PRACTICAL EXPERIENCE OF MACHINE TRANSLATION
V. Lawson (ed.)
North-Holland Publishing Company / © ASLIB, 1982

EUROTRA: AN ATTEMPT TO ACHIEVE MULTILINGUAL MT

M. King
ISSCO, University of Geneva
Switzerland

EUROTRA is an MT system currently being designed under the auspices of the Commission of the European Communities. A basic principle of its design is multilinguality, to be achieved by a modular design which permits a strict separation of mono-lingual analysis and generation modules from multi-lingual transfer modules. Communication between modules is performed via an interface structure, whose definition has been commonly agreed by the groups collaborating in the design. Integrity of the system is ensured by the interface structure, by the use of a common data structure to hold results throughout processing and by the use of common software to manipulate the data structure.

1. BACKGROUND

Planning the design of EUROTRA started in February 1978, at the instigation of the Commission of the European Communities. A group of European experts in machine translation were brought together and a working group set up to collaborate in specifying the design of the system. Fairly intensive work has continued ever since, with further groups associating themselves with the project. Up to now, around eighty people have been involved, although none works on a full time basis, coming from a wide range of University Institutes throughout the Member Countries.

2. CURRENT STATUS

By now, more than three years of hard work have gone into designing the system. The general framework has become stable, the main decisions concerning the software underpinning the system have been taken. A Council decision approving the next stage, implementation of the project, is expected shortly.

The rest of this paper is devoted to explaining the system design, and to a brief description of the software. I hope to show that coherence and system integrity have been ensured whilst, at the same time, a great deal of freedom has been left to the individual groups concerned with the linguistic parts of the system, which are, by their nature, crucial to the success of the system.

3. PRINCIPAL DESIGN CRITERIA

From the very beginning, the system has been conceived of as a multi-lingual system, intended to be capable of carrying out translation between all the language pairs of the Community languages. This has an immediate consequence on system design, since it means that it is impossible to take advantage, during analysis of the source language, of any "accidental" similarities between the source language and the target language. In a bi-lingual system, where, for example, if the source language is French and the target language is Russian, just enough and just the right kind of analysis of French can be done to get the right Russian translation as though French were being looked at through a pair of Russian spectacles. In a multi-lingual system, analysis is forced to be more thorough, since the result of the analysis must be adequate for any of the target languages in the system.

At the same time, the system must be extensible: it must be possible to add new language pairs and new domains of discourse without re-writing or disturbing what is already there. When the planning of EUROTRA started, there were six Community languages - Danish, Dutch, English, French, German and Italian. During the planning period, a seventh, Greek, has become an official language; in the future yet more may be added. It is clearly important that the extensibility criterion is maintained.

From the beginning, too, the system has been designed to be developed collaboratively, by groups working independently on the linguistic modules in the Member States, with a separate central group ensuring the coherence and the integrity of the system as a whole. The alternative would be to try to bring together into one place a large multi-national team, which would probably prove impracticable and would certainly be socially undesirable, since one of the motivations behind the project is to encourage research in this area throughout Europe, an aim that could hardly be achieved by entering into competition for skilled labour with local specialist groups.

4. MODULARITY

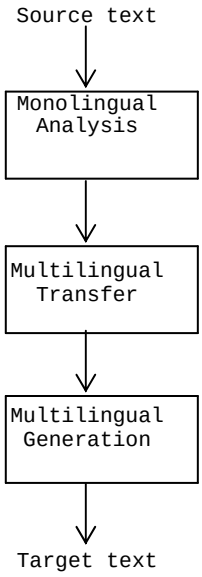
The three considerations spelt out in the last section lead almost inevitably to the design of a highly modular system, that is, a system which is built up from clearly distinguished independent parts, each with a specific task to perform. Independent support for modular design comes from work in computer science, where large programs - and machine translation systems are necessarily large - are always designed in modules. There is no need to be a computer programmer to see the advantages - the same advantages will hold for the organisation of any large and complex operation, mounting a conference, for example.

First, if the job can be broken down into separate sub-tasks, it is easier to specify exactly what each sub-task must achieve. Then, the people (or programmers) responsible for that sub-task can get on with one thing at a time, just that sub-task, without worrying about the interaction with the rest of the job. Thirdly, when something goes wrong, as it inevitably does, it is much easier to isolate the mistake and to correct it. And finally, when the sub-task is being satisfactorily performed, it is much easier to explain to someone else what the sub-task is and how it is being accomplished than it would be if the whole job were being carried out in a single more complex operation, perhaps with bits of sub-tasks mixed up together.

Note, though, that in any modular system, the interface between modules must be very well defined and rigorously adhered to. That means that each module must know its starting point and what it is expected to produce, since its results may well serve as the starting point for a different module. The analogy with organising a conference holds here too: if I am the module responsible for organising the agenda and someone else is the module for organizing printing, then the interface between us is the agenda, with the list of authors and titles of their papers, and if I do not produce the agenda, the person responsible for the printing cannot do his job properly. We shall come back to the interface between EUROTRA modules later, in section 6.

5. EUROTRA'S BASIC MODULES

If the job to be performed is translation by machine, and we want to split that job up into sub-tasks which can be performed independently, what are the easiest divisions to make? Multi-linguality comes into the answer to this question too, since there is a parallel question: how can the job be divided up so that some parts of the system can work without knowing the target language, others without knowing the source language? The obvious answer to that is to make the analysis of the source language a separate sub-task, and to make it mono-lingual, and to do the same thing for generation, so that analysis of Dutch, for example, can be carried out solely in terms of Dutch, the generation of the target language, Italian (again, for example) can be done solely in terms of Italian. But then clearly there has to be a link between the two, something which takes the results of the Dutch analysis and prepares it for the generation of Italian. This module, the only one which knows about more than one language, is the transfer module, since it "transfers" the Dutch results into the starting point for the Italian generation. The overall picture then is:



There are obvious economic reasons for trying to keep transfer as small as possible. With six languages there are six analysis modules, six generation modules, but 6x5=30 transfer modules. With seven languages, there are 7x6=42 transfer modules, and so on. For this reason, the generation modules of EUROTRA are designed to do considerably more than the generation modules of most translation systems, taking over a lot of the work hitherto done by transfer.

6. THE INTERFACE STRUCTURE

In the section on modularity, it was said that the interface between modules knew what to expect and what it must produce. Such an interface is rather conspicuously lacking in the diagram above, where the arrows go straight from analysis to transfer and from transfer to generation with no attempt to specify what the content of the interface is. Let us therefore consider the question here.

Clearly the result of analysis must be in some way a representation of the text, but giving more detail about its structure (unless, of course, we are concerned with word-to-word translation, where we shall produce perhaps a number of good jokes, but not an adequate translation). EUROTRA'S interface structure, in fact, contains at least four kinds of information about the text which has been analysed. The overall shape of the structure is a tree, built up according to a syntactic analysis of the text in terms of dependency grammar. Dependency grammar analyses the text into constituents, then picks out one item in each constituent as being the main item, or governor, of that constituent, to which all other items are related.

A dependency tree structure carries some information in itself, since it shows which constituents are related to which. But, apart from the fact that one constituent is given a special status and is taken to dominate the others, it gives no information about the nature of the relationships involved.

A EUROTRA interface structure therefore carries additional information on the kinds of relationships.

The first of these describes surface syntactic function - whether a constituent plays the role, in the surface structure of the text, of a subject, an object and so on. At this level of description, it is only the surface structure that counts. So, in

"John broke the window"

"John" is subject and "window" is object, whilst in

"The window was broken by John"

"The window" is subject and there is no object, despite the fact that at another level of description "John" and "window" play similar roles in both sentences.

This other level of description of relationships is that which is captured in the EUROTRA interface structure by the semantic relations. The easiest way to see why semantic relations are needed is to think of prepositional phrases. Both

"He built the boat with care"

and "He built the boat with wood"

have the same surface syntactic structure. Yet their translation into another language than English may involve producing quite different structures in the target language. It is only when the semantic relation of "with care", (MANNER), to the rest of its sentence is distinguished from the semantic relation of "with wood" (SOURCE) to the rest of its sentence, that it becomes possible to be sure of producing the right translation.

Another type of information which appears in the interface structure is information on whether a constituent is "valency bound" or not. Valency boundedness is an attempt to capture the intuitive feeling that some constituents are more intimately connected with the predicate than others. In

"George ate his lunch"

both "George" and "his lunch" are valency bound to "ate" whilst if we add

"George ate his lunch quickly on Friday"

neither "quickly" nor "on Friday" is valency bound.

In addition to these three dimensions of description - surface syntactic function, semantic relations, valency boundedness - the interface structure also contains morphological, morpho-syntactic and syntagmatic information. So for

"He went to school"

it is recorded that "he" is a third person singular personal pronoun forming a noun group, that "went" is the past tense of "go" and forms a verb group and so on.

These four kinds of information constitute the minimum amount of information about the text that should be calculated. Much more information can, and will, be added, for example, on definiteness, determinedness, emphasis and so on. In fact, there is no upper limit on the amount of information a group may store in the interface structure. But there is a lower limit: all groups are committed to an attempt to calculate at least the four kinds of information briefly described in this section.

Interface structures of this type serve as the means of communication between analysis and transfer, and between transfer and generation. In the ideal case, all transfer does is to replace the lexical units of the source language in the interface structure input to it by the lexical units of the target language, retaining the rest of the interface structure unchanged. The resulting interface structure is then handed over to generation.

7. THE COMMON DATA STRUCTURE

The interface structure guarantees a certain coherence throughout the system. No matter what transfer or generation module is being written, it knows what to expect as its starting point.

A further guarantee of coherence is an agreement to use a single data structure to represent intermediate results throughout the entire system.

One of the chief considerations in deciding on the nature of this data structure was a desire not to restrict the choice of linguistic strategy by the individual groups. Over the last fifteen years or so a great deal of attention has been paid to techniques for computer analysis of language, and a number of different techniques have emerged. Quite apart from the consideration that amongst the different EUROTRA groups, experience with a variety of such techniques can be encountered, so that clearly best use of experience can be made by allowing groups to use whatever technique seems to them best, there are good linguistic and pragmatic arguments for attempting to leave open as much as possible the choice of linguistic strategy.

The linguistic arguments come quite simply from the diversity of language. It is not a priori obvious that the best way to analyse German is also the best way to analyse Italian, for example. The intuition that specific languages may require specific tools is born out by practical experience.

The pragmatic arguments come from the speed with which computational linguistics has developed in the last few years. New analysis techniques frequently appear, and since each new model builds on past experience, prove to have advantages over their predecessors. There is no reason to believe that progress will slow down. Indeed, projects like EUROTRA should tend to stimulate work in these areas. So, once again, it makes sense to design a system which is not restrictive and which allows for research and testing of new approaches.

These and other considerations have led to the definition of a common data structure which is in essence very simple. It is based on allowing easy expression of alternatives at any level of description and at any point in the processing. Thus, if a constituent could be a noun group or a verb group, to take a simple example, it is possible to keep both possibilities open until sufficient information is available to choose between them.

Of course, the possibility to express alternatives does not oblige alternatives to be invented where they do not exist: thus a technique which is constructed on the principle of never having to change its mind is as possible as a technique which keeps all possibilities open until the last possible moment.

In a very strong sense, then, the common data structure is independent of any linguistic strategy. Its only imposition is that the results it expresses must conform to partial or whole interface structures.

It follows from this that it is also independent of the way linguistic facts about a specific language can be expressed. We shall see in the next section how this can be so.

8. MANIPULATING THE DATA STRUCTURE

Modifications to the data structure bearing intermediate results are done via the application to the structure of rules. The rules are independent of the structure itself, and it is they which constitute a description of the language being treated.

A rule consists of two parts. The first specifies a state of the data structure to be looked for, or, in other words, a partial result already achieved. The second specifies a change to the data structure to be carried out when the specified state has been found, or, in other words, a new intermediate result to be recorded on the data structure. As an example, there might be a rule which looks for a verb preferring an animate agent preceded by an independent noun phrase whose main constituent is marked as being animate and constructs out of the two a single verb phrase containing the noun phrase as a constituent, simultaneously marking the noun phrase as being in the semantic relationship of agent to the verb.

Although the basic pattern of a rule is very simple, rules themselves may of course be very complex, both in terms of the situation they are looking for and of the action to be carried out if that situation is found. There is no restriction on the type of information which can be asked for in the specification of the situation. A rule may simultaneously specify as conditions for its own application a particular morphological context, plus a particular semantic configuration, plus a particular dependency structure and so on. Thus there is no stratification inherent in the system: that is, there is no need to do morphological analysis independently of semantic analysis, or to complete all syntactic analysis before taking into account semantic considerations, and so on. The person(s) responsible for writing the linguistic rules to carry out a particular task are free to decide on the best linguistic way to organise their rules. Thus, in this way too, EUROTRA makes possible a very wide range of linguistic strategies.

Rules are grouped together into grammars, which may be of any size: there is no restriction on the number of rules in a grammar. So there is no need to write one very large grammar to accomplish one of the basic modules of analysis, transfer or generation. These tasks too can be broken down into sub-tasks, each sub-task being the responsibility of a separate grammar. Tools are provided to control the computational behaviour of the grammars, and thus to prevent infinite looping or combinatorial explosion.

As a brief summary of this section for the cognoscenti, the system as a whole constitutes a production system with external control mechanisms, and with the common data structure serving as the equivalent of a production system data base.

9. THE LIMITS OF THE SYSTEM

The system has been designed to put as few constraints as possible on the groups writing the linguistic modules. Most of the better known techniques for carrying out linguistic analysis have been considered during the design phase, and the system has been planned in such a way as to allow their use within EUROTRA.

Nonetheless, there are limits which come from linguistic problems which no-one yet knows how to solve. As an extreme example, consider the two sentences (Example based on Winograd):

The town councillors refused a permit to the women because

- they feared violence
- they advocated revolution.

Most of us would take the first "they" to be the town councillors, the second to be the women. But our judgement is based not only on the sentences themselves but on our knowledge of an entire culture. It has been known for people coming from a different culture to get these two "they's" the other way round.

EUROTRA is not intended to solve problems like these. At best, it may stimulate research aimed at their resolution and may provide a framework within which to try out possible solutions, but it will not count itself a defective system because it cannot get them right.

Other sorts of limits too should be taken into account. EUROTRA is not intended to be a fully automatic machine translation system in the sense that it aims at producing text which will need no post-editing. Such output is suitable only for very specific types of text in very specific applications. A great deal of thought is going into embedding EUROTRA in a wider context of advanced text-processing, part of which will try to make life as easy as possible for the post-editor.

It should be remembered too that the quality of a EUROTRA translation will depend on a number of different factors. It is quite possible to have several grammars or sets of grammars dealing with, say, the analysis of a particular language, where alternative grammars may be "tuned" to a specific text type. If this is the case, then selecting texts in such a way that a particular text is dealt with by the grammars tuned for its text type rather than by a general purpose set of grammars will clearly change the quality of output.

It is possible too to link modules and sub-modules together in different ways to take account of the use to which the target language version is to be put. Often the objective of the translation determines the quality required. The modular design of the system permits a great deal of flexibility in these respects too.

On the other hand, there will always be texts which by their nature are unsuitable for machine translation. Any text which relies on deliberate use of ambiguity, for example, should not be submitted to a process which tends to regard it a duty to disambiguate wherever possible. Any text where not only must the translation be an equivalent of the original but where also the inferences to be drawn from the original and the translation must be equivalent, as is the case with legal texts where both original and translation have equal status, should not be submitted to machine translation. Many more instances of texts unsuitable for machine translation could be found. But systems such as EUROTRA are not intended to translate perfectly every possible text: they are intended to remove some of the burden of banal everyday work from human translators who have plenty of more interesting work to do.

10. CONCLUSIONS

This paper has given a very sketchy outline of a multi-lingual machine translation system, EUROTRA, attempting to show how the intention to create a multi-lingual system affects not only the linguistic work to be carried out but the system design itself.

ACKNOWLEDGEMENT

Very many people have contributed to the design of EUROTRA, the members of the EUROTRA Co-ordination Group and of the language groups directly, others by making the results of their research freely available through publication. It would be both invidious and misleading to name individuals. Let it be enough to say that EUROTRA would not look the way it does without having had the benefit of a great deal of previous experience.