Tooling the Lexicon Acquisition
Process for Large-Scale KBMT

John R. R. Leavitt  Deryle W. Lonsdale  Kevin Keck  Eric H. Nyberg

Center for Machine Translation
Carnegie Mellon University
Pittsburgh, PA 15213
412/268-7282
jrrl@cs.cmu.edu
1 Introduction

In developing knowledge-based systems, the most time-critical step is almost invariably knowledge acquisition. This is especially true to knowledge-based natural language systems, in which the lexicon must grow in parallel with the coverage of the system. [Galinski, 1988] In this paper, we describe the specific case of lexical knowledge acquisition for the KANT machine translation system, the nature of the desired acquisition process, the tools we built to implement that process, and the resulting benefits.

2 Lexicon Acquisition for KANT

KANT is a knowledge-based machine translation system [Nyberg and Mitamura, 1992], built around an interlingual architecture (Figure 1). In such a system, the translation process consists of two stages: an analysis stage in which a source sentence is syntactically and semantically analyzed to create an intermediate or interlingua representation, and a generation stage in which the interlingua is used to construct a semantically equivalent syntactically correct sentence in the target language. The interlingua representation contains only the semantic information contained in the source sentence. The main advantage of interlingua systems is that the a given analysis module may be used for multiple generation modules (e.g., generation
into French, German, and Italian from English analysis) and that these same generation modules may be used with different analysis modules (e.g. translation of documents written in Japanese).

![Diagram](image.png)

**Figure 1: The KANT Interlingual MT Architecture**

KANT is designed to perform translation within well-specified technical domains, such as heavy equipment operation and maintenance manuals. Within such a domain, it is possible not only to restrict the meanings of ambiguous words (EXAMPLE?), but also to handle common phrases from the domain (e.g. suspension control valve housing) as single units. A phrasal lexicon of this sort greatly eases disambiguation during syntactic and semantic analysis, in that it allows the analyzer to cast out conflicting analyses based on preferences with respect to phrases. [Baker et al., 1994] Similarly, having very precise concepts in the interlingua facilitates the generation of the correct target languages expression for that phrase. [Leavitt et al., 1994]

For any such a benefit, however, there is almost always an associated cost,
which in this case is the burden of constructing the lexica on this large scale. The construction of the source language lexicon is largely automated from analysis of existing corpora [Mitamura et al., 1993], but the construction of the generation lexicon requires more effort both because it format is MORE VERBOSE and because online corpora are often not available in the target language.

The process by which the target language lexicon is acquire must have the following attributes:

- **It must be easy.** For most technical domains, the details of vocabulary usage will lie outside the domain of the knowledge engineer. This necessitates the employment of a domain expert to perform the majority of the knowledge acquisition. Since this person’s area of expertise is the technical domain, rather than machine translation, it is important that the knowledge acquisition process be both easy to learn and easy to perform.

- **It must leverage all knowledge assets.** Since the size of the vocabulary for a technical domain can easily exceed 50,000 terms, it is important to make maximal use of any available resources. This includes any existing translated documents from the domain, on-line dictionaries if available, and even the target lexicon itself as it is under development.

- **It must be time-effective.** There are two main lexical knowledge sources that
must be developed for each target language. These are the syntactic lexicon — what we have been calling and will continue to call “the lexicon” — which specifies a target language syntactic structure for each target language phrase, and the semantic mapping rules, which determine both which syntactic lexicon entries to use to realize the concepts within a given interlingua representation and how they are to be put together. Because of the relationship between these two knowledge sources (i.e., that the mapping rules need to be able to refer to the lexicon entries), they cannot be developed in parallel. The development cycle for these two knowledge sources is shown in Figure 2. Given the need to get past the “hump” of the syntactic lexicon development before any real work can begin on mapping rule development, the development time for the entire system depends heavily on the speed of the lexicon acquisition task. It can be easily seen (Figure 3) that shortening this stage shortens the entire development period.

The problem facing the KANT team, then, was how to implement the lexical knowledge acquisition in order to best meet these goals.
Figure 2: Development Cycle for Lexical Knowledge Sources

Figure 3: Improved Development Cycle for Lexical Knowledge Sources
3 Tooling the Process

In order to implement the lexical knowledge acquisition process according to the goals laid out in the previous section, we have broken the process into three steps and built an appropriate tool for each step. The steps are:

1. Get what you can for free. Extract whatever translations may already exist from previously translated documents.

2. Get the rest in way that is easiest on the domain expert. Provide a comfortable environment for manual knowledge acquisition.

3. Convert and clean up. Cast full syntactic lexicon entries from the acquired knowledge and clean up any remaining rough edges.

In the remainder of this section, we describe the tools that implement each of these steps.

3.1 Corpus Aligner and Bilingual Browser

Since one of our goals was to leverage all knowledge assets, we developed a tool to align on-line document pairs and to facilitate extraction of translations from them.

A corpus of source/target document pairs is collected and automatically aligned using extra-linguistic information [Lonsdale, 1994]. Such information, for example diagram references, list itemizations, measurements, numbers, and part-name
alphanumerics, does not change from source to target documents and therefore serves a useful purpose as alignment match tokens. When a given source term is entered by the user of the system, all instances of that term in the source corpus are retrieved, and each translated instance is retrieved via the alignment process. Finally, the paired occurrences are displayed side-by-side in a special browser developed in Common Lisp using the Motif widget library (Figure 4). Then, the user may determine source/target pairs and easily record them in a translations database. This process is very easy and can be performed by anyone competent in both the source and target languages, since the context and parallel source text should make the meaning of any unknown terminology clear.

The magnitude and quality of the results from this stage of acquisition are naturally limited by the size and availability and vocabulary coverage of source/target document pairs. If no such pairs are available, then this stage is skipped entirely. We have found, however, that with even a few documents, we have been able to extract translations for 10%-15% of the technical phrases.

3.2 Vocabulary Translation Editor

The output of the Bilingual Browser serves as an initial translation database, but it is likely (unless a very large corpus of source/target document pairs exist) that the majority of the technical terms do not yet have translations. This preliminary
Bilingual KWTC Browser

Figure 4: The Bilingual Browser
database is expanded by the domain expert using the Vocabulary Translation Editor (VTE).

VTE is a specially designed editing tool developed in Common Lisp using the Motif widget library to create an easy-to-use graphical user interface. For each term to be translated, the translator is shown (Figure 50 a definition and usage examples from the source language lexicon (top left), a Key-Word-In-Context (KWIC) browse of how the source term is used (bottom), and a list of any existing translations (top right). The length of context and number of examples shown in the KWIC browse is adjustable.

All translation-specific controls are placed in the top right portion of the window. New translations may be added or old ones edited. Multiple translations may be entered for a given source term to allow for the lexical fan-out that is common when a target language makes finer-grained distinctions than does the source language. Each translation may be flagged as accepted, rejected, or tentative. Rejected translations are not removed from the database, but remain as a reminder to the domain expert. Each translation may also have a comment attached to it. This is particularly useful for notes regarding usage and idiomatic expressions.

One feature of VTE that have proven very useful is the drafting mechanism, which allows the domain expert to leverage partial translations out of database itself. When the draft button is clicked, VTE looks for source terms that are wordwise
transmission lubrication circuit

b. oil pressure to the transmission lubrication circuit is good. Procedure: 1

- Go to the transmission lubrication circuit and from oil thrown area.
- For gear comes from the transmission lubrication circuit.
- The oil comes through the oil pump and magnet.

a. Oil pressure to the transmission lubrication circuit is good. B. Oil pressure to the transmission lubrication circuit is good during retarder.

Too much leakage in the transmission lubrication circuit (bad seals, etc.). *
substrings of the current term. For example, "control", "valve", "group", "control valve", and "valve group" are all wordwise substrings of "control valve group" that could also be terms in their own right. VTE looks in the database to determine which, if any, of these sub-terms have translations. If translations are found, the one corresponding to the right-most longest substring (a minor Anglocentrism that should be parameterized) is chosen as a "draft translation" for the current term. This is then displayed in the editing field so that the domain expert may make whatever changes are necessary to get a fully correct translation. In order to maximize the likelihood that a partial translation will be available, terms are kept sorted alphabetically by right-most words. That is, "battery" would also be presented for translation before "lamp battery" or "car battery" and "car battery" would be presented before "main car battery".

All transactions are logged to allow for easy identification of changed items. This is important for two reasons. First, it allows for the results to be pipelined into the tool that builds the actual syntactic lexicon entries. This prevents VTE from becoming a knowledge acquisition bottleneck. Second, VTE is also used when further clarification is needed from the domain expert. When this happens, the domain expert is asked to review and comment on certain translations he or she made in response to questions the knowledge engineer has. The knowledge engineer must be able to easily extract any changes — new comments, new translations, etc. —
that the domain expert makes during such a task.

3.3 Syntactic Structure Builder

The database constructed by the Bilingual Browser and VTE only specifies correspondences between source language terms and target languages terms. As mentioned above, the syntactic lexicon must specify full syntactic structures for each target language term, so the contents of the database by themselves are clearly inadequate. To facilitate the construction of full entries from the raw strings, we have developed a special editing mode for Lucid Emacs called the Syntactic Structure Builder. The construction code uses a simple grammar (less than a dozen rules), lists of closed-class items (determiners, prepositions, etc.), and some default assumptions regarding word order to determine the most probable syntactic structure for each phrase. When different structures are possible, such as when more than one prepositional phrase is present, the user is prompted to confirm the default. If the user rejects the default structure, then the next possible structure is shown. In most cases, the decision making involves only preposition phrase attachment, but occasionally more interesting structures such as relation clauses arise and these too are handled (Figure 6). Thus, the process of constructing syntactic entries is reduced to a few keystrokes for each entry. The process is, in fact, easy and quick enough that to correct mistakes, it is faster to rerun the code to generate
the entries than to edit the entries by hand.

---

no hidrostático’
((root "de"))
(lex "nando"
 (modifier (lex "hidrostático"
 (modifier (lex "no")))))

["superficie que no hace contacto"
 (lex 'superficie"
 (rel-clause ((rel-prof ((root "que")))
 (lex 'hace
 (modifier (lex 'no)))]
 ["noncurrent water lines group" "grupo de tuberías de agua antiguas"
 ["rope bar installation" "instalación de la barra delantera"
 ["rope housing" "caja de la nariz"]
 ["rope ball valve" "válvula de bola de tobera"
 ["rope bottom face" "cara inferior de la tobera"
 ["rope cap installation" "instalación de tapa de tobera"
 ["rope clamp bolt" "perno de abrazadera de la tobera"

---

<table>
<thead>
<tr>
<th>Snace</th>
<th>partab</th>
<th>LISP</th>
<th>---</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>12%</td>
</tr>
</tbody>
</table>

Is the complement of the rel clause a (n)oun or (a)djective phrase ("a" or "n")?

Figure 6: The Syntactic Structure Builder

The entries produced by the code do require one additional step of processing in order to neutralize any morphological inflections that were present in the input string, but this step is also automated. By using the morphology rules for the target language under construction and a tagged word list for the language (e.g. from an on-line dictionary), a table can be created to map inflected forms back to their base forms. Applying this table to the structures is a simple matter of recursively traversing the syntactic structures.

---

1The development of these rules is beyond the scope of this paper, but they are guaranteed to be present by this stage in the knowledge acquisition process. The syntactic structure builder needs to know the features and design of the syntactic structures before it can build them. Because of this requirement, the development of the generation grammar and morphology rules must be largely completed before the syntactic structures can be created. Thus, it is quite safe to assume that the morphology rules are present by this time.
4 Benefits from Tools

These tools have allowed us to develop a lexicon acquisition process that fits our design criteria:

- *It must be easy to use.* The Bilingual Browser and VTE both make use of graphical user interfaces both to reduce the training time and to make the acquisition process itself more intuitive. The Syntactic Structure Builder prompts the user whenever a decision is necessary and requires a minimal number of key strokes for each entry.

- *It must leverage all knowledge assets.* The Bilingual Browser allows for extraction of any translations available in on-line corpora. VTE presents as much information as it can about a given term to the domain expert and allows for the bootstrapping of “draft” translations from the very database that is being developed. The morphological correction in the Syntactic Structure Builder uses both static knowledge sources such as dictionaries, and the custom knowledge built into the morphology rules.

- *It must be time-effective.* All three of these tools have proven to enable very rapid lexicon development:

  - Bilingual Browser. Part-time students competent in both source and target languages are able to extract translations at rates varying from
several dozen to several hundred terms per hour. Higher rates were primarily encountered when the vocabulary was very technical (hence easily identifiable from context) and of low frequency (so that alignment and display proceeded quickly).

- **VTE.** Using a prior version of VTE that did not include an integrated KWIC browser, domain experts were able to supply between 500 and 1000 translations/day. The automatic browsing now built into VTE has removed a step (browsing in a separate tool) from the translation process, so the rate should be even high. Initial results have been excellent.

- **Syntactic Structure Builder.** Part-time students competent in the target language have been able to generate structures for 500 to 1000 translations/hour, easily keeping pace with the domain experts with only a small time commitment.

These tools have allowed us to develop target language lexica in a faster, more controlled, more maintainable fashion. **<SOMETHING ABOUT SPEED HERE>**

In terms of maintainability, our biggest success was the Syntactic Structure Builder. Prior to its development, we had used a full phrase structure parsing grammar to determine directly the structure of the phrases. While a competent user of this system could perform at rates similar to that using the Syntactic Structure Builder, extending the grammar was complicated and there was no provision for
default behaviors; each possible structure was presented in its entirety and the
used had to both find the differences (not always easy in 20 to 30 line structures)
and determine which was correct by hand. With the Syntactic Structure Builder,
the grammar rules can be changed and extended easily and the determination
process is greatly simplified. Furthermore, the Syntactic Structure Builder handles
morphology in a separate pass, whereas the full parsing grammar was forced to
include morphological parsing in the structure decision itself.

There is a additional benefit for both maintainability and space-efficiency to
the Syntactic Structure Builder. Our lexicon formalism includes a mechanism for
specifying entries based on simpler entries. Thus, the entry for the Spanish phrase
"válvula de agua" could be based on the entries for "válvula" and "agua." When
an entry is specified this way, the full syntactic structure is not expanded out until
it is needed at run-time. The SSB creates this sort of mosaic entry automatically.
This means that the structures are minimal and that any special information about
single words (e.g. gender, conjugation, etc.) is stored only once in the root entry for
that word. From a maintainability standpoint, this is optimal, since any changes
that need to be made to a word’s features need only be made once. From a space-
efficiency standpoint, this is also advantageous, since only the minimal amount of
information is stored for each entry. This seemingly small amount of savings can
become important when the number of lexicon entries grows to more than 50,000.
5 Conclusion

The task of developing a large-scale target language lexicon of technical phrases for machine translation is ********. In this paper, we described several tools used by the KANT team for developing this sort of lexicon and how the use of these tools greatly reduced the amount of effort necessary.

blah.

6 References


