[*Proceedings of a Workshop on Machine Translation*, July 1990, UMIST]

Rapportage from the discussion group 'Speech/NL Interaction and MT'

Kate MORTON
Department of Language and Linguistics
University of Essex
Wivenhoe Park
Colchester C04 3SQ, UK

July, 1990

SPEECH and MACHINE TRANSLATION

Introduction
Katherine Morton - Essex

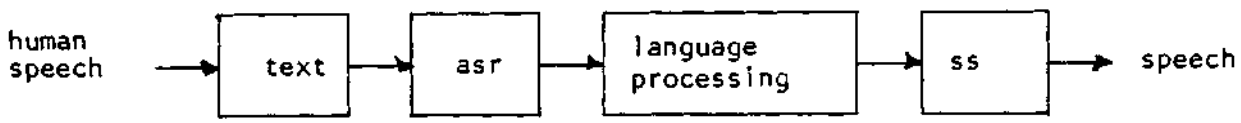
In a simple view, speech technology is concerned with automatic speech recognition and with speech synthesis. However, speech systems must be associated with language processing, machine translation, or dialogue construction in order to be useful. It is not yet clear how to efficiently interface speech and language processing systems; the first step is to define the problem as explicitly as possible. The session on a potential speech interface to machine translation addressed itself to discussing the general issue of the nature of the difficulties involved. To this end, Marcel Tatham presented some ideas about the type of speech systems which could be associated with formal language descriptions; John Connolly and Simon Lucas presented approaches to dealing with language processing that might be useful for providing a way to develop a complete translation system potentially involving speech.

1. Speech
Marcel Tatham - Essex

Speech technology covers the area of the development of voice input and output systems, respectively automatic speech recognition and speech synthesis. As yet the technology is by no means perfect, but it is already sufficiently well developed for incorporating into some useful applications which rely on language processing. A simple example would be the automatic translation of the spoken word, rather than, as is the case at the moment, translation of just text. It may be premature to think of automatic simultaneous translation, but eventually it will be possible to translate the spoken word into a voice output in real or near real time. Machine translation will have to improve somewhat for this to happen, but it is something that we should keep in mind.

The text that current machine translation and other systems which involve language processing is not a complete encoding of the meaning of the writer. When we write a text we expect it to be decoded by a human being, and part of the understanding of the text comes, of course, from the reader. Spoken language almost always contains more information than written language in the sense that nuances in speech, usually at the prosodic level, are able very often to convey the speaker's mood or intentions in a way that is not encoded in the actual words themselves. Automatic speech recognition devices are in principle able to detect such nuances and could provide the decoded information to the language processing part of the application: thus by using a speech input language processing has more information to work on.

Similar benefits accrue to speech processing when the two technologies are brought together. Speech synthesis, for example, is well-known for being unnatural in the way it sounds. This is partly due to inadequacies in the technology itself, but mostly it is due, once again, to the fact that speech synthesis systems usually have insufficient information available to produce a good simulation of a human being talking. The input to such systems is often plain text - that is, text which, as we saw above, does not adequately encode the author's intentions or mood. Ideally the input should be at the 'concept' level rather than the text level; that is, it should be an encoding of what was in the writer's mind before it was converted to words on paper. In as much as a language processing device is simulating the creation of text by the writer it should be possible to use this 'concept information' before it is lost altogether in the conversion to text. The diagram illustrates the point.



There are therefore several areas in which it should be possible to improve the results currently obtained from both language and speech processing systems by bringing to two together into an fully

integrated system. It may well be that a major reason why such systems fall short of the performance we would like is that they are currently treated as quite separate areas of technology and quite separate areas of supporting theory and research.

2. Functional Architectures for Spoken MT

John Connolly - Loughborough

Probably the two best known functional architectures that have been proposed for automatic speech recognition (ASR) systems are i) the 'conveyor belt' whereby the input is analysed at successively higher levels of linguistic structure, and (ii) the 'blackboard', whereby the analysis of the input is achieved through the cooperative activity of several separate knowledge sources under the control of a scheduling mechanism which generally does not simply proceed up the scale of linguistic abstraction. Either of these functional architectures is compatible with the incorporation of an ASR system into a MT system.

The output of the ASR module of an MT system needs to be in the form of a semantic interpretation (SR) rather than conventional text. This SR will be input to the transfer module, whose output will comprise the corresponding SR of the target language. The latter SR will form the input to a 'speech synthesis from concept' module, which will generate the appropriate spoken output.

Various questions remain unanswered in relation to the above. These include the following: (i) What is the best internal functional architecture for each of the components of the spoken MT system? (ii) How does one cope with the problem of representing and retrieving the encyclopaedic knowledge that such a system is likely to demand? (iii) What is the best way of representing/compartmentalising and utilising the linguistic knowledge within such a system? (iv) What kind of (probably non- classical) logic is best suited for such a system, and how should it be implemented? (v) What part (if any) should be played by automatic learning in the development of such a system? (vi) How should interactive discourse be managed within such a system? and so on.

Nevertheless, we may take comfort from the fact that much of the work that has been done in the past with reference to the functional architecture of monolingual knowledge-based speech systems does seem to be applicable also to multilingual systems. On the other hand, the path towards the development of accurate and fast spoken MT systems will not be easy, and will require much further research, both on functional architectures and on other matters.

3. Machine Translation with Syntactic Neural Networks

Simon Lucas - Southampton

Most machine-translation (MT) systems have employed the use of explicit translation rules, written laboriously by teams of experts. By contrast, our proposed approach is to train syntactic neutral networks (SNN's) to perform the task. The important features of SNNs are their underlying grammars and the unsupervised manner in which they are able to learn.

Syntactic neural networks were introduced in Lucas and Damper (1989). Since then they have been successfully applied to a number of diverse problems, such as isolated hand-written character recognition (Lucas and Damper, 1990), cursive script recognition (Lucas an Damper 1989) signature verification (Lucas and Damper 1990) and text-phonetics translation (Lucas and Damper 1990).

Our SNNs are not the first neural networks to have an interpretation in formal grammars, but they are perhaps the most useful Previous approaches have tended to force conventional formalisms into a connectionist framework, without suggesting any new learning algorithms or gaining any extra insight into the problems at hand. For example, Fanty (1986) gives a connectionist context-free parser that requires O(n3) (where n is the maximum length of input string), which implies impracticably large networks. In contrast, our full context-free parser requires only O(n2) neurons (Lucas and Damper 1990), the reduction being achieved via the appropriate use of time-delay elements. Further, we have suggested useful restrictions on the full context-free formalism, which reduce the growth to O(log n) - these networks may then deal with very long input patterns.

More importantly, the connectionist paradigm has suggested novel grammatical inference algorithms, based on attributed grammars (Knuth 1986) where each symbol may have attached attribute vectors which add meaning to the symbols. The attributes allow the use of clustering techniques which speed up the inference algorithms enormously; this is significant since grammatical inference algorithms are notoriously expensive e.g. the

well known inside/outside algorithm (Baker 1979) is 0(n3,m2), where m is the number of non-terminals. Furthermore, the use of clustering techniques means we no longer have to guess the number of non-terminals necessary at the outset; this will be decided by the clustering threshold (analogous to the

vigilance parameter of Grossberg, 1976) and the variability of the training set. Also, the learning occurs locally, which adds some biological plausibility to the procedure.

In pattern recognition applications, we choose attributes which are useful as gross pattern descriptors. For natural language processing, including MT, we plan to use attribute vectors similar to those employed in the work of McClelland and Kawomoto (1986) with their semantic micro-features. The idea is to train the system on multilingual descriptions of micro-worlds; thus for each sentence in English, we also give a semantic description in terms of the attribute vectors. After extensive exposure to the self- organising net, we will then use it generatively (we may do this because of the underlying grammar) to gain an insight into the type of language model that has been acquired. The ability to do this demonstrates a significant advantage over multi-layer perceptrons or self-organizing feature maps for example, where no underlying grammar is involved.

For MT we have an SNN for each required language. Each SNN is trained on sentences of its associated language together with a description of the meaning of each sentence, couched in terms of a fuzzy conceptual graph. As training proceeds, a grammar is inferred for each languag3e and an associative mapping is learned between stochastic parses of sentences and their meanings. To use the model for translation, we feed the input sentence into the appropriate SNN, which triggers the meaning in the conceptual graph. This may then propagate excitations to any of the other language SNNs, now running in generation mode, to produce the translated outputs