

# Frame Detection over the Semantic Web

Bonaventura Coppola<sup>2</sup>, Aldo Gangemi<sup>1</sup>, Alfio Gliozzo<sup>1</sup>,  
Davide Picca<sup>3</sup>, and Valentina Presutti<sup>1</sup>

<sup>1</sup> STLab, ISTC-CNR, Roma, Italy

{aldo.gangemi,alfio.gliozzo,valentina.presutti}@istc.cnr.it

<sup>2</sup> Department of Computer Science and Engineering, University of Trento, Italy  
coppola@disi.unitn.it

<sup>3</sup> Université de Lausanne, Switzerland  
davide.picca@unil.ch

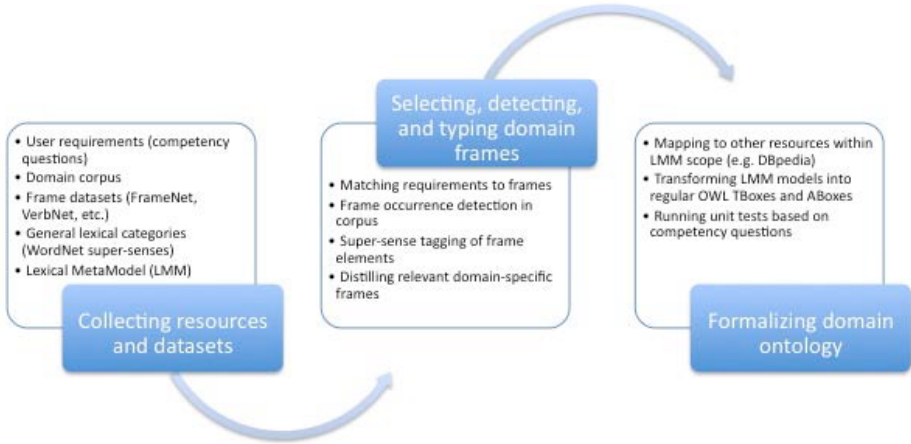
**Abstract.** In the past, research in ontology learning from text has mainly focused on entity recognition, taxonomy induction and relation extraction. In this work we approach a challenging research issue: detecting *semantic frames* from texts and using them to encode web ontologies. We exploit a new generation Natural Language Processing technology for frame detection, and we enrich the frames acquired so far with argument restrictions provided by a *super-sense* tagger and domain specializations. The results are encoded according to a Linguistic MetaModel, which allows a complete translation of lexical resources and data acquired from text, enabling custom transformations of the enriched frames into modular ontology components.

## 1 Introduction

Ontology learning from text has become an important functionality for ontology design, due to the demand for domain-specific knowledge that supports semantic solutions scaling from personal knowledge management to large organizations, without requiring substantial competence from domain experts or expensive assistance from knowledge engineers. This is specially true if we consider the large number of different domains involved in modelling knowledge at an organization or even at a Web scale.

The current state of art in ontology learning deals with taxonomical relations and simple binary domain relations, but research has just started dealing with more complex domain relations that can be able to answer realistic competency questions [20], such as *who is communicating what to whom and for what reason?*. Complex relations are called *semantic frames* in linguistics, and show interesting analogies with other data structures that are known in AI and the Semantic Web, such as *frames*, *microformats*, *infoboxes*, etc.

In this paper we present a methodology for detecting and learning frames and situations from textual domain corpora, for using them to generate modules of a domain ontology, and possibly for populating the ontology with named entities and facts extracted from the same corpora. The methodology aims at specializing and/or instantiating a set of frames from the FrameNet lexicon [3], by extracting and filtering configurations of data from text corpora. We want to use the



**Fig. 1.** A summary of the datasets, components, functionalities, and phases of the frame-based ontology learning method

extracted data in order to build domain-specific ontologies that show desirable characteristics such as cognitive and linguistic relevance, density, right granularity level, and inherent capability of being automatically evaluated based on expert requirements (in the form of competency questions). A previous attempt to achieve a similar objective has been reported by [4]. In contrast to that work we have used different text processing technology and knowledge representation standards.

The natural language processing technology for detecting occurrences of frames in corpora is presented in section 3.1), and their representation according to the FrameNet ontology is presented in section 2). Frame detection learns typical patterns from the occurrences in a corpus, providing raw material to start a domain adaptation process of frames by means of a WordNet Supersense tagging component (see section 3.2). The acquired information is then connected to a pile of reusable ontologies: the lexical ontologies OntoFrameNet [16] and OntoWordNet [18], DBpedia [1], and LMM [24,16], which allows a general representation model for lexical ontologies. The LMM representation is eventually converted into a regular ontology by means of *transformation patterns* (see section 4).

Results, described in the evaluation section (4) show that our workflow is sustainable and cost effective, being the speed of the adopted technology fast enough to work with mid-size corpora in reasonable time, and the accuracy of the data distilled at the end of the acquisition process good enough to be easily checked by an expert (or even automatically) with reference to competency questions. This turns out in a substantial reduction of the manual effort required for designing domain-specific ontologies that are design-ready to be evaluated against user requirements. Figure 1 summarizes the main components and functionalities involved in our workflow.

## 2 Representing Frames in LMM

Frame Semantics [15] is a linguistic theory that attempts to represent the conceptual interface between linguistic and real-world knowledge in the form of *semantic frames*, which are structures that describe particular types of situations, objects, or events along with the roles of their participating elements. Based on Frame Semantics, FrameNet [3] is a lexical knowledge base, consisting of a set of *frames*, which have proper *frame elements* and *lexical units*. Frame elements are unique to their frame, and can be optional ('non-core'). An occurrence of a frame consists in some piece of text whose words can be normalized as *lexemes*, and which have semantic roles dictated by the elements of that frame. A frame usually has only some of its roles actually lexicalized in texts. Types of linguistic realizations of a frame are called *lexical units*. Frames respectively frame elements are related between them, e.g. through the *subframe* relation. For example, a short definition of **CommerceScenario** in FrameNet is reported below:

### CommerceScenario

*Core Elements:* **Buyer, Goods, Money, Seller**

*non-Core Elements:* **Manner, Means, Purpose, Rate**

*Subframes:* **CommercialTransaction**

The intuition underlying Frame Semantics is not very dissimilar from that described in [22], but the latter developed in the AI context, and eventually acquired a formal semantics, as in [7]. [16] is a reconstruction of different semantics of frames, and how they can be reconciled. As an assumption that can be adopted for all different approaches to frame semantics, [16] suggests that frames can be represented as (intensional) n-ary relations, with typed arguments (either mandatory or optional). For example,

$$\text{Desiring}(x, y, e) \rightarrow \text{Agent}(x) \wedge \text{Agent}(y) \wedge \text{Event}(e) \quad (1)$$

An occurrence of a frame can be straightforwardly treated as an instance of an n-ary relation, e.g.:

$$\text{Desiring}(\text{Susan}, \text{Marko}, \text{ListeningToHer}) \quad (2)$$

The logical representation of frames as n-ary relations is easily generalizable and clear, but hardly manageable by automated reasoners on large knowledge bases. A hard design problem is constituted by the polymorphism of many frames/n-ary relations, which can vary in number of the arguments that can be taken by the relation. For example, the same frame **Desiring** can be assumed with four arguments:

$$\text{Desiring}(x, y, e, t) \rightarrow \text{Agent}(x) \wedge \text{Agent}(y) \wedge \text{Event}(e) \wedge \text{Time}(t) \quad (3)$$

This problem was originally evidenced by Davidson with reference to a logic of events [12]. A neo-davidsonian approach is actually taken by [6], where an attempt is made to reconstruct FrameNet in terms of DRT (Discourse Representation Theory), with *events* being the DRT correlate of frames, and *semantic roles* being generic correlates for equivalence classes of frame elements.

A limitation found by [6] lies in the fact that DRT semantic primitives are tightly associated with elements having a given syntactic category (e.g. it assumes that an event cannot be expressed by an adverb), while FrameNet semantics only requires that any grammatical *construction* expresses a frame, independently from its syntactic category. The DRT approach has anyway the advantage of simplifying some assumptions of Frame Semantics, for example the uniqueness of frame elements to their frame. A formal semantics for frames that is also computationally manageable has been provided by Description Logics (DL) [2], such as OWL(DL). Due to their limited expressive power, description logics represent frames as classes, with *roles* (binary relations) that link a class to the types of the arguments of the original n-ary relation. Those types are classes as well, so that a graph of frames emerges out of this semantics. The example in (3) can be reengineered in DL as follows:

$$T \sqsubseteq \forall R_1.\text{Agent}, T \sqsubseteq \forall R_1^-. \text{Desiring} \quad (4)$$

$$T \sqsubseteq \forall R_2.\text{Agent}, T \sqsubseteq \forall R_2^-. \text{Desiring} \quad (5)$$

$$T \sqsubseteq \forall R_3.\text{Event}, T \sqsubseteq \forall R_3^-. \text{Desiring} \quad (6)$$

$$T \sqsubseteq \forall R_4.\text{Time}, T \sqsubseteq \forall R_4^-. \text{Desiring} \quad (7)$$

$$\text{Desiring} \sqsubseteq (=1R_1 \sqcap =1R_2 \sqcap =1R_3 \sqcap =1R_4) \quad (8)$$

The computational features of description logics make them a reasonable choice to formally represent linguistic frames, and this is the approach followed e.g. by [27]. On the other hand, description logics cannot be used to formalize the meta-level part of the intended semantics of frames: relations between frames, between frame elements, and between frames and frame elements, lexical units, and lexemes can be hardly represented in a DL.<sup>1</sup> The Lexical Meta-Model (LMM) [24] contains a viable solution to keep both the (reified) meta-level constructs of FrameNet in an OWL(DL) ABox, and to deliver custom views of frames in an OWL(DL) TBox when needed. LMM provides a formal semiotic representation of lexical knowledge. It is implemented in OWL(DL)<sup>2</sup>. LMM facilitates the integration of heterogeneous lexical knowledge resources, such as WordNet [14] and FrameNet<sup>3</sup>, as well as of other semi-formal resources (thesauri, subject directories, etc.), with regular ontologies expressed in a formal language like OWL. LMM can be considered as a *semiotic façade* enabling the reengineering of any lexically-based resource to an ontology resource [16]. In the methodology reported in this paper, it is used to make different lexical data interoperable, and to transform the output of the three learning components *frame detector*, *super-sense tagger*, and *distiller* (see next sections) into an OWL ontology. The core LMM elements are `lmm:Expression`, which can `lmm:express` a `lmm:Meaning` (that is `lmm:conceptualizedBy` `lmm:Agent`), and which can

<sup>1</sup> F-logic [21] is another alternative that however has no direct correspondence to current standard semantic web languages.

<sup>2</sup> <http://www.ontologydesignpatterns.org/ont/lmm/lmm1.owl>

<sup>3</sup> The OWL ontology that encodes the alignment of OntoFrameNet to LMM can be downloaded at <http://www.ontologydesignpatterns.org/ont/lmm/ofn2lmm.owl>

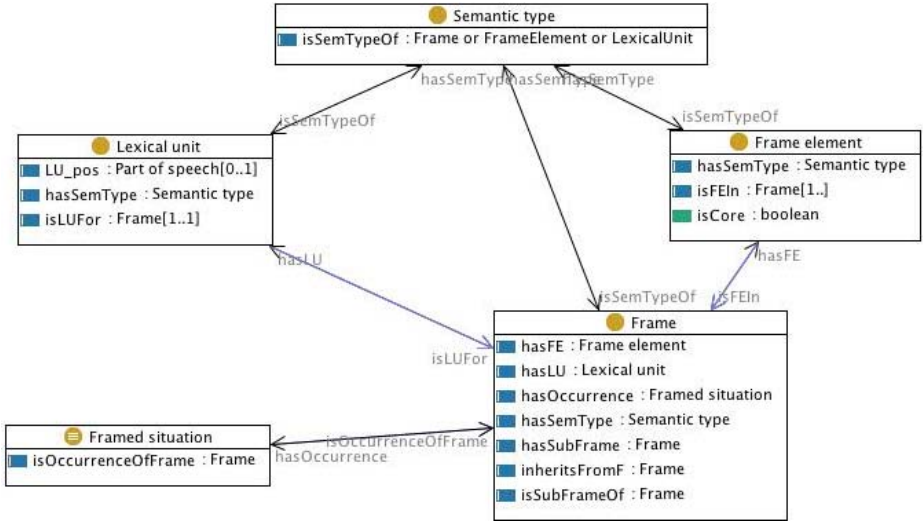


Fig. 2. OntoFrameNet: A full-fledged metamodel for FrameNet

`lmm:denote` a `lmm:Reference`; in addition, any entity can have a `lmm:Context` and can be `lmm:representedBy` a `lmm:FormalExpression`. Expressions can be any information object, and lexical items in particular; references can be any entity; meanings can be any information or conceptual entity that is used to describe or encode the (informal, natural) semantics of an expression, i.e. other expressions, concepts from a thesaurus, frames, topics, etc. A rich set of relations is defined in LMM to associate meanings between them, as well as to associate expressions, meanings, and references to formal constructs. A complete reengineering of FrameNet as a plugin to LMM can be found in the OWL version of OntoFrameNet<sup>4</sup>[16] (Fig. 2). A `ofn:Frame` has some `ofn:FrameElement`, is lexically realized by some `ofn:LexicalUnit`, and all these can be evoked by some `ofn:Lexeme`, and can have some `ofn:SemanticType`. Since semantic types do not cover all frame elements and lexical units,<sup>5</sup> we use the super-sense tagger component (section 3.2) to infer WordNet `wn:SuperSenses` as additional types for frame elements. All mentioned classes and relations are aligned to LMM; an excerpt of alignments is included here:<sup>6</sup>

$$\text{ofn:Lexeme} \text{ rdfs:subClassOf } \text{lmm:Expression} \quad (9)$$

$$\text{ofn:Frame} \text{ rdfs:subClassOf } \text{lmm:Meaning} \quad (10)$$

$$\text{ofn:FrameElement} \text{ rdfs:subClassOf } \text{lmm:Meaning} \quad (11)$$

<sup>4</sup> <http://www.ontologydesignpattern.org/ont/ofn/ofntb.owl>

<sup>5</sup> Currently the majority of frames (719 out of 795) has types for one or more of their frame elements (4187 out of 7124), while no lexical unit (about 10000) has semantic types assigned to the application of frame elements in their context.

<sup>6</sup> <http://www.ontologydesignpattern.org/ont/lmm/ofn2lmm.owl>

`ofn:evokes rdfs:subClassOf lmm:expresses` (12)

`ofn:FramedSituation rdfs:subClassOf lmm:Reference` (13)

A thorough discussion on alternative formal translations of FrameNet is contained in [16].

### 3 The Frame Learning Components

In this section, we propose an ontology learning method aimed at acquiring frames and situations from text. This is a novel method involving challenging parsing capabilities. Ontology learning from text is usually performed with three components: a text corpus of an arbitrary size, a set of lexical resources, and some algorithms to extract, rank, and reengineer selected invariances from texts. Preliminary text classification may be also performed [5]. In the literature, most of the effort has concentrated in learning the terminology for a specific domain, taxonomies of concepts and their instances, relations between instances, trying to augment and adapt the information contained in e.g. WordNet with domain specific data. Unfortunately, this is not enough for our purposes, since we are trying to learn structures with multiple related elements in the context of a frame, as occurring in a particular situation. Therefore, we need text processing components that are able to identify occurrences of frames in text, and to recognize the corresponding frame elements and their types, also allowing for the learning of domain-specific specializations and instantiations.

To this aim, we have integrated two advanced text processing components: a frame detector and a super-sense tagger, described in the following subsections. As a result, portions of text corresponding to lexical units and frame elements are collected for each frame, which describe particular *framed situations* within a text.

As an example of the workflow, we use here the **JudgmentCommunication** frame. A sample annotated sentence follows:

(S383) Public opinion and the press nowadays accuse us of being unavailable to give a response.

FRAME: **JudgmentCommunication**

**Communicator:** "Public opinion" noun.cognition Concept  
"and the press" noun.group-Concept"

**Target:** "accuse"\_verb.communication

**Evaluee:** "us"

**Reason:** "of being" verb.stative "unavailable"  
"to give" verb.possession "a response" noun.phenomenon-Concept"

The **Target** frame element is present in all FrameNet frames, and refers to the lexical unit (e.g. *accuse.v* in the previous example) that abstracts from the words that realize the frame in a particular text occurrence. This is actually used to assign that particular occurrence to one of the frames in FrameNet.

The remaining slots describe the frame elements identified in a text, and their corresponding types, as recognized by the super-sense tagger. We eventually

learn domain frames by matching the frame element names to the corresponding super-senses, and by filtering out low-ranked results. This process is described in subsection 3.3.

### 3.1 The Frame Detection System

The Frame Detection component takes care of analyzing plain English input text, locating possible candidate frames invoked along by the terms in a text, and then tagging individual Frame Element instances of such frames. Tagging involves both locating the exact text boundary of each Frame Element, and assigning its correct semantic label.

The Berkeley FrameNet Project currently includes the definitions of nearly 800 Frames, 4,000 Frame Elements, and 135,000 annotated English sentences. An example of sentence annotation for the **CommerceScenario** is reported hereafter:

*Ralemborg said [he]<sub>SELLER</sub> already had a [buyer]<sub>BUYER</sub> [for the wine]<sub>GOODS</sub>*

where the underlined word *buyer* is the target word (or *lexical unit* or predicate) which plays the role of *evoker* for this particular Frame.

To implement a FrameNet parsing system we adopted a multi-stage classification approach over natural language. Foundational studies in this area generally refer to Semantic Role Labeling [19]. In particular, our strategy extends the approaches in [23] to FrameNet and it is thoroughly described in [11]. Shortly, moving from previous pattern-based approaches [28], we exploit Support Vector Machines (SVMs) as a general statistical machine learning framework, in which we plug in Tree Kernels to obtain a similarity measure capable of working over structured objects as syntactic trees. Our strategy includes (1) **Target Word Detection**, where the semantically relevant words bringing predicative information are detected; (2) **Frame Disambiguation**, where the correct Frame for any target word has to be selected among several candidates; (3) **Boundary Detection**, where the sequences of words constituting the Frame Elements (arguments) are detected; and (4) **Role Classification**, which assigns semantic labels to the Frame Elements detected in the previous stage.

### 3.2 The Super-Sense Tagger

In this work, we exploit the Super-Sense Tagger (SST) described in [9] for a deeper semantic interpretation of texts finalized to the acquisition of structured knowledge in a frame-based representation. Supersenses are lexicographer's categories, used to provide an initial broad classification for the lexicon entries in WordNet [14]. Although simplistic in many ways, the supersense ontology has several attractive features for ontology engineering. In fact, supersenses correspond to the typical high level classes of most ontologies, reflecting distinctions between persons, locations, organizations, acts, etc. In this work we use supersenses to characterize the type of Frame Elements. Exploiting SST, we identify types in frame occurrences by filling the frame slots that are identified by the frame detection component.

SST<sup>7</sup> has been implemented using the SemCor corpus (a fraction of the Brown corpus annotated with WordNet word senses) and can be used for annotating large collections of English texts [8]. The tagger implements a Hidden Markov Model, trained with the perceptron algorithm introduced in [10]. The tagset used by the tagger defines 26 supersense labels for nouns and 15 supersense labels for verbs. The tagger outputs named entity information, but also covers other relevant categories and attempts lexical disambiguation at the supersense level. In recent work, we have extended the tagset by subdividing each category into two sub-categories, *Instance* and *Concept* so that now the term “president” is tagged as *noun.person\_Concept* and the term “Bill Clinton” as *noun.person\_Instance*. An example of the output of this tagger is reported below.

```
GunsB–noun.group_instance andI–noun.group_instance RosesI–noun.group_instance
playsB–verb.communication atO theO stadiumB–noun.location_concept
```

### 3.3 Distilling Knowledge from Tagged Frames

The output of the frame detector component provides a shallow semantic parsing of text, where text fragments are associated with frame elements and the lexical units are identified. After joining these data along with the tagging provided by the SST, the final output of the text processing components is illustrated in the example below:

FRAME: CommerceScenario

```
Buyer:"People_noun.group_Instance who
      read_verb.cognition this warning_noun.communication"
Target:"buy_verb.possession"
Goods:"these toys_noun.artifactInstance"
Recipient:"for their children_noun.person_Concept or
          grandchildren_noun.person_Concept"
```

The next step consists in converting this format into a formal representation. To this aim, the *distiller* component involves both semantic and statistical criteria:

**Syntactic constraints.** Occurrences of frame elements should only be nominals or noun phrases, while target lexemes can be verbs, nouns or adjectives

**Type constraints.** Most of the occurring elements that are recognized by the frame detector belong to a rather small number of super-senses (e.g. in a given domain buyers are either persons or groups, less likely they could be animals)

**Redundancy constraints.** Redundancy helps in recognizing fine-grained restrictions for frame elements with very high accuracy. Only those concepts occurring in the context of a particular frame with a certain degree of redundancy can be regarded as “valid” subtypes.

<sup>7</sup> The tagger is publicly available at:

<http://sourceforge.net/projects/supersensetag/>



At this stage, we have implemented a simple thresholding mechanism on frequency-based countings in order to model those assumptions, and in particular:

- The implementation of the syntactic constraints directly comes from the SST annotation, which distinguishes nouns, verbs and adjectives, and recognizes compound terms.
- Concerning type constraints, for each occurrence of a frame element, we count the occurrences of all different super-senses associated with the matched arguments, and we filter out those having less than 10 occurrences. For example, for the frame KILLING the possible types of the element VICTIM are as follows: `group_concept`: 90, `person_concept`: 79, `person_instance`: 41.
- Concerning redundancy constraints, for any occurrence of the frame elements, we keep the terms belonging to the super-senses selected at the first step. Since the SST distinguishes between concepts and instances, we select only those terms belonging to the concept types having more than three occurrences. For example, we have found that occurrences of the frame element VICTIM belonging to the **person\_concept** super-sense include notions like *child*, *woman* and *civilian*, while occurrences categorized by the **noun.state** super-sense, e.g. *recognition*, have a sensibly lower frequency, and are most likely to be errors.

The result of the three steps above is a list of “qualified”, domain-oriented semantic types for domain-specific frame element that can be easily leveraged to improve the quality of the frame representation. As a final step, we are interested in detecting specializations of frames. To this aim, we can adopt the results of the frame detector component, applying each of the cleaning steps described above. Among all possible occurrences of a certain frame in the corpus, we select only those having at least three of their element occurrences satisfying the constraints above, and we regard them as candidate *domain-specific* frames, like in the following case, accompanied by its generalization:

```
FRAME: Killing : Victim:smoker ; Killer:tobacco
FRAME: Killing : Victim:person ; Killer:artifact
```

The described procedure is fully automatic, and we have adopted the same threshold uniformly for three case study frames under analysis in this paper (Section 4.2). The results of this method include a set of domain-specific frames that can be candidate data for answering a competency question, and therefore for implementing an ontology that fits the user requirements.

## 4 Empirical Evaluation

### 4.1 NLP evaluation

**Frame Detection Evaluation.** We have tested the Frame Detection Component in the natural setting of the FrameNet Corpus. Version 1.3 of FrameNet was used for both learning and test. After preprocessing and parsing with the Charniak’s constituency-based syntactic parser we obtained 135,293 annotated and

parsed sentences. We split the data considering the part of speech of predicates, ending up with 782 different frames. The overall dataset was then partitioned into 90% training set and 10% test set.

We tested the plain polynomial kernel (*poly*, with a degree of 3) and the SubSet Tree Kernel (SST) respectively over standard [19] and structured features [23]. Also, their combination was tested. Please see [11] for full details.

Table 1 reports Precision, Recall and  $F_1$  measure of our classifiers over different tasks. The four rows in the table show in turn: (1) the “pure” performance of the Boundary Detection (BD) classifier, i.e. considering correct the classification decisions also when a correctly classified tree *node* does not exactly correspond to a valid sentence constituent. Such mismatch frequently happens when the parse tree (which is automatically generated) includes incorrect nodes and attachments; (2) the performance of the BD classification “projected” on the tree leaves, i.e. when matching not only the constituent node as in 1, but also the selected *words* (leaves) with those in the FrameNet gold standard. This implies an exact syntactic analysis being realized in the subtree; (3) the same as 1, with the argument role classification (RC) also performed (i.e. Frame Element labels must match as well); (4) the same as 2, with RC also performed.

As shown in the table, the SST+*poly* kernel achieves 1.0 Precision, 0.732 Recall and 0.847  $F_1$  on NS. These figures can be compared to 0.855 Precision, 0.669 Recall and 0.751  $F_1$  of the system described in [13], achieved with the same amount of training data. In conclusion, our best learning scheme is currently capable of tagging FrameNet data from noisy syntax with exact boundaries and role labels at 0.63  $F_1$ .

**Table 1.** Results for Frame Detection on the FrameNet dataset. SST+*poly* with 90% training and 10% test data.

Enhanced SST + poly			
Eval Setting	<i>P</i>	<i>R</i>	$F_1$
BD (nodes)	1.0	.732	.847
BD (words)	.963	.702	.813
BD+RC (nodes)	.784	.571	.661
BD+RC (words)	.747	.545	.630

**Super-Sense Tagger Evaluation.** We evaluated the performances of SST by adopting a n-fold cross validation strategy on the SemCor corpus exploited for training. Results for the chosen categories are illustrated in Table 2, reporting Precision, Recall and  $F_1$  for any Supersense. Looking at the table we can notice that for some categories the  $F_1$  is exceptionally high. Some of those best classified categories are really essential for ontology learning. For example, important labels such as *noun.person*, *noun.group* or *noun.location* achieve results higher than 0.70. We obtained a general  $F_1$  of 0.76. For some categories we got a  $F_1$  over 0.80: e.g. *noun.person\_Instance*:  $F_1$  0.90) or *noun.person\_Concept*:  $F_1$  0.81, etc.

**Table 2.** Super-Sense Tagger Evaluation Results: Recall, precision and F1 for each classification category

Category	Recall	Precision	F <sub>1</sub>
noun.animal_Concept	0.712	0.766	0.738
noun.animal_Instance	0.416	0.793	0.545
noun.artifact_Concept	0.726	0.737	<b>0.731</b>
noun.artifact_Instance	0.596	0.646	0.620
noun.food_Concept	0.687	0.720	0.703
noun.food_Instance	0.444	0.500	0.457
noun.group_Concept	0.729	0.731	<b>0.730</b>
noun.group_Instance	0.683	0.703	0.693
noun.location_Concept	0.682	0.653	0.667
noun.location_Instance	0.752	0.800	<b>0.775</b>
noun.person_Concept	0.838	0.804	0.821
noun.person_Instance	0.927	0.881	<b>0.904</b>

## 4.2 Knowledge Engineering Evaluation

We have tested our methods on the Europarl corpus<sup>8</sup>, comprising about 30 million documents, extracted from the proceedings of the European Parliament. It includes 11 official languages of the European Union. For our experiments, we have used the English part, including 1,461,429 sentences and 39,618,240 words. In step 1, sentences from the text corpus are annotated with super-senses by the SST, and with frames, frame elements, and lexical units by the Frame Detector component. For the sake of this experiment, we focused on the annotation of three frames: **Killing**, **JudgmentCommunication**, and **Commerce**. Results for each frame have been collected in tables reporting, for each frame occurrence recognized in the corpus, all the frame elements and their types, as recognized by SST. Occurrences of core frame elements show meaningful frequency in the corpus, so confirming the hypothesis that some elements are more central than others in the conceptualization of frames. About 1000 occurrences for each frame have been collected after step 1.

In step 2, the raw frame occurrence data have been filtered by applying the procedure described in section 3.3, consisting of identifying the most frequent super-senses for each frame element, in order to learn more specific types (e.g. **Person** or **Group**, and to filter out instances having different types (all super-senses can be assumed as disjoint classes of synsets). After applying the automatic distiller procedure based on combined frequency thresholds, data from only about 100 frame occurrences have been retained as candidate domain-specific frames for each generic frame. In addition to types, step 2 allows to learn sensible lexemes for the domain specialization of frame elements (e.g. *people*), and sensible lexical units (e.g. **Accuse** for the candidate frame specializations. An example of the outcome of the distiller after evaluation is shown in Table 3. Each distilled

<sup>8</sup> <http://www.iccs.inf.ed.ac.uk/~pkoeHN/publications/europarl-mtsummit05.pdf>

**Table 3.** A sample output of the distiller with manual evaluation

Communicator	TargetVerb	Evaluee	Reason	Validity
people	accuse	european union	-	<i>ok</i>
-	accused	security	violations	<i>ok</i>
member states	blame	problems	-	<i>ok</i>
-	blamed	fishermen	negotiations	<i>ok</i>
companies	charge	price	-	<i>no</i>
amnesty international	cited	-	torture	<i>ok</i>
commission	charged	-	failure	<i>ok</i>
-	charged	authorities	crime	<i>ok</i>
-	charged	group	report	<i>ok</i>
employment	charged	-	-	<i>no</i>

domain frame is formalized as an `owl:Individual rdf:type ofn:Frame`, thus resulting to be also `rdf:type lmm:Meaning`, according to the guidelines provided in section 2. For example, in one filtered occurrence we first specialize the generic frame (14), and then add a new super-sense typing over its frame elements (15).

```
europarl:AccuseOfViolation lmm:specializes
                             ofnabox:JudgmentCommunication (14)
```

```
ofnabox:Communicator ofn:hasSemType europarl:People) (15)
```

After a manual evaluation on the domain-specific frames detected for the three generic frames chosen, 62% of distilled domain-specific frames resulted to be valid, where validity is assumed as respecting the conceptualization of the generic frame, in terms of extensional interpretation of the classes derived from target verbs, and of properties derived from frame elements. In this section we explain the workflow used to make this interpretation viable.

In this rapid ontology design process, an ontology designer is only asked to validate the detected frames that present a desirable design property, *density* i.e. the resulting ontology corresponding to each domain-specific frame is a labeled connected graph with nodes labeled by `owl:Class` names and edges labeled by `owl:ObjectProperty` names. Density helps preserving the conceptual structure of the generic frame, which is known to feature cognitive relevance and linguistic appropriateness. Notice that typical ontologies produced by traditional learning methods result to lack density and do not feature the expected formal structure implicit in experts' conceptualization of their domain.

Besides desirable design properties, we are interested in designing ontologies that *work*. Quality is therefore evaluated in terms of fitness to *competency question*-based (or task-based) ontology design [20], as assessed in the ontology evaluation framework described in [17]. We notice that frame-based ontology learning has also this very desirable property. Since frames are formally equivalent to n-ary relations at design time (section 2), and competency questions can be formalized as n-ary relations at query time [25], it is straightforward to match

a linguistic competency question like: *what are the reasons for those events?* to the knowledge base of domain-specific frames.<sup>9</sup>

Domain frames can be used for two different design tasks: the first one is producing a traditional TBox domain ontology, e.g. consisting of OWL classes like `AccuseOfViolation`; the second one is to classify named entities that instantiate the domain frame (*framed situations*), which are formalized as OWL individuals.

The generation of a TBox from the LMM-encoded domain frame and framed situations is performed by means of customizable *transformation patterns* [25], which contain rules for translating each instance of an LMM type (Meaning, Expression, etc.) into an OWL element.

An additional advantage of frame-based ontology learning is the ability to generate small ontologies that respect the characteristics of highly-reusable *content ontology design patterns* [26,25]: small size, density, inherent task-based evaluability, cognitive and linguistic relevance.

As a concrete example, we show a diagram depicting a proposed design patterns that has been derived from the domain-specific frame **AccuseOfViolation** (Fig. 3). The transformation patterns adopted are exemplified here as a set of OWL(Full) axioms over the OntoFrameNet TBox, and the OWL metamodel:<sup>10</sup>

```

ofn:Frame rdfs:subClassOf
  (owl:Restriction(owl:onProperty trans:transformableTo
    (owl:allValuesFrom owl:Class)))
  (16)

```

```

ofn:FrameElement rdfs:subClassOf
  (owl:Restriction (owl:onProperty trans:transformableTo
    (owl:allValuesFrom owl:ObjectProperty)))
  (17)

```

```

ofn:SemanticType rdfs:subClassOf
  (owl:Restriction (owl:onProperty trans:transformableTo
    (owl:allValuesFrom owl:Class)))
  (18)

```

```

ofn:hasFE rdfs:subClassOf
  (owl:Restriction (owl:onProperty trans:transformableTo
    (owl:allValuesFrom owl:Restriction)))
  (19)

```

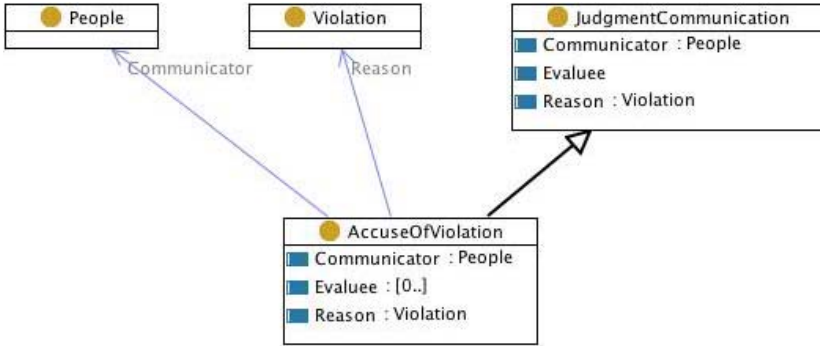
```

ofn:hasSemType rdfs:subClassOf
  (owl:Restriction (owl:onProperty trans:transformableTo
    (owl:allValuesFrom rdfs:range)))
  (20)

```

<sup>9</sup> Several domain-specific frames can be retrieved which have evidence in the corpus: **AccuseOfViolation**, **BlameNegotiation**, **ChargeForFailure**, **ChargeForCrime**, etc.

<sup>10</sup> Transformation patterns are based on alignments between OntoFrameNet and LMM (cf. section 2), and between LMM and the FormalSemantics vocabulary: <http://ontologydesignpatterns.org/ont/dul/FormalSemantics.owl>



**Fig. 3.** A content ontology design pattern derived from the domain-specific frame **AccuseOfViolation**, based on the transformation patterns

For example, given the previous axioms 14 and 15, and the following axioms extending OntoFrameNet ABox,<sup>11</sup>:

$$\text{europarl:AccuseOfViolation ofn:hasFE ofnabox:Communicator} \quad (21)$$

$$\text{europarl:AccuseOfViolation ofn:hasFE ofnabox:Reason} \quad (22)$$

$$\text{ofnabox:Reason ofn:hasSemType europarl:Violation} \quad (23)$$

we obtain the following TBox axioms,<sup>12</sup> after applying the transformation patterns:

$$\begin{aligned} &\text{europarl:AccuseOfViolation rdfs:subClassOf} \\ &(\text{owl:Restriction} (\text{owl:onProperty ofnowl:Communicator}) \\ &\quad (\text{owl:allValuesFrom europarl:People})) \end{aligned} \quad (24)$$

$$\begin{aligned} &\text{europarl:AccuseOfViolation rdfs:subClassOf} \\ &(\text{owl:Restriction} (\text{owl:onProperty ofnowl:Reason}) \\ &\quad (\text{owl:allValuesFrom europarl:Violation})) \end{aligned} \quad (25)$$

Task-based fitness can be proved by launching a unit-test that contains data about the `europarl:AccuseOfViolation` competency question: *what accuses of violation any people has made for what reason?* From the n-ary relation hypothesis, we know that a frame has direct translations into competency questions, queries, and ontologies: if the TBox produced is able to support the proper queries, then the TBox fits the task. This is the case actually, since the classes and properties that have been built (Fig. 3) are able to encode the query derived from the competency question:

$$\begin{aligned} &\text{SELECT DISTINCT ?x ?y ?z WHERE \{?x a Accuse.} \\ &\text{?y a People . ?z a Violation . ?x :Communicator ?y . ?x :Reason ?z\} \end{aligned} \quad (26)$$

<sup>11</sup> <http://www.ontologydesignpatterns.org/ofnframes/accuseofviolation.owl>

<sup>12</sup> <http://www.ontologydesignpatterns.org/owlframes/accuseofviolation.owl>

## 5 Conclusions

In this paper we have presented an implemented method for frame-based ontology learning by using components for frame detection and super-sense tagging on a text corpus, the FrameNet and WordNet lexical knowledge bases, and the LMM metamodeling framework.

The method can be used to generate candidate *domain-specific* frames suggested by qualified corpus evidence, and to populate an ontology with complex facts extracted from the corpus. The method fits a task-based analysis and evaluation of ontologies, as it is being developed in the NeOn project, where we are implementing novel functionalities for pattern-based design.

The method presented shows some significant improvements in terms of functionalities and technology with respect to existing ontology learning components, not only because we are acquiring more advanced knowledge structures (i.e. frames, instead of simple taxonomies or binary relations) but also do we provide an easier alignment to existing semantic web resources: the use of LMM makes the results more easily mappable to different kinds of data, such as datasets from Linking Open Data.

Our experiment proves that we can build a resource of reusable design patterns also by learning them from domain text corpora. There are challenging research issues in this area of ontology design. An important one is: since the amount of domain-oriented patterns can be huge, and FrameNet alone provides less than 1000 generic frames, are they complete, so that all domain patterns specialize them? Should we consider a procedure to learn patterns without previous knowledge of generic ones?

## Acknowledgements

We thank Daniele Pighin and Alessandro Moschitti at University of Trento for their critical support in developing the Frame Detection module. This work has been partly supported by the EU projects NeOn, funded within the 6th IST Framework Programme, and BONY, funded within the Lifelong Learning Programme 2007.

## References

1. Auer, S., Bizer, C., Lehmann, J., Kobilarov, G., Cyganiak, R., Ives, Z.: DBpedia: A Nucleus for a Web of Open Data. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 722–735. Springer, Heidelberg (2007)
2. Baader, F. (ed.): The Description Logic Handbook: theory, implementation, and applications. Cambridge University Press, Cambridge (2003)
3. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The Berkeley FrameNet Project. In: Proceedings of the 17th International Conference on Computational Linguistics, Morristown, NJ, USA, pp. 86–90 (1998)

4. Basili, R., Giannone, C., De Cao, D.: Learning domain-specific framenets from texts. In: ECAI Workshop on Ontology Learning and Population (2008)
5. Basili, R., Moschitti, A., Paziienza, M.T.: A text classifier based on linguistic processing. In: Proceedings of the IJCAI 1999 Workshop on Machine Learning for Information Filtering (1999)
6. Bos, J., Nissim, M.: Combining Discourse Representation Theory with FrameNet. In: Favretti, R.R. (ed.) *Frames, Corpora, and Knowledge Representation*, pp. 169–183. Bononia University Press (2008)
7. Brachman, R.J.: *A Structural Paradigm for Representing Knowledge*. Ph.d. thesis, Harvard University, USA (1977)
8. Ciaramita, M., Altun, Y.: Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In: Proceedings of EMNLP 2006, Sydney, Australia, pp. 594–602 (2006)
9. Ciaramita, M., Johnson, M.: Supersense tagging of unknown nouns in wordnet. In: Proceedings of EMNLP 2003, Sapporo, Japan, pp. 168–175 (2003)
10. Collins, M.: Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In: Proceedings of EMNLP 2002 (2002)
11. Coppola, B., Moschitti, A., Pighin, D.: Generalized framework for syntax-based relation mining. In: Proceedings of the IEEE International Conference on Data Mining (ICDM 2008), Pisa, Italy (2008)
12. Davidson, D.: *The Logical Form of Action Sentences*. In: *The Logic of Decision and Action*, 2nd edn. University of Pittsburgh Press, Pittsburgh (1967)
13. Erk, K., Pado, S.: Shalmaneser - a flexible toolbox for semantic role assignment. In: Proceedings of LREC 2006, Genoa, Italy (2006)
14. Fellbaum, C. (ed.): *WordNet. An Electronic Lexical Database*. MIT Press, Cambridge (1998)
15. Fillmore, C.J.: The Case for Case. In: Bach, E., Harms, R.T. (eds.) *Universals in Linguistic Theory*, pp. 1–210. Holt, Rinehart, and Winston, New York (1968)
16. Gangemi, A.: What's in a Schema?. In: Huang, C.R., Calzolari, N., Gangemi, A., Lenci, A., Oltramari, A., Prevot, L. (eds.) *Ontologies and the Lexicon*. Cambridge University Press, Cambridge (2009)
17. Gangemi, A., Catenacci, C., Ciaramita, M., Lehmann, J.: Modelling Ontology Evaluation and Validation. In: Sure, Y., Domingue, J. (eds.) *ESWC 2006*. LNCS, vol. 4011, pp. 140–154. Springer, Heidelberg (2006)
18. Gangemi, A., Navigli, R., Velardi, P.: The OntoWordNet Project: Extension and Axiomatization of Conceptual Relations in WordNet. In: Meersman, R., Tari, Z., Schmidt, D.C. (eds.) *CoopIS 2003, DOA 2003, and ODBASE 2003*. LNCS, vol. 2888, pp. 820–838. Springer, Heidelberg (2003)
19. Gildea, D., Jurafsky, D.: Automatic Labeling of Semantic Roles. *Computational Linguistics* 28(3), 245–288 (2002)
20. Gruninger, M., Fox, M.: *The role of competency questions in enterprise engineering* (1994)
21. Kifer, M., Lausen, G., Wu, J.: Logical Foundations of Object-Oriented and Frame-Based Languages. *Journal of Association for Computing Machinery* 42(4), 741–843 (1995)
22. Minsky, M.: *A Framework for Representing Knowledge*. In: Winston, P. (ed.) *The Psychology of Computer Vision*. McGraw-Hill, New York (1975)
23. Moschitti, A., Pighin, D., Basili, R.: Tree kernels for semantic role labeling. *Computational Linguistics* 34(2), 193–224 (2008)



24. Picca, D., Gangemi, A., Gliozzo, A.: LMM: an OWL Metamodel to Represent Heterogeneous Lexical Knowledge. In: Proc. of the International Conference on Language Resources and Evaluation (LREC), Marrakech, Morocco (2008)
25. Presutti, V., Gangemi, A., et al.: Library of design patterns for collaborative development of networked ontologies. Deliverable D2.5.1, NeOn project (2008)
26. Presutti, V., Gangemi, A.: Content Ontology Design Patterns as Practical Building Blocks for Web Ontologies. In: Li, Q., Spaccapietra, S., Yu, E., Olivé, A. (eds.) ER 2008. LNCS, vol. 5231. Springer, Heidelberg (2008)
27. Scheffczyk, J., Baker, C.F., Narayanan, S.: Reasoning over Natural Language Text by means of FrameNet and Ontologies. In: Ontologies and the Lexicon. Cambridge University Press, Cambridge
28. Tanev, H., Kouylekov, M., Negri, M., Coppola, B., Magnini, B.: Multilingual pattern libraries for question answering: a case study for definition questions. In: Proceedings of LREC 2004, Lisbon, Portugal (2004)