BEDE: A MICROPROCESSOR-BASED MACHINE TRANSLATION SYSTEM

H.L. Somers and R.L. Johnson

Centre for Computational Linguistics, University of Manchester
Institute of Science and Technology

The proposed paper describes an on-going research pro-
ject being carried out by staff and students at the Centre
for Computational Linguistics to design a limited-syntax
controlled-vocabulary machine translation system of sophistic-
ated design to run on a microprocessor.

1. Background

Bede is essentially a research project: we are not
immediately concerned with commercial applications, though
such are clearly possible if the research proves fruitful.
Work on Bede at this stage though is primarily experimental.
The aim at the moment is to investigate the extent to which
a microprocessor-based M.T. system of advanced design is
possible, and the limitations that have to be imposed in order
to achieve a working system. This paper describes the overall
system design specification to which we are currently working.

In the basic design of the system we attempt to incor-
porate as much as possible features of large-scale M.T. sys-
tems that have proved to be desirable or effective. Thus, Bede
is multilingual by design (i.e. not based on language pairs)
(cf. King, 1981:12); algorithms and linguistic data are
strictly separated (cf. Johnson, 1979:140); and the system is
designed in more or less independent modules (cf. Vauquois,
1965:33).

The microprocessor environment means that criteria of size are important: data structure both dynamic (created by and manipulated during the translation process) and static (dictionaries and linguistic rule packages) are constrained to be as economical in terms of storage space and access procedures as possible.Limitations on in-core and peripheral storage are important considerations in the system design.

In large _general purpose_ M.T. systems, it is necessary to assume that failure to translate given input correctly is generally not due to incorrectly formed input, but to insufficiently elaborated translation algorithms. This is particularly due to two problems: the lexical problem of choice of appropriate translation equivalents, and the strategic problem of effective analysis of the wide range of syntactic patterns found in natural language. The reduction of these problems via the notions of _controlled vocabulary_ and _restricted syntax_ seem particularly appropriate in the microprocessor environment, since the alternative of making a system infinitely extendable is probably not feasible. Both notions have been tried with bigger systems, resulting both in better results from the M.T. system itself, and in increased legibility from a human point of view of source texts (cf. Ducrot, 1972; Elliston, 1978; Lawson, 1979:81-2; Sommers and McNaught, 1980:49).

Given these constraints, it seems feasible to achieve translation via an 'interlingua' (cf. Veillon, 1969; Hutchins, 1978:131), in which the canonical structures from the source language are mapped directly onto those of the target language(s), avoiding any language-pair oriented 'transfer' stage. Translation thus takes place in two phases: analysis of source text and.synthesis of target text.

2. Brief description

A description of the system forms the second half of the proposed paper. For the sake of clarity and brevity in

this summary, we refer to the attached schematic representation of the translation process in Bede. In the full version of this paper, each step is to be outlined in rather more detail.

The analyser uses a chart-like structure (cf. Kaplan, 1973) to produce the interface trees of the abstract interlingual representation. These trees serve as input to synthesis, where they are rearranged into valid surface structures for the target language.

The source text is translated sentence by sentence (or equivalent). Text is first subjected to a two-stage morphological analysis. In the first stage the text is compared word by word with a stop-list of frequently occurring words (mostly function words); words not found in the stop-list undergo morphological analysis, again on a word by word basis. Morphological rules form a finite-state grammar of affix-stripping rules ('A rules') and the output is a chart with labelled arcs indicating lexical item and possible interpretation of stripped affixes, as confirmed by dictionary look-up. The morphological analysis phase also creates a temporary 'sentence dictionary', consisting of copies of the dictionary entries for (only) those lexical items found in the current translation unit.

The chart then undergoes a two-stage syntactico-semantic analysis. In the first stage, context-sensitive phrase-structure rules ('E rules') work towards creating a single arc spanning the entire translation unit: arcs are labelled with appropriate syntactic class and syntactico-semantic feature information and a trace of the lower arcs which have been subsumed. In the second stage, the tree structure implied by the labels and traces on these arcs is disjoined from the graph and undergoes general tree-to-tree-transduction rules ('T rules') resulting in a single tree structure representing the canonical form of the translation unit. With source-language lexical items replaced by unique multilingual-dictionary

addresses, this is the interlingua which is passed for syn-
thesis into the target language(s).

Synthesis consists of a combination of T rules which re-
assign new order and structure to the interlingua, and of con-
text-sensitive rules which can be used to assign mainly syn-
tactic feature labels to leaves ('L rules'), for example for
the purpose of assigning number and gender concord (etc.). The
resulting list of labelled leaves (the superior branches are
no longer needed) is passed to morphological synthesis where
a finite-state grammar of morphographemic and affixation
rules produces the target string.

As can be seen, the system is strictly modular, and at
each interface only a small part of the data structures used
by the donating module is required by the receiving module.
The 'unwanted' data structures are written to peripheral store
to enable recovery of partial structures in the case of fail-
ure or mistranslation, though automatic back-tracking to
previous modules by the system as such is not envisaged as a
major component.

The 'static' data used by the system consist of the
different sets of linguistic rule packages, plus the diction-
ary. The system essentially has one large multilingual diction-
ary from which numerous software packages generate various
subdictionaries as required either in the translation process
itself, or for lexicographers working on the system. Alpha-
betical or other structured language-specific listings can be
produced, while of course dictionary updating and editing
packages are also provided.

3. Implementation details

The system will run on any microprocessor system which
runs under the CP/M operating system and at C.C.L. is implem-
ented on the Intertec Superbrain with twin $5\frac{1}{4}$" double density
floppy disk drives, giving a total of 320k bytes of on-line

store. Programs are written in Pascal/M (Sorcim, 1979), a
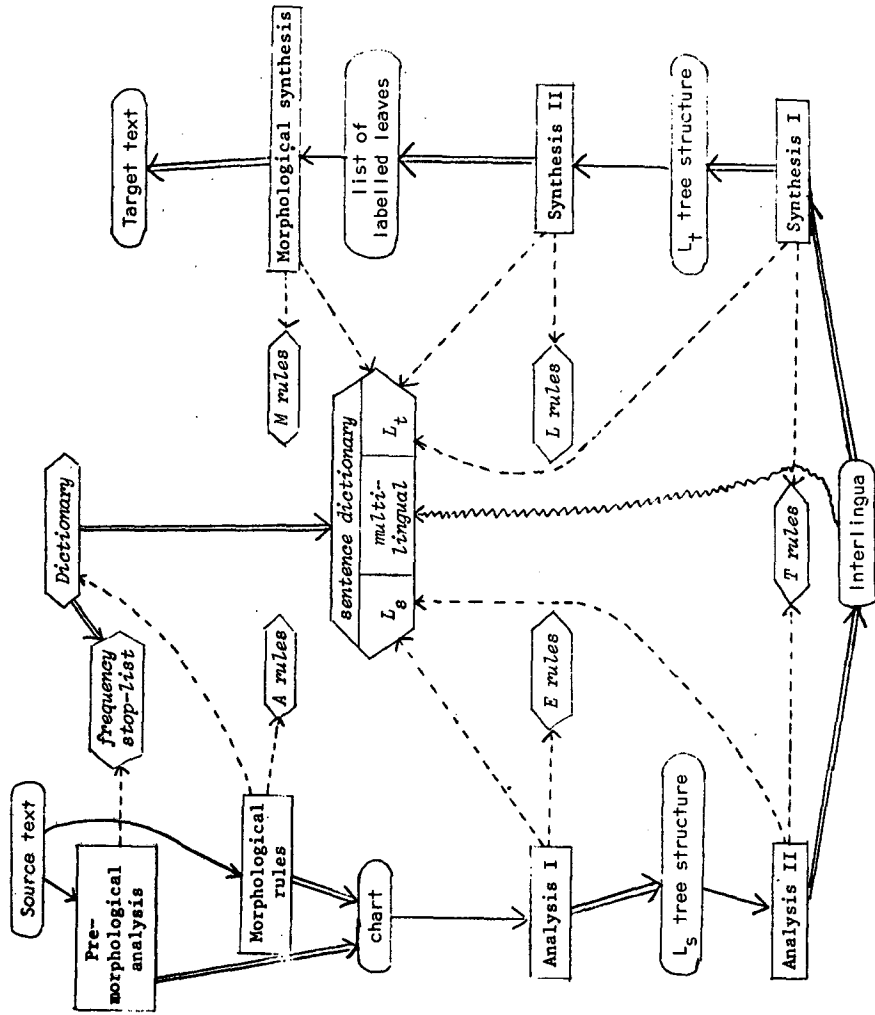Pascal dialect closely resembling UCSD Pascal.

4. References

Ducrot, J.M. (1972) - Research for an automatic translation
    system for the diffusion of scientific and technical
    textile documentation in English-speaking countries:
    Final report. Boulogne-Billancourt: Institut Textile
    de France.

Elliston, J.S.G. (1978) - Computer aided translation: a busi-
    ness viewpoint. In Snell, B.M. (ed.) - Translating and
    the computer: Proceedings of a Seminar, London, 14th
    November, 1978. Amsterdam (1979): North-Holland. 149-
    158.

Hutchins, W.J. (1978) - Machine translation and machine aided
    translation. Journal of Documentation 34, 119-159.

Johnson, R.L. (1979) - Contemporary perspectives in machine
    translation. In Hanon, S. and Pedersen, V.H. (eds.) -
    Human translation machine translation: Papers from the
    10th Annual Conference on Computational Linguistics in
    Odense, Denmark, 22-23 November, 1979 (Noter og Kommen-
    tarer 39). Odense (1980): Romansk Institut, Odense Uni-
    versitet. 133-147.

Kaplan, R.M. (1973) - A general syntactic processor. In Rustin,
    R. (ed.) - Natural Language Processing (Courant Computer
    Symposium 10). New York: Algorithmics Press. 193-241.

King, M. (1981) - EUROTRA - a European system for machine
    translation. Lebende Sprachen 26, 12-14.

Lawson, V. (1979) - Tigers and polar bears, or: translating
    and the computer. The Incorporated Linguist 18, 81-85.

Somers, H.L. and McNaught, J. (1980) - The translator as a
    computer user. The Incorporated Linguist 19, 49-53.

Sorcim (1979) - Pascal/M user's reference manual, Walnut
    Creek, CA: Digital Marketing.

Vauquois, B. (1975) – <u>La traduction automatique à Grenoble</u>
(Document de Linguistique Quantitative 24), Paris:
Dunod.

Veillon, G. (1969) – Description du language pivot du système
de traduction automatique de CETA. <u>T.A. Informations 1,</u>
8-17.

Key to the scheme (see overleaf):

(data structure)   ⟶   enters

⟨ dictionary/grammar ⟩   ⟹   creates

[module]   - - -⟶ uses

                            ⤳   is linked to

Target text

Morphological synthesis

list of labelled leaves

Synthesis II

L_t tree structure

Synthesis I

M rules

L rules

Dictionary

sentence dictionary

multi-lingual

L_t

L_s

Interlingua

frequency stop-list

A rules

E rules

T rules

Source text

Pre-morphological analysis

Morphological rules

chart

Analysis I

L_s tree structure

Analysis II

Schematic representation of the translation process in Bede