Tooling the Lexicon Acquisition Process for Large-Scale KBMT

John R. R. Leavitt — *jrrl@cmu.edu*
Deryle W. Lonsdale — *lonz@cmu.edu*
Kevin Keck — *kk30@andrew.cmu.edu*
Eric H. Nyberg — *ehn@cmu.edu*

Center for Machine Translation
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Large-scale lexical knowledge acquisition is one of the most time critical steps in developing a knowledge-based machine translation system. In particular, developing the syntactic lexicon for the target language can be an unwieldy task, as on-line knowledge assets are likely to be more scarce than for the source language. This paper addresses this problem within the KANT machine translation system and describes how we structure the KA process to address this problem. This was done by first determining the nature of the desired process and then developing tools to implement that process. The tools themselves and the ways in which the helped us to realize out design goals are described. We conclude that, while the problem of lexical acquisition can be formidable, it can be overcome with proper foresight and tool design.

1 Introduction

In developing knowledge-based systems, the most time-critical step is almost invariably knowledge acquisition. This is especially true for knowledge-based natural language systems, in which the lexicon must grow in parallel with the coverage of the system [2]. In this paper, we discuss the specific case of target language lexical knowledge acquisition for machine translation in the KANT machine translation system. In this case, the knowledge to be acquired is syntactic structures for target language words and phrases. We describe the problem this presents, 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 [6], 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 from the source sentence. The main advantage of interlingua systems is that 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).

KANT is designed to perform translation within well-specified technical domains, such as heavy equipment operation and maintenance manuals. Within such as a domain, it is possible not only to restrict the meanings of ambiguous words (e.g. "washer" being taken to mean a small round gasket, rather than something or someone that washes), 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 [1]. Similarly, having very precise concepts in the interlingua facilitates the generation of the correct target language expression for that phrase [3].

To achieve these benefits, however, there is an associated cost, which in this case is the burden of constructing the source and target language lexica on a large scale. The construction of the source language lexicon is largely automated from analysis of existing corpora [5], but the construction of the generation lexicon requires more effort, both because its format includes more explicit syntactic information (i.e. a full f-structure for each target language term), and because online corpora are often not available in the target language. It is the acquisition of the target language lexicon taht we discuss in this paper.

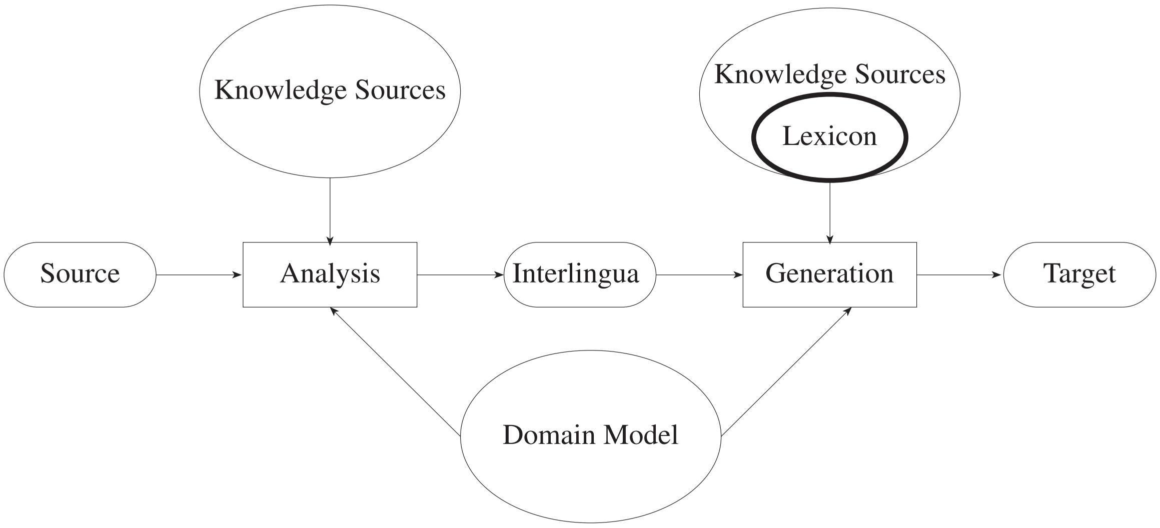

Figure 1: The KANT Interlingual MT Architecture

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

- *It must be intuitive.* 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 a bootstrap during 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 — henceforth "the lexicon" — which specifies a target language syntactic structure for each target language phrase, and the semantic mapping rules, which determine both the syntactic lexicon entries to use to realize the interlingua concepts and how they are to be conjoined. Because the mapping rules need to be able to refer to lexicon entries, mapping rules cannot be developed until a certain portion of the lexicon is in place. By expediting the acquisition of the syntactic lexicon, the critical portion is established sooner. This means that the development of the majority of the mapping rules can begin earlier, and the entire project development period can be reduced.

The problem facing the KANT team, then, was how to implement the lexical knowledge acquisition in order to best meet these goals.

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 (almost) free.* Extract whatever terminology translations may already exist from previously translated documents.

2. *Get the rest in a way that is easiest on the domain expert.* Provide a comfortable environment for manual entering of translations for the remaining terms.

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

Note that the last step is critical, as a set of source-target translation pairs is not a sufficient lexicon for an interlingua system. The underlying syntactic structure of the target language phrases must be extracted and is associated only with the interlingua concept with which the source phrase was associated. If a different source language were used (i.e. we started analyzing Japanese documents), the interlingua concepts and the target language lexicon entries would remain the same.

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 terminology translations from them.

A corpus of source/target document pairs is collected and automatically aligned using extra-linguistic information [4], such as diagram references, list itemizations, measurements, numbers, and part-name alphanumerics. This information does not change during translation and therefore serves well as anchoring information for alignment. 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 2). Then, the user may determine source/target pairs and easily record them in a terminology translations database. This process can be performed by anyone competent in both the source and target languages, since the context and parallel source text clarify the meaning of any unknown terminology.

The magnitude and quality of the results from this stage of acquisition are naturally limited by the size, 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 terminology translations for 10%-15% of the technical phrases.

3.2 Vocabulary Translation Editor

The output of the Bilingual Browser serves as an initial terminology translation database, but it is likely (unless a large corpus of source/target document-pairs exist) that the majority of the technical terms do not yet have translations. This preliminary 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 3) 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 for the source term (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 terminology translations may be added or old ones edited. Multiple translations may be entered for a given source term to allow for the fact that the target language make make finer-grained distinctions than 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 has proven very useful is the drafting mechanism, which allows the domain expert to leverage partial terminology translations from the database itself. When the draft button is clicked, VTE looks for source terms that are wordwise 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 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 be presented to be translated 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.

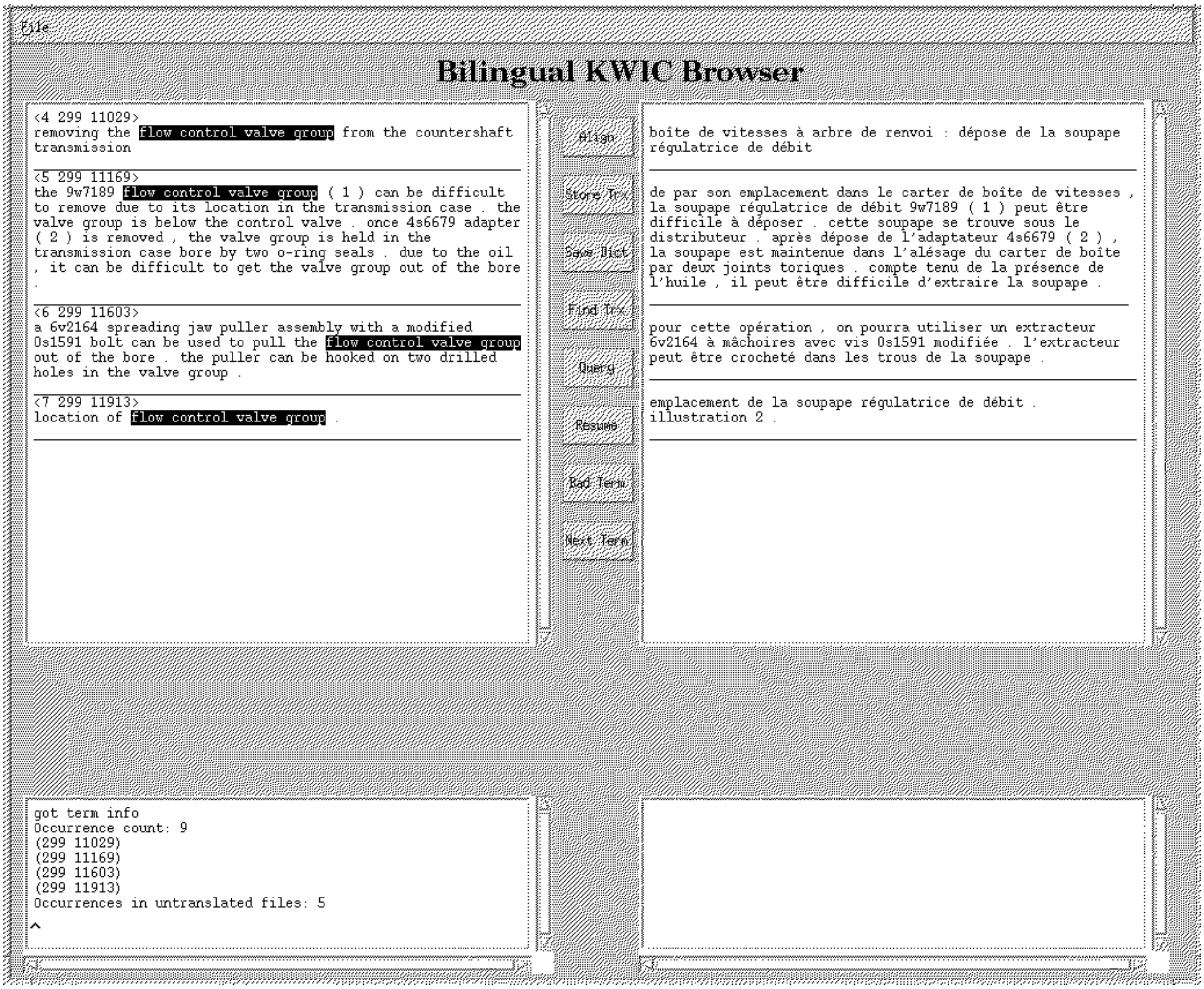

Figure 2: The Bilingual Browser

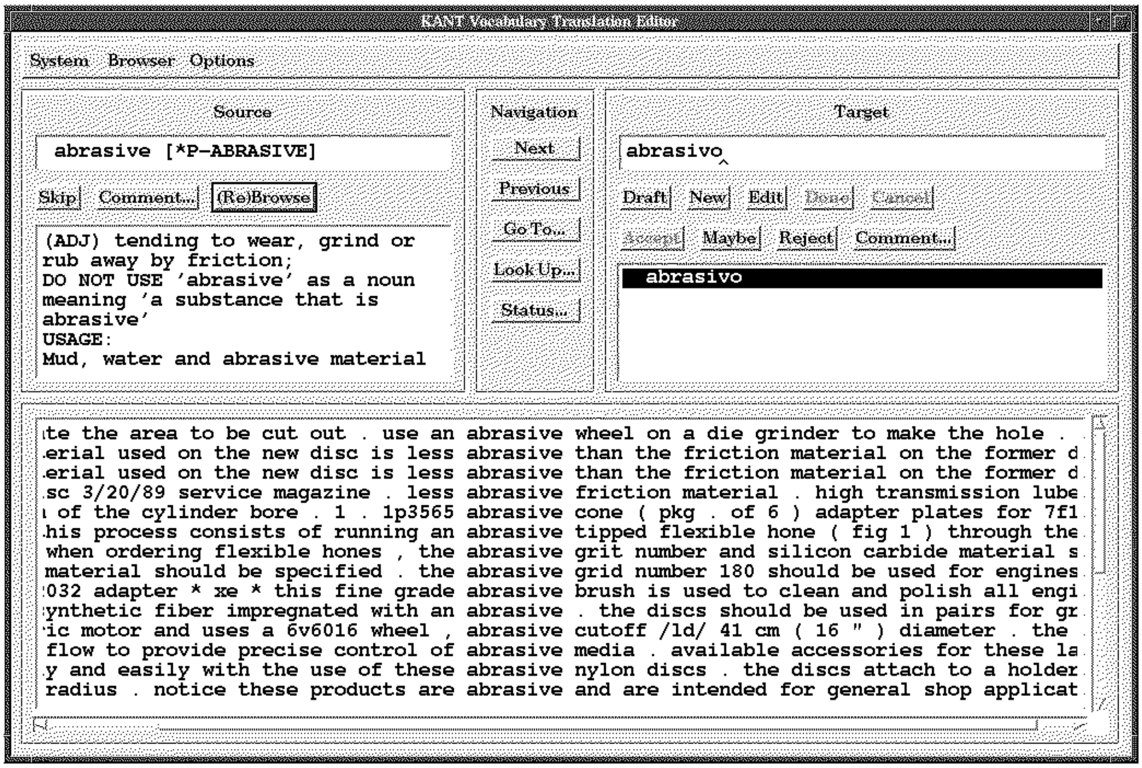

Figure 3: The Vocabulary Translation Editor (VTE)

3.3 Syntactic Structure Builder

The database constructed by the Bilingual Browser and VTE only specifies correspondences between source language terms and target language 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 (SSB). 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 target language term. 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 relative clauses arise and these too are handled (Figure 4). Thus, the process of constructing syntactic entries is reduced to a few keystrokes for each entry. In fact, the process is efficient enough that it is easier to rebuild syntactic entries with SSB that to edit them by hand, when changes are necessary.

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[1]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.

4 Benefits from Tools

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

- *It must be intuitive.* The Bilingual Browser and VTE both make use of graphical user interfaces both to reduce the training time and to make the acquisition

[1] The 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, by virtue of the development schedule for and dependencies between the other knowledge source development tasks.

```
                 no hidrostático"

                ((root "de")))
                 (:lex "mando"
                       (modifier (:lex "hidrostático"
                                       (modifier (:lex "no")))))))))


("superficie que no hace contacto"
 (:lex "superficie"
       (rel-clause ((rel-pron ((root "que")))
                    (:lex "hace"
                          (modifier (:lex "no"))
("noncurrent water lines group" "grupo de tuberías de agua antiguas")
("nose bar installation" "instalación de la barra delantera")
("nose housing" "caja de la nariz ")
("nozzle ball valve" "válvula de bola de tobera")
("nozzle bottom face" "cara inferior de la tobera")
("nozzle cap installation" "instalación de tapa de tobera")
("nozzle clamp bolt" "perno de abrazadera de la tobera")
--**-Emacs: partab              (Lisp)----12%------------------------------------
Is the complement of the rel clause a (n)oun or (a)djective phrase ("a" or "n")?-
```

Figure 4: The Syntactic Structure Builder (SSB)

process itself more intuitive. SSB 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 terminology 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 SSB 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. 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 translate between 500 and 1000 terms per 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 higher. The current version of VTE has been deployed with domain experts in Canada (for French), Germany and Austria (for German), and in the US (for Spanish, Portuguese, Italian). A version that supports the cyrillic alphabet (for Russian) is expected to be deployed later this year. Performance results indicate that its performance is as good if not better than the previous version.

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

These tools have also allowed us to develop target language lexica in a more maintainable and space-efficient fashion.

In terms of maintainability, our biggest success was SSB. 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 SSB, 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 SSB, the grammar rules can be changed and extended easily and the determination process is greatly simplified. Furthermore, SSB handles morphology in a separate pass, whereas the full parsing grammar was forced to include morphological parsing in the structure decision itself.

There is an additional benefit for both maintainability and space-efficiency provided by SSB. 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 conglomerate 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 savings becomes significant with lexicons containing more that 50,000 entries. On this scale, a megabyte of space is saved overall for every 20 bytes saved per entry.

5 Conclusion

The task of acquiring a large-scale target language lexicon for machine translation can be daunting. We have shown that an efficient process for this acquisition may be developed by first determining the desired features of the process and then building efficient tools to facilitate different steps within the process. These tools have allowed us to make the acquisition process both manageable and effective.

Acknowledgements

We would like to thank Claude Dore, Don Farquharson, and Sarah Law for their invaluable assistance in working with these tools, both to discover problems, and to suggest possible improvements.

References

[1] Baker, K., Franz, A., Jordan, P., Mitamura, T., and Nyberg, E., (1994). Coping with ambiguity in a large-scale machine translation system. In *COLING-94*.

[2] Galinski, C. (1988). Advanced terminology banks supporting Knowledge-Based MT. In Maxwell, D., Schubert, K., and Witkam, A. P. M., editors, *New Directions in Machine Translation*. Foris Publishers.

[3] Leavitt, J., Lonsdale, D., and Franz, A. (1994). A reasoned interlingua for knowledge-based machine translation. In *Canadian Artificial Intelligence Conference*, Banff, Canada.

[4] Lonsdale, D. (1994). Extraction d'un vocabulaire bilingue: outils et méthodes. In Clas, A. and Bouillon, P., editors, *Actes du Colloque Lexicologie, Terminologie et Traduction*. Les Presses de l'Université de Montréal.

[5] Mitamura, T., Nyberg, E., and Carbonell, J. (1993). Automated corpus analysis and the acquisition of large, multi-lingual knowledge bases for MT. In *5th International Conference on Theoretical and Methodological Issues in Machine Translation*, Kyoto, Japan.

[6] Nyberg, E. and Mitamura, T. (1992). The KANT system: Fast, accurate, high-quality translation in practical domains. In *COLING-92*.