Two approaches for integrating translation and retrieval in real applications

Cristina Vertan
University of Hamburg
Research Group "Computerphilology"
Von-Mell Park 6, 20146 Hamburg, germany
cristina.vertan@uni-hamburg.de

Abstract

In this paper we present two approaches for integrating translation into cross-lingual search engines: the first approach relies on term translation via a language ontology, the other one is based on machine translation of specific information.

1 Introduction

The explosion of on-line available multilingual information during the last years, raised the necessity of building applications able to manage this type of content. People are more and more used to search for information not only in English, but also in their mother tongue and often in some other languages they understand. Moreover there are dedicated web-platforms where the information is a-priori multilingual, like eLearning Systems and Content Management Systems. eLearning systems are used more an more as real alternatives to face-to-face courses and include often materials in the mother languages and also English (either because a lot of literature is available in English or because the content should be made available to exchange students). Content management systems used by multinational corporates, share materials in several languages as well.

On such platforms the search facility is an essential one: usually the implemented methods are based on term indexes, which are created per language. This prohibits or at least makes very difficult the access to multilingual material: the user is forced to repeat the query in several languages, which is time consuming and error – prone.
Cross-lingual retrieval methods are only slowly introduced in real applications like those ones quoted above. In this paper we will describe two applications and two different ways of combining term-translation and information retrieval. In the first one, an eLearning system, we implement a language ontology on which we map the multilingual lexical entries. The search engine makes then use of the mapping between the lexical material and the ontology. The second application is a content management system, in which we use machine translation as backbone to the search engine

The rest of the paper is organised as follows: in Section 2 we describe the eLearning environment in which we embedded the search engine and present this one. In Section 3 we describe the content management system and the symbiosis between the machine translation and the search engines. In Section 4 we conclude with some observations on these two approaches and introduce possible approaches for further work.

2 Crosslingual search based on language independent ontology and lexical information

The system we describe in this section was developed within the EU-Project LT4eL – Language Technology for eLearning (http://www.lt4el.eu). The main goal of the project was to enhance an eLearning system with language technology tools. The system dealt with nine languages (Bulgarian, Czech, Dutch, English, German, Maltese, Polish, Portuguese, Romanian). eLearning documents were processed through language specific pipelines and keywords and definitions were automatic extracted. The kernel of the system is however the crosslingual semantic search engine which makes use of a language

Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pages 59–64,
Avignon, France, April 23 - 27 2012. ©2012 Association for Computational Linguistics

independent ontology and mapping of language specific lexicons.

As prototype we implemented a domain specific ontology of about 1000 concepts, from the field „Computer Science for non computer science specialists". The concepts were not collected from English texts, but from analyzed keywords from al involved languages. In this way we avoided a bias towards English specific concepts. For the keywords in each language each partner provided an English translation (one word, one expression or even several sentences). The analysis of these translations conducted to the construction of the ontology. The concepts were represented in OWL-DL. The domain specific ontology was mapped on the DOLCE-upper ontology as well as WordNeT to ensure consistency.

The two main components that define the ontology-to-text relation necessary to support the crosslingual retrieval are: (terminological) lexicon and concept annotation grammar (Lemnitzer et. Al, 2007).

The lexicon plays twofold role in the architecture. First, it interrelates the concepts in the ontology to the lexical knowledge used by the grammar in order to recognize the role of the concepts in the text. Second, the lexicon represents the main interface between the user and the ontology. This interface allows for the ontology to be navigated or represented in a natural way for the user. For example, the concepts and relations might be named with terms used by the users in their everyday activities and in their own natural language (e.g. Bulgarian). This could be considered as a first step to a contextualized usage of the ontology in a sense that the ontology could be viewed through different terms depending on the context. For example, the color names will vary from very specific terms within the domain of carpet production to more common names used when the same carpet is part of an interior design.

Thus, the lexical items contain the following information: a term, contextual information determining the context of the term usage, grammatical features determining the syntactic realization within the text. In the current implementation of the lexicons the contextual information is simplified to a list of a few types of users (producer, retailer, etc). With respect to the relations between the terms in the lexicon and the concepts in the ontology, there are two main problems: (1) there is no lexicalized term for some of the concepts in the ontology, and (2) there are lexical terms in the language of the domain which lack corresponding concepts in the ontology, which represent the meaning of the terms. The first problem is overcomed by writing down in the lexicon also non-lexicalized (fully compositional) phrases to be represented. Even more, we encourage the lexicon builders to add more terms and phrases to the lexicons for a given concept in order to represent as many ways of expressing the concept in the language as possible.

These different phrases or terms for a given concept are used as a basis for construction of the annotation grammar. Having them, we might capture different wordings of the same meaning in the text. The concepts are language independent and they might be represented within a natural language as form(s) of a lexicalized term, or as a free phrase. In general, a concept might have a few terms connected to it and a (potentially) unlimited number of free phrases expressing this concept in the language

Some of the free phrases receive their meaning compositionally regardless their usage in the text, other free phrases denote the corresponding concept only in a particular context. In our lexicons we decided to register as many free phrases as possible in order to have better recall on the semantic annotation task.

In case of a concept that is not-lexicalized in a given language we require at least one free phrase to be provided for this concept. We could summarize the connection between the ontology and the lexicons in the following way: the ontology represents the semantic knowledge in form of concepts and relations with appropriate axioms; and the lexicons represent the ways in which these concepts can be realized in texts in the corresponding languages.

Of course, the ways in which a concept could be represented in the text are potentially infinite in number, thus, we could hope to represent in our lexicons only the most frequent and important terms and phrases. Here is an example of an entry from the Dutch lexicon:

```
<entry id="id60">
<owl:Class
rdf:about="lt4el:BarWithButt
ons">
<rdfs:subClassOf>
<owl:Class
rdf:about="lt4el:Window"/>
</rdfs:subClassOf>
</owl:Class>
<def>A horizontal or
vertical bar as a part of a
window, that contains
buttons, icons.</def>
<termg lang="nl">
<term
shead="1">werkbalk</term>
<term>balk</term>
<term type="nonlex">balk met
knoppen</term>
<term>menubalk</term>
</termg>
</entry>
```

Each entry of the lexicons contains three types of information: (1) information about the concept from the ontology which represents the meaning for the terms in the entry; (2) explanation of the concept meaning in English; and (3) a set of terms in a given language that have the meaning expressed by the concept. The concept part of the entry provides minimum information for formal definition of the concept.

The English explanation of the concept meaning facilitates the human understanding. The set of terms stands for different wordings of the concept in the corresponding language. One of the terms is the representative for the term set. Note that this is a somewhat arbitrary decision, which might depend on frequency of term usage or specialist's intuition. This representative term will be used where just one of terms from the set is necessary to be used, for example as an item of a menu. In the example above we present the set of Dutch terms for the concept `lt4el:BarWithButtons`.

One of the term is non-lexicalized - attribute type with value nonlex. The first term is representative for the term set and it is marked-up with attribute shead with value 1. In this way we determine which term to be used for ontology browsing if there is no contextual information for the type of users. The second component of the ontology-to-text relation, the concept annotation grammar, is ideally considered as an extension of a general language deep grammar which is adopted to the concept annotation task. Minimally, the concept annotation grammar consists of a chunk grammar for concept annotation and (sense) disambiguation rules. The chunk grammar for each term in the lexicon contains at least one grammar rule for recognition of the term.

As a preprocessing step we consider annotation with grammatical features and lemmatization of the text. The disambiguation rules exploit the local context in terms of grammatical features, semantic annotation and syntactic structure, and alsp the global context such as topic of the text, discourse segmentation, etc. Currently we have implemented chunk grammars for several languages.

The disambiguation rules are under development. For the implementation of the annotation grammar we rely on the grammar facilities of the CLaRK System (Simov et al., 2001). The structure of each grammar rule in CLaRK is defined by the following DTD fragment:

```
<!ELEMENT line (LC?, RE, RC?,
RM, Comment?) >
<!ELEMENT LC (#PCDATA)>
<!ELEMENT RC (#PCDATA)>
<!ELEMENT RE (#PCDATA)>
<!ELEMENT RM (#PCDATA)>
<!ELEMENT Comment (#PCDATA)>
```

Each rule is represented as a line element. The rule consists of regular expression (RE) and category (RM = return markup). The regular expression is evaluated over the content of a given XML element and could recognize tokens and/or annotated data. The return markup is represented as an XML fragment which is substituted for the recognized part of the content of the element.

Additionally, the user could use regular expressions to restrict the context in which the regular expression is evaluated successfully. The LC element contains a regular expression for the left context and the RC for the right one. The element Comment is for human use. The application of the grammar is governed by Xpath expressions which provide additional mechanism for accurate annotation of a given XML document.

Thus, the CLaRK grammar is a good choice for implementation of the initial annotation grammar. The creation of the actual annotation grammars started with the terms in the lexicons for the corresponding languages. Each term was lemmatized and the lemmatized form of the term was converted into regular expression of grammar rules. Each concept related to the term is stored in the return markup of the corresponding rule. Thus, if a term is ambiguous, then the corresponding rule in the grammar contains reference to all concepts related to the term.

The relations between the different elements of the models are as follows. A lexical item could have more than one grammar rule associated to it depending on the word order and the grammatical realization of the lexical item. Two lexical items could share a grammar rule if they have the same wording, but they are connected to different concepts in the ontology. Each grammar rule could recognize zero or several text chunks.

The relation ontology-to-text implemented in this way is the basis fort the crosslingual search engine which works in the following way:
Words in any of the covered languages can be entered and are looked up in the lexicon; the concepts that are linked to the matching lexicon entries are used for ontology-based search in an automatic fashion.

Before lexicon lookup, the words are orthographically normalised, and combinations for multi-word terms are created (e.g. if the words "text" and "editor" are entered, the combinations "texteditor", "text editor" and "text-editor" are created and looked up, in addition to the individual words). For each of the found concepts, the set of all its (direct or indirect) subconcepts is determined, and is used to retrieve Learning Objects (Los) .

The use of these language-independent concepts as an intermediate step makes it possible to retrieve LOs in any of the covered languages, thus realising the crosslingual aspect of the retrieval. When the found LOs are displayed, at the same time the relevant parts of the ontology are presented in the language that the user prefers. In a second step, the user can select (by marking a checkbox) the concept(s) he wants to look for and repeat the search. If an entered word was ambiguous, the intended meaning can be explicated now by selecting the appropriate concept. Furthermore, by clicking on a concept, related concepts are displayed; navigation through the ontology is possible in this way. A list of retrieval languages (only LOs written in one of those languages will be found) is specified as an input parameter. The retrieved LOs are sorted by language. The next ordering criterion is a ranking, based on the number of different search concepts and the number of occurrences of those concepts in the LO. For each found LO, its title, language, and matching concepts are shown.

3 Crosslingual search based on machine Translation

The second case study is the embedding of a crosslingual search engine into a web-based content management system. The system is currently implemented within the EU-PSP project ATLAS (http://www.atlasproject .eu) and aims to be domain independent. Thus, a model as presented in section 2 is impossible to be realised, as the automatic construction of a domain ontology is too unreliable and the human construction too cost effective. Also a general lexicon coverage is practically impossible.

Therefore in this project we adopted a different solution (Karagiozov et al 2011), namely we ensure the translation of keywords and short generated abstracts, and all these translations are part of the RDF-generate index. The ATLAS system ensures the lingustic processing of uploaded documents and extraction of most important keywords. A separate module generates short abstracts. These two elements can be further submitted for translation.

For the MT-Engine of the ATLAS –System on a hybrid architecture combining example (EBMT) and statistical (SMT) machine translation on surface forms (no syntactic
trees will be used) is chosen. For the SMT-component PoS and domain factored models as in (Niehues and Waibel 2010) are used, in order to ensure domain adaptability. An original approach of our system is the interaction of the MT-engine with other modules of the system:

The document categorization module assigns to each document one or more domains. For each domain the system administrator has the possibility to store information regarding the availability of a correspondent specific training corpus. If no specific trained model for the respective domain exists, the user is provided with a warning, telling that the translation may be inadequate with respect to the lexical coverage.

The output of the summarization module is processed in such way that ellipses and anaphora are omitted, and lexical material is adapted to the training corpus.

The information extraction module is providing information about metadata of the document including publication age. For documents previous to 1900 we will not provide translation, explaining the user that in absence of a training corpus the translation may be misleading. The domain and dating restrictions can be changed at any time by the system administrator when an adequate training model is provides.

The translation results are then embedded in a document model which is used further for crosslingual search.
Each document is thus converted to the following format

```
<foaf:Document
rdf:about=http://atlas.eu/item
#20>
<dc:title>Internet Ethics
</dc:title>
<dc:creator
rdf:resource=http://atlas.eu/p
ers#950 />
<atlas:summary
xmnls:lang=“en”>
Default english summary
<atlas:summary>
<atlas:summary
xmnls:lang=“de”>
 Deutsche Zusammenfassung
</atlas:summary>
</foaf:Document>
<foaf:Personrdf:about=http://atla
s.eu/pers#950>
<foaf:name>Name </foaf:name>
<foaf:mbox>    name@some.address.eu
</foaf:mbox>
</foaf:Person>
```

This ist he basis for creation of the RDF-Index. The crosslingual serch engine is in this case a classic Lucene search engine, which operates however not with word-indexes but with these RDF-indexes, which automatically include multilingual information. This engine is currently under construction.

4 Conclusions

In this paper we presented two approaches of embedding multilingual information into search engines.

One is based on the construction of a language independent ontology and corresponding lexical material, the other one on machine translation.

The first approach relies on a manual constructed ontology, therefore it is highly domain dependent and requires the involvement of domain specialists.

The second approach relies on machine translation quality, and also lacks a deep semantic analysis of the query.

However the mechanism can be implemented completely automatically, and is domain independent (assuming that the machine translation engine contains domain adaptation models)

Therefore it is difficult to asses which approach performs better. Further work concerns the selection of a certain domain and comparison of retrieval quality fort the two approaches.

5 Acknowledgements

The work described here was realized within two projects: LT4eL (http://www.lt4el.eu) and ATLAS (http://www.atlasproject.eu).

The author is indebted especially to following persons who contributed essentially for the fulfillment of the described modules: Kiril Simov and Petya Osenova (Bulgarian Academy of Sciences), Eelco Mosel (former at the university of Hamburg), Ronald Winnemöller (University of Hamburg).

References

Karagiozov, D. Koeva,S. Ogrodniczuk, M. and Vertan, C. *ATLAS — A Robust Multilingual Platform for the Web.* In Proceedings of the German Society for Computational Linguistics and Language Technology Conference (GSCL 2011), Hamburg, Germany, 2011

Lemnitzer, L. and Vertan, C. and Simov, K. and Monachesi, P. and Kiling A. and Cristea D. and Evans, D., *„Improving the search for learning objects with keywords and ontologies“.* In Proceedings of Technologically enhanced learning conference 2007, p. 202-216

Niehues J. and Waibel,A. Domain Adaptation in Statistical Machine Translation using Factored Translation Models, Proceedings of EAMT 2010 Saint-Raphael, 2010

Simov, K., Peev, Z., Kouylekov, M., Simov, A., Dimitrov, M., Kiryakov, A. (2001). CLaRK - an XML-based System for Corpora Development. In: Proc. of the Corpus Linguistics 2001 Conference. Lancaster, UK.