

# Enriching an Ontology with Multilingual Information

Mauricio Espinoza<sup>1</sup>, Asunción Gómez Pérez<sup>1</sup>, and Eduardo Mena<sup>2</sup>

<sup>1</sup> UPM, Laboratorio de Inteligencia Artificial, 28660 Boadilla del Monte, Spain  
asun@fi.upm.es, mespinoza@delicias.dia.fi.upm.es

<sup>2</sup> IIS Department, Univ. of Zaragoza, María de Luna 1, 50018 Zaragoza, Spain  
emena@unizar.es

**Abstract.** Organizations working in a multilingual environment demand multilingual ontologies. To solve this problem we propose LabelTranslator, a system that automatically localizes ontologies. Ontology localization consists of adapting an ontology to a concrete language and cultural community.

LabelTranslator takes as input an ontology whose labels are described in a source natural language and obtains the most probable translation into a target natural language of each ontology label. Our main contribution is the automatization of this process which reduces human efforts to localize an ontology manually. First, our system uses a translation service which obtains automatic translations of each ontology label (name of an ontology term) from/into English, German, or Spanish by consulting different linguistic resources such as lexical databases, bilingual dictionaries, and terminologies. Second, a ranking method is used to sort each ontology label according to similarity with its lexical and semantic context.

The experiments performed in order to evaluate the quality of translation show that our approach is a good approximation to automatically enrich an ontology with multilingual information.

**Keywords:** Ontology localization, Multilingual ontologies.

## 1 Introduction

The Semantic Web offers the most appropriate scenario for exploiting the potentialities of ontologies due to the large amount of information which is to be exposed and accessed. However, most of the ontologies on the Web are in English and although there are a few exceptions (like EuroWordNet [13]), it is necessary to guarantee that the same knowledge be recognizable in different natural languages.

Currently, more and more organizations working in multilingual environments demand ontologies supporting different natural languages. In the framework of the NeOn project [7], all case studies have expressed the need for multilingual ontologies. One case study is led by the Food and Agriculture Organization of the United Nations (FAO), an international organization with a huge amount

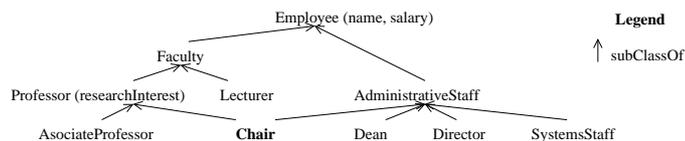
of multilingual resources, some of them in more than ten languages. The second use case is concerned with the pharmaceutical industry in Spain, and requires ontologies in the different languages spoken in the country. Consequently, the inclusion of multilingual data in ontologies is not an option but a must.

In this paper we introduce *LabelTranslator*, a system that automatically localizes ontologies in English, Spanish and German. The Ontology Localization Activity (OLA) consists of adapting an ontology to a concrete language and culture community, as defined in [6]. We describe here the features and design aspects of the current prototype of our system and discuss some of the innovations we are planning for a future prototype. *LabelTranslator* takes as input an ontology whose labels are expressed in a source natural language, and obtains the most probable translation of each ontology label into a target natural language.

The rest of this paper is as follows. In Section 2 we describe the main components of our approach. In Section 3 we show how the system obtains the translation of each ontology label from different linguistic resources. The translation ranking method used to rank the translations of each ontology label according to its context can be found in Section 4. Some experimental results are presented in Section 5. Related work can be found in Section 6. Finally, conclusions and future work appear in Section 7.

## 2 Overview of the System

As a motivating example, let us consider the extract of the sample university ontology shown in Figure 1. Let us suppose that the user wants to translate the term *chair* from English into Spanish. According to the domain of the sample ontology, the correct translation of the selected term should be in the sense of the position professor, nor in the sense of a place where one person can sit down and nor an instrument of execution by electrocution, etc.



**Fig. 1.** Extract of the sample university ontology.

In the following, we first show the main components of our system and then we describe how these components with interact each other.

### 2.1 Structural Overview

There are three important aspects to be considered to automatically localize monolingual ontologies to other natural languages and to allow users to access

to multilingual information: 1) obtaining the possible translations for ontology labels, 2) disambiguating label senses, and 3) ranking translations. Certainly, all multilingual results have to be appropriately stored and presented to the user. Figure 2 shows the main components of our system and illustrates the process for enriching an ontology with linguistic information:

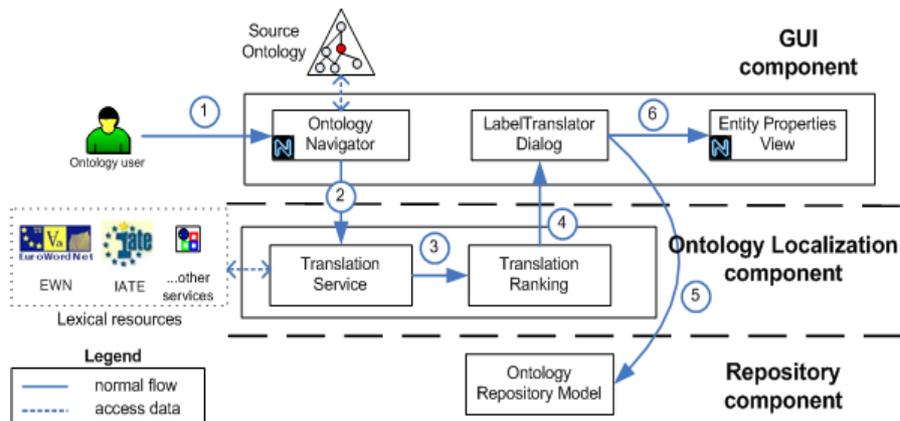


Fig. 2. Main components of *LabelTranslator* plug-in.

- **GUI component:** This component controls the GUI in order to show the multilingual results appropriately. Once invoked, *LabelTranslator* uses some views<sup>3</sup> of the Neon ToolKit to load the ontology and store the multilingual results respectively. Due to space limitations, the user interfaces used by our system are not shown here, but can be found in [4].
- **Ontology Localization Component:** This component is responsible of obtaining the most probable translation for each ontology label. It relies on two advanced modules. The first, *translation service*, automatically obtains the different possible translations of an ontology label by accessing to different linguistic resources. This service also uses a compositional method in order to translate compound labels (multi-word labels). A more detailed description of the translation service can be found in Section 3. The second module, *translation ranking*, sorts the different translations according to the similarity with its lexical and semantic context. The method relies on a relatedness measure based on glosses to disambiguate the translations. This is done by comparing the senses associated to each possible translation and their context. More details about the ranking method can be found in Section 4.

<sup>3</sup> In the NeOn ToolKit a view is typically used to navigate a hierarchy of information, open an editor, or display properties for the active editor.

- **Repository Component:** This component captures all the linguistic information associated with concepts. *LabelTranslator* supports the linguistic model [10] designed for the representation of multilingual information in ontologies. In the current version of our system, the link we establish between the terms in the ontology and their associated translations is characterized by simple references between concepts and labels (as offered by the standard owl:comment and rdfs:label properties). The representation of the multilingual information obtained by our tool is out of scope of this paper, however, a detailed description of this multilingual model can be found in [10].

The current version of LabelTranslator has been implemented as a NeOn plug-in, but it can easily become an independent module, for example a web service.

## 2.2 Functional Overview

This section briefly describe the interaction of the components shown in the Figure 2. A more detailed description of the execution process of our system can be found in [4]. The main activities can be summarized as follows:

1. the user chooses the label of the ontology term(s) to be translated.
2. the *translation service* access bilingual or multilingual linguistic resources to look for the possible translations of the selected label(s).
3. the *ranking method* compares the obtained translation possibilities against the label(s) in the original language and its ontological context.
4. the user confirms the translation proposed by the ranking method, or he/she chooses the translation that better fits in the ontological context.
5. the system updates the ontology model with the selected translation(s) and additional linguistic information.
6. finally, the linguistic information associated with ontology term(s) is shown to the user.

More details on the features of the *ontology localization component*, are given in the following sections.

## 3 Translation service

In this section we provide the details that show how the system obtains the different translations of an ontology label (which can name different kinds of ontology terms: concepts, properties or relations) using different linguistic resources.

The *translation service* takes as input an ontology label  $l$  described in a source language and returns a set of possible translations  $T = \{t_1, t_2, \dots, t_n\}$  in a target language. The current prototype supports translations among English, Spanish, and German. In order to discover the translations of each ontology label, the system accesses different lexical resources: 1) remote lexical databases as EuroWordNet [13], 2) multilingual dictionaries as GoogleTranslate<sup>4</sup>, Wiktionary<sup>5</sup>,

<sup>4</sup> <http://www.google.com/translate.t>

<sup>5</sup> <http://en.wiktionary.org/wiki/>

Babelfish<sup>6</sup>, and FreeTranslation<sup>7</sup>, and 3) other lexical resources as IATE<sup>8</sup>. A *cache* stores previously translations to avoid accessing the same data twice.

The algorithm used by the translation service is summarized in the following: 1) If the selected ontology label is already available in the target language in our cache, then LabelTranslator just displays it, with all the relevant available information, 2) If the translation is not stored locally, then it accesses remote repositories to retrieve possible translations. A compositional method may be needed to translate compound labels (explained in the Section 3.2). If no results are obtained from the two previous steps, then the user can enter his/her own translation (together with the definition).

In our approach, the translation of an ontology label denoted by  $t$ , is a tuple  $\langle trs, senses \rangle$ , where  $trs$  is translated label in the specific target language, and  $senses$  is a list of semantic senses extracted from different knowledge pools. In the following we briefly describe the task of automatically retrieving the possible semantic senses of a translated label.

### 3.1 Semantically Representating a Sense

In order to discover the senses of each translated label ( $t_i$ ), we have considered the approach proposed in a previous work [12]. Our system takes as input a list of words (each  $t_i$ ), discovers their semantics in run-time and obtains a list of senses extracted from different ontology pools; it deals with the possible semantic overlapping among senses. We summarize here the key characteristic of the sense discovering process:

1. To discover the semantic of the input words, the system relies on a pool of ontologies instead of just a single ontology.
2. The system builds a sense (meaning) with the information retrieved from matching terms in the ontology pool.
3. Each sense is represented as a tuple  $s_k = \langle s, grph, descr \rangle$ , where  $s$  is the list of synonym names<sup>9</sup> of word  $k$ ,  $grph$  describes the sense  $s_k$  by means of the hierarchical graph of hypernyms and hyponyms of synonym terms found in one or more ontologies, and  $descr$  is a description in natural language of such a sense.
4. As matching terms could be ontology classes, properties or individuals, three lists of possible senses are associated with each word  $k$ :  $S_k^{class}$ ,  $S_k^{prop}$  and  $S_k^{indv}$ .
5. Each word sense is enhanced incrementally with the synonym senses (which also searches the ontology pool).

<sup>6</sup> <http://babelfish.altavista.com/>

<sup>7</sup> <http://ets.freetranslation.com>

<sup>8</sup> <http://iate.europa.eu/iatediff/SearchByQueryLoad.do?method=load>

<sup>9</sup> The system extracts the synonym names of a term by consulting the synonym relationships defined in the ontology of such a term.



### 3.2 Compositional Method to Translate Compound Labels

Compound labels which have an entry in linguistic ontologies such as EuroWordNet [13] (for example “jet lag”, “travel agent” and “bed and breakfast”) are treated in our system as single words. Others like “railroad transportation”, which have no entry in the previous resources, are translated using a compositional method. This method split the label into tokens (“railroad” and “transportation” in the example); the individual components are translated and then combined into a compound label in the target language. Care is taken to combine the components respecting the word order of the target language. A set of lexical templates derived from different ontologies are used to control the order of translation. The main steps of the algorithm are:

1. The compound label is normalized, e.g., rewriting in lowercase, hyphens are removed, it is split into tokens, etc.
2. A set of possible translations is obtained for each token of the compound label using the translation service.
3. Since translations between languages do not keep the same word order, the algorithm creates candidate translations in the target language using lexical templates<sup>11</sup>. Each lexical template contains at least a pair of patterns, namely ‘source’ and ‘target’ patterns. A source pattern is a template to be compared with the *tagged compound label*<sup>12</sup>, described in the source language, while the target pattern is used to generate the label in the target language. If no applicable template is found, the compound label is translated using the translation service directly.
4. All the candidate labels that fulfill the target pattern are returned as candidate translations of the compound label.

The senses of each candidate translation are discovered using the sense discovering process described in the section 3.1. If not results are obtained, the method tries to discover the senses of each token separately.

In the following we describe the process to learn the lexical templates used to control the order of translation of compound labels.

#### Learning Lexical Templates from Ontological Labels

We believe that lexical templates used to translate compound labels are a necessary component to produce high quality translations because 1) it guarantees grammatical output and, 2) it makes sure that the structural source language meaning is preserved. In our approach, we used a semi-automatic process to obtain the lexical templates. As we explained before, each lexical template is composed of source and target patterns. The ontology labels used to learn the

---

<sup>11</sup> The notion of lexical template proposed in this paper refers to text correlations found between a pair of languages.

<sup>12</sup> We use TreeTagger [11] in order to annotate the compound labels with part-of-speech and lemma information.

source patterns were extracted from different domain ontologies expressed in English, German, or Spanish. Each label was tokenized and tagged using the language independent part-of-speech tagger proposed in [11]. On the other hand, the labels used to learn the target patterns were extracted either from the multilingual information associated with each ontological term or by means of a manual translation process. The same process used to annotate part of speech (POS) in the labels of the source patterns was used to annotate the labels of the target patterns. The empirical results collected during the learning of lexical templates are briefly described below:

- *Existing ontologies share the same lexical patterns.* For instance, approximately 60% of the labels that describe an ontological concept makes use of an adjective followed by a noun (e.g. spatial region, industrial product, natural hazard, etc.). Other labels use as lexical pattern ( $\approx 30\%$ ) a noun followed by another noun (e.g., transport vehicle, knowledge domain, etc.).
- *Ontology labels usually have less than four tokens.* Approximately 85% of labels fulfill this. Thus, for the current prototype we only focus on the definition of lexical templates for compound labels of two or three tokens.

A repository is used to store all the lexical templates obtained for each pair of languages. Due to space limitations, in Table 1 we show only a sample list of the lexical templates learned to translate compound labels from English into Spanish.

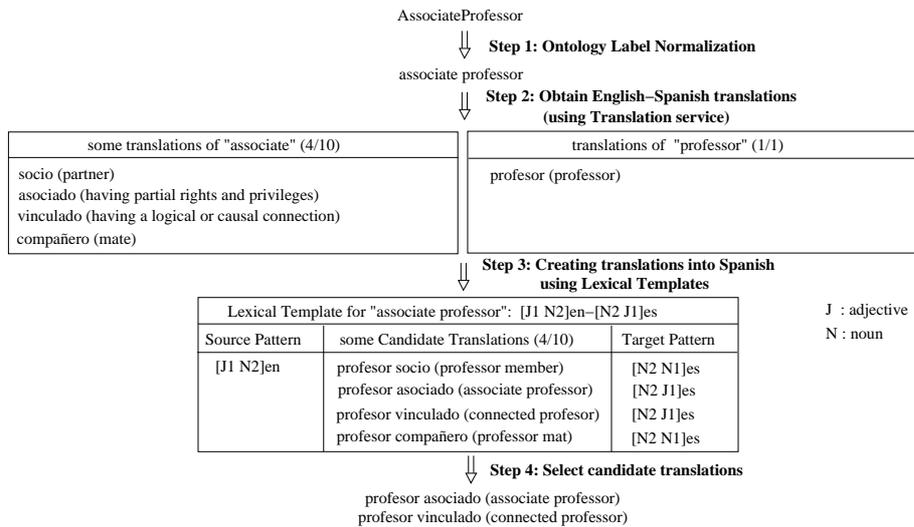
**Table 1.** Some lexical templates to translate a compound label from English into Spanish.

<i>Templates (4/25)</i>	<i>Samples of source and target patterns</i>	
	<i>English</i>	<i>Spanish</i>
$[J_1 N_2]_{en} \rightarrow [N_2 J_1]_{es}$	spatial region→ industrial product→ natural hazard→	región espacial producto industrial peligro natural
$[N_1 N_2]_{en} \rightarrow [N_2 \langle pre \rangle N_1]_{es}$	transport vehicle→ knowledge domain→ research exploration→	vehículo de transporte dominio del conocimiento exploración de la investigación
$[J_1 VB_2]_{en} \rightarrow [VB_2 \langle pre \rangle J_1]_{es}$	remote sensing→	detección remota; detección a distancia
$[J_1 N_2 N_3]_{en} \rightarrow [N_2 \langle pre \rangle N_3 J_1]_{es}$	associated knowledge domain→	dominio de conocimiento asociado

J: adjective; N: noun; VB: verb

As an illustrating example of the compositional method, we show in Figure 4 the steps of the algorithm when collecting Spanish translations for the English compound label “AssociateProfessor”, which was introduced in our motivating example (see Figure 1). Our prototype finds ten translations for the token “associate” and one for “professor” (normalized in the first step). In the next step, our tool searches a lexical template (in our repository) to create candidate translations. In the template found,  $[J_1 N_2]_{en}$  represents the source pattern in English

whilst  $[N_2 J_1]es$  represents the target pattern in Spanish. In both cases, numbers represent the position of each token of the compound label. Notice that, in the last step the candidate translations “profesor socio” (professor member) and “profesor compañero” (professor mat) are discarded because they do not fulfill the target pattern.



**Fig. 4.** Algorithm to translate the compound label “AssociateProfessor” into Spanish.

## 4 Translation Ranking Method

In this section we explain the *ranking method*, which sorts the list of translations according to similarity with the context of the label to be translated. The method takes as input the set of translations  $T$  obtained in the previous step. From this set of translations, the ranking method uses a disambiguation algorithm described in [9] to sort the translations. Once all the translations are ranked, the method allows two operation modes:

- Semi-automatic mode: It shows a list with all the possible translations sorted decreasingly. The method proposes the most relevant translation to be selected first although the user can change this default selection.
- Automatic mode: It automatically selects the translation with the highest score.

Next, we first describe how the system obtains the context of each ontology label, and then we describe the disambiguation algorithm used to sort the translations according to similarity with their context.

#### 4.1 Determining the Context of an Ontology Term

We defined context as the information/knowledge that can be used additionally to perform some task. In our approach, the context of an ontology term is used to disambiguate the lexical meaning of a ontology term. To determine the context of an ontology term, the system retrieves the labels of the set of terms associated with the term under consideration. The list of context labels, denoted by  $C$ , comprises a set of names which can be direct label names and/or attributes label names, depending on the type of term that is being translated.

In order to mitigate risks associated with system performance, the *ranking method* limits the number of context labels used to disambiguate the translated label. Every context label  $c \in C$  is compared with the ontology label  $l$  using a measure based on Normalized Google Distance [1] (NGD). NGD measures the semantic relatedness between any two terms, considering the relative frequency in which two terms appear in the Web within the same documents. Those labels with the higher values of similarity are chosen (maximum 3). To discover the senses of each context label (denoted by  $S_c$ ), the system performs the same process used to discover the senses of each translated label (as explained in the previous section).

In Figure 5, on the left, the dashed area represents all the context labels found for the ontology label “chair”. Our prototype finds five labels, but only selects three (see the dotted area) to disambiguate the term. In the table on the right, we show for each type of ontology term (concept, attribute, or relation) the context labels that could be extracted. For instance, for the concept “chair” the system retrieves its hypernyms, hyponyms, attributes, and sibling concepts.

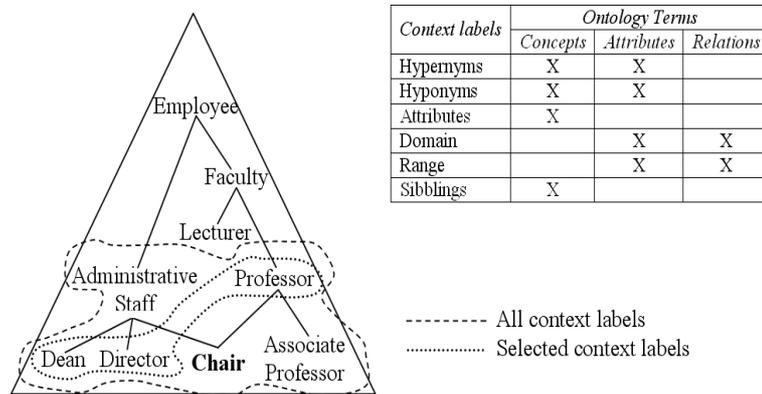


Fig. 5. Context of the ontology label “chair”.

## 4.2 Disambiguating the Senses of the Translations

In some works [8,9] the glosses are considered as a very promising means of measuring relatedness, since they can be used: 1) to make comparisons between concepts semantically different, and 2) to discover relations of which no trace is present in the resource they come from. For the current version of the prototype, the ranking method relies on a measure based on glosses proposed in [9] to sort the translations according to their context. However, we recognize that glosses are by necessity short and may not provide sufficient information on their own to make judgments about relatedness. Therefore, we make use of the hierarchical graph of the sense to extend the gloss with the relatedness glosses of their ontological terms.

We carry out disambiguation in relation to the senses of each translated label and the senses of the context labels. In the following we describe the method: let us suppose that the ontology label  $l$  after executing the translation process has yielded  $n$  translations:  $T = \{t_1, t_2, \dots, t_n\}$ . For each translation the system retrieves its corresponding senses, for example the first translated label ( $t_1$ ) to be disambiguated has  $n$  senses  $S_{t_1} = \{s_{t_1}^1, s_{t_1}^2, \dots, s_{t_1}^n\}$ . We use the notation  $TSC$  (translation sense collection) in order to group the senses of all translated labels.

$$TSC = \{S_{t_1} \cup S_{t_2} \cup \dots \cup S_{t_n}\}$$

where  $S_{t_j}$ ,  $t_j \in T$ , represents all senses corresponding to  $j^{th}$  translated label.

Now, suppose that the ontology label  $l$  has the context  $C$  which comprises several labels:  $c_1, c_2, c_3$ . Each of these context labels has a list of corresponding senses, for instance,  $c_i$  has  $m$  senses:  $S_{c_i} = \{s_{c_i}^1, s_{c_i}^2, \dots, s_{c_i}^m\}$ . We use the notation  $CSC$  (context sense collection) in order to group the senses of each context label.

$$CSC = \{S_{c_1} \cup S_{c_2} \cup S_{c_3}\}$$

where  $S_{c_j}$ ,  $c_j \in C$ , represents all senses corresponding to  $j^{th}$  context label.

The goal of the disambiguation algorithm is to select one of the senses from the set  $TSC$  as the most appropriate sense of the translation of label  $l$ . The algorithm performs word sense disambiguation by using a measure of semantic relatedness that is given by:

$$\max_{j=1}^{|TSC|} (SenseScore(TSC_j, CSC))$$

where  $TSC_j$  is the representation of one of the senses of each translated label. The chosen sense is the one with the greater value of SenseScore, defined as:

$$SenseScore(TSC_j, CSC) = \sum_{k=1}^{|CSC|} (Similarity(TSC_j, CSC_k))$$

where  $CSC_k$  is the representation of each sense of the different context labels.

In order to compute the similarity between the senses of each context and the translated label, the method applies an overlap scoring mechanism. Details about this process are available in [9] as it is not the main goal of this paper.

In our example, “cátedra” (cathedral) in the sense of “the position of professor” is ranked as first translation of the ontology label “chair”. Once the right sense has been selected, the system updates the linguistic information of the corresponding ontological term.

## 5 Experimental Evaluation

The ontology corpus used for the evaluation was selected from the set of KnowledgeWeb [2] ontologies used to manage EU projects. The corpus statistics are given in Table 2. In order to illustrate the utility of the tool to a broad community of ontology users, we are working in the evaluation of prominent ontologies such as those used in the use cases the NeOn project. The results obtained are ongoing work.

**Table 2.** Ontologies corpus statistics.

<i>Ontology Domain</i>	<i>Number of Ontological Terms</i>			<i>% Compound labels</i>	
	<i>concepts</i>	<i>attributes</i>	<i>relations</i>	<i>≤ 3 tokens</i>	<i>&gt; 3 tokens</i>
Documentation-Meeting	42	61	22	44%	25.6%
Person&Project	25	18	12	47.2%	10.9%
Organization	10	7	11	46.4%	7.1%
Office	20	12	8	12.5%	0%
University	30	10	12	17.3%	0%

### 5.1 Experimental Results

The goal of the experiments is to evaluate some aspects of the translation ranking method (described in Section 4) which tries to select the most appropriate translation for each ontology label. In particular, we evaluated three aspects of the algorithm 1) the quality of the output when the algorithm automatically suggest an translation, 2) the quality of all the set of translations, and 3) the quality of translation of the compound labels.

The manual evaluation that we decided to apply was done by Spanish speakers with a good level of English. In all the experiments a reference translation (gold standard) provided by the evaluators was used. The “gold standard” allows users to compare the quality of the translations provided by an expert with the translations provided by the algorithm. Next, we give a short overview of each experiment and show the obtained results.

**Experiment 1: Accuracy.** In order to evaluate the quality of the output of the ranking method in automatic operation mode we propose a measure of accuracy. The accuracy measures the capacity of the algorithm of translation to get in an automatic way a correct translation according to context. To measure the accuracy of the algorithm, we counted the number of times the first translation was correct.

**Experiment 2: Precision and Recall.** The previous evaluation does not allow checking the completeness of the translations since it does not observe the behavior of all the translated labels. Thus, we have measured *precision* as the number of correct translations of all the translations provided by the system and divided by the total number of translations provided by the system. To measure the *recall*, we divided the number of correct translations of all the translations provided by the system into the number of correct translations (provided by the gold standard).

**Experiment 3: Adequacy and Fluency.** In order to measure the quality of the translation of compound labels we propose a subjective 1-5 score for adequacy and fluency. The adequacy measures the capability of the translation algorithm to determinate the quantity in which the meaning of a correct translation is preserved. On the other hand, the fluency measures the capability of the algorithm to determine how good the corresponding language is. In this experiment, each evaluator assigned fluency and adequacy ratings for each translated label. Each score ranges from one to five (with one being the poorest grade and five the highest). The adequacy and fluency scores of two evaluators for each sentence were averaged together, and an overall average adequacy and average fluency score was calculated for each evaluated ontology.

**Table 3.** Results obtained in the three experiments.

<i>Ontology Domain</i>	<i>Spanish</i>					<i>German</i>				
	<i>Accu.</i>	<i>Prec.</i>	<i>Rec.</i>	<i>Adeq.</i>	<i>Flu.</i>	<i>Accu.</i>	<i>Prec.</i>	<i>Rec.</i>	<i>Adeq.</i>	<i>Flu.</i>
Documentation	0.51	0.47	0.39	68%	75%	-	-	-	-	-
Person&Project	0.73	0.35	0.81	89%	93%	-	-	-	-	-
Organization	0.81	0.41	0.78	87%	95%	0.73	0.33	0.64	73%	69%
Office	0.79	0.49	0.77	93%	95%	0.78	0.34	0.74	67%	76%
University	0.80	0.36	0.87	96%	93%	0.71	0.23	0.71	69%	73%

In Table 3 we show the results achieved by the prototype in each experiment. The values are organized by target language. All the percentages of adequacy and fluency shown in this table correspond to those translations punctuated with a value greater than 4. The experimental results show that our system is a good approximation to enhance the linguistic expressivity of existing ontologies. For example, in average our system suggest the correct translation 72% of the times. Also, the values of recall suggest that a high percentage of correct translations are part of the final translations shown to the user in the semi-automatic operation mode. Moreover, the obtained results in each metric help us to analyze which components need improvement. The main limitations discovered are:

- *Translation service is highly dependent on the types of resources used and their domain coverage.* The worst values of precision and recall were obtained by the documentation ontology, because the domain of this ontology is covered by the resources used for the translation only partially.

- *The lack of learning of new lexical patterns limits the scalability of our tool.* The percentages of adequacy and fluency obtained for English-German compound label translations are in general lower than the percentages of the English-Spanish ones. Our explication is that a major effort was put (in the current version) for the learning of templates between English-Spanish languages. However, this situation can be improved by allowing users to provide, in runtime, new lexical templates when these do not exist yet in any repository.

## 6 Related Work

Our work enhances the work presented in [3], where a system for supporting the multilingual extension of ontologies expressed in just one natural language was proposed. This tool is used to support “the supervised translation of ontology labels”. Therefore, the tool offers a semi-automatic strategy. In our approach we have implemented an automatic method to reduce human intervention while enriching an ontology with linguistic information.

In [8] they propose a framework for adding linguistic expressivity to conceptual knowledge, as represented in ontologies. They use two lexical resources for the linguistic or multilingual enrichment; WordNet, for the linguistic enrichment of ontologies with English labels, and DICT dictionaries, for the linguistic and multilingual enrichment of ontologies. In this work, they do not describe the process to translate compound ontology labels, which are often not contained in linguistic ontologies as WordNet. In our work, we use a compositional method which first searches for translation candidates of a compound label and then builds the translations for the candidates using lexical templates.

In [5] a method to give support to multilingual ontology engineering is developed. In this work some software tools have been used for supporting the process of term extraction and translation. In particular the translation process requires sentence aligned parallel text, tokenized, tagged and lemmatized. In our opinion, obtaining a corpus aligned is not a simple task. Unlike this work, we rely on some multilingual translation services and extend them by using lexical templates.

## 7 Conclusions and Future Work

In this paper we have presented LabelTranslator, a system that automatically localize ontologies, providing translations across different natural languages. The main features of this tool are the following:

1. It uses a translation mining service to obtain from different linguistic resources the possible translations of each ontological label. A compound label is translated using a compositional method that searches candidate translations of each lexical component and then builds a translation using lexical templates.

2. It uses a disambiguation method that ranks the possible translations of each ontology label. A gloss overlap scoring mechanism is used to calculate the similarity between two senses.

The experimental results obtained encourage us to tackle further improvements and tests on our system.

### Acknowledgements

This work is supported by the European Commission's Sixth Framework Program under the project name: Lyfecycle support for networked ontologies (NeOn) (FP6-027595), the National Project "GeoBuddies" (TSI2007-65677C02), and the spanish CICYT project TIN2007-68091-C02-02.

### References

1. R. L. Cilibrasi and P. M. Vitnyi. The Google Similarity Distance. *IEEE Transactions on Knowledge and Data Engineering*, 19(3):370–383, march 2007.
2. O. Corcho, A. López-Cima, and A. Gómez-Pérez. The odesew 2.0 semantic web application framework. In *Proceedings of the 15th International Conference on World Wide Web (Edinburgh, Scotland, May 23 - 26, 2006)*. WWW '06, pages 1049–1050. ACM Press, New York, NY, 2006.
3. T. Declerck, A. Gómez-Pérez, O. Vela, Z. Gantner, and D. Manzano-Macho. Multilingual lexical semantic resources for ontology translation. In *Proceedings of LREC 2006*, 2006.
4. M. Espinoza, A. Gómez-Pérez, and E. Mena. Labeltranslator - a tool to automatically localize an ontology. In *Proc. of 5th European Semantic Web Conference (ESWC'08), Tenerife, (Spain), June 2008*.
5. K. Kerremans and T. R. Towards multilingual, termontological support in ontology engineering. In *Proceedings Workshop on Terminology, Ontology and Knowledge representation, January 22-23, 2004, Lyon, France*.
6. M. C. Suarez-Figueroa, (coordinator). NeOn Development Process and Ontology Life Cycle. NeOn Project Deliverable 5.3.1 (2007).
7. Neon Project., 2006. <http://www.neon-toolkit.org/>.
8. M. T. Pazienza and A. Stellato. Exploiting linguistic resources for building linguistically motivated ontologies in the semantic web. In *Second Workshop on Interfacing Ontologies and Lexical Resources for Semantic Web Technologies (OntoLex2006), held jointly with LREC2006, May 24-26, 2006, Genoa, (Italy)*, 2006.
9. T. Pedersen, S. Banerjee, and S. Patwardhan. Maximizing Semantic Relatedness to Perform Word Sense Disambiguation. Research Report UMSI 2005/25, University of Minnesota Supercomputing Institute, March 2005.
10. W. Peters, E. Montiel-Ponsoda, and G. Aguado de Cea. Localizing ontologies in owl. In *OntoLex2007*, 2007.
11. TreeTagger., 1997. <http://www.ims.uni-stuttgart.de/projekte/corplex/>.
12. R. Trillo, J. Gracia, M. Espinoza, and E. Mena. Discovering the semantics of user keywords. *Journal on Universal Computer Science. Special Issue: Ontologies and their Applications, ISSN 0948-695X*, 2007.
13. P. Vossen. Eurowordnet: a multilingual database for information retrieval. In *Proceedings of the workshop on Cross-language Information Retrieval, Zurich*, 1997.