_{\bar{w}}$ is the set of training documents not containing w.

[4] http://www.xmleges.org

Entropy in information theory measures the amount of bits necessary to encode the class of a generic element from a labelled set, and thus depends on the dispersion of labels within the set.

Information Gain measures the decrease of entropy obtained by dividing the training set basing on the presence/absence of the term, thus preferring terms which produce subsets with more uniform labels. Basically it measures the discriminative power of a term, with respect to different classes; in other words it measures the effectiveness of an attribute in classifying the training data. In fact, given a term w and a set of data, labelled as positive (S_+) and negative (S_-) examples, the optimal case for the information gain value of w is represented by the situation in which all the documents containing w belong to a single specific class, say S_+, and all the documents which do not contain w belong to S_-, (in our case the entropies of the two sets of documents $H(D_w)$ and $H(D_{\bar{w}})$ would be 0 and the information gain $ig(w)$ maximum ($ig(w) = H(D)$)). This method basically allows to select terms with the highest discriminatory power among a set of classes.

Once basic terms have been defined, a vocabulary of terms T can be created from the set of training documents D, containing all the terms which occur at least once in the set. A single document d will be represented as a vector of weights $w_1, \ldots, w_{|T|}$, where the weight w_i represents the amount of information which the i^{th} term of the vocabulary carries out with respect to the semantics of d. We tried different types of weights, with increasing degree of complexity:

- a *binary* weight *δ(w,d)* indicating the presence/absence of the term within the document;
- a *term-frequency* weight tf(w,d) indicating the number of times the term occurs within the document, which should be a measure of its representativeness of the document content;
- a combination of *information gain* and *term-frequency* ($ig(w,d) \times tf(w,d)$);
- a *tf-idf* weight which indicates the degree of specificity of the term with respect to the document. Term Frequency Inverse Document Frequency (Buckley and Salton 1988) is computed as

$$tfidf(w,d) = tf(w,d) \times log(|D_w|^{-1})$$

where $|D_w|$ is the fraction of training documents containing at least once the term w. The rationale behind this measure is that term frequency is balanced by *inverse document frequency*, which penalizes terms occurring in many different documents as being less discriminative.

A wide range of experiments was conducted over a dataset made of 258 examples (text fragments containing Rules), collected by legal experts, distributed among 6 classes representing as many types of provisions (Table. 13.1).

Table 13.1 Dataset of provision types

Class labels	Provision types	Number of documents
c_0	Definition	10
c_1	Liability	39
c_2	Prohibition	13
c_3	Duty	59
c_4	Permission	15
c_5	Penalty	122

After terms preprocessing, we tried a number of combinations of the document representation and feature selection strategies previously described. We employed a *leave-one-out* (loo) procedure for measuring performances of the different strategies and algorithms. For a dataset of n documents $D = \{d_1, \ldots, dn\}$, it consists of performing n runs of the learning algorithm, where for each run i the algorithm is trained on $D \backslash d_i$ and tested on the single left out document d_i. The loo accuracy is computed as the fraction of correct tests over the entire number of tests. Table 13.2 reports loo accuracy and train accuracy, which is computed as the average train accuracy over the loo runs, of the Multiclass Support Vector Machine algorithm for the different document representation and feature selection strategies. The first three columns (apart from the index one) represent possible preprocessing operations. The fourth column indicates the term weighting scheme employed (binary (δ), term frequency (*tf*), infogain $\times$ term frequency ($ig \times tf$), term frequency-inverse document frequency (*tf–idf*)). The two following columns are for feature selection strategies: the unsupervised *min frequency* and the supervised *max infogain*, which

Table 13.2 Detailed results of MSVM algorithm for different document representation and feature selection strategies.

#	repl. digit	repl. alnum	Use stem	Weight scheme	Min freq sel.	Max IG sel.	Loo acc (%)	Train acc (%)
0	no	no	no	δ	2	500	89.53	100
1	yes	no	no	δ	2	500	88.76	100
2	yes	yes	no	δ	2	500	88.76	100
3	yes	yes	yes	tf	2	500	91.09	100
4	yes	yes	yes	tf-idf	2	500	89.15	100
5	yes	yes	yes	ig	2	500	89.15	100
6	yes	yes	yes	ig $\times$ tf	2	500	89.15	100
7	yes	yes	yes	δ	2	250	89.92	100
8	yes	yes	yes	δ	2	100	82.55	100
9	yes	yes	yes	δ	2	50	82.17	96.12
10	yes	yes	yes	δ	2	1000	90.31	100
11	yes	yes	yes	δ	0	500	92.24	100
12	yes	yes	yes	δ	2	500	92.64	100
13	yes	yes	yes	δ	5	500	92.24	100
14	yes	yes	yes	δ	10	500	89.92	100

Table 13.3 Confusion matrix for the best MSVM classifier

Classes	c_0	c_1	c_2	c_3	c_4	c_5
c_0	122	0	0	0	0	0
c_1	1	9	4	0	1	0
c_2	0	3	55	0	1	0
c_3	2	0	1	6	1	0
c_4	1	1	3	0	8	0
c_5	0	0	0	0	0	39

actually indicates the number of terms to keep, after being ordered by Information Gain. Finally, the last two columns contain loo and train accuracies.

Replacing digits or non alphanumeric characters does not improve performances, while the use of stemming actually helps clustering together terms with common semantics. The simpler binary weight scheme appears to work better then term frequency, probably for the small size, in terms of number of words, of the provisions in our training set; this fact makes statistics on the number of occurences of a term less reliable. Only slight improvements can be obtained by performing feature selection with Information Gain, thus confirming how SVM algorithms are able to effectively handle quite large feature spaces.

Finally, Table 13.3 shows the confusion matrix for the best classifier, the MSVM indexed 12, reporting details of predictions for individual classes. Rows indicate true classes, while columns indicate predicted ones. Note that most errors are committed for classes with fewer documents, for which poorer statistics could be learned.

13.4.1.2 Automatic Provision Arguments Extraction

A tool called xmLegesExtractor (Biagioli et al. 2005) of the xmLeges family has been implemented for the automatic detection of provision arguments.

The purpose of xmLegesExtractor[5] is to select relevant text fragments (lexical units) corresponding to specific semantic roles that are relevant for the different types of provisions. xmLegesExtractor is realized as a suite of Natural Language Processing tools for the automatic analysis of Italian texts (see Bartolini et al. 2004a, b, 2002), specialized to cope with the specific stylistic conventions of the legal parlance. A first prototype takes in input single legislative texts paragraphs in raw text, coupled with the categorization provided by the xmLegesClassifier, and identifies lexical units corresponding to provision arguments.

The approach follows a two–stage strategy. The first stage consists in a syntactic pre–processing which takes in input a text paragraph, 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 chunked sentence, however,

[5] xmLegesExtractor has been developed in collaboration with the Institute of Computational Linguistics (ILC-CNR) in Pisa (Italy)

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.

The second stage consists in a semantic annotation phase, basically in the identification of all the lexical units acting as arguments relevant to a specific provision type. It takes in input a chunked representation of legal text paragraphs and identifies semantically relevant structures by applying a specific provision type oriented grammar, locating relevant patterns of chunks which represent entities with specific semantic roles within a provision type instance Fig. 13.1.

Some experiments testing the reliability of xmLegesExtractor have been carried out on a dataset of 209 provisions.

The aim of this evaluation is to assess, for each provision type, the system reliability in identifying all the relevant semantic roles foreseen by the model. For each class of provisions in the dataset, the total number of semantic roles to be identified are collected in a gold standard dataset; 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. Some results are reported in Table. 13.4: here, Precision is scored as the number of correct answers returned by system over

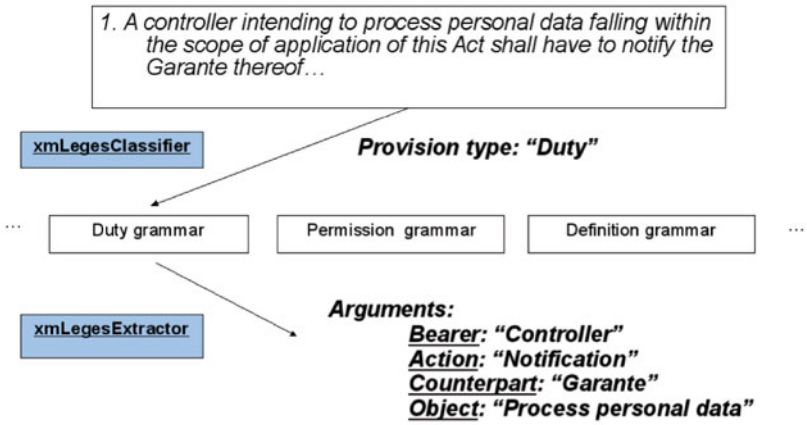

Fig. 13.1 The combination of xmLegesClassifier output and the grammar approach used by xmLegesExtractor

Table 13.4 xmLegesextractor experiments

Class labels	Provision type	Dataset	Precision	Recall
c_2	Prohibition	13	85.71%	92.30%
c_3	Duty	59	69.23%	30.50%
c_4	Permission	15	78.95%	100.00%
c_5	Penalty	122	85.83%	89.34%
	Total	209	82.80%	73.68%

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

13.4.2 DK Construction

Lexical units and their roles within a provision identified by xmLegesExtractor represent language-dependent lexicalizations of provision arguments. More information on the identified entities, as well as their relations within a specific domain, can be obtained by mapping lexical units to concepts in existing Domain Knowledges (DKs), if any. On the other hand such information can be consider as a ground to construct domain knowledges (in terms of thesauri or domain ontologies).

Actually the construction of DKs is not a specific task of *legal* ontologists, but of ontologists *tout court*, since a Domain Knowledge has to contain information on entities of the domain independently form a legal perspective. This is an important aspect to underline, in order to design a knowledge architecture whose components can be reused.

A DILK-DK learning approach only suggests language-dependent lexical units for DKs, which can be implemented by projecting lexical units on a large text corpora of a specific domain, inferring conceptualizations by term clustering, as well as using statistics on recurrent patterns for discovering term relationships. This issue is out of the paper scope; a vast literature exists on this topic, therefore the interested reader can refer to (Buitelaar and Cimiano 2008).

13.5 Conclusions

A knowledge modelling approach for the legal domain, called DILK-DK approach, has been presented. It aims to keep distinct domain knowledge from its legal perspective. Moreover an automatic approach based on machine learning and NLP techniques to support bottom–up knowledge acquisition from legislative texts within the DILK-DK framework has been shown.

The proposed learning approach for legal knowledge acquisition can provide several benefits:

- it contributes to implement taxonomies or suggest concepts for hand-crafted ontologies (Walter and Pinkal 2009; Lenci et al. 2009)
- it contributes to bridge the gap between authoritativeness and consensus for legal rules representation, since it is able to extract rules directly form legislative texts, which are authoritative sources (by definition), nevertheless promoting consensus, since rules are automatically extracted from legal sources, limiting human interaction.

References

Agnoloni, T., L. Bacci, E. Francesconi, W. Peters, S. Montemagni, G. Venturi (2009). A Two-Level Knowledge Approach to Support Multilingual Legislative Drafting. In J. Breuker, P. Casanovas, M. Klein, E. Francesconi (Eds.) *Law, Ontologies and the Semantic Web*, vol. 188 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam, 177–198.

Bacci, L., P. Spinosa, C. Marchetti, R. Battistoni (2009). Automatic Mark-Up of Legislative Documents and Its Application to Parallel Text Generation. In N. Casellas, E. Francesconi, R. Hoekstra, S. Montemagni (Eds.) *Proceedings of the 3rd Workshop on Legal Ontologies and Artificial Intelligence Techniques joint with 2nd Workshop on Semantic Processing of Legal Texts*. Huygens Editorial, Barcelona, 45–54.

Bartolini, R., A. Lenci, S. Montemagni, V. Pirrelli (2002). The Lexicon-Grammar Balance in Robust Parsing of Italian. In *Proceedings of 3rd International Conference on Language Resources and Evaluation*. Las Palmas, Canary Islands, Spain.

Bartolini, R., A. Lenci, S. Montemagni, V. Pirrelli, C. Soria (2004a). Automatic Classification and Analysis of Provisions in Italian Legal Texts: A Case Study. In *Proceedings of the Second International Workshop on Regulatory Ontologies*. Larnaca, Cyprus.

Bartolini, R., A. Lenci, S. Montemagni, C. Soria (2004b). Semantic Mark-Up of Legal Texts Through Nlp-Based Metadata-Oriented Techniques. In *Proceedings of 4rd International Conference on Language Resources and Evaluation*. Lisbon, Portugal

Bentham, J., H.L.A. Hart (1970). *Of Laws in General.* Athlone, London, (1st ed. 1872).

Biagioli, C. (1991). Definitional Elements of a Language For Representation of Statutory. *Rechtstheorie*, 11: 317–336.

Biagioli, C. (1997). Towards a Legal Rules Functional Micro-Ontology. In *Proceedings ofworkshop LEGONT '97*. Melbourne, Australia.

Biagioli, C., F. Turchi. (2005). Model and Ontology Based Conceptual Searching in Legislative Xml Collections. In *Proceedings of the Workshop on Legal Ontologies and Artificial Intelligence Techniques*, Bologna, Italy, 83–89.

Biagioli, C., E. Francesconi, A. Passerini, S. Montemagni, C. Soria (2005). Automatic Semantics Extraction in Law Documents. In *Proceedings of International Conference on Artificial Intelligence and Law*, Bologna, Italy, 133–139.

Breuker, J., R. Hoekstra (2004a). Core Concepts Of Law: Taking Common-Sense Seriously. In *Proceedings of Formal Ontologies in Information Systems*. Torino, Italy.

Breuker, J., R. Hoekstra (2004b). Epistemology and Ontology In Core Ontologies: Folaw and lricore, Two Core Ontologies For Law. In *Proceedings of EKAW Workshop on Core ontologies*. CEUR. Whittlebury Hall, Northamptonshire, UK.

Breuker, J., S. van de Ven, A. El Ali, M. Bron, S. Klarman, U. Milosevic, L. Wortel, A. Forhecz (2008). Developing Harness. ESTRELLA Deliverable 4.6/3b, European Commission.

Breuker, J., P. Casanovas, M. Klein, E. Francesconi (Eds.) (2009). *Law, Ontologies and the Semantic Web. Channelling the Legal Information Flood*, vol. 188 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam.

Buckley, C., G. Salton (1988). Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing and Management*, 24(5): 513–523.

Buitelaar, P., P. Cimiano (Eds.) (2008). *Ontology Learning and Population: Bridging the Gap Between Text and Knowledge*, vol. 167 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam.

Buitelaar, P., P. Cimiano, B. Magnini (2005). Ontology Learning From Text: An Overview. In Buitelaar et al. (Eds.) *Ontology Learning from Text: Methods, Evaluation and Applications*, vol. 123 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam, 3–12.

Bylander, T., B. Chandrasekaran (1987). Generic Tasks for Knowledge-Based Reasoning: The "Right" Level Of Abstraction For Knowledge Acquisition. *International Journal of Man-Machine Studies*, 26(2): 231–243.

Bench Capon, T.J.M., P.R.S. Visser (1997). Ontologies in Legal Information Systems; The Need For Explicit Specifications of Domain Conceptualizations. In *Proceedings of the 6th International Conference on Artificial Intelligence and Law*. ACM Press, New York, NY, 132–141.

Casellas, N. (2008). *Modelling Legal Knowledge through Ontologies. OPJK: The Ontology of Professional Judicial Knowledge*. Ph.D. thesis, Institute of Law and Technology, Autonomous University of Barcelona.

Chandrasekaran, B. (1986). Generic Tasks in Knowledge-Based Reasoning: High-Level Building Blocks for Expert System Design. *IEEE Expert*, 1(3): 23–30.

Cimiano, P. (2006). Ontology Learning and Population From Text. In *Algorithms, Evaluation and Applications*. Springer, Berlin.

Clancey, W.J. (1981). The Epistemology of a Rule-Based Expert System: A Framework for Explanation. Technical Report STAN-CS-81-896, Stanford University, Department of Computer Science.

Euzenat, J., P. Shvaiko (2007). *Ontology Matching*. Springer, Berlin.

Francesconi, E., A. Passerini (2007). Automatic Classification of Provisions in Legislative Texts. *International Journal on Artificial Intelligence and Law*, 15(1): 1–17.

Francesconi, E., S. Faro, E. Marinai (2008). Thesauri Alignment for Eu Egovernment Services: A Methodological Framework. In *Proceedings of the JURIX 2008 Conference*. IOS Press, Amsterdam, 73–77.

Gangemi, A., N. Guarino, C. Masolo, A. Oltramari, L. Schneider (2002). Sweetening Ontologies With Dolce. In A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, L. Schneider (Eds.) *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW02)*, LNCS, vol. 2473. Springer, Sigüenza, Spain.

Gruber, T. (2006). Where the Social Web Meets the Semantic Web (Keynote Abstract). In I.F. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, L. Aroyo (Eds.) *The Semantic Web – ISWC 2006, Proceedings of the 5th International Semantic Web Conference*, LNCS, vol. 4273. Springer, Berlin, 994.

Guarino, N. (1997). Semantic Matching: Formal Ontological Distinctions For Information Organization, Extraction, and Integration. In M.T. Pazienza (Ed.) *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*, LNCS, vol. 1299. Springer, Berlin, 139–170.

Hart, H. (1961). *The Concept of Law*. Clarendon Law Series. Oxford University Press, Oxford.

Hoekstra, R., J. Breuker, M. Bello, A. Boer (2009). Lkif Core: Principled Ontology Development for the Legal Domain. In J. Breuker, P. Casanovas, M. Klein, E. Francesconi (Eds.) *Legal Ontologies and the Semantic Web*. IOS Press, Amsterdam.

Hohfeld, W.N. (1913). Some Fundamental Legal Conceptions as Applied in Judicial Reasoning. I. *Yale Law Journal*, 23: 16–59.

Hohfeld, W.N. (1917). Some Fundamental Legal Conceptions as Applied in Judicial Reasoning. II. *Yale Law Journal*, 26: 710–770.

Kelsen, H. (1991). *General Theory of Norms*. Clarendon Press, Oxford.

Lame, G. (2005). Using Nlp Techniques to Identify Legal Ontology Components: Concepts and Relations. *Lecture Notes in Computer Science*, 3369: 169–184.

Lenci, A., S. Montemagni, V. Pirrelli, G. Venturi (2009). Ontology Learning from Italian Legal Texts. In J. Breuker, P. Casanovas, M. Klein, E. Francesconi (Eds.) *Law, Ontologies and the Semantic Web*, vol. 188 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam, 7594.

Quinlan, J.R. (1986). Inductive Learning of Decision Trees. *Machine Learning*, 1: 81–106.

Rawls, J. (1955). Two Concepts of Rule. *Philosophical Review*, 64: 3–31.

Raz, J. (1977). *Il Concetto di Sistema Giuridico*. Il Mulino, Bologna.

Ricciardi, M. (1997). Constitutive Rules and Institutions. In *Meeting ofthe Irish Philosophical Club and the Royal Institute ofPhilosophy,* Ballymanscanlon.

Ross, A. (1968). *Directives and Norms.* Routledge, London.

Saias, J., P. Quaresma (2005). A Methodology to Create Legal Ontologies in a Logic Programming Based Web Information Retrieval System. *Lecture Notes in Computer Science*, 3369: 185–200.

Searle, J.R. (1969). *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, Cambridge, MA.

Sebastiani, F. (2002). Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, 34(1): 1–47. URL http://faure.iei.pi.cnr.it/fabrizio/Publications/ACMCS02.pdf.

Studer, R., V. R. Benjamins, D. Fensel (1998). Knowledge Engineering: Principle and Methods. *Data Knowledge Engineering*, 25(1–2): 161–197.

van Heijst, G. (1995). *The Role of Ontologies in Knowledge Engineering*. Ph.D. thesis, Social Science Informatics, University of Amsterdam.

Walter, S., M. Pinkal (2006). Automatic Extraction of Definitions From German Court Decisions. In *Proceedings of the COLING-2006 Workshop on Information Extraction Beyond The Document*, Sidney, 20–28.

Walter, S., M. Pinkal (2009). Definitions in Court Decisions – Automatic Extraction and Ontology Acquisition. In J. Breuker, P. Casanovas, M. Klein, E. Francesconi (Eds.) *Law, Ontologies and the Semantic Web*, vol. 188 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam, 95–113.

Yang, Y., J.O. Pedersen (1997). A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc., San Mateo, CA, 412–420.