A computer Translation System for Authors

Joerg Schuetz

IAI

Saarbruecken

1. TRANSLATION

We have found various definitions for translation in the linguistic literature. In our opinion translation transforms the meaning of a given text into another language. This definition is based on Saussure's statement that a language is a system of terms of form (signifiant) and meaning (signife); and implies that to translate means to change the form of the terms, preserving their meaning.

Given a linguistic term in the source language, the goal of translation then is to determine its meaning and to find a linguistic term in the target language which renders that meaning. Terms may refer to various units such as morpheme, word, phrase, clause, sentence, paragraph, chapter, and text.

The application of a computer for the translation process involves the necessity of formulating the rules explicitly. Rules can only work when well defined knowledge is accessible. Thus translation is based on both rules and knowledge. Human translators can (and often do) use the language rules intuitively.
A computer manipulates symbols. If translation means replacing the form of terms, it comes down to the manipulation of symbols. However, manipulating forms is not enough. The crucial goal is to preserve the meaning of the text to be translated. The computer is well suited to manipulate the form of linguistic terms, but there is no simple link to the meaning side.

Meaning analysis need not go deeper than to establish correspondences between forms to fulfil that function. For a computer system, there is also no reason to imitate the way a human translator works. A computer system relies much more than a human does on formal rules rather than on rules of meaning.

2. DIALOGUE

The problems of syntactic ambiguity and word choice are resolved by means of semantic and/or pragmatic knowledge about meaning inside and outside the language system. If all these efforts fail to lead to a safe conclusion, there must be a way out. It is generally agreed that fully automatic high quality translation is ultimately unattainable, though it remains the ambition of many researchers (Slocum, 1985).

A translating computer system needs human help. In the system we have designed, this help is provided in the form of a sytem-initiated interactive dialogue with the user who enters a personal text. And essentially, this is the main principle of our system: we make use of the author's knowledge about his/her own text to navigate the system's processing (this implies the title) of disambiguation.
The advantage compared to postediting (the mechanism used by all MT-systems) is that our user (author) does not need to know any other language but the language of his/her text. The user need not be a translator or even have the slightest knowledge about the language into which the text will be translated, nor about the internal representation, which is invisible to the user. (User knowledge about the target language could in fact help the system in finding the right word in the transfer phase; we think, for example, of technical terminology.)

Our system is based on the traditional divisions of computer

translation systems: analysis, transfer and synthesis. As a first approach we designed a system for translating German into English.
Before the dialogue can be started the processing has passed from the monolingual analysis stage into the bilingual transfer stage. The rule system controlling the dialogue must therefore be bilingual as well. Our decision, to wait until transfer, is based on experiences we have had with systems which started the dialogue with the user earlier, e.g. after syntactic and/or relational analysis stages. For German this approach seemed promising, especially for case ambiguities, a consequence of German free word order. Dialogue sequences took place after each step of the analysis process if ambiguities of a predefined type occurred.
For not designing an entirely new translation system, our main goal is to elaborate the computational aspects of dialogues for computer translation, we used the LFG approach (c-structure and f-structure dialogues) for analysis on the one hand (Bresan/Kaplan,1983), and the new EUROTRA approach (ECS-, ERS- and IS-dialogues) for doing our tests with a computer translation system on the other hand (Arnold et al., 1986). As a consequence of a lot of data housekeeping these systems were slow, and not of interest in terms of user-friendly interactive applications; interaction needs a real time scenario.

In the following we give some exemplification of problems arising in German analysis, which are well suited for dialogue disambiguation. We have these case ambiguities:

a) nominative/accusative ambiguity

"Die Recognition-Matrix berechnet das Verfahren."
(Das Verfahren berechnet die Recognition-Matrix.)
*"The recognition matrix computes the process."
(The process computes the recognition matrix.)

b) dative/genitive ambiguity

"Ein weiterer Rechenschritt fuegt die Werte der Tabelle hinzu."
(*An additional computing step adds the values the table to.)
"An additional computing step adds the values to the table."

c) the classification of a genitive nominal complex

"... aktiviert das Modul des Programms."
(*... activates the module the programm.)

Another problem of German is that called "Ausklammerung" (separation):

"Den Ableitungsbaum zu erzeugen zu versuchen war die Absicht des Erstellers des Algorithmus."
(*The derivation tree to generate to try was the intention the designer the algorithm.)

To disambiguate prepositions we need to know their semantic interpretation; moreover discourse specific knowledge,

"Diese Prozedur definiert die Schnittstelle zum Systemkern."
(This procedure defines the interface to the system kernel.),

and/or complex knowledge, i.e. system internal information,

"... wird erreicht durch den Aufruf der Prozedur P."

(... is achieved by calling the procedure P.),

is also useful for disambiguating prepositions.

The function of the dialogue with the user is to resolve syntactic ambiguities in the source text and problems of word choice, if they cannot be resolved by other means within the system. It is thus the task of the user to choose among a number of alternative semantic representations (trees) which differ in words, labels or structure. This choice must be requested without bothering the user with either internal semantics or tree structures. The user will see only questions in plain German.

We have the following rule systems, and in this system the data is drastically reduced:

i) a TREE COMPARATOR which finds out those nodes at which the alternative trees differ;
ii) a CATEGORISOR which decides on the types of the alternatives;
iii) a QUESTION GENERATOR which transfers the internal representation phrases into the source language; this is a sort of retranslation, where appropriate words of the source sentence can and should be used, while the crucial ones are paraphrased;
iv) a PARAPHRASE GENERATOR which formulates the crucial parts of sentences.
v) a DIALOGUE HANDLER which coordinates the dialogue and interacts with the above generators.

The system works on semantic representations of natural language expression with the following steps:

i) locate the differences of the trees (TREE COMPARATOR);
ii) categorisation of the differences (CATEGORISOR);
iii) decision if a user interaction should be activated;
iv) start of the appropriate dialogue form if necessary (DIALOGUE HANDLER).

At the moment we have only three dialogue forms:

i) yes/no dialogue,

e.g. "Is it true that the recognition matrix computes? (y/n)";

ii) menu-driven dialogue,
which means the dialogue consists of dialogue sequences; and

iii) user choice dialogue,

e.g. "Is the word ...
a) human, or
b) animate?".

As of yet, the system is not fully implemented. We have experimented with prototype implementations in Prolog as well as with Comskee (Mueller et al., 1984) implementations.
Future systems should also be based on the proposals by Steiner (Steiner et al., 1986) and Zelinsky (Zelinsky-Wibbelt, 1987) for the semantics of computer translation systems.

Another very interesting subject of such a system is the handling of errors (user and system errors). For the user we can summarize the following:
word order, word choice, spelling and inflexion errors;
whereas for the system:
lexicon errors (no entry or incomplete entry in the lexicon), grammar (rule) errors for analysis, transfer and synthesis.
To recognize these errors and to generate appropriate error diagnostics we have to use other mechanisms and/or information representation.

We end this paper by ennumerating some applications we have in mind for our system:

- small translation offices,
- scientific documentation (e.g. manuals),
- university (scientific articles, abstracts, ...).

3. REFERENCES

Arnold, D., S. Krauwer, L. des Tombes, and N. Varile. 1986.
"The EUROTRA Reference Manual", Version 1.2, Utrecht.

Bresan, J. and R. Kaplan. 1983.
"The Mental Representation of Grammatical Relations", Cambridge, Mass.: MIT-Press.

Gazdar, G., E. Klein, G. K. Pullum, and I. A. Sag. 1985.
"Generalized Phrase Structure Grammar", Basil Blackwell, Oxford.

Mueller, A., T. Kretschmer, J. Messerschmidt, M. Ries, and J. Schuetz. 1984.
"The Programming Language Comskee", LA SFB 100, Heft 10, University of Saarbruecken

Saussure, F. 1916.
"Cours de Linguistic Generale", in M. Bierwisch, Kursbuch 5, 1966, Wagenbach: Berlin.

Slocum, J. 1985.
"A Survey of Machine Translation", in Computational Linguistics.

Steiner, E., U. Eckert, B. Weck, and J. Winter. 1986.
"The Develepment of the Eurotra-D System of Semantic Relations", IAI Working Papers, Saarbruecken.

Zelinsky-Wibbelt, C. 1987.
"Eine semantische Metasprache fuer die Interprtation von Bedeutungen", paper submitted for GWAI 87.