Semi-automatic construction of multilingual lexicons

Lynne Cahill*
ITRI, University of Brighton,
Lewes Rd,Brighton BN2 4GJ,UK
Lynne.Cahill@itri.bton.ac.uk

Abstract

The construction of lexicons for NLP applications is a potentially very expensive task, but a crucially important one, especially in multilingual applications. The automation of the task from generic data sources or corpora is as yet largely impractical for most *applied* systems. In this paper we describe a methodology for the semi-automation of the task, used in the CLIME project to develop bilingual lexicons for generation in a restricted domain. We go on to discuss ways in which the same methodology has been used to develop lexicons for a range of applications.

1 Introduction

Despite a large variety of research in recent years addressing issues of the construction of large lexical resources in a range of languages, it is still the case that most NLP applications do not make use of such resources, but produce tailor-made lexicons for each application. Projects such as ACQUILEX (Copestake et al, 1995), GENELEX (GENELEX Consortium, 1994), EDR (EDR, 1990) and MULTILEX (MULTILEX (1993)) have made great advances in the creation of lexical resources, but practical applied NLG systems, for example, almost invariably make use of relatively small, manually produced specialised lexicons (Cahill, 1998b). We stress here that we are not addressing lexicon building for the purpose of MT, but for other multilingual NLP tasks, namely NLG and NLU. As we shall see, it is often the case in practical NLP tasks that sophisticated theories of semantic relations are not required for adequate performance, in contrast to MT.

There has been a significant amount of work on the structuring, development and maintenance of lexicons for NLP, particularly in the tradition of non-monotonic inheritance. Daelemans and Gazdar (1992) and Briscoe et al. (1993) bring together

*This work was supported by the CLIME project (Computerised Legal Information Management and Explanation), EU project number EP25414. See http://www.bmtech.co.uk/clime for more details of the project. The contents of this paper were presented in a seminar at the ITRI. I am grateful to the audience for their comments on that occasion.

much of this work on the application of inheritance networks to lexical description, while Cahill and Evans (1990) discusses the issue in relation to the practical goal of making lexicons more portable and extendable.

Other discussions of the development of lexical resources include work on extraction of information from corpora, such as Garside et al (1997); and work on the extraction of information from machine-readable dictionaries, such as Boguraev and Briscoe (1989). However, what is required for the application we have in mind is a semantically much less complex set of lexical information that nevertheless would benefit from shared cross-linguistic information.

In this paper we discuss the methodology we adopted in developing the lexicons needed for an applied NLG system and the reasons for it. This methodology involved a combination of manual and automated development and has resulted in a set of tools that will enable a non-linguist domain expert to enter the required lexical information to port the lexicon to a new language. We first look at the particular lexical requirements for the CLIME system interface. We then discuss the approach we adopted in the development of English and French lexicons for the CLIME interface before considering similar approaches to lexicons for different NLP tasks. We argue that this type of approach is the most likely way forward in exploiting the wide range of generic lexical resources in NLP applications, as it permits the system developer to combine any number of distinct resources while also tailoring the output to the particular application at hand.

2 The CLIME Project requirements

The CLIME project is developing a legal reasoning system which can be used by ship surveyors to query a database of legal regulations. The user interface to this is the WYSIWYM (Power, Scott and Evans, 1998) system, which is implemented primarily in ProFit (an extension of Prolog). The user formulates questions by manipulating on-screen texts. These texts contain spans which can either optionally or obligatorily be expanded by the use of menus. In the domain we are modelling, the maritime domain, there are around 3300 *concepts* that have been identified by our partners at the University of Amsterdam as occurring in the portion of the rules they have modelled to date. Each of these concepts needs a lexical entry, providing the syntactic and realisational information needed to generate sentences about the concept. Given the presence of a concept *bilge pump* in the ontology of the system, the WYSIWYM interface will allow the user to phrase such questions as *What is a bilge pump?*, *What are all the parts of a bilge pump?*, *What are the things connected to a bilge pump?* and so on. When the answers to the question are returned by the other modules of the system, the response is generated in the chosen language by a back-end generation module.

We need lexical entries for these concepts in both English and French, but we do not require any subtle semantic information for the range of questions that the user can sensibly ask the system. We simply need one form for English and one for French for each concept.

The task of finding simple one-to-one, domain specific translations of the concept set we wanted to represent proved more difficult than we had hoped. On-line dictionaries could be found which gave us the translations we (thought we) wanted, but only amongst several others which we clearly didn't want. In addition, we found that *we* didn't always know which of the translations returned we wanted – this was knowledge that only experts in the domain could reliably provide.

It must be stressed that the implementation of the system makes certain simplifying assumptions about the differences between English and French that prove acceptable in the current application, but which would not be acceptable in an application to perform a different NLP task, such as Information Extraction. These assumptions result in virtually identical grammars for English and French, grammars that are sufficient to generate the limited range of language required for this interface. With the exception of certain rules for the handling of English plurals, the only differences between the languages are handled in the lexicon, either as word forms or as fixed phrases. It is our assumption that any differences that required more sophisticated grammatical treatment would require a (computational) linguist to implement, while the domain specific lexical forms require a domain expert. However, in the model we propose here, the two tasks are entirely separated, so that porting the lexicon to a different domain, or just extending it, can be performed after the linguist has finished development and the system has been deployed.

3 The CLIME lexicons

The CLIME system has two NLG modules – one which the user interacts with to compose a query and the second which generates the linguistic version of the answer to the query. As discussed above, the first of these uses the WYSIWYM system (Power, Scott and Evans, 1998), which is implemented in ProFit, an extension of Prolog. The system currently generates English and French, and will shortly be extended to include Italian. The core parts of the lexicons for the NLG modules were entered manually, including the core lexemes for each language – i.e. determiners, common nouns, auxiliaries, fixed phrases for the domain etc.. For the NLG modules to function, however, it is vital that there is a lexical entry for each concept in the domain model. The domain model for the NLG is derived from an ontology (the Legal Knowledge Repository or LKR) that is used by the legal reasoning system. We subsequently devised a system for automatically extending all the lexicons required to cover all of the concepts in the ontology.

The ontology comes to us in HTML format. From this we derive two things: a database consisting of the concept name and the major syntactic category; and a prolog theory consisting of subtype definitions. To this database, we manually added French translations of the concepts[1], together with their gender. There was no obvious alternative to this manual translation effort, because the translations we required were very domain specific. As we discussed above, we could not find any machine-readable dictionary that could provide for us the single most appropriate translation for terms

such as "bilge pump" or "horizontal bulkhead". This is an area where domain experts are needed, but we did not want to force these experts to get their hands dirty entering the translations into a structured lexicon, nor did we want to have to enter all of the (3000+) translations manually ourselves. Thus, we opted for the best compromise, where the French experts entered the translations into a simple database (in fact it was done in an Excel spreadsheet which we subequently dumped out into ASCII) from which we could then automatically generate the structured lexicons required.

From this database, a set of inheritance-based hierarchically structured lexicons, were produced, with the top structure manually crafted and the bulk of the lexemes at the leaves automatically generated[2]. These included the sharing of cross-linguistic information. In contrast to the PolyLex model (Cahill and Gazdar, 1999), in which shared information is contained in a separate multilingual hierarchy, the default hierarchy in this case was the English one. The main reason for this was simply the practical consideration that we had started with the English lexicon and then extended it to French. However, this also carries the benefit of being able to use the English word where the French translation is not available. Although not an ideal situation, it was felt that it was better to have an English term appearing in the French text than to have the system fail to produce a text at all if some French translations were missing. It is also the case in this particular application that many of the concepts are actually abbreviations (e.g. "cbt", "ice_i"), for which it does not make sense to have a translation.

The next stage of generation of the lexicons combines the hierarchically organised information with the prolog subtype information to construct ProFit entries as required by the WYSIWYM system. The subtype information is used to determine whether a noun is mass or count – subtypes of "ship", "equipment" , "system" etc. are count, while subtypes of "notation", "state" etc. are mass[3]. Let us look at an example lexical entry.

The NLG part of the WYSIWYM system is written in ProFit, and consists of grammar rules that the generator attempts to instantiate by realising the "right-hand side" where the meaning matches the "left-hand side". The lexicon is essentially a set of declarative rules that define sets of feature-value pairs that correspond. In generation terms, this means that we index on (primarily) the `meaning` feature, and the output is the value of the `cset` feature. The WYSIWYM lexicon needs entries like the following:

```
word(english, meaning!cargo_ship &
              syntax!(category!noun &
                      opening!consonant &
                      form!common &
                      noun_type!count) &
                      cset!'cargo ship').
```

Here, the ProFit defines a set of feature/value pairs such as `noun_type` (feature) and `count` (value). In the automatically derived section of the lexicon in DATR the corresponding entry for 'cargo ship' in English is:

```
E_Cargo_ship:
    <> == Noun
```

```
    <syntax category> == noun
    <opening> == consonant
    <form> == common
    <noun_type> == count
    <cset> == 'cargo ship'.
```

In French, this is:

```
F_Cargo_ship:
    <> == E_Cargo_ship
    <gender> == masc
    <cset> == 'navire cargo'.
```

From these basic DATR entries, lexical entries are generated for both languages, for two different types of entry that are used for asking different types of question. In addition, the second NLG module requires slightly different lexical entries again, and these too can be generated from the same DATR entries. We therefore generate six separate lexicons from these entries, the main WYSIWYM lexicons, the concept lexicons used by one more specific part of the WYSIWYM interface and the lexicons for the back-end generation.

The whole process is illustrated in figure 1. In the figure, the solid boxes are what we consider to be non-lexical databases or information sources[4]. The dashed boxes are lexicons. The solid arrows between boxes are fully automatic derivation, while the dashed arrows indicate manual derivation.

4 Other NLP applications

The methodology described above can be viewed as having at least two stages: the first moving from a (largely unstructured) database to a more highly structured lexicon and the second from this structured lexicon to an application specific lexicon which may be less structured again, but which may have more highly structured (and programming language specific) individual entries.

In this section we briefly discuss two different lexicon building processes that each undertake one of these two levels. The PolyLex automatic extension process takes the largely unstructured CELEX database to extend the highly structured PolyLex multilingual lexicons. The lexicons for the POETIC project were constructed as highly structured lexicons, from which less structured, application specific lexicons were automatically derived. We shall look at each of these in turn.

4.1 The PolyLex lexicons

The aim of the PolyLex project was not to build lexicons for a particular application or application type, but to develop hierarchically structured lexicons that organised the information about related languages in a way that permitted sharing of information

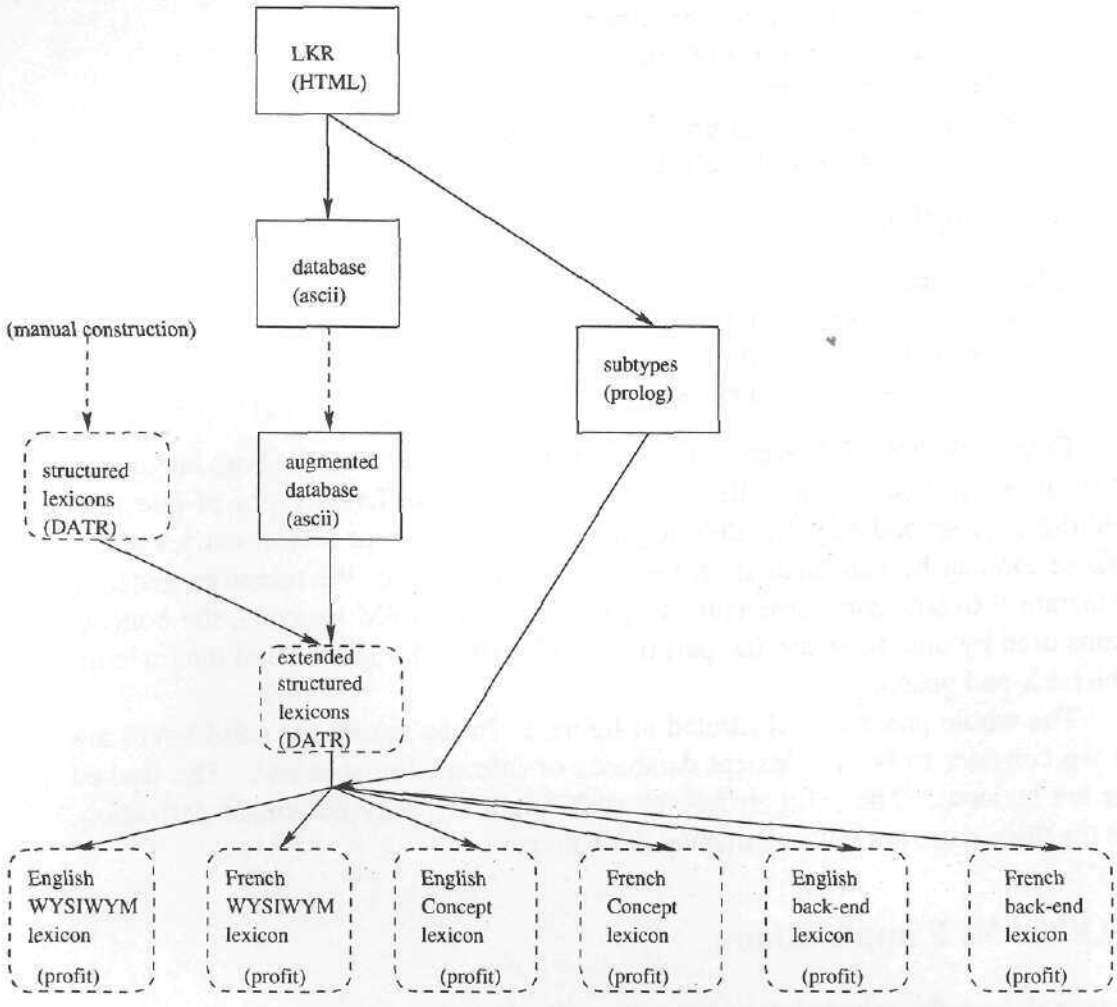

Figure 1: The automatic creation of the CLIME lexicons

across all different levels of linguistic description (Cahill and Gazdar, 1999). The resulting lexicons covered morphological, morphophonological and phonological information primarily, with some syntactic and orthographic information. The information common to two or more of the three languages covered – Dutch, English and German – was contained in a *multilingual* hierarchy[5], with the individual language hierarchies inheriting this information by default and overriding it where necessary.

The methodology employed in developing the lexicons was to first develop a core multilingual lexicon including around 300 words for each language. These items were chosen because they were representative of all of the different *morphological* classes, and so they included most of the irregular words of each language. These were developed as default inheritance hierarchies, implemented in the lexical knowledge representation languages, DATR (Evans and Gazdar, 1996), with the lexemes as the leaf nodes of the hierarchy. In order to then extend the lexicons to the intended level of 3000 words for each language, it was decided to automatically induce the lexical entries from

a combination of the CELEX lexical database and manual translations from English into German and Dutch. This manual translation was chosen again because of the difficulty in finding automatically simple one-to-one translations. The translations were done by bi-lingual speakers of English/Dutch or English/German who could most reliably give the most straightforward translations of the list of common words.

This automatic extension assumed that the words could be added to existing morphological classes, so the structured morphological information at the top of the hierarchy had to be in place. The information contained in this hierarchy also had to be employed, albeit in this case in a different form, in the extension algorithm itself, since information from CELEX about the different word forms of the different lemmas was used to deduce the morphological class. Thus, for instance, in German nouns, the nominative singular, nominative plural, accusative singular, genitive singular and dative plural were all examined to infer the inflectional class of the noun. Any words which did not fit one of the classes was defined as a member of the default (regular) class and also placed in a list of entries to be checked manually.

As well as this type of monolingual deduction, the automatic extension algorithm decomposed the root forms into their syllable constituents and extracted cross linguistic commonalities across these constituents.

The semi-automatic extension of the PolyLex lexicons resulted in a fairly substantial set of lexicons for the three languages addressed. It demonstrated the use of largely unstructured databases to induce the leaves of manually constructed highly structured lexicons. However, it also has its limitations, especially from the point of view of applied NLP.

In the first place, it could be claimed that the lexicons were actually constructed from other lexicons, as the CELEX databases, although not highly structured, are nevertheless a non trivial collection of specialised linguistic data. Indeed, the availability of such sources for other languages is variable, to say the least. Secondly, the resulting lexicons themselves are probably not suitable for use in any NLP applications in their present form, due to their rather abstract nature. This suggests that we might want to consider a model of lexical construction that does not have "input sources" and "output lexicons" but rather a multi-layered model that may have a variety of different sources being "refined" and combined into a variety of ultimate output lexicons.

In such a view, the PolyLex lexicons are somewhere in the middle of the layering, being a refinement of a set of already quite sophisticated lexical databases, but needing further "refinement" to make them useable for a NLP application.

4.2 The POETIC lexicons

The POETIC project (Evans et al, 1995) was a follow-on project, further developing the TIC message understanding system to be more extendable and portable. The system takes police reports of incidents that are logged by operators and uses Information Extraction techniques to build a picture of any incidents that may affect traffic, broadcasting automatically to motorists about any relevant incidents.

The lexicon in the original system was a simple lookup table, giving syntax

and semantics for each domain specific or very common English word. However, in contrast to the requirements of the CLIME NLG system described above, there was a need to have potentially several different forms for each meaning, so that all the forms that might arise in the input texts could be recognised. The revised lexicon structure, designed to simplify porting the system to a new police force sub-language, had to ultimately produce the same output as the original system. It was decided that, for these reasons, together with reasons of efficiency, the lexicons would be defined as highly structured inheritance based lexicons whose content was "dumped" out into simple lookup tables as were in the original system. This meant that we could adapt various aspects of the lexicons, including adding quite substantial sets of new entries, relatively simply, by adding leaves to the inheritance trees. This enabled the new entries to inherit all of the more general information from higher points in the hierarchy, while the complete lexical entries required by the system were automatically generated on the basis of this hierarchically organised information.

5 Conclusions

We have presented a lexical architecture for NLP systems that involves potentially numerous layers of information, of possibly different granularity as well as different form. We have also presented examples of how these layers may be automatically or semi-automatically constructed. Thus, in the example of the CLIME system described above, the derivation of the monolingual database from the ontology is fully automatic. The extension of this database to be bilingual is entirely manual. The construction of the DATR lexicons from this database is fully automatic, but the next stage, to produce the ProFit lexicons, is only semi-automatic, relying on a hand-coded lexical hierarchy to be in place for the automatically derived leaves to attach to.

Essentially the same methodology was employed in the POETIC NLU system to produce lexicons for the different sublanguages used by different police forces. We believe that this kind of approach to lexical development is the way forward, allowing the use of many and varied sources at different levels to (semi-)automatically construct, or at least extend, lexicons for genuine multilingual applications.

Notes

[1]The translations were provided by our project partner, Bureau Veritas in Paris.

[2]These lexicons were defined in the lexical representation language DATR (Evans and Gazdar, 1996).

[3]Although this is a simplification, it is one which works a large proportion of the time. All of the automatic lexicon construction described here assumes that some checking may be necessary to deal with certain lexical exceptions. In some cases there are ways of dealing with this explicitly. For example, the automatic construction of the PolyLex lexicons (Cahill 1998a) produces a separate file for words whose morphological behaviour does not exactly match any of the available classes, while those words are given default morphological values in the automatically produced lexicons.

[4]Of course, the boundary between these different types of resource are unclear. The HTML LKR, for instance, is not strictly a lexical resource, but it nevertheless contains a large proportion of the information

required by a lexicon.

[5]This should more properly be described as a set of hierarchies, as the different levels of information tend to be defined in essentially separate, although possibly interacting, hierarchies.

References

Baayen, Harald, Richard Piepenbrock & H. van Rijn (1995). The CELEX Lexical Database (CD-ROM), Release 2. Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA.

Briscoe, Ted, Valeria de Paiva & Ann Copestake, eds. (1993). *Inheritance, Defaults, and the Lexicon,* Cambridge: Cambridge University Press.

Boguraev, Branimir and Ted Briscoe (eds) (1989). *Computational Lexicography for Natural Language Processing*, London: Longman.

Cahill, Lynne (1998a). "Automatic extension of a hierarchical multilingual lexicon" *2nd Workshop on Multilinguality in the Lexicon, ECAI-98*, 16-23.

Cahill, Lynne (1998b). "Lexicalisation in applied NLG systems". ITRI technical report, ITRI-99-04, obtainable via
`http://www.itri.brighton.ac.uk/projects/rags/`.

Cahill, Lynne & Roger Evans (1990). "An application of DATR: the TIC lexicon" *Proceedings of ECAI-90*, Sweden, August 1990, 120-125.

Cahill, Lynne and Gerald Gazdar (1999). "The PolyLex architecture: multilingual lexicons for related languages" *Traitement automatique des langues*, **40:2**, pp. 5-23.

Copestake, Ann, Ted Briscoe, Piek Vossen, Alicia Ageno, Irene Castellon, Francesc Ribas, German Rigau, Horacio Rodriguez and Anna Samitou (1995). "Acquisition of lexical translation relations from MRDs" *Journal of Machine Translation*, **9:3**, 1-35.

Daelemans, Walter & Gerald Gazdar, eds. (1992). *Computational Linguistics* **18.2** & **18.3**, special issues on inheritance.

EDR (1990). Bilingual dictionary. Technical Report **TR-029**, Tokyo: Japanese Electronic Dictionary Research Institute Ltd.

Evans, Roger & Gerald Gazdar (1996). "DATR: A language for lexical knowledge representation" *Computational Linguistics,* 22.2, 167-216.

Evans, Roger, Robert Gaizauskas, Lynne Cahill, John Walker, Julian Richardson and Anthony Dixon (1995). "POETIC: A system for gathering and disseminating traffic information" *Natural Language Engineering*, **1:4**, pp. 363-387.

Garside, Roger, Geoffrey Leech and Anthony McEnery (1997) *Corpus annotation: linguistic information from computer text corpora*, London: Longman.

Genelex Consortium, Report sur le multilinguisme, Version 2.0, December 1994

Hajicŏvá, Eva & Zdeněk Kirschner (1987). "Fail-soft ("emergency") measures in a production oriented MT system" *Proceedings of the third conference of the European Chapter of the Association for Computational Linguistics*, Copenhagen, 104-108.

Kameyama, Megumi (1988). "Atomization in grammar sharing" *26th Annual Meeting of the Association for Computational Linguistics,* 194-203.

MULTILEX (1993). Linguistic description of the MULTILEX standard. Boulogne-Billancourt: CAP GEMINI INNOVATION.

Power, Richard, Donia Scott & Roger Evans (1998). "What You See Is What You Meant: direct knowledge editing with natural language feedback" *Proceedings of ECAI-98*, UK, August 1998, 677-681.

Poznański, Victor, John L. Beaven and Pete Whitelock (1995). "An efficient generation algorithm for lexicalist MT" *33rd Annual Meeting of the Association for Computational Linguistics,* 261-267.