The KANT Machine Translation System: From R&D to Initial Deployment[1]

Eric Nyberg, Teruko Mitamura, Jaime Carbonell

Center for Machine Translation
Carnegie Mellon University
Pittsburgh, PA 15213
ehn@cs.cmu.edu

http://www.lti.cs.cmu.edu/Research/Kant

1. Introduction

The KANT system (Knowledge-based, Accurate Natural-language Translation) is a set of software tools for automatic and interactive analysis of source text and generation of target text (Mitamura, et al., 1991). It has been primarily targeted towards the translation of technical text in controlled subdomains (Mitamura and Nyberg, 1995). Initially an outgrowth of research ideas following the completion of the KBMT-89 system at CMU (Goodman and Nirenburg, 1990), KANT has been scaled up for multilingual document production in an industrial setting. In this paper, we discuss the facets of KANT which are of potential interest to the LISA audience: its technical foundation, its intended domain(s) of application, its current performance, and our future plans regarding its application and commercialization.

2. The KANT Architecture

KANT is a knowledge-based machine translation (KBMT) system. It uses explicitly coded lexicons, grammars, and semantic rules to perform translation. KANT is also an interlingual system, making use of an explicit intermediate representation which acts as a "pivot" between the source and target languages.

There are three main advantages of this architectural approach:

- **Increased accuracy of translation.**
  By allowing the creation of lexicons and grammars that can be as simple or as complex as the application requires, KANT supports a high degree of accuracy in both the source analysis and target generation phases.

- **Efficient support of multiple target languages.**
  Through use of an intermediate interlingua representation, KANT's architecture allows straightforward extension to new target languages – in fact, the interlingua produced from a single source text may be used to generate several target texts.

- **Separation of code and knowledge bases.**
  All of the software in KANT (analysis and generation modules) is independent of the language pair being translated. When a new output language or language pair is desired, all that is required is "plugging in" the new knowledge sources; the system code remains constant. Unlike transfer-based systems, the addition of a new target language does not require any replication or redesign in the analysis/transfer phase of translation; rather, new grammars and lexicons are added for each new target language, which use the intermediate interlingua representation as their input.

[1] Paper presented at the LISA Workshop on Integrating Advanced Translation Technology, Hyatt Regency Crystal City, Washington D.C., June 3-4, 1997.

3. KANT Application Domains

KANT is designed for dissemination of texts authored in a domain sublanguage to several target languages. This implies that KANT is a natural fit for large customers with the following translations needs:

- There is a need for consistency in source grammar and terminology, such that the use of a domain sublanguage is justified beyond its use for accurate MT;

- The texts to be translated focus on an area of technical information (such as computers or motor vehicles), such that it is feasible to design a domain-specific approach to writing (terminology and grammar);

- A high degree of output accuracy is required, such that the benefits of knowledge-based translation are cost-justified;

- The volume of text to be translated (and the implied cost of manual translation) is large enough that the effort to develop an automatic machine translation system is a worthwhile investment;

- Input texts are created by a specific organization, rather than being an amalgam of texts authored by many suppliers, such that it is feasible to implement a domain sublanguage standard for authoring;

- The input texts should be translated to more than one target language, such that the benefits of KANT's modular architecture and interlingua approach are realized.

Companies which tend to fit this profile include large manufacturers of complex machinery with world-wide sales, with translations volumes over ten thousand pages per year.

4. KANT and the Translations Workflow

KANT can play two roles in the document production workflow:

- **Grammar checking during text creation**.
  The KANT Analyzer can be embedded inside the author's editing interface. During authoring, KANT can provide source terminology and grammar checking functionality, which helps the author to write texts which conform to a given sublanguage specification and hence produce more accurate machine translation output. During grammar checking, KANT has the ability to recognize unresolved ambiguity in the source text, which might lead to inaccurate translation if left unresolved. KANT has an API (application program interface) which allows it to notify editing tools of the location and type of ambiguity, so that the author may be queried to resolve the ambiguity. The grammar checking API allows the KANT Analyzer to run as a separate process, and communicate with any tool(s) which support the API.

- **Full batch-mode translation**.
  Once source documents are authored, the full KANT system (Analyzer and Generators) is used in an off-line batch capacity to translate source documents into the desired target languages. In batch mode, KANT interacts with other tools primarily through a file I/O specification. The software supports SGML input and output, and uses an intermediate interlingua file format to store the results of Analysis. The interlingua file is the input to the generation module, which typically produces SGML-marked output text.

5. Accuracy vs. Complexity

Our initial goal was to build a system which could create a single, unambiguous and correct interlingua for each input sentence; this, along with a comprehensive set of grammar rules and lexical entries for the target language, would allow KANT to produce output so accurate that little or no post-editing would be required.

In reality, we have found that large-scale application domains of practical interest are rarely simple enough for this goal to be fully achieved. In practice, it is difficult if not impossible to limit the meanings of source words and sentences such that there is only one interpretation possible. For example, in the heavy equipment domain there are several meanings for the word *valve*, any of which might be appropriate in a given context. If the system cannot resolve this ambiguity by interacting with the author, or by accessing internally-stored knowledge about the domain, the quality of the output translation may suffer.

A more realistic goal is to limit the effects of complexity through careful definition of the domain sublanguage. When it is possible to guide the writer in using the most straightforward prose possible, and to resolve the important ambiguities in an interactive human-machine dialog, then any residual ambiguity or complexity which results in loss of accuracy can be dealt with in a minimal post-editing phase. It has been our experience that the amount of effort required to produce a perfect knowledge-based translation in a large practical domain is probably not justified. Generating mostly accurate translations requiring a small amount of postediting is generally more cost-effective with the right tools and training.

6. Explicit Support for Document Markup

KANT is designed to take advantage of document markup, such as SGML or HTML. Rather than stripping out this important information, and trying to re-insert the tags after translation, KANT analyzes markup directly as an integral part of the input text, and represents it in the interlingua, so that it can be properly generated in the correct position by the target generator. Because information about the markup tags is represented separately from the source grammar, it is possible to update the markup processing easily when adjustments are made in the markup language. Another advantage of KANT's sophisticated markup support is that it allows for straightfoward integration of MT into the document production workflow.

The use of SGML is also advantageous for high-accuracy MT, since tags can be used to identify various open-class items (e.g., serial numbers) which would otherwise be ambiguous or difficult to process.

7. Initial Deployment

The KANT system has been applied to the domain of heavy-equipment documentation – documents describing the operation, maintenance, and assembly of vehicles (e.g., tractors, backhoes, etc.) and their subcomponents (e.g., engines, hydraulics, etc.). We have developed a domain sublanguage with a vocabulary of about 65,000 words and technical phrases. At document creation time, the KANT Analyzer is integrated with Carnegie Group's ClearCheck tool and a commercial SGML authoring tool used by the customer's technical writers. At document translation time, the KANT Analyzer and Generator run in batch mode on a translation server machine, translating individual sections of documents as they are authored.

The application has been in operation for English-to-French translation for over a year; initial deployment of English-to-Spanish is now underway. The system is being used in an industrial setting, both for grammar checking and batch translation of documents. The translations volume is measured in hundreds of thousands of pages per year.

8. Why was KANT Chosen?

Our industrial customer had several reasons for choosing KANT:

- The degree of accuracy required by the customer could not be obtained with existing translations tools, which do not take advantage of domain sublanguage specialization;

- KANT supports the use of a controlled language for consistency in source authoring and reduced complexity of translation;

- The KANT system provides a flexible API for integration with other support tools (checkers, editors, etc.);

- Explicit support for SGML markup was desired, which was not available with any other system at the time;

- A high (costly) volume of technical translation was projected for several target languages of interest.

9. How Well Does KANT Perform?

We conducted an internal postediting productivity study, comparing the time required to manually translate SGML text (English to French) with the time required to post-edit the KANT translation of the same text. Postediting was carried out according to a set of "minimal postediting" guidelines, which are designed to limit the scope of postediting to just those corrections required to produce understandable, accurate output (thus limiting purely stylistic rewrites). The postediting was done by a French native speaker with no special expertise in the application domain, but with a good understanding of the editing interface and the postediting guidelines. The results were quite encouraging – texts which took close to an hour to translate manually took 10 to 15 minutes to post-edit when KANT was used, implying possible productivity gains of 4 or 5 to 1. The minimally-postedited output was reviewed by the translations end-user (a product dealer in France), who verified that the postedited MT output would be highly useful if made available on a large scale.

Following our initial evaluation, the customer performed a usability study to test the English-to-French system in a similar manner, comparing manual translation times to postediting times for a range of experienced to inexperienced translator/posteditors. The customer's initial results were mixed, and initial productivity gains of 2 to 1 were reported on average. It is clear that the two main areas of difficulty for human posteditors are a) learning the shortcuts for effective use of the editing tool interface; and b) staying within the minimal postediting guidelines. It seems that building a productive MT tool is only the beginning of the integration process, for human translators who are accustomed to complete creative control over the output have a hard time limiting themselves to minimal touch-up of computer text. The customer believes there is opportunity to improve productivity to a much greater degree, once the translators' perspective shifts to productive postediting and effective use of the postediting environment.

10. Current Challenges

Our initial deployment of the KANT system for grammar checking and translation has identified a set of challenges to be addressed in ongoing work:

- **Evolving Requirements**.
  Our customer has a need for constantly evolving technical terminology, so the system's requirements change from release to release. To a lesser extent, changes are also made in the

SGML markup language and the domain sublanguage grammar. All of these changes must be supported in the KANT Analyzer, and in the Generator modules for each target language.

- **Maintenance Cost and Turnaround Time**.
  Because of the constant change in system requirements, it is crucial that the system support streamlined (low-cost) maintenance, with rapid turnaround time. A significant effort has been expended in setting up a change request database and a corresponding issue resolution procedure in support of effective maintenance and delivery of systems.

- **Workflow Integration**.
  The deployment of an entirely new system for authoring and translating across a large technical information department has given rise to technical integration issues as well as human process issues. The effect on the author and translator is that their productivity may decline when the new tools are introduced, due to system slowness and process inefficiencies.

- **What Level of Control is Achievable?**
  Asking authors and translators to work with a domain sublanguage can be like asking them to learn a whole new dialect of their language, and a substantial learning curve is to be expected. Once the technical goals of the MT system are achieved, the issues of author and translator acceptance of the system are still to be resolved. In a large-scale document production environment, productivity, consistency and accuracy are the goals, rather than creativity, but the cultural change can be difficult – especially for translators who have a literary perspective on their work.

- **Increasing System Complexity**.
  Although many of the training and cultural issues during deployment can be resolved independently of the translations system, there are situations where it is necessary to increase the sophistication of the software, so that some controls can be relaxed. Our initial goal was to provide a very tightly controlled domain sublanguage, with no lexical ambiguity and a simplified grammar. To achieve acceptable quality and productivity, we have had to relax some of those requirements along the way, thus adding to the complexity of the task that the MT software must perform. We believe to have found the right tradeoff between freedom/expressibility on the one hand, and consistency/control on the other.

11. Plans for the Future

Based on our initial positive experience with English-to-French deployment, we are now working on deployment of KANT for English-to-Spanish in the heavy equipment domain. We feel strongly that there is great potential for continued commercialization of KANT in new domains and languages. In order to take advantage of all we have learned during the initial deployment, the system has been redesigned for an object-oriented implementation, which will improve maintainability, integration, and run-time efficiency. Specifically, the reimplementation targets the following goals:

- **Better Tools for Development and Update**.
  An effective set of developer tools is a crucial aspect of streamlined (low-cost) development and update, as well as rapid turn-around time. Good tools also facilitate smooth transfer of system maintenance to non-developer personnel at the customer site.

- **Better Run-time Performance**.
  By reimplementing the current software (originally written in Lisp) in C++, we expect significant gains in run-time performance of the system. This is especially critical for support of fast grammar checking, because the author's working environment typically integrates several

complex tools for text, graphics, etc., and the grammar checking application must be as small and fast as possible.

- **Better Integration with Production Environments**.
  In addition to porting the system software to C++, we are reimplementing our terminology databases (flat files) as Oracle/Forms applications. This will improve the overall completeness and consistency of the terminology data, while providing a streamlined interface for terminology updates.

- **Generalized Knowledge Bases**.
  While the KANT software is reimplemented, we are also taking steps to generalize our existing knowledge bases, so that they can be used as a basis for new applications in any domain of interest. We're working on core knowledge bases for French, Spanish, German, Italian, Portuguese, and Chinese at present.

12. Summary

We have described the initial deployment of the KANT system for English-to-French translation, in a large-scale application for heavy-equipment documentation. The KANT system is specifically designed for translation of source texts written in a controlled sublanguage into several target languages. By making use of a knowledge-based, interlingual architecture, KANT supports a separation of code and knowledge, as well as language-independence; these are two crucial characteristics for efficient development of multi-lingual systems. Through use of sublanguage constraints, and explicit grammars and lexicons for each language, KANT achieves a high level of accuracy. SGML markup is processed as part of the translation itself, making KANT a highly suitable translations engine for applications which rely heavily on SGML or HTML markup. In addition, the KANT API allows the system to be used as a grammar checker in parallel with existing editing tools.

Based on our experience during the initial deployment of the system, we are in the process of reimplementing the KANT system in an object-oriented framework (C++). We are also reimplementing our terminology support tools as Oracle/Forms applications, for better integration with production environments. Future applications of the system will take advantage of these new software tools, which will support more efficient system development/update and better run-time performance.

13. Future Applications

In parallel with ongoing improvements to the KANT software, we are also striving to integrate KANT with the Multi-Engine Machine Translation (MEMT) system (Frederking et al., 1994). The MEMT framework allows us to combine KBMT systems (such as KANT) with other translations techniques, such as translation memory, example-based machine translation (EBMT), and glossary-based translation. In unrestricted translation domains, a combination of techniques is generally more effective than relying solely on KBMT, which does not work as well when the input is unrestricted and potentially errorful.

We have begun to investigate applications beyond technical documents, such as real-time and off-line translation of broadcast captioning, and bi-directional translation of medical records for on-line information systems. In both of these areas, we have completed initial prototype systems; we are in the process of proposing full-scale implementations, which will use the reimplemented KANT software if and when they are developed.

14. References

Frederking, R. et al., “Integrating Translations from Multiple Sources within the Pangloss Mark III Machine Translation System,” *Proceedings of the First Conference of the Association for Machine Translation in the Americas*, Columbia, MD.

Goodman, K. and S. Nirenburg, eds. (1991). *A Case Study in Knowlege-Based Machine Translation*, San Mateo, CA: Morgan Kaufmann.

Mitamura, T., E. Nyberg and J. Carbonell (1991). “An Efficient Interlingua Translation System for Multi-lngual Document Production,” *Proceedings of the Third Machine Translation Summit*, Washington, D.C.

Mitamura, T. and E. Nyberg (1995). “Controlled English for Knowledge-based MT: Experience with the KANT System,” *Proceedings of TMI-95*, Leuven, Belgium.