Use of Linguistic Resources like Translation Memories in Machine Translation Systems

Lee Humphreys

Abstract

The context and general technical strategy for MAHT and MT at ERLI is presented. After presenting the commercial context of translation, we outline the architecture of AlethTR, ERLI's assisted-translation platform.

Special attention is paid to the question of linguistic resources : general lexicon, terminology, translation memory. We consider briefly the integration of translation memory with MT and its extension to the sub-sentence level.

Lee Humphreys

Lee Humphreys currently heads the Grammar group in the Research and Development Department of ERLI. He is also involved in the design of translation systems. He was previously responsible for French-English MT development at SITE-EUROLANG (Paris) (1992-1994). Prior to this he was part of the CL/MT Group in the Department of Language and Linguistics at the University of Essex (UK), working on MT Evaluation and linguistic specifications in the EUROTRA project.

GSI-Erli

Created in 1977, ERLI is Europe's leading language and document engineering company. With a turnover of MFF 31 in 94, the company has a team of 60 computational linguists and software engineers. The principal areas of activity are natural-language based indexation and document retrieval (AlethIR), text generation (AlethGen), controlled-language checking (AlethCL), translation (AlethTR), linguistic knowledge manipulation (AlethKES), terminology management (AlethGT) and general lexical management (AlethGD).

Lee Humphreys
ERLI
1 place des Marseillais
F-94227 Charenton-le-Pont CEDEX
France

Tel: +33 (01) 48 93 81 21, Fax: +33 (01) 43 75 79 79
E-mail: info@erli.fr
WWW: http://www.erli.fr

Introduction

AlethTR™ is a translation support tool from ERLI that integrates

- bilingual terminology management
- identification and default translation of candidate new terms
- translation memory (TM)
- light machine translation (MT)

together with text handling, project costing software, translation utilities and user interfaces.

As has become standard in translation engineering, the MT engine is used to translate - if required - text which is not found in translation memory. The grammar of the current engine can be customised by ERLI to suit the particular type of text routinely handled by a customer.

Translation Context

ERLI's translation tool engineering effort has historically concentrated on supporting high-volume professional technical translation. It is worthwhile for the translator to invest a considerable amount of time and effort in a project preparation phase since this is recouped by improved quality and speed during the translation phase.

A project typically involves from one hundred to several thousand pages, where all the texts in the project belong to the same domain and text type. The French texts we have looked at have syntactic and lexical restrictions characteristic of sub-languages:

restrictions on verb form

Verbs restricted to 3rd person, mostly present tense, limited use of past and future, perhaps complete exclusion of the French passé simple. Interrogatives are restricted to very specific document parts such as fault-finding e.g.

Is the main circuit stop valve closed?

anaphora

Pronouns almost always represent things rather than persons

restrictions on relatives

Object relatives are rare

determination

Greater use of zero determination than in standard French e.g. before a deverbal noun e.g.

Avant montage du joint, nettoyer ...

Sentence and part sentence repetition rates are high, often because they correspond to standard warnings or advice, or because they describe frequently repeated operations / states. For example, we have found in a series of related projects for the same customer (same domain, same text type) that typically half the text is covered by exact matches in the TM.

General vocabulary is nearly closed and polysemy reduced. For example, whilst a French dictionary may give several senses to the pronominal verb *s'afficher*, the sense meaning *flaunt* as in

Elle s'affiche partout

is most unlikely in a technical text.

Translating with AlethTR™

In this section we rapidly sketch the various steps involved in a translation project with AlethTR™. The object is to highlight the tools available and their contribution to the translation process.

Project Preparation Activities

Validate terminology

One of the first thing a professional translator does before tackling a translation project - be it machine-assisted or otherwise - is to draw up a list of the terms in the source text and their translations.

Although the lexicon of AlethTR™ may contain some appropriate terms at the start of the project, their translation must be validated. AlethTR™ lemmatises and syntactically tags the source text, identifies known terms and presents their translations for validation.

Identify potential new terms

Even for a familiar domain, the texts in a new translation project are likely to contain new terms. AlethTR performs a syntactic analysis of the text and uses a template-based search algorithm on the noun phrases in the resultant structure to allow identify instances of candidate multiword terms. For example, a template for French might be

N de/à N

as in *bac de récupération* or *filtre à huile*. Instances of candidate terms found in the text e.g. *bacs de récupération, filtres à huile*, are normalised to standard term form and statistically processed, taking into account such factors as the frequency of the candidate term in the text(s). The translator validates the resultant list.

Find translations for new terms

default term translation rules

Many new terms entering into a domain tend to translate compositionally e.g.

N1 de/a N2 => N2 N1

AlethTR™ exploits such rules to propose translations for the candidate terms it has found in the source text. The proposed translation can be modified before being entered into the term translation dictionary.

statistical identification of candidate translations

In another approach currently under development we align source and translation in the customer's legacy corpus. Statistical techniques find segments in the translations which correspond to terms in the source text. Although computationally quite heavy,

this approach has the advantage that the target terms identified need not be compositional translations. (Not yet available in the standard AlethTR™ product.)

Identify repeating text

AlethTR™ identifies repeating text segments in source text. It can provide a default translation for repeating sequences, using existing translation memory, terminology translation and the translation engine. The sequences with their translations are made available to an editing interface, allowing the translator to select appropriate sequences and to edit the proposed translation. The result is used to create a project Translation Memory.

Identify unknown words

AlethTR™ signals the presence of unknown wods and attempts to predict their category. The translator can enter the words in the dictionary and supply an appropriate translation.

Identify possible translations for general language words

The general language dictionary can be organised to allow the user to partition translations on semantic grounds. A nice example of this is the French word *fraise*, which can be translated as strawberry (fruit) or reamer (engineering tool). Since it is possible to arrange for (say) fruit and vegetables to be grouped into a sublexicon, inappropriate references to strawberries can be avoided in engineering text translations by selection of the appropriate sublexicon.

However, no amount of chopping and changing of sublexica will entirely eliminate the problem of multiple translations for general language content words. Rather than trying to calculate very elaborate lexical selection constraints in the translation engine, AlethTR™ encourages direct control by the translator during the preparation phase. Working from a lemmatised and tagged internal representation, AlethTR generates a list of all the general language word types in the text, together with examples of source text context and their possible translations as given by the bilingual dictionary. It is very easy to rapidly scan through this list and indicate to AlethTR™ the preferred translations.

Of course, the preferred translation of a general language word can easily vary from one context to another. However, recall that in technical texts the number of senses (and hence translations) of a word is usually limited with respect to general language, and that often a potentially polysemous word turns out to have only one sense in a particular text(s). Hence this translation preselection step turns out to be surprisingly cost-effective. (Surprising, that is, for MT professionals, who are used to devoting a great deal of effort to the lexical selection problems posed by general purpose translation).

Costing

Using a sample text(s), and various user definable cost parameters, AlethTR generates a fully detailed quotation with estimates for the contribution of Translation Memory to the projected project. This allows an immediate assessment of the cost effectiveness (which is closely related to the repetitivity of the documents).

Translation and Revision Phases

After preparation, translation can involve

- Translation of terminology alone
- Translation using exact-match or exact- and fuzzy-match translation memory
- Translation using MT

The result is made available in a revision/translation interface, which displays the source text and its translation (line by line). Colour codes and underlining highlight client terms, base terms, terms with several translations, terms with usage notes, unknowns, phraseology, exact match TM, fuzzy-match TM and MT. This interface allows entirely free navigation and editing i.e. unlike with some other products, the user is not required to translate/revise in a strict sequence from the top to the bottom of the file.

Update of TM

When the translation has been revised and corrected, AlethTR creates an extension to the Translation Memory. The compiled TM can be used for other texts in the project or in other projects.

Billing

At the end of the project, AlethTR^{TM} generates a customer bill.

Directions in Linguistic Resource Management

AlethTR^{TM} has a linguistic knowledge base (monolingual and bilingual vocabulary together with terminology information) which can be directly inspected and modified by the user.

The AlethTR^{TM} linguistic knowledge base fits into a high-level approach to lexicographic resources at ERLI. Our general strategy involves the creation of very rich lexical resources in a generic lexicon model - GENELEX. A powerful lexicon manager and lexicographer workbench - AlethGD - provides

- OO management of large lexical resources according to the GENELEX model
- graphical display and navigation of lexical information e.g. semantic networks
- special facilities (including a linguist-usable OO dictionary programming language) for

  - importing dictionary data in virtually any external format
  - exporting dictionary data to
    * ERLI applications
    * other end users or dictionary managers

All general language lexical resources are created, validated and maintained in the GENELEX format by the Dictionary team. ERLI applications such as AlethTR, AlethCL (a Controlled Language checker), AlethGEN (a language generator) and AlethIR (a natural language front end to document management systems) call upon the same GENELEX resource via specific export programs. The approach allows us

this one source to supply linguistic processors which use completely different underlying grammatical theories such as

- LFG
- Meaning-Text Theory Dependancy Grammar
- Simple Constituency Grammars

and so on

GENELEX

The GENELEX model (MENON 94) is extensively described in the three reports (Consortium 93a; Consortium 93b; Consortium 94) and will not be presented in detail here. With relatively minor modifications, this model forms the basis of the EC's LE PAROLE project, which involves the construction of lexica for 20,000 or more Ums in 12 European languages. Given the level of activity based on this model, it can be considered to be a de facto European standard.

The model is intended to be multi-theoretical, providing a descriptive formalism which allows linguistic facts to be drawn from different linguistic theories. The GENELEX model provides for a very high level of factorisation of linguistic information.

There are three layers of representation : morphology, syntax, and semantics. The morphological layer relates a given morphological unit (a UM) to its variant written forms (UMGs) e.g. *bosun* and *boatswain* are variant spellings of the form *boatswain* in English. A given UM is associated with a inflection paradigm.

A given UM has one or more USYNs - syntactic units. Each such unit describes a single syntactic behaviour of the UM. Typically a verb UM e.g. *tackle* might have several USYNs, each USYN describing a particular syntactic complementation pattern for that verb. (The internal structure of a USYN is quite rich, and allows full description of local trees. Interestingly, the described element does not necessarily have to be the root of the local tree.)

Finally, a given USYN can have one or more semantic units (USEMs), where each semantic unit corresponds to the notion of a word sense. Since a given sense can have more than one syntactic realisation, a USEM can itself point back to more than one USYN. A USEM can be linked to a predicate-argument structure. USEM-USEM links allow the construction of a classic semantic network.

The multilingual part of the GENELEX model (Consortium 95) establishes reversible bilingual links between two monolingual models : these links can be at the usyn level, at the usem level, at both usyn and usem level, or at the predicate level.

Terminology - TRANSTERM

The GENELEX model was particularly intended for the representation of general language. An extension of GENELEX which adds further information for representing the usage of terms was developed in the context of the TRANSTERM project (EC LRE project TRANSTERM - Creation, reuse, normalisation and integration of terminologies in natural language processing systems). The TRANSTERM approach allows a terminologist or ordinary user - who does not

necessarily have the theoretical linguistic expertise of an GENELEX / AlethGD user - to enter terminological data without entering into the full linguistic richness of the GENELEX model.

Just as the GENELEX model has its associated lexical DBMS - AlethGD - intended for specialists, so also TRANSTERM has a corresponding DBMS - AlethGT™ - intended for use by customers. Like AlethGD, AlethGT™ is based on an OO DBMS with specialised navigation and lexicographic facilites.

AlethGT™ also stocks an entire GENELEX-conformant general language dictionary. When the user creates or modifies his/her terminology, AlethGT™ automatically hunts down in the GENELEX part of the dictionary all the morphological, syntactic and semantic information associated with the component parts of the term. This information is essentiel for the operation of linguistic processors such as AlethTR™. The AlethGT™ terminology manager hides all this complexity from the user.

Lifecycle aspects

In the design of the TRANSTERM model and the resultant DBMS AlethGT™, particular attention was paid to the problem of ensuring consistency between customer supplied and managed terminology, on the one hand, and ERLI supplied and maintained general language dictionary information on the other. The customer manages his/her terminology from day to day: at periodic intervals, ERLI or other third-parties might issue new and extended versions of the GENELEX general lexicon. Using the AlethGT™ terminology manager, the customer can download this new dictionary version - thus improving the performance of the linguistic processor - with minimal disruption to his/her terminology.

AlethGT™ is just coming on stream and will be progressively integrated with ERLI products.

Linguistic Knowledge Extraction - KES

As we have seen, AlethTR™ provides a number of tools to aid the translator in the preparation of the lexicon for a project e.g. tools for the identification of new terms. These tools are frequently used on a historical corpus for a given client.

Extraction of lexicographic and grammatical information from corpora is an activity carried out in the context of all types of NLP applications and products, not just translation (OGONOWSKI 94). For example, the construction of an appropriate thesaurus is an important prelude to optimised domain-specific information retrieval with AlethIR.

ERLI's AlethKES™ responds to this need, integrating facilities for corpus analysis and the manipulation of hypotheses. One typical use is to allow a terminologist to progressively construct a conceptual network appropriate for a particular domain. Starting from initial hypotheses, the structure of this network can be updated, modified or completely restructured in the light of additional information or further insight. The resultant network could be used as a basis for lexicographic coding in AlethGT™. (AlethKES™ is very closely coupled to AlethGT™ and shares the same underlying software architecture.)

It is intended to progressively integrate in AlethKES^{TM} a variety of specific corpus exploration tools, including, for example, the repetition analysis tools currently found in AlethTR^{TM}.

Directions in Machine Translation and Translation Memories

ERLI is developing a new robust general purpose translation engine based on a new analyser. The first component in the analyser is a constraint-based morpho-syntactic analyser (KARLSSON 90; KARLSSON 95; ZAYSSER 96). This analyser attempts to assign appropriate categories and basic morpho-syntactic features (e.g. number, tense, mood etc) to words in a sentence by intersecting an initial word automaton produced by lexical lookup with a finite-state automaton compiled from a set of linguistic rules in the form of regular expressions.

An additional ruleset in this analyser allows the calculation of a *presyntax* - the assignment of a surface syntactic function such as **main-verb, subject, object, preposition-complement, noun-premodifier** and so on to each word in the sentence. In the case of complex sentences, further rules indicate the location of clause boundaries.

A special lifting algorithm parses the presyntax and creates a dependency graph. This graph is similar to a classic dependency tree structure except that a given node made have more than one potential governor, reflecting residual functional ambiguities and attachment ambiguities characteristic of these surface functions. Subsequent operations establish which of the component trees in this graph maximises the satisfaction of syntactic, semantic and general heuristic constraints. The selected optimal surface dependency is the result of the analysis (GRAAL project carried out in conjunction with Aérospatiale).

A relatively classic transfer phase transforms the source language surface tree into its target language equivalent. Thereafter a generation component - based on the linguistic part of AlethGEN - creates the corresponding surface string.

Integration with Translation Memory

Translation Memory techniques do not have to respect natural linguistic boundaries. It is perfectly appropriate to stock paragraphs, sentences, or fragments of sentences in TM provided that these correspond to a certain level of repetition.

However, most existing integrations of MT and TM only pass the MT engine segments delimited by strong punctuation when these are unrecognised by the TM: the MT engine is not asked to translate arbitrary fragments of sentences. If it was, having no idea what sort of fragments it is supposed to be looking at, it is likely to produce poor results. We are exploring two approaches to this problem.

Repetition-driven segment identification

The first approach is to go ahead and identify TM candidate segments by carrying out the classical repetition analysis. Once these segments are obtained, we then try to automatically identify an appropriate category for the segment. For example, the segment

In a scalar context

as in

In a scalar context, function blah returns the number of items in the list

might be classified as an Adverb.

We would then supply to the MT engine something like

proADV, function blah returns ...

where *proADV* is a generic adverb placeholder world in the MT lexicon. It is replaced in the translated output by the TM translation.

One simple approach to automatic categorisation is to try to analyse - or at least tag - the segment in one (or perhaps more) of its simplest use contexts and then to compare the result with a set of templates e.g

```
START_SENT PREP DET? ADJ? N --> ADV
```

This, of course, is just an extension of familiar term candidate recognition techniques.

The features in placeholders may need to be quite rich. Thus

allows users

as in

The cancel command allows users to cancel print requests ...

requires a placeholder with features *verb, 3p, singular, indicative, active, SUBJ+TO-INF*. The success of the approach depends on the reliability with which we can carry out the automatic classification.

Clause Boundary identification

Another approach that we are starting to investigate uses clause boundary information. Recall that our constraint-based morpho-syntactic analyser has a rule set which identifies clause boundaries in complex sentences. For example

The Prime Minister advised people † not to work

As is well known † detergents can damage the skin

Smoking cigarettes † is dangerous

The watched him † smoking cigarettes

It is said † that detergents can damage the skin

where † indicates a clause boundary. When repetitions are found, it is possible to search back in their sentential contexts to see whether they correspond to clause boundaries.

As before, we replace the clause in the matrix sentence by a placeholder. Since analysers expect clauses to be more complex than single words, we use multiword skeletal clause structures where important clause elements are replaced by placeholder words.

proVing proN † *is dangerous*

It is said † **proThat proN proVfin**

(If we used single words as placeholders rather than multiwords, we would have to change our analyser grammars.)

After passing through the translation engine, the skeleton is replaced by the TM translation. If for some reason the skeleton contains insertions induced by the translation process, the whole MT translation is rejected.

The approach depends on an underlying assumption that clauses typically have a translation which is not affected by context i.e. by the matrix sentence in which they are found. This is a question we are only just starting to look at.

Conclusion

AlethTR™ provides powerful facilities for the translation professional. In the context of a global strategy for linguistic resource management based on generic dictionaries, we intend to integrate it with a new-generation terminology manager (AlethGT™) and linguistic knowledge management tool (AlethKES™).

In two further lines of development we are

- preparing a completely new translation engine
- exploring ways to improve MT-TM integration.

Acknowledgements

The techniques and tools described in this paper are the result of work by many people at ERLI. My particular thanks to:

Sophie Corbel, Sylvie Greverend, Marie-Claude Guérin, Béatrice Marchand, Dominique Maret, Laurent Roussarie, Simon Sabbagh

and to Veronika Lux of Aérospatiale for information on the linguistics of Aircraft Maintenance manuals.

References

(Consortium 93a) GENELEX Consortium. Report on the morphology layer. Technical report, 1993.

(Consortium 93b) GENELEX Consortium. Report on the syntactic layer. Technical report, 1993.

(Consortium 94) GENELEX Consortium. Report on the semantic layer. Technical report, 1994.

(Consortium 95) GENELEX Consortium. Rapport sur le multilinguisme. Technical report, 1995.

(KARLSSON 90) F. KARLSSON. Constraint grammar as a framework for parsing running text. In *COLING-90. 13th International Conference on Computational Linguistics, vol. 3 Kargren, H. (ed.) Helsinki, Finlande,* 1990.

(KARLSSON 95) F. KARLSSON. Constraint grammar: a language-independent system for parsing unrestricted text. 1995.

(MENON 94) B. MENON. Eureka project GENELEX: an overview. In *Journées du Génie linguistique: actes, Paris - La Défense,* 1994.

(OGONOWSKI 94) A. OGONOWSKI. Constraint grammar as a framework for parsing running text. In *COLING-94. 15th International Conference on Computational Linguistics, vol2, Kyoto,* 1994.

(ZAYSSER 96) L. ZAYSSER. Representing morpho-syntactic ambiguity. In *MIDDIM-96 Seminar, Le Col de Porte, France,* 1996.