EXPERT SYSTEMS AND OTHER NEW TECHNIQUES IN MT SYSTEMS

Christian BOITET - René GERBER
Groupe d'Etudes pour la Traduction Automatique
BP n° 68
Université de Grenoble
38402 Saint-Martin d'Hères
FRANCE

ABSTRACT

Our MT systems integrate many advanced concepts from the fields of computer science, linguistics, and AI : specialized languages for linguistic programming based on production systems, complete linguistic programming environment, multilevel representations, organization of the lexicons around "lexical units", units of translation of the size of several paragraphs, possibility of using text-driven heuristic strategies.

We are now beginning to integrate new techniques : unified design of an "integrated" lexical data-base containing the lexicon in "natural" and "coded" form, use of the "static grammars" formalism as a specification language, addition of expert systems equipped with "extralinguistic" or "metalinguistic" knowledge, and design of a kind of structural metaeditor (driven by a static grammar) allowing the interactive construction of a document in the same way as syntactic editors are used for developing programs. We end the paper by mentioning some projects for long-term research.

INTRODUCTION

In this paper, we assume some basic knowledge of CAT (Computer Aided Translation) terminology (MT, MAHT, HAMT, etc.). The starting point of our research towards "better" CAT systems is briefly reviewed in I. In II, we present 3 lines of current work : improving current second-generation methodology by incorporating advanced techniques from software engineering, moving toward third-generation systems by incorporating expert systems, and returning to interactive techniques for the creation of a document.

I - IMPORTANT CONCEPTS FROM EXISTING SYSTEMS

For lack of space, we only list our major points, and refer the reader to (3,4,5,6,15) for further details.

1 - Computer science aspects

1) Use of Specialized Languages for Linguistic Programming (SLLP), like ATEF, ROBRA, Q-systems, REZO, etc.

2) Integration in some "user-friendly" environment, controlled by a conversational interface, and managing a specialized data-base composed of what we call "lingware" (grammars, dictionaries, procedures, formats, variables) and corpuses of texts (source, translated, revised, plus intermediate results and possibly "hors-textes" -- figures, etc.).

3) Analogy with compiler-compiler systems : rough translation is realized by a monolingual analysis, followed by a bilingual transfer, and then by a monolingual generation (synthesis).

2 - Linguistic aspects

1) Only linguistic levels (of morphology, syntax, logico-semantics, modality, actualisation, ...) are used, leading to some implicit understanding, characteristic of second-generation MT systems.

2) Hence, the extralinguistic levels (of expertise and pragmatics) which furnish some degree of explicit understanding are beyond the limits of second-generation CAT systems.

3) During analysis of a unit of translation, computation of these (linguistic) levels is not done sequentially, but in a cooperative way. Analysis produces the analog of an "abstract tree", namely a multilevel interface structure to represent all the computed levels on the same graph (a "decorated tree").

4) Lexical knowledge is organized around the notion of lexical unit (LU), allowing for powerful paraphrasing capability.

5) The texts are segmented into translation units of one or more paragraphs. This allows for intersentential resolution of anaphora in some not too difficult cases.

3 - AI aspects

1) During the structural steps, the unit of translation is represented by the current "object tree", which may encode several competing interpretations, like the "blackboard" of some AI systems.

2) This and the SLLPs' control structures allow for some heuristic programming : it is possible to explicitly describe and process ambiguous situations in the production rules.

This is in contrast to systems based on combinatorial algorithms which construct each interpretation independently, even if they represent them in a factorized way.

II - DIRECTIONS OF CURRENT WORK

1 - Linguistic knowledge processing

The experience gained by the development of a Russian-French translation unit of a realistic size over the last three years (6) has shown that maintaining and upgrading the lingware, even in an admittedly limited second generation CAT system, requires a good deal of expertise. Techniques are now being developed to maintain the linguistic knowledge base. Some of them deal with the lexical data-base, others with the definition and use of specification formalisms ("static grammars") and verification tools.

Lexical knowledge processing

In the long run, dictionaries turn out to be the costliest components of CAT systems. Hence, we are working towards the reconciliation of "natural" and "coded" dictionaries, and towards the construction of automated verification and indexing tools. Natural dictionaries are usually accessed by lemmas (normal forms). Coded dictionaries of CAT systems, on the other hand, are accessed by morphs or by lexical units. Moreover, the information the two types of dictionaries contain is not the same. However, it is highly desirable to maintain some degree of coherency between the coded dictionaries of a CAT system and the natural dictionaries which constitute their source, for documentation purposes, and also because these computerized natural dictionaries should be made accessible to the revisors.

Let us briefly present the kind of structure proposed by N. Nedobejkine and Ch. Boitet at an ATALA meeting in Paris in 1983. The central idea here is to start from the structure of modern dictionaries, which are accessed by the lemmas, but use the notion of lexical unit. Each item may be considered as a tree structure. Starting from the top, selections of a "local" nature (on the syntactico-semantic behavior in a phrase or in a sentence) give access to the "constructions". Then, more "global" constraints lead to "word senses".

At each node, codes of one or more formalized models may be grafted on. Hence, it is in principle possible to index directly in this structure, and then to design programs to construct the coded dictionaries in the formats expected by the various SLLP. Up to this level, the information is monolingual and usable for analysis as well as for generation. If the considered language is source in one or more language pairs, each word sense may be further refined, for each target language, and lead to equivalents expressed as constructions of the target language, with all other information contained in the dictionary constructed in a similar way for the target language. For lack of space, we cannot include examples.

This part of the work thus aims at finding a good way of representing lexical knowledge But there is another problem, perhaps even more important. Because of the cost of building machine dictionaries, we need some way to transform and transport lexical knowledge from one CAT system to another. This is obviously a problem of translation. Hence, we consider this type of "integrated structure" as a possible lexical interface structure. Research has recently begun on the possibility of using classical or advanced data base systems to store this lexical knowledge and to implement the various tools required for addition and verification. VISULEX and ATLAS (1) are first versions of such tools.

Grammatical knowledge processing

Just as in current software engineering, we have long felt the need for some level of "static" (algebraic) specification of the functions to be realized by algorithms expressed in procedural programming languages. In the case of CAT systems, there is no a priori correct grammar of the language, and natural language is inherently ambiguous. Hence, any usable specification must specify a relation (not a function) between strings and trees, or trees and trees : many trees may correspond to one string, and, conversely, many strings may correspond to one tree.

Working with B. Vauquois in this direction, S. Chappuy has developed a formalism of static grammars (7), presented in charts expressing the relation between strings of terminal elements (usually decorations expressing the result of some morphological analysis) and multilevel structural descriptors. This formalism is currently being used for all new linguistic developments at GETA. Of course, this is not a completely new idea. For example, M. Kay (13) proposed the formalism of unification grammars for quite the same purpose. But his formalism is more algebraic and less geometric in nature, and we prefer to use a specification in terms of the kind of structures we are accustomed to manipulating.

2 - Grafting on expert systems

Seeing that linguistic expertise is already quite well represented and handled in current ("closed") systems, we are orienting our research towards the possibility of adding extralinguistic knowledge (knowledge about some technical or scientific field, for instance) to existing CAT systems. Also, because current systems are based on transducers rather than on analyzers, it is perfectly possible that the result of analysis or of transfer (the "structural descriptors") are partially incorrect and need correction. Knowledge about the types of errors made by linguistic systems may be called metalinguistic.

In his recent thesis (9), R. Gerber has attempted to design such a system, and to propose an initial implementation. The expertise to be incorporated in this system includes linguistic, metalinguistic, and extralinguistic knowledge. The system is constructed by combining a "closed" system, based only on linguistic knowledge (a lingware written in ARIANE-78), and two "open" systems, called "expert corrector systems". The first is inserted at the junction between analysis and transfer, and the second between transfer and generation.

The control structure of a corrector system is as follows :

(1) transform the result of analysis into a suitable form ;
(2) while there is some error configuration do
    solve (using meta- or extralinguistic knowledge) ;
    if solving has failed then exit endif ;
(4) perform a partial reconstruction of the structure, according to the solution found ;
endwhile ;
(5) output the final structure in ARIANE-78 format.

(2) relies on metalinguistic knowledge only. The implementation has been done in Foll-PROLOG (8). The lingware used corresponds to a small English-French system developed for teaching purposes. Here are some examples.

Example 1 : ADJ + N N

(1) Standard free-energy change is calculated by this equation.

The analyzer proposes that "standard" modifies "change", while "free-energy" is juxtaposed to "change", hence the erroneous translation :

"La variable standard d'énergie libre est calculée par cette formule".

In order to correct the structure, some knowledge of chemistry is required, namely that "standard free-energy change" is a ... standard notion. With this grouping, (1) translates as :

"La variation d'énergie libre standard est calculée par cette formule".

Example 2 : (ADJ) N and N N

(2) The mixture gives off dangerous cyanide and chlorine fumes.

(2') The experiment requires carbon and nitrogen tetraoxyde.

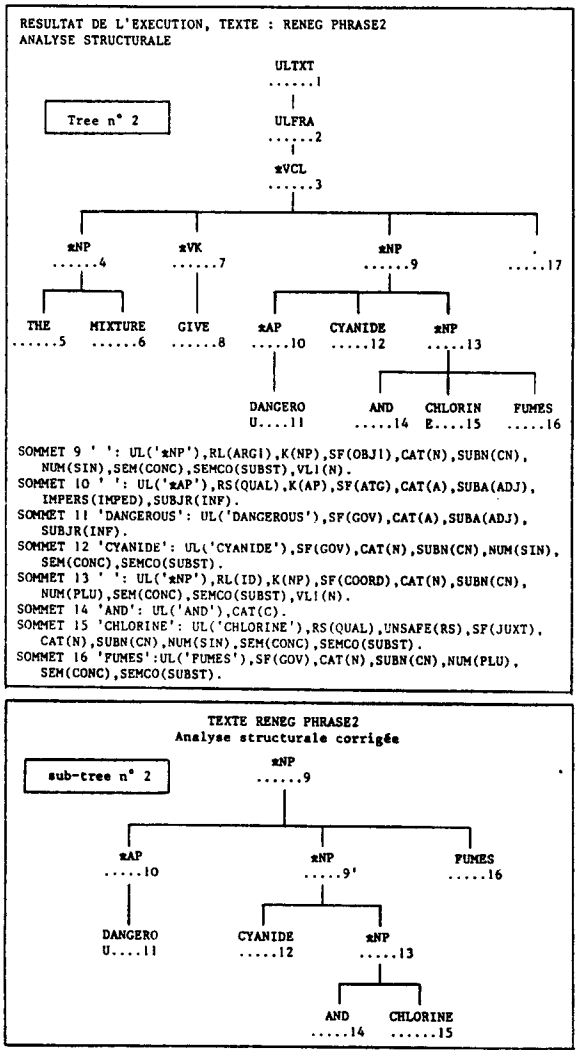
Let us develop this example a little more. Sentence (2) presents the problem of determining the scope of the coordination. The result of analysis (tree n° 2) groups "dangerous cyanide" and chlorine fumes", "chlorine" being juxtaposed to "fumes" (SF(JUXT) on node 12). Hence the translation :

"La préparation dégage le cyanure et la vapeur de chlore dangereux".

But, if we know that cyanide is dangerous as fumes, and not as crystals, we can correct the structure by grouping "(cyanide and chlorine) fumes" (see subtree n° 2). The translation produced will then be :

"La préparation dégage la vapeur dangereuse de cyanure et de chlore".

Of course, some more sophisticated analyzers would (and some actually do) use the semantic marker "chemical element" present on both "chlorine" and "cyanide", and then group them on the basis of the "semantic density" (e.g., number of features shared). But this technique will fail on (2'), because there is no "carbon tetraoxyde" in normal chemistry ! Hence, without extralinguistic knowledge, this more sophisticated (linguistic) strategy will produce :
"L'expérience demande du tétraoxyde de carbone et d'azote".
instead of :
"L'expérience demande du carbone et du tétraoxyde d'azote".



Example 3 : Antecedent of "which"

(3) The water in the beaker with which the chlorine combines will the poisonous.

The analyzer takes "beaker" instead of "water" as antecedent of "which". The corrector may know that chlorine combines with water, and not with a beaker.

Examples 4 & 5 : Antecedent of "it" within or beyond the same sentence

(4) The state in which a substance is depends on the energy that it contains. When a substance is heated the energy of the substance is increased.
(5) The particles vibrate more vigorously, and it becomes a liquid. (5') It melts.

In order to choose between "substance" and "state" (4), one must make some type of complex reasoning using detailed knowledge of physics -- and one may easily fail in a given context : it is not correct to simply state (as we did to solve this particular case), that a substance may possess energy, while a state cannot. Here, perhaps it is better to rely on some (metalinguistic) information on the typology, which may be included in a (specialized) linguistic analyzer, or in the expert corrector system. For (5), there are simple, but powerful rules like : if the antecedent cannot be found in the sentence, look for the nearest possible main clause subject to the left.

3 - Aiding the creation of the source documents

Lingware engineering may be compared with modern software engineering, because it requires the design and implementation of complete programming systems, uses specification tools, and leads to research in automatic program generation. Starting from this analogy, a group of researchers at GETA have recently embarked on a project which could converge with still another line of software engineering, in a very interesting way. The final aim is to design and implement a syntactico-semantic structural metaeditor that uses a static grammar given as parameter in order to guide an author who is writing a document, in much the same manner as metaeditors like MENTOR are used for writing programs in classical programming languages.

This could offer an attractive alternative to interactive CAT systems like ITS, which require a specialist to assist the system during the translation process. As a matter of fact, this principle is a sophisticated variant of the "controlled syntax" idea, like that implemented in the TITUS system. Its essential advantage is to guarantee the correctness of the intermediate structure, without the need for a large domain-specific knowledge base. It may be added that, in many cases, the documents being written are in effect contributing some new knowledge to the domain of discourse, which hence cannot already be present in the computerized knowledge base, even if one exists.

III - CONCLUSION : SOME LONG TERM PERSPECTIVES

There are many areas open for future research. The introduction of "static grammars" suggests a new kind of design, where the "dynamic grammars" would be generated from the specifications and from some strategies, possibly expressed as "metarules".

"Multisliced decorated trees" (16) have been introduced as a data structure for the explicit factorization of decorated trees. However, there remains to develop a full implementation of the associated parallel rewriting rule system, STAR-PALE, and to test its linguistic practicability.

Last but not least, the development of true "translation expert systems" requires an intensive (psycholinguistic) study of the expertise used by human translators and revisors.

REFERENCES

(1) Bachut D. - Vérastégui N. "Software tools for the environment of a computer aided translation system". COLING-84.

(2) Barr A. - Feigenbaum E., eds. "The Handbook of Artificial Intelligence (vol 1,2). Pitman, 1981.

(3) Boitet Ch. "Research and development on MT and related techniques at Grenoble University (GETA)". Tutorial on MT, Lugano, April 1984, 17 p.

(4) Boitet Ch. - Guillaume P. - Quézel-Ambrunaz M. "Implementation and conversational environment of ARIANE 78.4, an integrated system for translation and human revision". Proc. of COLING-82, Prag, July 1982, North-Holland, 19-27.

(5) Boitet Ch. - Nédobejkine N. "Recent developments in Russian-French Machine Translation at Grenoble. Linguistics 19, 199-271, 1981.

(6) Boitet Ch. - Nédobejkine N. "Illustration sur le développement d'un atelier de traduction automatisée". Colloque "L'informatique au service de la linguistique", Université de Metz, juin 1983.

(7) Chappuy S. "Formalisation de la description des niveaux d'interprétation des langues naturelles". Etude menée en vue de l'analyse et de la génération au moyen de transducteurs. Thèse de 3ème cycle, USMG, Grenoble, juillet 1983.

(8) Donz Ph. "Foll, une extension au langage PROLOG". Document CRISS, Grenoble, Université II, février 1983.

(9) Gerber R. "Etude des possibilités de coopération entre un système fondé sur des techniques de compréhension implicite (système logico-sémantique) et un système fondé sur des techniques de compréhension explicite (système expert). Thèse de 3ème cycle, Grenoble, USMG, janvier 1984.

(10) Hayes-Roth F. - Waterman D.A. - Lenat D.B. eds. "Building expert systems". Reading MA, London Addison-Wesley, 1983.

(11) Hobbs J.R. "Coherence and co-reference". Cognitive sciences 3, 67-90, 1979.

(12) Isabelle P. "Perspectives d'avenir du groupe TAUM et du système TAUM-AVIATION". TAUM, Université de Montréal, mai 1981.

(13) Kay M.
"Unification grammars". Doc. Xerox, 1982.

(14) Laurière J.L. "Représentation et utilisation des connaissances". TSI 1(1,2), 1982.

(15) Vauquois B. "La traduction automatique à Grenoble". Document de Linguistique Quantitative n° 29, Dunod, 1975.

(16) Vérastégui N. "Etude du parallélisme appliqué à la traduction automatisée par ordinateur. STAR-PALE : un système parallèle". Thèse de Docteur-Ingénieur, USMG & INPG, Grenoble, mai 1982.