The Development and Progress of Machine Translation Systems

Derek Lewis

The involvement of computers in natural language translation began after the Second World War with Warren Weaver's famous memorandum in 1949 which suggested that translation could be handled by a machine as a variation of a coding task. The vast injection of resources which now followed, especially in the USA, until the ALPAC progress report of 1966 which recommended on the basis of what had been achieved so far that MT should be relativized to one area of experimentation within the general field of computational linguistics and urged specific support to the more practical aspects of machine-aided translation such as the mechanisation of text-production, computer-based word and phrase look-up facilities, etc., is well known. (ALPAC 1966; for early reviews of the state of the art see Bar-Hillel 1960, Booth 1967, Josselson 1971). From the mid 1960s onwards, however, a number of fully automated MT systems were developed, some of which achieved the status of commercial and government service. The results of this experience and the realisation that fully automated systems are probably a thing of the future has provoked an interest during recent years in so-called limited language and interactive systems which require some sort of human intervention during or before the translation process in order to correct or pre-empt the computer's deficiencies. At a recent ASLIB conference dedicated to practical experience of machine translation the interest of the professional translating fraternity in already operational and commercially viable systems of machine assisted translation (MAT) emerged very clearly (Lawson 1982; see also Commission of the European Communities 1976). Much work, too, has been done in the area of computerized terminology banks and thesauri, although this review will concentrate on systems which aim to use the computer for some degree of consistent analysis, transfer and synthesis rather than as a large data bank for periodic terminological access.

An MT system has to do — in one way or another — the following things:

(1) Load itself with the SL text and store all the words: it must, of course, be able to segment the words into morphological stems and affixes, to look up each word in its own dictionary or lexicon to determine what part of speech the word is, what grammatical and, perhaps, semantic information is associated with it.

(2) Analyse the SL text, usually a sentence at a time, and assign to each sentence a PHRASE STRUCTURE — the computer thus represents to itself the grammatical structure of each SL sentence. To do this it has to scan the words in order, look at the information it has about these words and decide on a structure. Different systems use different phrase structure representations. This stage is called ANALYSIS.

(3) In the SYNTHESIS stage of the system, each sentence is translated into the TL. The dictionary is consulted for the TL words and phrases, and changes in word order are made, inflectional endings are added, and so on, as required. This process at its simplest can be conceived as a mapping of TL words or structures onto the phrase structure analysis produced by stage (2).

Information on SL morphology, syntactic categories and semantic features will be contained in one or more dictionaries for consultation during analysis, and the synthesis stage will require access to bilingual and/or monolingual (TL) lexica. Large single bilingual dictionaries are more often found in older, direct translation systems (see below). Separate dictionaries for idioms, high-frequency words and lexically restricted sublanguages are also implemented (Hutchins 1978). Single monolithic bilingual dictionaries can lead to faster processing times but are more difficult to compile and alter and tend to be system-specific (for general descriptions of stages and procedures see Hutchings 1978, Sigurd 1978). The theory and practice of terminology and nomenclature in sublanguages, the organisation of lexical data bases and the problems of equivalence are the objects of recent reported research in the Soviet Union. Here, also, work is being undertaken on techniques of text analysis without a full dictionary and on algorithmic methods of overcoming gaps in the lexicon which go further than the 'not found' message (Knowles 1984).

Parsers

Parsers have been a particular object of programming and linguistic interest. (Tennant 1981, King 1983, Jones and Wilks 1983), especially from an MT point of view (Johnson 1983).

Syntax-orientated MT systems (and these account as far as I am aware for all operational ones at the present time) rely on a parsing component for structural analysis of input text. There are a number of features instrinsic to computational parsers. They consist typically of a set of rules describing the language (the grammar), a set of procedures for applying the rules to a particular text (the recognizer), and facilities for outputting a structural representation of the sentence parsed (parsers doing these three things are often said to be transducers). Parsing strategies can, moreover, be classified in a number of ways. They can be context-free or context-sensitive (since Chomsky context-free parser have generally been considered at best inefficient describers of language structure, although recent voices have dissented from this view (Gazdar 1983, Sampson 1983b)). They can operate from the top of a tree down or from the bottom upwards, i.e. given a sentence to analyse they can start from the S constituent and test the input sentence against a series of expected structures until the correct one is found or, typically working from left to right at terminal string level, they can try to build up structures one word at a time until a series of possible structures is found, most of which have to be discarded later on as higher level structures are assigned. Depth-first parsers, moreover, explore one rule re-write possibility at a time to progress up or down the tree as far as possible, backtracking to try another possibility if it cannot complete the parse. Breadth-first parsers, on the other hand, will follow up and retain all possibilities on a horizontal level while proceeding through the tree. Backtracking is superfluous as bad structures are discarded from the parallel set as they emerge. Related to these features is that of determinism. Non-deterministic parsers allow decisions during the computational process to be changed or several alternative decisions on structure to be held in parallel. A deterministic parser, however, claims to build only structures which will be irrevocable: once a decision is made it cannot be altered and there is no provision for backtracking. Finally, a parsing algorithm is said to be decidable if it can determine whether a given input string is either a well-formed string according to the grammar or not (in contrast to a parser which will only halt on recognizing well-formed sentence input and will search indefinitely if the string is ungrammatical, being unable even to give a signal of failure). In practice parsers may combine more than one strategy (e.g. a basically bottom up search for structure can be interrupted with the occasional top-down scan from the S constituent in order to filter out structures which could not contribute to the eventual analysis) but for most cases one strategy predominates: thus ATNs are top-down depth-first and chartparsers are invariably bottom-up breadth-first (De Roeck 1983).

Some current problems and issues in parsing theory can be summarized as follows:

(i) Problems of transformational parsing (a transformation parser is one whose components are similar to those of the classical transformational grammar and in particular use a transformational component to map tree structures onto different tree structures).

The METALS MT project in particular found that T-rules as formulated by Chomsky involved excessively complex programming. Although the Chomsky model was designed to generate sentences, i.e. principally surface strings from base structures, it proved difficult to construct parsers which could apply T-rules in reverse in order to generate complete but unique tree structures from terminal strings: in particular reverse T-rules were unable to recover deleted deep structure (King 1983). Ultimately METALS went over to an earlier, pre-Chomsky version of transformational grammar.

(ii) The advantages and disadvantages of ATNs.

Augmented Transition Networks or ATNs (invented by Woods 1970) are basically top-down parsers which start at the constituent S and move down the tree, testing for each node and backtracking to try another structure as or if a previous test fails. Although fundamentally a context-free device, it is given an appreciable degree of context-sensitivity for natural language by the use of registers which store information on what has been tested already and which can be retrieved and compared with information currently available or yet to come. It should be noted that ATNs have been implemented in different host languages and that the implementer is free to change many functions, with the result that there is probably no standard ATN form or even notation in this highly flexible system. Disadvantages are that backtracking can lead to repeated parsing of constituents and that the whole system may be halted if the parser cannot proceed beyond a problem input (structures to the right of the unparsable component are never reached). Co-ordinate complex structures have also presented serious hurdles (Boguraev 1983) and on the whole ATNs are most efficient in small systems (MT systems do not usually fall into this category).

(iii) Charts (or data-driven parsers).

The chart, developed by Kay (1976) and Kaplan (1970), is a data structure which is designed to reduce the complexity of having to retain a number of alternative structures as the parser (say, an ATN), moves through the sentence. Charts therefore are representations of the possible structure assignments (i.e. the tree nodes and the relationships between the nodes) as they stand at any point during the parsing process. Charts can be combined with Q-graphs to provide a valuable tool for MT. Q-graphs, designed by Colmerauer (1970), enable trees to be altered and mapped onto other trees according to whether structural descriptions are met, thus simulating a key component of transformational grammar. (Varile 1983).

(iv) Marcus Parsers

Marcus' deterministic parser (Marcus 1980) works from left to right and dispenses with the need to backtrack (as in ATNs) or retain parallel parses: it never builds structure which will not be eventually used in the final parse. It does this primarily by means of a limited look-ahead facility (the buffer), which scans three constituents ahead to determine the structure of the current node(s). A limitation of the system is that a finite buffer can always overflow, thus preventing a correct scan of garden path sentences (e.g. 'I told the boy the dog bit Sue would help him' or 'The horse raced past the barn fell') which, in a deterministic parser unable to change structure already assigned, is a major disadvantage. Although it has been questioned whether the distinction between backtracking and limited look-ahead is substantial, Marcus himself claimed some psychological reality for his parsing model based on the existence of Chomskyean universal features of language which at least some of the processing strategies of this particular parser are supposed to reflect (Varile 1983). Deterministic parsers may also be more useful in coping with ungrammatical input because they are able to pinpoint the stage at which grammatical input in terms of the rules of grammar have given over to ungrammatical (ATNs on the other hand would not be so efficient at this since the acceptance of previous structure depends more heavily on what lies ahead and there is no clear point at which ungrammaticality is perceived to occur (Sampson 1983, Briscoe 1983).

(vi) Syntax versus Semantics

The most far-reaching debate concerns the whole future of syntax-driven parsers. Some semantic component is vital if they are to be at all efficient and the claim has been made (in its strongest form, perhaps, by Yorick Wilks) that they have been an unnecessary, and from a psycholinguistic point of view, counter-intuitive diversion: natural language can be parsed more effectively and completely by a model which relies entirely on semantic categories. The domination of purely syntactic parsers during the 1960s was challenged from the early 1970s by models claiming to be wholly semantic sentence analysers (Wilks 1972, Riesbeck 1975a, 1975b, Schank 1975). Recent positions in parsing theory appear to accept the need for semantic processing and to be concerned with the levels and frequency during parsing at which semantic information is optimally involved (e.g. immediately after broad syntactic constituents are isolated, at frequent intermediate intervals during parsing, or as soon as possible after non-terminal nodes have been built: see Ritchie (1983).

Classification of MT Systems

In view of the number and diversity of components in a large MT system, differences in restrictions on text input and the quality of translation produced, it is not surprising that there exists no straightforward typology of systems. A classification based on the evolution from so-called direct (or first generation) to interlingual and transfer systems (second and third generation) has the merit of providing convenient historical perspective, although early systems have increasingly adopted features (notably modularisation of components) of later developments and cannot accurately be considered total prisoners of their original conception. This explains why — as can be seen in the following account — it is often the earlier systems which are now at the forefront of commercial implementation.

Direct Translation Systems

The first MT systems adopted the Direct Translation approach, i.e. they were designed for a single language pair (usually Russian to English) and undertook the minimum amount of analysis required to convert the SL into the TL, without clear separation of the processes and without a clear or adequate model of linguistic structure representation. The major disadvantage of this approach was that improvements and modifications to the system as well as incorporation of further languages proved increasingly complex in view of the lack of structure and modularity. Such systems rapidly approached a ceiling of efficiency and adaptability. Examples of this approach are the Wayne State University, the Georgetown University and Logos systems.

The GEORGETOWN UNIVERSITY SYSTEM, the largest in the US for many years, was developed from 1954 till about 1964 and is a forerunner of SYSTRAN. A Russian to English system it was delivered in 1964 to the US Atomic Energy Commission at Oak Ridge and to Euratom in Italy (both were in operation until recently, the system for Euratom being replaced by SYSTRAN in 1976). Despite the early nature of the system it has been reported that over 90% of users of both systems rated the unedited output as acceptable and recommended it for further use (Hutchins 1978). In 1960 the Georgetown project under Leon Dostert consisted of four sub-groups experimenting with syntactic analysis: Zarechnak provided the basic parser and A. F. R. Brown the programming method (known as the Simulated Linguistic Computer or SLC). The syntactic analysis component recognized word classes (e.g. nouns, verbs, adjectives, etc.) and scanned only constituents immediately to the left and the right to resolve ambiguities and deal with homographs (e.g. 'control' as a noun or verb). Semantic analysis did not amount to much, but used features such as +− concrete, +− liquid to collocational ambiguity (e.g. 'the foot of the stairs' v. 'the foot of the mountain'). Analysis of syntax and lexis served to identify word-for-word translation equivalents and to specify the need for some changes in word order in the target language. Typically of such first generation systems, the lack of separation of the components, in particular of the language data from the grammatical rules or even the SL analysis from the TL translation, rendered the system too complex to modify after installation in 1964 (Macdonand 1963, Josselson 1971, Kay 1973, Hutchins 1982).

From 1968, however, SYSTRAN underwent considerable development by LATSEC Inc in the light of second generation systems approaches (see below) and was used in its later versions by NASA during the Apollo-Soyuz project, as well as replacing the early Georgetown System Euratom in 1976 and being adopted by the European Community for trial evaluations of English-French translations. In 1977 the EC assessed the unrevised output as 'far from satisfactory' on the basis of intelligibility scores of 44% to 66% compared with manual translation, the latter higher figure achieved after the system's dictionaries had been updated. Revised output was, however, assessed as economically competitive with manual translation within the framework of a large department employing full-time translators. In statistical terms SYSTRAN output required 26%–31% revision (based on proportion of text), compared with 3%–6% for manual translation: scientific texts, however, required the least revision when output by MT (25.84%) but the most post-editing if produced manually (6.64%) (Chaumier 1977).

The USAF Russian-to-English system, operational since 1970, is a very large software package which is designed to handle random text input (although mainly of a technical nature) with no pre-editing and minimal post-editing. From 1978 a program module COMPUP (comparator) was added in order to monitor and review the effects of transitional changes to the system by comparing output, although difficulties are acknowledged in that improvements in one area can generate degradation in another — the so-called ripple-effect (Bostad 1982).

The WAYNE STATE UNIVERSITY SYSTEM, developed from 1959 to 1972, had a clearer model of linguistic structure, using an extension of Garvin's 'fulcrum' model to represent relationships between constituents in terms of dependancies (in which model one constituent of a phrase is taken to dominate all the others, e.g. a noun for a noun phrase, verb for verb phrase, etc.). However, the system was never able to parse a Russian sentence with more than a single finite verb, for which the straight phrase structure model itself is considered responsible (Garvin 1972).

LOGOS is perhaps the most advanced of the earlier systems. It separates analysis and synthesis, as well as the programming processes, the linguistic rules and the language data, although its model of language is said to be inexplicit. It has been used to translate English aircraft manuals into Vietnamese and is being adapted for English into French, Spanish and German (Bruderer 1977). A German to English version exists on Wang hardware and has been the object of a recent informal evaluation (Ordish 1984). An existing vocabulary of 100000 words is claimed. Pre-editing entails scanning proposed input for words not in the dictionary and entering them accordingly, and entering proper nouns in inverted commas. Although there is an upper limit on sentence length of sixty words, the system is claimed to be able to translate 20000 German words per hour. In general the evaluation concluded that there were surprising lexical 'gaps' in the system's knowledge and that it could not cope realistically with difficult, literary-type texts: with updated dictionary information, however, LOGOS could probably perform adequately with fairly standard texts containing some degree of repetition. In the words of the evaluator: 'A logical system cannot be expected to have a human translator's feel for language. Consequently, for all but the driest and most factual of texts, results are so mediocre that editing is scarcely quicker than translating the text onself in the first place, since problems have been left unsolved. With such texts, then, it is not merely a question of editing the finishing touches and inspired subtleties into a basically sound translation ...'

The Wang version of LOGOS is at present marketed only in West Germany and Switzerland, although an English to German program is already installed in the United States and LOGOS software should ultimately be transferable to other systems. The system is about to be tested alongside SYSTRAN for German to English raw output in a six-months trial instituted by the European Commission. There are, however, already over a dozen installations in the Federal Republic of Germany, most of them very recent and providing highly usable output of technical texts with concomitant increases in the productivity of human translators. Each user must face the initial problems of setting up dictionaries, although this is supposed to be cheaper than for SYSTRAN (for whom trained dictionary decoders are required) and, once done, results in a welcome standardisation of terminology. Each dictionary entry includes up to 250 subject matter codes, five of which may be specified in order of preference for an input text in order to facilitate the selection of correct meanings in context (Lawson 1984).

Interlingual Systems

Second generation systems in the 1960s were more influenced by contemporaneous developments in linguistic theory and on syntactic structure as a basis for language analysis and generation. These systems aimed ultimately at analysing the TL into a universal machine interlingua which would serve as the basis for conversion into the SL. Thus further languages could be added, i.e. programs written to convert them into or from the universal deep structure interlingua representation, without disrupting what the system could do already and without the need to write individual translation programs for each addition SL-TL combination. It was presumed, too, that an existing SL-TL translation direction could be more easily reversed. The notion of a universal interlingua was already suggested by Warren Weaver in his 1949 memorandum. His famous metaphor was that of communication among people living in large towers: 'the way to translate ... is not to attempt the direct route, shouting from tower to tower ... ⟨but⟩ ... to descend, from each language, down to the common base of human communication ... and then re-emerge by whatever particular route is convenient.' These so-called 'indirect' systems also became more strictly modular in structure, separating the dictionaries and grammars for each language, the linguistic data bases and the actual processing programs. Systems implemented on these principles include METALS (Texas) and CETA (Grenoble).

METALS, developed at the University of Texas from 1960 to 1975, translated from German into English, although it was intended to include other languages in time. It began by attempting to adopt Chomsky's transformational grammar as its model of operation, although the programming team discovered that the original form of transformation rules would not work adequately in syntactic analysis by computer. The analysis stage consisted of three parts. Firstly the input sentence is analysed to alternative provisional 'strings' of grammatical categories using the information in the SL dictionary. Next, potential phrase structure trees are constructed for each string, the unacceptable ones being rejected. Thirdly semantic information from the dictionary is used to test collocational acceptability across constituents (e.g. subject-verb coherence), whereupon accepted phrase structures are converted into deep structure representations using an argument-predicate model derived from propositional logic. The METALS interlingua is limited and cannot be considered truly universal: it consists of a set of deep syntactic structures designed to facilitate transfer between specifically English and German, has little or no semantic component and translates at lexical level via a straightforward English-German dictionary. The deep structure representation or so-called 'normal form' is arrived at during the analysis stage by three 'grammars' which successively scan for and set up tentative tree structures, unacceptable trees being filtered out by a sequence of syntactic and, finally, semantic compatibility tests. TL lexical forms are inserted onto the 'normal form', from which surface strings are produced. Problems in the METALS system included the production of too many tentative strings and the heavy reliance on syntax, which meant that, when deriving TL German strings, for example, from a single normal form, it would be difficult to decide between alternative syntactic realisations of the same semantic equivalent. The METALS project demonstrated the lesson that, in the current state of the art, the immediate problems of translating between individual languages overshadowed the ambitious goal of creating a universal interlingua (Grishman 1976, Lehmann and Stachowitz 1972–5).

While it seemed theoretically attractive to some early researchers to use a natural language as the interlingua, this has not emerged as a practical proposition in implemented systems. The EEC, however, has financed the initial stages of an investigation into the feasibility of Esperanto as a pivot language in MT. The principle advantage of Esperanto is alleged to be its lack of ambiguity, although problems are presented by the language's paucity of technical vocabulary and the existence of competing terminology in the various national variants of Esperanto itself. The use of Esperanto can even increase ambiguity, when, for example, relationships between nouns in nominal groups expressed in the SL by syntax or inflection are lost because Esperanto does not possess the necessary grammatical markers. No MT processing has been undertaken yet, although it is planned to implement programs for translating from Esperanto to English (LM 1984, No 5).

Transfer Systems

The next developmental phase was the 'transfer' approach which adapted the notion of the universal interlingua into separate interlingual representations for both SL and TL linked by transfer programs. These deep structure representations are language specific (and hence do not require a universally interlingual degree of abstraction) being designed purely in order to permit the transfer programs to perform successfully on particular language pairs. Despite the fragmentation of the originally proposed single interlingua into separate language-specific ones, the transfer model was claimed to possess similar advantages. Thus only one program of analysis and only one of synthesis is required for each language. Unfortunately this has yet to be established for several languages working in the same system. Examples of transfer systems include TAUM, CETA and SUSY.

CETA, developed on an IBM 7044 at the University of Grenoble between 1961 and 1970, is one of the best known indirect systems in Europe. Designed to translate Russian scientific articles of over 300000 words into French, it is very similar to METALS in overall concept. The linguistic model employed for deep structure representation is a dependency model expressing relationships between lexical items in terms of head constituents and their subordinates, the theoretical basis of which is derived from the work of Tesnieres and Melchuk. CETA exhibited similar problems and characteristics as METALS, in particular the reliance on a syntax-based deep representation produced an excessive number of semantically incoherent parses as well as loss of surface structure information vital for resynthesization. A further complication ensued from the rigidity of the system. A partial or incorrect parse endangered subsequent processing and it proved impossible to translate 'bit by bit' up to or around the point of failure. In general the model of grammar showed itself to be too complex to handle more than simple sentences. (Hutchins 1982, Vauquois 1975).

TAUM is another well-known transfer system. Supported by the Canadian Government at the University of Montreal from the mid-1960's (funding has now been withdrawn), its purpose is to translate technical Air Force manuals from English into French (intended to become operational during the 1980s). The first or analysis stage identifies morphology and base forms of words, to which grammatical information from a dictionary is attached, before moving on to syntactic analysis (using a Fortran version of Q-systems), from which phrase structure constituents in canonical forms are constructed (e.g. passive sentences are made passive and verbs are placed before subjects and objects). Information about the surface string is, however, retained, so that we are not dealing with a pure and highly abstract universal interlingua. During the transfer stage French lexical items are substituted for the English and syntactic structure is altered to conform to the TL. In synthesis surface structures and correct morphology are output. On the whole the system is claimed to be linguistically less complex than CETA or METALS. Once again, however, the model is syntax-based with little semantic analysis to resolve syntactic ambiguity (TAUM 1977, ref. in Hutchins 1978, 1982).

METEO has been translating weather forecasts from English into French since 1976. It is a greatly simplified TAUM system (made feasible by the restricted vocabulary, syntax and the stereotyped nature of meteorological reports): there is no morphological analysis of input text (the forms being so highly standardized) and only one dictionary (for TAUM's three) is used to convert from English into French (base forms and morphological items inclusive). The parsing and synthesis programs were also simplified. 20% of the weather reports are not translated by the system, most of which is put down to unedited input containing typing, spelling and punctuation errors (Chandioux 1976, Chevalier et al 1978, Thouin 1982).

POLA (Project on Linguistic Analysis) was a Chinese to English MT project at the University of California, Berkeley, which was similar to the TAUM transfer system except for special procedures designed to handle the peculiarities of Chinese. Thus, in the absence of such indicators in the SL itself, the program attempted to segment Chinese text into sentences on the basis of punctuation and syntactic markers (prepositions, conjunctions, etc.) and set up word boundaries on the principle of the longest match of a series of characters. Progress was made on automatic recognition of Chinese characters before the project ended in 1975 with an incomplete program for synthesis into English. The grammar, however, consisted of a series of sub-grammars enabling sections to be re-written separately (cf. GETA below) and the programming language itself (GASP) could be adapted to other computers (Wang & Chan 1974, 1975, Wang, Chan & Isou 1973).

GETA was established in 1971 as the child of CETA. With a new computer (an IBM 360/67), however, fresh programs were devised in preference to merely translating CETA (with all its problems) to the new system. The components and functions of GETA may be summarized as follows:

ATEF is a morphological analyser which uses a push-down stack non-deterministically (decisions can therefore be changed and proceed in parallel). Maximally six dictionaries can be called upon alongside a dictionary of idioms and ATEF also has procedures for coping with unknown words. From 1974 a further component, TRANSA, was added to facilitate the processing of unknown proper nouns in Russian (this worked by constructing a new lexical unit on the basis of what remained to be analysed instead of the whole form). Furthermore, changes were effected involving the handling of idioms (search priority was given to the longest) and, in particular, enabling the user to intervene in order to correct unrecognized forms, add words to the dictionary, and prevent the program from halting. The CETA component as such now handles tree diagram transformations for analysis and synthesis. Receiving the output of ATEF it transforms this into a PIVOT language (not a universal interlingua, but an abstract deep structure specific to each language). CETA produces tree structures, altering them according to transformational rules, which can be applied in alternative modes, i.e. the unitary mode (the rules are applied once only) and the exhaustive mode (the rules are applied repeatedly until no more structural descriptions are met); rules can also apply recursively and exhaustively to a portion of the whole tree diagram. The system is also supposed to handle discontinuous idiomatic phrases by associating them with a single lexical unit and point in the tree. In contrast to CETA, which has no access to dictionaries, TRANSFER uses a transfer dictionary to assign directly to points of the source language tree (generated by CETA1) target language nodes which will ultimately point to TL lexical units: e.g.

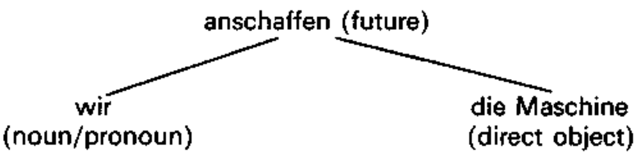
'DRUG'=$SN5J/1(2) / 1:'AUTRE'; 2: 'UN' / / / 'AMI'
(example from Boitet 1977)

In this example N5J is a so-called condition-procedure for the SL lexical unit 'DRUG' (Russian for either 'the one and the other' or 'friend'). 1/2 represents a tree diagram with two points. If the point or node is not derived from the Russian structure for 'one and the other' NJ5 is not verified and TRANSF replaces the source node by a TL node for which the lexical unit will be 'friend'. It is worth noting that resolution of ambiguities during analysis can thus be effected at semantic level by reference to dictionary information.

TRANSFER itself is a further component which superseded CETA in 1976. CETA2 applies transformations after the TRANSFER process to produce a pivot form: each SL has its own pivot representation. At this point SYGMOR (ready from 1976) acts upon a labelled tree diagram output from TRANSFER/CETA2 to 'flatten' it into a linear string of variables or combination of variables which is finally transformed into a surface string of characters with the aid of external data drawn from variable values and conditions, dictionaries and a grammar. SYGMOR is a finite-state deterministic system reflecting the lesser degree of complexity at this stage of synthesis.

GETA is especially interesting for the levels if its linguistic analysis and the flexibility of the group's approach. The linguistic levels, for instance, are separated into the grammatical (concerned with the formal properties of units), the conceptual (the formalized relationships between language units which result from the universe of reference) and the pragmatic (communication in situation). It is presumably the intention of the group to incorporate and semantic and referential and possibly even situative information into their system. As far as the flexibility of the system is concerned, the project has for the time being rejected powerful and complex systems such as Q-systems and ATN parsers in favour of simple manipulation of strings in morphological analysis and synthesis. Furthermore, the linguist is able to construct his own grammars of T-rules and their conditions for tree-transformations: different linguistic models and so-called sub-grammars can furthermore be specified for different parts of the process. It is even felt that non-transformational models (e.g. Wilks' semantics-based analyser) could be implemented here. The overall long-term aim of the GETA group is a third generation multilingual translation system. Languages which have been under study include English, Portuguese, and Japanese. (Boitet 1977, Chaucé 1975).

The Saarbrücken Automatic Translation System (SUSY) originated in 1967 as research into a Russian-to-German system, with projects involving French, English and Esperanto into German started during the 1970s. The translation system's components are as follows. The LESEN operator or module reads the text to be translated, assigning words identification and serial numbers within the sentence. This program can operate on any language input. WOBUSU, which is language-specific, consults root dictionaries and a word form index (containing mainly functional words and covering about 50% of the text) to break down the output from LESEN into basic forms and grammatical information by analysing inflections. The strategies vary according to the language, but in all cases fixed sequences of word forms can be grouped into a single unit. The LEMMAT operator reduces syntactic word class ambiguities to a minimum by contextual analysis, although it does not aim to invest excessive programming time in a complete resolution of ambiguities. Although LEMMAT draws on language-dependent sub-programs the intention is to develop algorithms which are independent of grammar (so-called control systems). SEGMENT reduces complex sentences to co-ordinate and subordinate units and draws together discontinuous structures. SEGMENT'S algorithm and grammar are separate. The NOVERA operator establishes the structure of noun phrases. Interestingly it uses a dependency model to describe the sequence and subordination of elements, with adjectives understood as predicates governing noun phrases:



='die von uns anzuschaffende Maschine'
(Maas 1977)

Not only does such a structure stand for a variety of syntactic equivalents —

(a) die von uns anzuschaffende Maschine
(b) die Maschine, die wir anschaffen werden (sollen, müssen)
(c) die Maschine, die von uns anzuschaffen ist
(d) die Maschine, von uns anzuschaffen (?)

— but it is claimed to be easy to generate alternative TL structures in accordance with the possibilities of that language. Incidentally, NOVERA also produces an inventory of the verbal elements of a subsentence. The SYNAN operator has a number of complex functions. These include the classification of noun groups or subsentences under a verbal node with an indication of their relationships (subject, object, complement, etc.), analysis of surface verbal elements, establishment of deep structure verbal and noun components, and the resolution of pronoun reference. The recovery of information on cancelled constituents is admitted to be a problem here, as is third person pronoun reference. A deep structure having been set up, the TRANSFER operator maps onto it target language equivalents drawn from a lexicon: some semantic information is available to inform the selection of equivalents and SL case is changed to TL case, although no further structural alterations take place. The SYNTHESE program converts by means of TL transformational rules the deep structure output of TRANSFER into a linear surface structure translation (Mass 1977, see also Schirmer 1981).

Interactive Systems

Acknowledged machine assisted systems (otherwise known as limited language or interactive systems) represent a fairly recent development. In place of the goal of total automation of the translation process with free text input (yet to be realized) MATs rely on human intervention at some stage and incorporate this into the general design concept. Intervention can occur in the form of pre-editing of text, consultation between computer and human operator during the running of the programs, or post-editing of the translated output.

Current examples of machine assisted translation systems with in-built interaction include: TITUS, CULT, ALPS, WEIDNER, MIND, SLUNT.

TITUS, developed from 1972 and operating since 1974, is used by the French, Institute of Textiles to translate in this field from and into English, French, German and Spanish. The system relies on pre-editing, in the form of abstracts written in a so-called 'canonical documentation language', i.e. a sub-language of limited syntactic and semantic complexity. The translation is performed with the use of a simple interlingua. (Ducrot 1973).

CULT, researched from 1969 and operating from 1975 at the Hong Kong Chinese University, is a direct translation system used for converting Chinese mathematical and physics texts into English. Human intervention occurs at two stages: viz. during analysis of input text, when the user is consulted in order to resolve ambiguities and homographs, and during the synthesis process for the determination of TL articles, verb tenses and moods.

In more detail, the system consists of a main program which summons a series of subroutines. Firstly, a subroutine scans the sentence and consults a dictionary to establish words from series of characters on the 'largest match' principle. A second subroutine derives some functional information for each word from the dictionary and assigns unique grammatical data, e.g. identifies nouns and verbs. After checking for punctuation, distinguishing interrogative and affirmative sentences and locating prepositional phrases, each word of which is translated at this stage, the verb is established. During further syntactic analysis, translation is combined and intermixed with the analysis itself, so that, for example, conjunctions at the beginning of the sentence are immediately translated, as are subjects, before the verb or predicate phrase are completely analysed. If an error occurs at any stage of the translation, an error routine is called, information pertaining to the error is displayed alongside grammatical data available, and the program moves on to another sentence. Difficulties are acknowledged, not with the translation itself, but with the lack of adequate terminology available for scientific terms. However, the translation speed is about 2–3 sentences per CPU second, and the developers are studying the application of the system to other subject areas and languages (Loh Kong 1977, Loh 1976).

The Automated Language Processing System (ALPS, developed initially at Brigham Young University) translates from English to French, German, Spanish and Portuguese. An interlingual system, it relies primarily on human assistance during analysis into deep structure representations (the model used is so-called 'junction grammar', a variation of transformational grammar pioneered by Eldon Lytle of the Translation Sciences Institute at Brigham Young). It has recently become commercially available and provides a good example of a system whose conception shifted during the 1970s from the goal of fully automatic production of high quality output to real-time interactive processing as a cost-effective aid to the human translator (Kingscott 1984).

The WEIDNER machine aided translation system is one of the most recent interactive systems to enter the commercial market. It was purchased by the MITEL Corporation (Canada) in 1980–1, who have provided an interesting recent assessment (Hundt 1982). MITEL purchased the system to translate and update manuals on telephone switch systems from English into French, German and Spanish. Key components in the Weidner system are VOCABULARY SEARCH and DICTIONARY UPDATE. The former establishes words in the input text which the computer does not recognize and the latter enables them to be entered into the machine's lexicon. Glossaries on specific areas can thus be effectively built up. For each new word DICTIONARY UPDATE asks a series of questions to establish grammatical information associated with the word (e.g. is the word a noun, verb, adjective, etc., if a noun, what gender is it and is it a noun of time or place, etc., or if a verb, is it reflexive, what kinds of object does it take, what prefixes are associated with it, and so on). The procedure for entering idioms is especially interesting. The computer asks what each translated word in the idiom is, requesting the user to specify (a) elements which do inflect as opposed to those which do not (e.g. in 'es regnet in Strömen', 'Strömen' will never change, and (b) a key word in the idiom (preferably the least frequent in order to speed up recognition time), which will be used to check for the occurrence of the whole phrase in a source text. Phrases arise with phrases which may or may not be ambiguous (e.g. 'to ring a bell'). The solution here is either to assume that, in the technical context, the phrase will or will not be an idiom (bells tend to ring literally in telephone companies) or that the idiom will be most probably betrayed by the occurrence of a single, specified, key word (e.g. 'name' in 'that name rings a bell'). During the actual translation process, the user can follow the machine's progress by summoning the TRANSLATION PROCESS MONITOR. Mitel have found that the system is economic. Once the dictionary is entered draft translations are produced and can be post-edited more rapidly than if a human translator had had to undertake the whole process himself.

MIND, developed from the early 1970s, is a transfer system designed for research purposes rather than practical implementation. After what is alleged to be a powerful parsing stage of analysis, by which a deep structure representation of morphology and syntax is reached, the human user is consulted to assist in the resolution of ambiguities and homographs. This is achieved by means of a dialogue between the user and a component called the SYNTACTIC DISAMBIGUATOR. Thus, in order to decide the structure of 'They filled the tank with gas', the computer could ask the following questions:

DOES THE WORD 'TANK' REFER TO

1. A MILITARY VEHICLE?
2. A VESSEL FOR FLUIDS?

Similar questions would establish the referent for the pronoun 'they' on the basis of the nouns in the immediate context.

SLUNT (Spoken Languages Universal Numeric Translation), an MT package written in a subset of COBOL, designed for simple English to German sentences and stored at the University of Essex although a considerable interest abroad is claimed, is a restricted system in which all texts are pre-edited, ambiguities are resolved, idioms re-written in a form acceptable to the computer, and unusual and complex sentences are simplified: the program rejects text it cannot handle. In more detail, the program converts input to an INTERMEDIATE LANGUAGE or NUMBER LANGUAGE (NL). The NL is claimed to be entirely unambiguous, consisting wholly of numerals. The NL version of the SL message becomes the official version, and it is hoped that all languages may be defined accurately in terms of NL numbers, which does not presuppose that there is one-to-one correspondence between NLs and individual words or between words across different languages: thus an NL number may represent one word in one language but require several words to define it in another: similarly a single word in English may need several NL numbers to express multiple meanings. Accurate dictionary definitions are vital to the system and the programmer is expected to decipher messages by referring to a SLUNT lexicon of NL numbers for his language in questions of doubt. (Goshawke 1977).

One of the largest MT systems under development is EUROTRA, which is intended to handle simultaneously various languages of the European Community. Planning originated in 1978 under the auspices of the European Commission, who brought together groups of experts now numbering about eighty. The design concept reflects the current state of the art, employing a transfer system, distinct modules and strictly separating the language data and the computing processes. It will accept data from other systems (e.g. dictionaries from SYSTRAN) and will also be compatible with future systems. Source language analysis is comprehensive or 'deep' and does not exploit chance similarities between specific language pairs, thereby facilitating the transfer process to future TLs. A far as the modules are concerned, the transfer modules are intended to be as small as possible, with monolingual SL generation modules at the synthesis stage taking over a lot of work hitherto done by the transfer modules. Original to the Eurotra system are the INTERFACE STRUCTURES which are designed to facilitate communication between analysis and transfer and between transfer and generation. Essentially, monolingual analysis creates a phrase structure tree expressing dependency relationships, i.e. what constituents dominate others, but no precise further information about the nature of the relationships. This is supplied by the interface structure and can take four forms:

(i) Surface syntactic function: e.g. whether a noun functions as subject or object at either the 'deep' or 'surface' level (as in active/passive structures).
(ii) Semantic relations not accessible by syntactic structure: e.g. in 'he built the boat with care' and 'he built the boat with wood' we need semantic information to distinguish the manner phrase ('with care') from the source phrase ('with wood') in order to produce a correct translation.
(iii) Information on valency: e.g. to distinguish the degree of closeness of constituents to the predicate (in 'George ate his lunch quickly on Friday', the elements 'quickly' and 'on Friday' are less strongly bound to the verb phrase than 'his lunch').
(iv) Various morphological and morpho-syntactic information: e.g. expressing person, number, verb tense, etc.

Although (iv) represents the lower limit of expressible information, there is supposedly no upper limit, so that definiteness, emphasis and focus are categories which can all be mapped onto the phrase structure tree. Ideally the transfer module merely receives the output of the interface structure and replaces SL lexical units with TL elements. The rest of the interface structure is unaltered and can proceed to the generation stage.

A further feature of Eurotra's design is that a single and uniform data structure is used to represent the intermediate results during the processing and throughout the system: it is kept separate from the linguistic rules and hence permits different linguistic strategies to be used for the description of different languages. The common data structure also facilitates the retention of possible alternative structures for the same string until a final decision is reached. The linguistic rules are conceived to be as flexible as possible. Typically they consist of two parts, the first of which specifies a data state to be looked for (this may be a partial result) and the second the change of modification to the data structure to be carried out. There is no restriction to the type of information to which the rule applies (it can be morphological or semantic feature or a high level dependency structure, etc.) and there is no inherent order in which information must be looked for and acted upon: thus semantic analysis can be implemented before morphological or syntactic analysis.

There are linguistic problems which EUROTRA cannot handle, for instance, cross clause pronoun reference (as for 'they' in 'The town councillors refused a permit to the women because they feared violence/because they advocated revolution' — the reference of 'they' to either women or councillors may depend on the cultural situation of the utterance). Postediting is required and attention is being given to incorporating EUROTRA into a carefully managed scheme of text-processing where the MT component will relieve the human translator of many routine tasks. Further developments to EUROTRA involve the use of different sub-grammars for distinct text types in place of a general all-purpose grammar for all text inputs. It may also be possible to adapt the system to take account of the purpose to which text output will be put, although it is acknowledged that some texts will probably always be unsuitable for MT, notably deliberately ambiguous texts and legal documents in which the inferences of the SL and TL must be equivalent and the translation must have equal status with the SL original (King 1982).

Microcomputers

The rapid development of microcomputer technology has not been without consequences for MT although the immediate response of the developers seems to have been to convert existing mainframe systems to operation on micro- or minicomputers.

The Weidner system, for instance, has been transferred from a DEC computer to the ICL micro for potential public use and was also launched commercially in June 1984 under the name of MicroCAT for use on an enhanced IBM PC XT Personal Computer. The Weidner software offered is currently for English to French, Spanish, German or Portuguese, and French to English and Spanish to English (a Japanese to English version is also in the offing). Each language pair has a core dictionary of about 10000 words and idioms, with facilities for updating, deleting or adding new terms and for setting up bilingual glossaries for specific purposes. Raw translation, it is claimed, is produced at rates of over 1600 words an hour, which can be post-edited on a split screen displaying the source text and the raw translation in parallel. Analysis is at sentence level and idioms or phrases of up to nine words can be dealt with, as can a maximum of six alternative meanings. No pre-editing is required and it is claimed that a trained translator can process (i.e. post-edit) between 600 and 1000 words an hour depending on the completeness of the dictionary, the style of the original and his own skill. As for cost, the hardware (an enhanced IBM PC XT with 10 megabyte hard disk version with 640 K RAM) is currently priced at £5244 and the software from 8000 for the first language pair with reductions for each additional pair. The developers of SYSTRAN, moreover, claim that within two years their MT system will be available on both large microcomputers equivalent to the IBM XT and smaller domestic machines. The 'large' version, TEXTUS 1, will have a basic dictionary of 35000 entries and the option of add-on specialized dictionaries of up to 20000 entries. The smaller edition, TEXTUS 2, is claimed to be inexpensive with a maximum dictionary size of 20000 entries. SYSTRAN will probably continue to dominate the field with its versatile large scale system for some time to come.

The Future: Semantics and Artificial Intelligence

MT systems hitherto (and largely those described above) have been syntax-orientated. A component of semantic analysis, where present, has operated on already generated syntactic structures, usually in order to resolve an ambiguous or otherwise imperfect parse (Kay 1973, Hutchins 1976).

It is possible that fully automated MT systems which rely on a primary syntactic component will never be completely successful. Apart from Riesbeck, whose semantics-based expectation parser ELI builds structure on the principle of expected meanings of input words and phrases (Riesbeck 1975, Riesbeck & Schank 1976, Gershmann 1977; also Tennant 1981), Yorick Wilks has experimented with a semantic analyzer which bypasses syntax altogether and is able to handle phenomena, such as reference across sentences and unusual usages, which present considerable problems for the syntactic analyzer alone. Wilks' system, however, has not been tested on a large scale. The basis of Wilks' system is a PREFERENCE SEMANTICS model in which dictionaries do not specify obligatory semantic features but preferred ones (e.g. 'drink' prefers, but does not insist on animate subjects). So-called COMMON SENSE INFERENCES are also used to link pronouns with their antecedents. In more detail, Wilks' suggestion for an English-French system involved the following processes: The first stage, called FRAGMENTATION, divides the text up at punctuation points and keywords (e.g. conjunctions and prepositions). Each fragment is tested against a series of TEMPLATES, which consist of semantic frames representing the general meaning of sentence fragments in the form of triples of semantic features. Thus the template MAN HAVE THING would be matched onto a sentence or sentence part such as 'John owns a car'. The items MAN, HAVE, THING, are semantic primitives which are found to be the main semantic categories (or 'heads') in the semantic formulae underlying the elements John, own and car. A semantic formula is built up on the basis of a finite number of features (or 'elements') hierarchically ordered by brackets: e.g.

((*ANI SUBJ)((FLOW, STUFF)(OBJE)(*ANI IN)((-THIS(*ANI(THRU PART)))TO)(BE CAUSE))))) —

Which means that an action, preferably done by human beings (*ANI SUBJ) to liquids (FLOW STUFF OBJE), of causing the liquid to be in the animate thing (ANI IN) and via (TO) a particular aperture of the animal thing (THRU PART), i.e. the mouth. Preferential semantics entails the selection of preferred (but not exclusive) subjects and objects and an analysis which need not bother to distinguish the mouth aperture from other apertures. Items not included in the templates are then investigated to see if they can be incorporated into a dependency relation with elements already defined. Thus, in 'John gave Mary the book', 'Mary' would be attached to 'gave' and 'the' to 'book' in the template 'John — gave — book'. Dependency ties are searched for both within and across sentence boundaries, so that the fragment itself (the 'phrase') is the unit of analysis, not the sentence. Common sense inference rules account for ties such as the assignment of 'them' to 'women' in 'the soldiers fired at the women and we saw several of them fall': the system infers from 'hit' that what it affects is likely to fall. The structure in Wilks' model is claimed to be purely semantic — there are no nouns, verbs or other syntactic categories although the system in practice performs a preliminary segmentation of basic units of input text which could be regarded as the equivalent of a shallow syntactic parse. In the example, 'Small men sometimes father big sons', the homograph father is assigned a (semantic) verbal sense with the feature CAUSE as its head as a result of a match with other head features in a template (see Wilks 1972 and subsequent work).

A semantics-based MT system (ATLAS) is also under development in Japan. A full scale English-Japanese mainframe system (ATLAS I) is about to be launched, capable of translating at 20000 words per hour, and a more advanced system, ATLAS II, is underway.

For non-MT systems which, to a greater or lesser extent, rely on semantic processing see SHRLDU (Winograd 1972), Woods' AIRLINE GUIDE (Woods 1968) and LUNAR (Woods 1973), ROBOT (later INTELLECT: Harris 1977), LIFER (Hendrix 1977), LSP (Linguistic String Project: Sager 1967 onwards), PLANES (Waltz 1975, Waltz Goddman 1977), and RENDEZVOUS (Codd 1974, 1978).

Recent work in artificial intelligence is directed towards the representation of semantic knowledge in computer memory. A number of natural language programs (not specifically MT) have used the concept of picture stereotypes (variously called frames, scripts, schemas or methods) to store information about objects or even events which are used as referential schemes to recognize and process input information for analysis. It is a reasonable assumption that future MT systems will be incorporating strategies of this sort (see Tennant 1981 for an overview of systems such as KLONE, FRL, GUS and SAMPAM).

Using such a model devised by SCHANK (Schank 1975, Schank and Abelson 1975, Schank 1977) CARBONELL (YALE UNIVERSITY) has developed a simple MT system which uses internal representations of basic situations (such as what normally happens in a car accident) to analyze a simple English text and extrapolate from it a representation of the contents which serves as input to a Russian or Spanish text generating component. The transfer and synthesis processes, however, are similar to those used in some of the more recent systems described above.

In more detail, conceptual frames may organize information and be inter-linked in a variety of ways. A frame, for example, of a motor car may be rather vague but it will contain the information that the default image of a car consists of a structure with four wheels, and engine, fuel-driven, etc. The FRAME usually consists of groups of SLOTS, each bearing an aspect of description of the concept which the frame represents. Thus for the frame CAR, there are slots for a manufacturer, engine size, date of manufacture, owner, and so on. The SLOTS can carry default values which are assumed unless contrary information is given. Slots may also have PROCEDURES attached which may be AUTOMATIC (so-called DEMONS, which automatically check to establish a valid value for a slot, for example, to verify that a manufacturer for a car is given or that a given manufacturer is one of a known list) or executed on demand (so-called SERVANTS, which may only be invoked by another procedure or if a previous SERVANT failed to supply a piece of information, for example, in order to deduce from either the context of information already given or even from the user himself the particular value for a slot). Values for slots are often pointers to other frames representing other concepts. Frames can be linked in terms of subset/superset, generality/specificity. Thus the frame for CAR has a superset pointing to the more general frame VEHICLE. This may involve specialization mapping (transferring selected features of one frame onto another, implying that all features of the more general concept are relevant to the first) or distinguishing generic (prototype) from instance frames (thus a particular object CAR3 is identifiable and describable by comparison with the class frame CAR). All frames within a system need not, however, be linked through a single hierarchical structure for which there is, incidentally, little psychological justification. One system (KRL) distinguishes basic frames (descriptions of mutually exclusive sets of things), abstract frames (e.g. a journey, travelling or a family) and specialization frames which may refine either of the previous frames (e.g. TRAVELLER is a specialization frame of the basic frame PERSON). The user may even define frames as he wishes controlling which concepts are abstract, basic, etc. Basic frames can limit searching times as mutually exclusive frames need not be searched completely and there is also provision for partial descriptions in the form of so-called manifestation frames (e.g. an individual whose identity is not known). KRL can place limits on search time in concept comparison and on the depth of analysis.

The Script Applier Mechanism (SAM) by Schank and Abelson (1977) encodes a 'knowledge' of particular events in the form of scripts, or prototype descriptions of what constitutes these events. The program scans a text to infer from an analysis of the words what script event the text refers to. It is thus able to infer an event from partial information. Scripts are activated if a set of key references to a situation is detected and deactivated by reference to likely exit points in the situation (e.g. in the RESTAURANT event by the guests leaving or the finish of the meal). The Plan Applier Mechanism (PAM) is an extension to SAM which is designed to handle situations which SAM has failed to understand. It achieves this by inferring the 'goals' and 'plans' of actors in the events which are held in standardized form in the memory and combining them to generate (postulate?) paraphrases and summaries of text input.

An example may throw light on this process. If the computer is confronted with the phrase 'gas-powered car' in a text, it can only 'understand' the phrase by being able to confirm that the phrase conforms to a FRAME within its memory containing information or details (= slots with values, etc.) which match what the phrase is about (obviously this is a limited, mechanized understanding which presupposes no similarity with human thought processes). It will do this by being able to activate the key FUEL slot in the frame CAR through the word 'gas' (the information that 'gas' is a form of fuel may be achieved by a SERVANT procedure as described above which will scan the text for one of the potential matching values of the FUEL slot for the frame CAR). Now if the computer is programmed to assume that it has a car on its hands on the basis of the FUEL slot alone it may have problems with phrases such as 'gas powered plane, bus, jet, truck', and so on unless it has frames distinguishing these vehicles.

Complex representations can lead to involved concept comparison, for example, where the program is operating on a PATTERN and a CANDIDATE. The program has to decide whether the candidate matches the pattern exactly, whether it is equivalent to the pattern, or what subconcepts of the candidate have to match before the candidate can be deemed equivalent to the pattern. In other words, if a match is to be made, how would constituents of the candidate match onto constituents of the pattern? For example, consider the pattern 'green Buick' and the candidate 'green Buick'. Obviously we have a perfect match. But if the pattern is 'green car' then the candidate 'green Buick' is not an identical match unless 'Buick' is known to the program to be a subset of car. If the pattern is 'green vehicle' then the deduction process becomes even more complicated for the machine and the system is faced with the prospect of combinatorially explosive searches through large masses of data. The computer may also have to decide between EXTENSION and INTENSION. The former refers to concepts which are members of a class and the latter to what it means to be a member of that class. Thus the extensional features of a car are its component parts (engine, wheels, brakes, etc.), whereas the intensional description is the structural information whereby it is known when the components interrelate or are held together to constitute the total entity 'car'. In practice this means that a pile of brakes, wheels and cogs do not necessarily constitute a car and that a computer must distinguish a possible heap of scrap from a limousine (see Tennant 1981 for a review of such problems).

The construction of intelligent analyzers faces a variety of problems, some of which are linguistic and others more to do with knowledge of the world or at least of the immediate non-linguistic context. The following examples serve to illustrate the range of difficulties involved:

(1) The Ford Fiesta is quite a large small car.

(2) A round square is a puzzling idea.

(3) It's hard to describe but I'll try anyway. It's a big house with a lot of wings ...

(4) I've run out of ideas (for recipes) — I'll have to grill my aunt the next time I see her.

(5) Everyone misses this turn/Can you please pass the salt.

(6) My uncle ran the London marathon.

(Examples from Christie 1980, Riesbeck 1983, Wilks 1983).

In (1) and (2) the phrases 'large small car' and 'a round square' could be rejected by a machine unless it is able to decide when to disable local semantic checking procedures. If (3) is an answer to a request for directions, the referent of the first pronoun 'it' (unlike the second 'it') is not the object of destination but the route to that object. (4) Could produce comical results if the machine draws obvious inferences between 'recipes' and 'grill' (the example is claimed to be an authentic utterance with no humour intended). The examples under (5) raise the interesting question of the variety of ways in which a language expresses warnings and requests, etc. 'by apparently referential devices'. Finally 'ran' in (6) can mean compete or organize, depending, perhaps, on what we know already about the uncle and his interests.

As may be expected, the gap between research and implemented systems of MT is still considerable and current fully implemented and commercially available systems such as SYSTRAN and LOGOS owe much, in fact, to earlier stages of development. An interesting corollary to this is the question of how far current systems are able to become both more efficient and more innovative. Systems which rely on pre-edited or streamlined text input may be especially vulnerable to the danger of running into the blind alley of becoming less and less susceptible to improvement and adaptation along more open-ended linguistic lines as they grow more efficient at processing texts of a restricted nature (see Masterson 1984 for a discussion of this issue). A conclusion on this, however, will have to await the results of the further development of existing systems and, furthermore, the practical realization of the potential of semantics-based approaches. Given the scale of hardware and software involved it was also inevitable that investment in MT systems remained the province of large commercial or governmental organizations (which is reflected, too, in the concentration on the major languages of international commerce and technology) although there are signs that some of the larger personal computers will be able to accommodate at least scaled-down MT systems. MT is, however, certainly here to stay and has become 'of age' and, as a tool for human use, may well be considered to possess neither more nor fewer limitations than computer assisted language learning programs and packages in the current state of the art. That this is indeed so is reflected in the interest in the economical viability of MT and in the growing disciplines of pre- and post-editing. Not just the theory, but the practical experience of MT and the evaluation of its output is now the object of publications and conferences. It may also be worth noting that it is the conversion from typewriters to general word processing and computerized text handling techniques accompanying the installation of an MT system which, as much as the MT component on its own, is bringing about a quiet revolution in commercial translation that will certainly outlive the success or failure of MT itself.

Bibliographies

Bar-Hillel, Y., 'The Present State of Automatic Translation of Languages,' in *Advances in Computers*, I (1960), 91-163.

Boguraev, B. K., 'Recognising Conjunctions within the ATN Framework,' in Jones and Wilks 1983, 39-45.

Boitet, C., 'Un essai de réponse a quelques questions theoriques et pratiqués liées à la traduction automatique. Definition d'un système prototype.' *Thèse d'etat*, Grenoble 1976.

Boitet, C., 'Où en est le GETA debut 1977?', in T.A. Informations, XVIII (1977), 3-20.

Booth, A. D., *Machine Translation*, Amsterdam, 1967.

Bostad, D. A., 'Quality Control Procedures in Modification of the Air Force Russian — English MT System,' in Lawson 1982, 129-136.

Briscoe, E. J., 'Determinism and its Implementation in PARSIFAL,' in Jones and Wilks 1983, 61-68.

Bruderer, H. E., 'The Present State of Machine and Machine-assisted Translation,' in *Overcoming the Language Barrier*, 1977, 529ff.

Bruderer, H. E., *Handbuch der maschinellen und maschinenunterstützten Sprachenübersetzung*, Munich 1978.

Chandioux, J., 'An Operational System for the Translation of Public Weather Forecasts,' in *FBIS Seminar on Machine Translation. American Journal of Computational Linguistics*, XLVI (1975), 27-36.

Chandioux, J., 'METEO: un système opérational pour la traduction automatique des bulletins météorologiques destinés au grand publique.' in *META* XXI (1976), 127–133.

Chauce, J., 'The ATEF and CETA systems,' in *American Journal of Computational Linguistics* XLVI (1975), 21–39.

Chevalier, M., Dansereaux, J. and Poulin, G., *TAUM-METEO — description du système Janvier 1978*, Montreal 1978.

Codd, E. F. et al, 'Rendezvous Version 1: An Experimental English-Language Query Formulation System for Casual Users,' *IBM Research Laboratory Report*, California 1978.

Colmerauer, A., 'Les Systèmes Q,' in TAUM 71, 1973, 1–44. *Commission of the European Communities: State of the Art on Multilingual Activities in the Field of Scientific and Technical Information*, 2 Volumes, Brussels 1976.

De Roeck, A., 'An Underview of Parsing,' in King 1983.

Ducrot, J. M., 'Le System TITUS II,' in *A.N.R.T. Information et Documentation*, 4th October 1973, 3–40.

Eggers, H., *Elektronische Satzanalyse der deutschen Gegenwartssprache, Verfahren zur Analyse beliebiger deutscher Sätze, entwickelt 1965–68 Saarbrücken*, Tübingen 1969.

Garvin, P. L., *On Machine Translation: Selected Papers*, The Hague 1972.

Gazdar, G., 'NLs, CFLs and CF-PSGs,' in Jones and Wilks 1983, 81–93.

Gershmann, A. V., 'Conceptual Analysis of Noun Groups in English,' in *Proceedings of the International Joint Conference on Artificial Intelligence*, Cambridge Mass. 1977.

Grishman, R., 'A Survey of Syntactic Analysis Procedures for Natural Language,' in *American Journal of Computational Linguistics*, microfiche XLVII (1976).

Harper, K. E., 'Machine Translation,' in *Current Trends in Linguistics, Vol 1: Soviet and East European Linguistics*, The Hague 1963, 133–142.

Harris, L. R., 'User-Oriented Data-Base Query with the ROBOT Natural Language Query System,' in *International Journal of Man-Machine Studies*, IX (1977), 697–713.

Hendrix, G. G., 'Human Engineering for Applied Natural Language Processing,' in *Proceedings of the International Joint Conference on Artificial Intelligence*, Cambridge Mass., 1977.

Herzog, R., *Computer in der Übersetzungswissenschaft*, Frankfurt am Main 1981.

Hundt, M. G., 'Working with the Weidner Machine-aided Translation System,' in Lawson 1982, 45–52.

Hutchins, W. J., 'Machine Translation and Machine-aided Translation,' in *Journal of Documentation* XXXIV (1978), 111–159.

Hutchins, W. J., 'The Evolution of Machine Translation Systems,' in Lawson 1982, 21–38.

Hutchins, W. J., 'Linguistic Models in Machine Translation,' *University of East Anglia Papers in Linguistics*, IX (1979), 29–52.

Johnson, R. L., 'Parsing — an MT perspective,' in Jones and Wilks 1983, 32–38.

Jones, K. S. and Wilks, Y., (eds), *Automatic Natural Language Parsing*, Chichester 1983.

Josselson, H. H., 'Automatic Translation of Language since 1960: A Linguist's View,' in *Advances in Computers*, XI (1971), 2–59.

Kaplan, R. M., *The MIND System: a Grammar Rule Language*, Santa Monica 1970.

Kay, M., 'Automatic Translation of Natural Languages,' in *Daedelus*, CII 1973, 217–229.

Kay, M., 'The MIND System,' in R. Rustin (ed.), *Natural Language Processing*, New York 1973, 155–188.

King, M., 'Eurotra: an Attempt to Achieve Multilingual MT' in Lawson 1982, 139–147.

King, M. (ed.), *Parsing Natural Language*, London 1983.

King, M., 'Transformational Parsing,' in King 1983, 19–34.

Kingscott, G., 'ALPS Moves to Improve European Sales for Computer-aided Translation,' in *Language Monthly*, XIV (1984), 27–29.

Knowles, F. E., 'Soviet Union Holds Major MT Conference,' in *Language Monthly*, VI (March 1984), 12–13.

LM = *Language Monthly*, published Nottingham and London 1983.

Lawson, V. (ed.), *Practical Experience of Machine Translation: Proceedings of a Conference — London 1981*, Amsterdam 1982.

Lawson, V., 'Users of Machine Translation System Report Increased Output,' in *Language Monthly*, XI (1984), 6–9.

Lehmann, W. P. and Stachowitz, R., *Development of English-German Machine Translation System, Final (Annual) Report(s)*, Austin, University of Texas, Linguistics Research Center (1972–1975).

Locke, W. N., 'Machine Translation', *Encyclopedia of Library and Information Science*, Vol. XVI, New York 1975, 414–444.

Loh, S. C., 'Machine Translation: Past, Present and Future,' *ALLC Bulletin*,IV (1976), 105–109.

Loh, S. C. and Kong, L., 'Computer Translation of Chinese Scientific Journals,' in *Overcoming the Language Barrier*, 1977, 631–645.

Maas, H. D., 'The Saarbrücken Automatic Translation System,' in *Overcoming the Language Barrier* 1977, 585–592.

Macdonald, R. R., *Georgetown University Machine Translation Research Project, General Report 1952–1963*, Washington DC 1963.

Masterman, M., 'The Limits of Innovation in Machine Translation,' in Lawson 1981.

Ordish, R., 'Personal Impressions of a Machine Translation System,' in *The Incorporated Linguist*, XXIII, 3 (1984), 136–145.

Overcoming the Language Barrier. Third European Congress on Information Systems and Networks in Luxemburg, May 1977, Munich 1977–1978.

Pulman, S. G., 'Generalised Phrase Structure Grammar, Earley's Algorithm and the Minimisation of Recursion,' in Jones and Wilks 1983, 117–131.

Riesbeck, C. K., 'Computational Understanding: Analysis of Sentences and Context,' *Ph.D. Thesis*, Stanford University 1974.

Riesbeck, C. K., 'Conceptual Analysis,' in R. C. Schank, *Conceptual Information Processing*, 1975, 83–156.

Riesbeck, C. K., 'Some Problems for Conceptual Analysers', in Jones and Wilks 1983, 178–181.

Riesbeck, C. and Schank, R. C., 'Comprehension by Computer: Expectation-Based Analysis of Sentences in Context,' *Research Report 78*, Yale University 1976.

Ritchie, G., 'Semantics in Parsing,' in King 1983, 199–218.

Roberts, A. H. and Zarechnak, M., 'Mechanical Translation,' in *Current Trends in Linguistics*, XII: *Linguistics and Adjacent Arts and Sciences*, Part 4, The Hague, 1974, 2825–2868.

Rolling, L. N., 'The Second Birth of Machine Translation', paper given at the 7th Cranfield International Conference on Mechanised Information Storage and Retrieval Systems, July 1979.

Sager, N., 'Syntactic Analysis of Natural Language,' in *Advances in Computers*, VIII (1978).

Sager, N., 'Language Information Formatting: The Automatic Conversion of Texts to a Structural Database,' in *Advances in Computers*, XVII (1978), 89–163.

Sampson, G., 'Deterministic Parsing,' in King 1983, 91–116.

Sampson, G., 'Context-free Parsing and the Adequacy of Context-free Parsing Grammars,' in King 1983, 151–170.

Schank, R. C., *Conceptual Information Processing*, Amsterdam 1975.

Schank, R. C. and Abelson, R., 'Scripts, Plans and Knowledge, in *Advance Papers of the Fourth International Joint Conference on Artificial Intelligence*, Cambridge, Mass. 1975.

Schank, R., *Plans, Goals and Understanding*, New Jersey 1977.

Schirmer, B., 'Erfahrungen bei der Entwickling eines maschinellen Übersetzungssystems (English-Deutsch),' in R. Herzog 1981, 25–45.

Sigurd, B., 'Machine Translation — State of the Art,' in L. Grahs (ed.), *Theory and Practice of Translation, Proceedings of the Nobel Symposium 39 in Stockholm 1976*, Stockholm 1978.

Tennant, H., *Natural Language Processing*, New York 1981.

Thouin, B., 'The Meteo System,' in Lawson 1982, 39–44.

Toma, P., 'SYSTRAN as a Multilingual Machine Translation System,' in *Overcoming the Language Barrier*, 1977.

Toma, P., 'SYSTRAN: ein maschinelles Übersetzungssystem der 3. Generation,' in *Sprache und Datenverarbeitung,* IX (1977), 38–46.

Van Slype, G. and Pigott, I., Description du système de traduction automatique SYSTRAN de la Commission des Communautés Europeennes, *Documentaliste,* XVI (1979), 150–159.

Varile, G. B., 'Charts: a Data Structure for Parsing,' in King 1983, 77–87.

Vauquois, B., *La traduction automatique à Grenoble,* Paris 1975.

Wang, W. S. Y., Chan, .S. W. and Tsou, .B. K., 'Chinese Linguistics and the Computer, in *Linguistics,* CXVIII (1973), 89–117.

Wang, W. S. Y. and Chan, S. W., *Development of Chinese-English Machine Translation System,* Berkeley, California 1974.

Wang, W. S. Y. and Chan, W. W., *Chinese-English Machine Translation System. Final Technical Report, Sept. 1972 — Aug. 1974,* Griffiss AFB, New York, Rome Air Development Center, 1975 (April).

Weaver, W., 'Translation,' in W. N. Locke and A. D. Booth (eds), *Machine Translation of Languages,* New York 1955, 15–23.

Weissenborn, J., 'The Role and Form of Analysis in Machine Translation: the Automatic Analysis of French at Saarbrücken,' in *Overcoming the Language Barrier,* 1977, 593ff.

Wilks, Y., *Grammar, Meaning and the Machine Analysis of Language,* London 1972.

Wilks, Y., 'An Artificial Intelligence Approach to Machine Translation,' in R. C. Schank and K. M. Colby, *Computer Models of Thought and Language,* San Francisco 1973, 114–151.

Wilks, Y., 'Does Anyone Really Still Believe This Kind of Thing?,' in Jones and Wilks 1983, 182–189.

Winograd, T., *Understanding Natural Language,* New York 1972.

Woods, W. A., 'Procedural Semantics for a Question-answering Machine,' in *Proceedings of the Fall Joint Computer Conference,* Montvale, New Jersey 1968.

Woods, W. A., 'Transition Network Grammars for Natural Language Analysis,' in *CACM,* XIII (1970), 591, 606.

Woods, W. A., 'Progress in Natural Language Understanding — An Application to Lunar Geology,' in *Proceedings of the National Computer Conference,* Montvale, New Jersey 1973.

Woods, W. A., 'Semantics and Quantification in Natural Language Question Answering,' in *Advances in Computers,* XVII (1978), 2–86.

Waltz, D. L., 'Natural Language Access to a Large Data Base,' in *Advance Papers of the Fourth International Joint Conference on Artificial Intelligence,* Cambridge, Mass. 1975.

Waltz, D. L. and Goodman, B. A., 'Writing a Natural Language Data Base System,' in *Proceedings of the International Joint Conference on Artificial Intelligence,* Cambridge, Mass. 1977.