AIR FORCE INSTITUTE OF TECHNOLOGY

AIR UNIVERSITY
UNITED STATES AIR FORCE

FRENCH TO ENGLISH
MACHINE TRANSLATION SYSTEM
BASED UPON DIGITAL COMPUTER
SOFTWARE PROGRAMS (SYSTRAN)

THESIS

GE/BE/70-9 Andre R. Gouin
Capt. CAF

SCHOOL OF ENGINEERING

WRIGHT-PATTERSON AIR FORCE BASE, OHIO

FRENCH TO ENGLISH MACHINE TRANSLATION SYSTEM BASED UPON DIGITAL COMPUTER SOFTWARE PROGRAMS (SYSTRAN)

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology

Air University

in Partial Fulfillment of the
Requirements for the Degree of

Master of Science

by

Andre R. Gouin, BASc.
Captain Canadian Armed Forces

Graduate Electrical Engineering

March 1970

NOTE: Controls lifted ref. 2130-4-1(D Secur) 17 January 1973

Preface

The human eye easily and accurately recognizes written or printed characters or signs already defined in the brain. The human ear does similarly for sounds. Much effort is presently directed towards modeling these functions under such titles as "Pattern Recognition" or "Optical Character Recognition" and others. These exhibit specific functions of the brain and the solution of any one or all of the problems involved would contribute much to the understanding of the human brain by providing models which could be, more or less, closely correlated to the way the brain behaves in these areas. One singular aspect of this work can be noticed in the "Re" of recognition. An higher level of study of the brain can thus be seen to be in the study of its cognitive ability. A simple example of this is the generation of a completely new and never seen or heard before sentence, an everyday process. Thus the study of languages, how they reflect known and accepted facts, how they are used as vehicles of thought, generate ideas and so on, is one conscious, or otherwise, attempt at defining one cognitive aspect of the brain function. The translation of two languages from one into the other exhibits some of the fundamental actions of the brain in a very restrictive sense, but nevertheless worthwhile of investigation. Machine Translation of languages is an attempt at modeling this process, even if the activity is often obscured by the more immediate goal of establishing a system to provide translation per se. The disfavor that Machine Translation finds itself in at the present time (Ref 5:3-9) should be removed or greatly reduced if computational linguists, who deal with one language at a time, would only consider machine translators as themselves computational linguists dealing with two languages at a time instead of only one and if machine translators would approach their work with some of the goals of computational linguistics in mind. Viewed in this light, the analysis of a machine translation system, operating on two specific languages, is highly interesting and very promising in shedding some light on the subject of thought process modeling, but is often limited in its ability to do so by the many artifacts used to achieve translation which, in themselves, obscure the similarity between the machine and brain processes.

The investigation reported in this thesis is based on the possibility that much useful results might be uncovered by the adaptation of a presently operating machine language translation system to accept one different language. If success can be achieved in changing only those portions of the system concerned with one of the two languages and still obtain some results in the other, then two very important conclusions could be inferred. One, that the system is more closely related to the human process, than had been apparent in the two-language-only system, by being capable, as the human, to translate from more than one language into another language, and, more importantly, that the unchanged language has inherent in its structure a set of indicators or rules (likely finite) which are also part of the two other languages. This inherent set of indicators or rules is sometimes referred to as an "underlying language" or "universal language" (universal in the sense of applicable to all, not in the sense of unique). The implications of the discovery of such a language are extremely important and challenging as one possibility for the understanding of the human thought process.

This discovery or even an attempt at such, is clearly

beyond the bounds of the present investigation and of the capabilities of this investigator. The purpose of this present project is therefore much more restricted in scope but is nevertheless one step in that general direction. In this regard, I wish to acknowledge my indebtedness to Dr. Matthew Kabrisky, for the continued, patient, thought-provoking and stimulating suggestions which he made, throughout the development of this study, as my thesis advisor. I also wish to express my sincere thanks to Dr. William Key and Captain Greg Alexander, of the Foreign Technology Division, for their appreciated support and encouragement as members of the sponsoring laboratory of this project and to Mr. Peter Toma, President of Latsec Inc. the owner of SYSTRAN, and Mr. D. Perwin, Programmer with Latsec, without whose assistance this investigation would not have been possible.

Andre R. Gouin

Contents

Contents

List of Figures

Abbreviations

A.	Adjective
Adj.	Adjective
Attr.	Attribute
Aux.	Auxiliary
B	Adverb
C	Coordinate conjunction (coord. conj.)
Chg.	Change
Cl. Ind.	Clause Indicator
Cmp. WP =	Compare word pointed by the word pointer named to the word stated.
Comp.	Complement
Cond.	Conditional
CW	Current word
D	Determiner
E	Relative Pronoun
E	Equal (when on exit of decision block in flow chart)
F	Framer comma
Fem.	Feminine
FB/WP	Function byte of word pointed by word pointer named
Fut.	Future
G	Negation
I	Intensifier
Imp.	Imperative
Imper.	Imperfect

Ind.	Indicative
Inf.	Infinitive
Intr.	Introduce
J	Inverter comma
L	Linking verb
LB	Left boundary
Masc.	Masculine
MC	Merge common
Mng.	Meaning
MT	Machine translation
N	Noun
N	No (when used on exit of decision block in flow chart)
N	Neuter (for French = can be either masculine or feminine)
NE	Not equal
Neg.	Negative
NP	Noun Phrase
NZ	Not zero
P	Preposition
Plur.	Plural
Part.	Participle
Pred.	Predicate
Prep.	Preposition
Pres.	Present
Prim.	Primary
Pron.	Pronoun

P. S.	Past simple
R	Pronoun
RB	Right boundary
Refl.	Reflexive
Rel. WP1/WP2	Relate the words pointed by the word pointers named
Req.	Require
S	Subordinate conjunction (subord. conj.)
Sing.	Singular
Subj.	Subject
Subjun.	Subjunctive
TB/WP	Test byte or bytes named of word pointed by word pointer named
TM/WP	Test merge byte or bytes named of word pointed by word pointers named and store common bits in 'MC'
Tr.	Transitive
SB	Subject byte (when used with "set" only)
SB	Sentence begin
SE	Sentence end
Set WP=	Set the word pointed by the word pointer named equal the function named
SL/WP1 '___'=WP2	Seek left of word pointed by word pointer named 'WP1' for function '___' and call that word with that function 'WP2'. (If WP1, WP2 not specifically named then 'CW' is implied)
SR/WP1 '___'=WP2	Same as SL/WP1 except to the right

SSU	Sub-sentence unit
STR/WP	Set translated the word pointed by the word pointer named
Subroutine WP1/WP2	Do subroutine with word pointer no. 1 on the governor, word pointer no. 2 on the dependent
S/WP ='Word/WP'	Set the word pointer named on the word or word pointer named
V	Verb
VP	Verb phrase
WA	Word after (WP name)
WB	Word before (WP name)
WC,WD,WE	Other words (WP names)
WP	Word pointer
X	Auxiliary
Z	Zero
1,2,3	Indicate person of pronoun or verb

Abstract

SYSTRAN, a Russian to English Machine Translation operating system forms the basis for a French to English MT adaptation. SYSTRAN is summarily explained followed by a development of the new dictionary, lexical routines and French structural analysis to the extent necessary to test the adaptability of SYSTRAN to accept French as a new source language. Comment sheets, flow charts and programs on the changed portions of SYSTRAN are included as appendices. The French test corpus, a human English translation and the machine translation are also included in the appendices.

FRENCH TO ENGLISH MACHINE TRANSLATION SYSTEM BASED UPON DIGITAL COMPUTER SOFTWARE PROGRAMS (SYSTRAN)

I. Introduction

The purpose of this investigation is to attempt to exhibit the adaptability of SYSTRAN, an operational Russian to English Machine Translation system, to accept French as another source language. The study of languages, as the vehicles of thought, offers interesting possibilities in understanding the human thought process. Translation, which involves two languages almost simultaneously, reflects the use of two different, varied and apparently incongruent such vehicles of thought. Translation by machine is thus an attempt at modeling the human translator. One text translated by eleven human translators generally results in eleven different translations. It is no surprise therefore, that the same text translated by machine is also different from all the others.

Two major hurdles prevent the field of Machine Translation from advancing at present, as it should. One is the input bottleneck. It is of little use to have a system capable of translating four or five hundred thousand words an hour if the preparation of all this text takes months or costs a fortune to prepare in a decent time frame. Work on Pattern Recognition, at the Air Force Institute of Technology, using spatial filtering of finite discrete Fourier transforms (Ref 8) offers, among other approaches, great promise in solving this major problem. The other obstacle to MT is the general misconception of the purpose of machine translation. If the translation wished is to retain all the flowery rhetoric of the original text, then poetic human translators, although they might not agree between themselves, nevertheless retain the upper hand on the machine. If, however, the translation is to convey scientific or technological facts, to a knowledgeable audience, then machine translation affords the best, if not the only, recourse for the scientific community to come out alive from under the so-called "information explosion".

Besides these immediate and necessary benefits obtainable from Machine Translation of languages, a more profound reason for continuing to advance this field is certainly its inherent possibility to exhibit, in part at least, some of the intricacies of the human thought process. That, Machine Translation has not been able to give perfect or near-perfect results to date should not be taken as an inherent inability of the machines to perform in this area, but rather, it might very well be, that the human himself has not yet found the ability to express concisely and precisely something (the thought process) that he feels is going on within him, but for which he cannot find the words. Outputs from machine translation might give him a clue. The reader interested in finding out more about Machine Translation is referred to Victor Yngve's chapter "MT at M.I.T. 1965" (Ref 12: 451-523), where a comprehensive list of references is given.

The approach adopted here, to achieve the more restricted goal of testing SYSTRAN's adaptability, is reflected in the text of this report. SYSTRAN is summarily explained in the next chapter to the extent needed to appreciate the amendments introduced later to adapt the system. The following chapter deals with the French-English machine dictionary. The word "dictionary" is an unfortunate term, used in MT, to describe what is very much different from a desk-top dictionary. This chapter will reflect some of these differences. Following this, the principle of the lexical routine, as employed in SYSTRAN to resolve the syntactical function-dependence of meaning for certain words, will be elaborated and applied to the incomplete but sufficient solution of one French example. Chapter V then will deal with the French structural analysis program which is developed to the extent necessary to replace the Russian one in SYSTRAN. Chapter VI then summarizes the conclusions drawn from the study and suggests some approaches for future investigation in this area.

II. SYSTRAN

A third generation computer program package, SYSTRAN was developed by LATSEC Inc. under an RADC contract (Ref 10) to provide a capability for translating Russian into English. The system was conceived to be improvable in its primary task but also, and more importantly, to be adaptable to accept, with minor modifications, other languages. To appreciate fully the impact of the versatility of SYSTRAN to accept a new language, an extensive explanation of the complete system, in its primary task, would be required. Such an explanation however, would necessitate more space than this report is allowed, and besides, would take the reader, as well as the writer, outside of the immediate purpose of this investigation. Nevertheless a minimal explanation needs to be included so that the reader may be able to follow the logic of the following chapters. Therefore, SYSTRAN as an operational system for Russian to English machine translation is introduced here while later chapters indicate the modifications made to adapt the system for French to English translation.

The Basis of SYSTRAN

The computer programs forming the basis of SYSTRAN consist of four main categories: Macros, support, pre-translation phase, and translation phase programs. The Macros of the first group are used to simplify and expedite the task of writing the programs of the last three categories. These latter programs are all written in the System 360 assembly language, excellently explained in N. Chapin's book (Ref 2). Each category is now summarized, with an example of one or two members of each group, to the extent sufficient to make the following chapters understandable.

Macros

The Macros consist of a set of approximately forty commands for the use of linguists to relieve them of the finer but necessary details of programing, such as: memory housekeeping, bit movement, length of register contents, address of contents in storage, and many more. These commands might be considered on the programing language level of FORTRAN or COBOL with the major difference that they are oriented towards string manipulation rather than computations. Their purpose is somewhat similar to the COMIT language developed by V. Yngve, for the 7090 type computers (Refs 6, 11). The Macros are compiled into memory by a compiler specially designed by LATSEC, and any one or more can be called by the other main programs. As these Macros are yet unpublished, proprietary rights prevent this writer from giving a thorough explanation of all those available. However, an example to indicate their function is necessary to facilitate, for those interested, the reading of the main program included as Appendix J, at the end of the report.

SETWP and TESTX Macros As an example of two of the SYSTRAN Macros, consider the problem of examining the immediate environment of a word for the syntactical function of the adjacent words. To accomplish this, a means of pointing at the word and its adjacents is necessary. The SETWP (set word pointer) Macro is used for this purpose. It allows the linguist to set pointers at any word and to ask questions about the adjacent word of interest. These questions may be asked using the TESTX Macro, which allows him to compare or test the contents of the different specific bytes which contain the information on the word. A simple example will make this clear.

Example A word has the pointer "CW" and it is desired to determine if the word before is a determiner. The commands to do this appear as:

```
SETWP    WB,CW,-1
TESTX    WB,B01,CE8,NE—NØTDET
```

In the example, the first command sets the pointer, called WB, on the word pointed by CW minus one (i.e. the word before CW). The second command then tests the content of the word at WB, at byte 01 (B01), to verify if it contains the hex character $E8_x$ ($CE8_x$, $E8_x$ being the code assigned to determiners, the x subscript indicating that the two characters which precede are in the hexadecimal numbering system) and, if it does not, (NE, for Not Equal) branch to the command named NØTDET, which would appear later in the program. If it is equal, that is, byte 01 of WB does contain $E8_x$, then drop through occurs (i.e., the next instruction in the sequence is executed). This simple example would require the linguist to write four lines of assembly language instructions without the Macros, besides keeping track of the registers where the information is located. The kind of situation examined here arises continuously in machine translation indicating the value and necessity of these relatively simple-to-use Macros.

At this writing approximately forty Macros have been developed by LATSEC to meet different linguistic needs and more are being considered. The program developed for this thesis uses these Macros exclusively. With this basic understanding of the Macros and with the comments included opposite most of the commands used in the program, the interested reader can later read through the program of Appendix J and gain as much understanding as the reading of any program can give to any one other than the programer who wrote it. The minimality of the program, developed here, to amend SYSTRAN to accept French as a source language, clearly emerges upon considering the other categories of programs of the system.

Support Programs

The second category of programs in SYSTRAN comprises the Support Programs. These are primarily concerned with organizing the information required by the remaining two categories. All the information, used by these latter two groups, can be traced back, one way or another, to the dictionary. The organization of the dictionary is thus seen to be a major determinant of the quality, correctness and speed of translation.

As will be seen later, in Chapter III on the French-English dictionary, a machine translation dictionary is vastly different from a desk-top bilingual dictionary. For a good discussion on machine translation dictionaries the reader is referred to a chapter by Lamb and Jacobson Jr. in a book edited by D. G. Hays on "Language Processing" (Ref 3:51). The SYSTRAN dictionary is divided into a number of sections: idioms, high-frequency words, limited semantic tables, general and topical glossaries. The open-ended nature of SYSTRAN necessitates this diversity in the dictionary and this in turn generates the need for a major portion of the support programs group, to organize each of the different sections, as well as, to provide easy access later for additions and amendments. The need for additions is obvious on considering the dynamism of a natural language, while that for amendments is necessary also when SYSTRAN is appreciated as an evolutionary system continuously being improved as further linguistic advances become available. Nearly twenty support programs are available to expedite the tasks just mentioned in addition to others. Some of these are glossed over quickly below.

HFCONV Program The purpose of one of these, the HFCONV (high frequency conversion) program, is to update the high-frequency word dictionary. The program accepts SYSTRAN coding sheet format input cards (to be discussed in Chapter III) and compiles them into assembly language format ready for inclusion in the high-frequency word table.

DICTCRT Program Another of the support group is the DICTCRT program (dictionary create) which accepts SYSTRAN coding sheet format input cards and organizes a general SYSTRAN dictionary. This program, with no change, is used to build the limited French-English dictionary needed for this study.

Many of the other programs of this group are for specialized tasks not applicable where French is the source language. Others, like the HFCONV, are available and valid but not necessary because of the size of the test corpus. Once the dictionaries are organized the next two categories of programs can be called on.

Pre-translation Phase

Before entering into the translation phase proper, the source language corpus must be organized for dictionary look-up. For example, a word must be made up of an item (a string of uninterrupted characters) followed by a blank, each word must be given a text sequence number, the word must be looked up in the idiom table, or in the high frequency word table, and if not found, it must be prepared for general or topical dictionary look-up. This latter preparation consists in ordering the complete input corpus into the sequence of the dictionary in memory. All these functions and others are performed by the pre-translation phase programs, an example of which follows.

LOADTEXT and CORESRCH Programs The LOADTEXT program, as the name implies, prepares the input text by loading it into the machine for idiom and dictionary look-up. The program separates each word, assigns the text sequence number and puts the word in the required format to enter the CORESRCH program, which searches the memory core and determines the idiom and high frequency word content of the input text.

Some three other programs then prepare what remains for the general dictionary search. The final result of these programs is that each word is back in the original text sequence with the information, obtained from the dictionaries, necessary to begin the translation phase. It is interesting to note at this point that everything dealt with so far has been without concern for either the source or the target languages, the programs being independent of language. The phase which is now entered into, however, deals with the portions of SYSTRAN upset by a change of source language.

Translation Phase

Among a dozen programs in this group of SYSTRAN the ESYNPRG (English synthesis program) can be retained, without change, since the target language is the same in this investigation, provided however, that the necessary inputs to the program remain the same or equivalent. These inputs, the basis for the synthesis program to establish the structural relationship between the words, are obtained in great majority, from the STRUAN (structural analysis of

Russian) program. The change from Russian to French requires a complete change of STRUAN and is the subject of Chapter V. This translation phase also has a LEXICAL program which deals with the meaning of multiple meaning words on the basis of their syntactical function. This program also cannot be used as is and the basis for a minimal replacement program is given in Chapter IV. Other programs of this group in SYSTRAN are the limited semantic recognition programs which determine the translation of words in context. These cannot be used without total change and are not included here as they go beyond the immediate purpose of this study. Others, specifically required for Russian, as the article insertion program to resolve the absence of articles in the Russian language are also deleted from the system in this test.

Conclusions on SYSTRAN

From the above very summarized exposition of the SYSTRAN system one highly important fact should be retained. A major proportion of the system is independent of the languages which indicates clearly its potential for adaptability to other languages. The forty Macros and approximately forty programs of SYSTRAN comprise about 36,000 assembly language instructions, thus it can be appreciated that the success of this present project is highly dependent on the full use of as much of SYSTRAN as possible. With this in mind then, the next chapter on the dictionary will prepare the way for the later chapters.

III. FRENCH-ENGLISH DICTIONARY CODING

The analysis of SYSTRAN leads to the conclusion that twenty four of its programs can be used unchanged for French to English translation; six are not needed; eight require major changes which cannot be accomplished here because of time primarily but also because their exclusion does not prevent the attainment of the goal of this study; and, two need to be changed completely. This chapter, on the dictionary, lays the groundwork for the two new programs, each of which is dealt with in a separate subsequent chapter.

As mentioned earlier, the prime source of information for all the programs in SYSTRAN can be traced back to the dictionary. The limited corpus used in the test here only requires the DICTCRT program to build the French-English dictionary. Each dictionary entry forms a part of an array of codes used by the later programs. What follows will explain the codes specifically used to define the French language.

Synthesis Array

Before going any further into the definitions of codes and their manipulation it is necessary to introduce, what will be called here, the synthesis array. A requirement later in the translation phase is best understood if associated with the idea of an array. In that later phase, an analysis area will be set up in core memory to analyze, manipulate, set indicators, establish governors and dependents, and so on for each word of one sentence at a time. The organization of the results of these later programs, as well as the information assigned by the dictionary, is best appreciated if the sentence under analysis is pictured associated to an array. The array has as many columns as there are words in the sentence. The number of rows is dependent on the machine used, but for SYSTRAN there are 128 rows. Each row is assigned a specific function, for example, row number two represents byte number 01, previously mentioned in the example on Macros (the row numbers are one higher than the byte numbers because the latter begin with number 00). Each element of the array is a hexadecimal number of two characters, representing one byte of eight binary bits. Thus for a ten-word sentence the reader can imagine a 10 x 128 array of hexadecimal numbers or, if imagined in the binary system, as an equivalent cube with 10 x 128 x 8 positions for "zero" or "one" bits. This picture should help the reader appreciate where and how the vast number of codes, about to be defined and developed later, are stored and retrieved when required.

<u>Part of Speech</u>

A basic requirement of the English synthesis program is for a set of indicators on the principal words and their dependents. The establishment of a word as a governor depends largely on its part of speech relative to that of the words in its environment. Therefore a requirement exists for classifying the words according to their part of speech which is developed for the French words, along similar lines as for Russian, in Appendix A. Within the same part of speech, words can act differently in varied functions, thus a subclassification into basic part of speech is included in the same appendix. For these two divisions the codes are named, as in SYSTRAN, the POS and BPQ codes respectively. An examination of the appendix may reveal, to some readers, an apparent arbitrariness in the choice of codes. The deviations, from the systematic approach used for Russian, are necessary to avoid possible conflicts with the Russian word coding.

Stem and Endings

To save memory space and expedite dictionary look-up, it is advantageous to use stem and endings for certain classes of words. One such class in French is the verb, where one stem is used unchanged with many different endings and where also the ending contains the person, tense and mood information. A table of such endings for the French verbs is given in Appendix B and the Paradigmatic Set Table (PST), developed from it, is included as Appendix C. Also included in the latter table is the set of endings used to recognize the feminine and plural form of French nouns and adjectives. This particular portion was not used however because it would have required the development of a morphological program by LATSEC for very few words in this test. Therefore all words are entered full form for this test and this point is included here for completeness.

French Diacritics

Non-standard characters of the French language are defined on the model of K.H.V. Booth (Ref 1:55) with the accute accent "´" having a "1" placed in front of the accented letter, the grave accent "`" a "2", and the circumflex accent "^" a "3". The diaeresis (¨) and cedilla (¸) are not included here as they are for phonetic purposes in words which would not otherwise appear without them.

Coding Sheet

The SYSTRAN coding sheet is reproduced on the next page, with the kind permission of LATSEC, Inc., and its use to code the French-English dictionary is detailed in Appendix D. The columns marked with an "x" are those

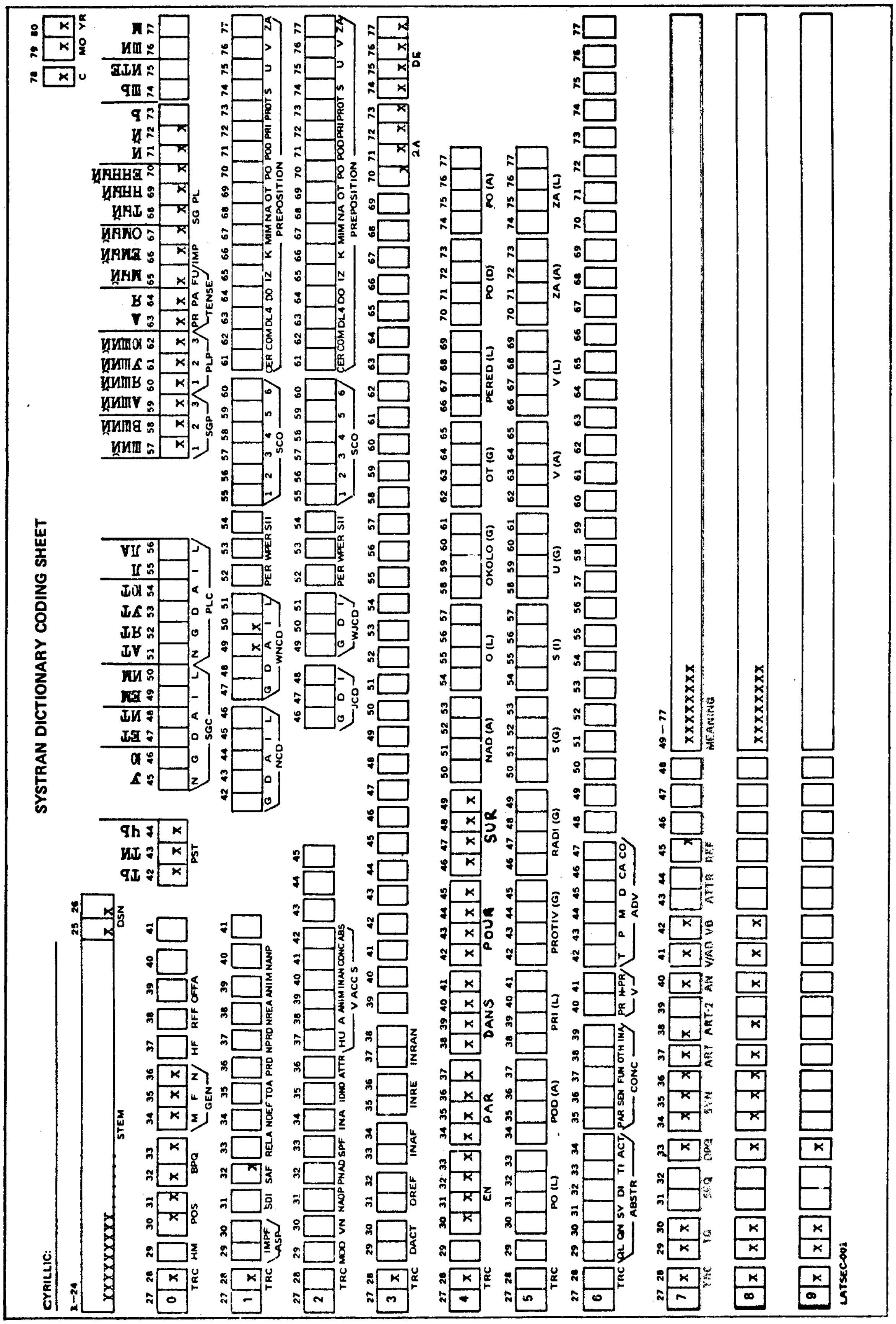

Figure 1. SYSTRAN Dictionary Coding Sheet.

applicable to this study. A column not assigned a specific set of codes implies that a "1" is entered in that column when it is applicable.

Starting with column 70 on card 3 (the card number in column 27) to the end of card 5 is a type of coding which differs somewhat from the rest and is used for preposition government. This means that each French preposition is assigned a basic meaning in the dictionary which is the one most likely to apply. However, the meaning often changes from the basic one and is generally governed by the word which immediately precedes or follows the preposition. A preposition translation program, part of the translation phase group of SYSTRAN, provides for these changing meaning and requires that the preposition governor be coded accordingly. For example, the French preposition "de" has the basic English meaning "OF". However, when the word "libérer" (liberate) is followed by "de", the meaning of "de" changes to "FROM". Thus the corresponding code for "FROM" is placed in columns 74-75 of card 3 of the word "libérer", the blocks reserved for the preposition "de".

Cards 7, 8, and 9 deal with the English equivalent coding, with the information entered here used by the synthesis program and others. Proprietary rights prevent entering into details on this portion of SYSTRAN but the method is the same.

Idioms

Another program of SYSTRAN, not mentioned yet, allows for the handling of idiomatic expressions. LATSEC Form 002, reproduced by permission on the next page, is used for coding idiomatic and limited semantic expressions. The latter function, as mentioned earlier, is not developed or used here and so the form is applicable only for the idioms appearing in Appendix E. The list will be seen to include expressions not normally considered as idioms. Their inclusion eliminated the need to develop some programs which would have contributed nothing to this investigation. Obviously this would be changed in a more complete system and programs would be developed to take care of these peculiarities.

The above, plus the referenced appendices cover the minimum requirement for coding the French language to the extent necessary to test the adaptability of SYSTRAN. On the basis of these definitions, each word of the test corpus is coded and the resultant input to the DICTCRT program is given in Appendix F. One very important situation not provided for yet is when a word has its meaning dependent on the syntactical function it serves in the sentence. In fact, W. N. Locke points out (Ref 4:13) that this type of words makes up over 30% of a study of some 400,000 French running words. Guided by the frequency of occurrences of these words, as given in Table 4 of the above reference, the following chapter deals with those most likely to occur in almost any text and explains the development of a "Lexical" routine.

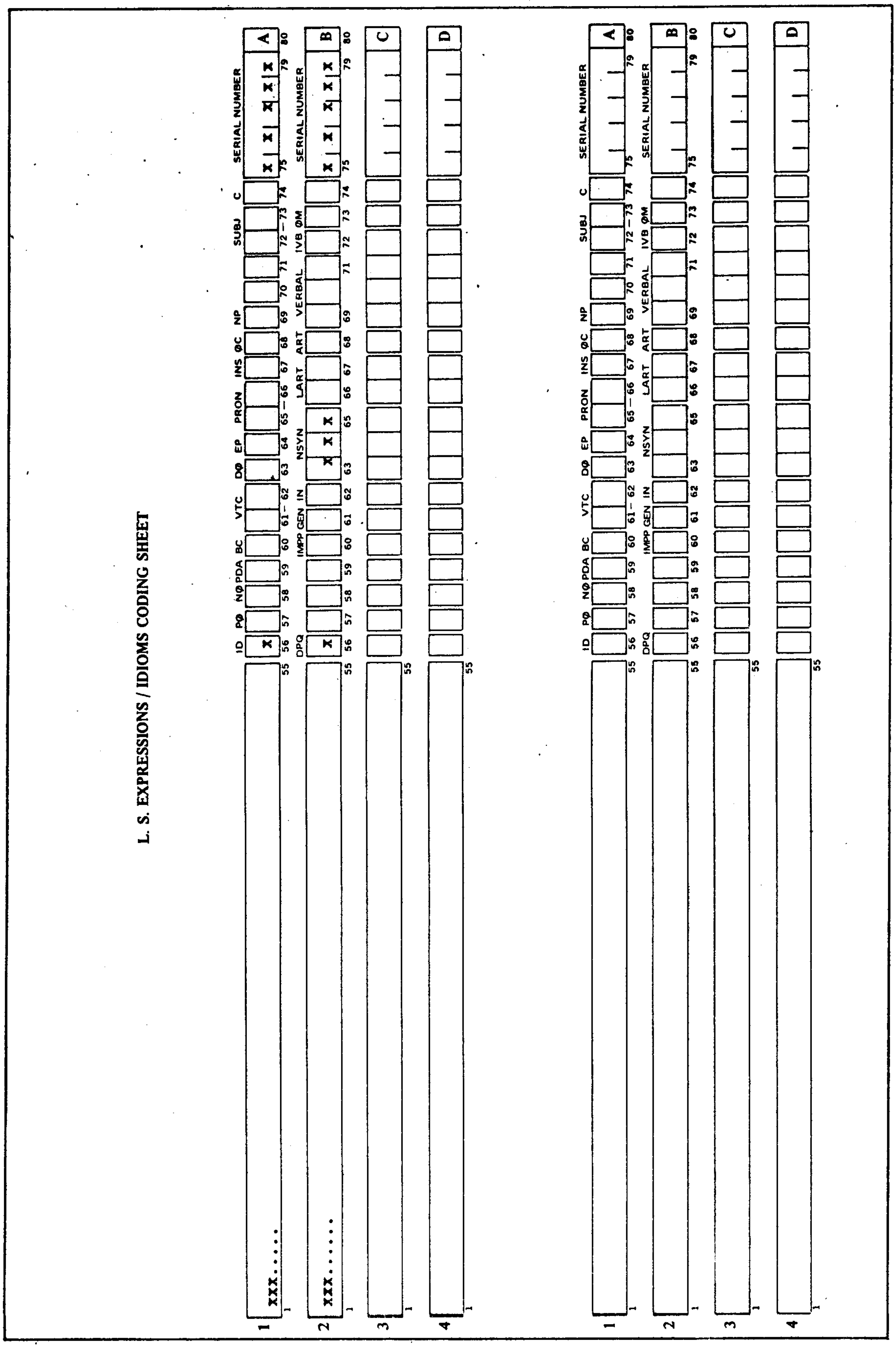

Figure 2. L.S. Expressions/Idioms Coding Sheet.

LATSEC-002

IV. LEXICAL ROUTINES

Many of the monosyllabic words of the French language have either lost their original specific meanings or have acquired various additional ones depending on the syntactical function of the word. The method developed in SYSTRAN, to deal with the equivalent situation in Russian, is called the LEXICAL program. The program there is made up of a large number of sub-routines each initiated by the appearance of the problematic word or words and is quite advanced and elaborate. The limited test corpus and time element prevents a development of the routines here to a similar extent, however, what follows indicates the basic principle in the make-up of one of the routines sufficient to allow a continuing effort in this area in the future.

Basic Principle

The development of a lexical routine begins with the study of the word, in general grammars and dictionaries of the language, and in context. The grammars and dictionaries indicate the various functions of the word and some examples are usually found, for the more common words, pointing to possible further avenues of investigation. The word in context is best studied from the results of concordances where the word is listed, at it appears in a running text, with its immediate environment.

Comment Sheet By observing the environment of the word, in its different syntactical functions, a general trend can usually be discovered fairly early which indicates the requirement on the adjacent words for a particular function to be defined for the word. A comment sheet is in this way developed including as much as is known as possible about the word and where decisions are doubtful this is noted for use in the flow chart development later. The beauty of this approach is in its self-sufficiency. By this is meant that each routine is independent by itself, it does not depend on any input, other than the word to trigger it, and its output is not a necessary condition for any other programs to run. Thus, any portions of a routine can be deleted, amended or added with no ill effect to the rest of the system. This valuable asset of the routines is taken full advantage of in this development since French concordances were not immediately available thereby forcing sometimes quite arbitrary decisions based more on intuition than fact.

Flow Chart The problem analysis portion of the lexical routine development is included in the comment sheet on each of the lexical routines in Appendix G. The next step in the routine development is to make up a flow chart of the pr blem to help in programing the solution. The flow chart is primarily used to organize the available information and to aid programing. The main underlying factor in the development of the flow chart is the absolute necessity for ease of future amendments. In other words, a flow chart presented here for the solution of a lexical problem only represents the first step of a vast number of steps needed to obtain a routine general enough to handle most occurrences of the word. Thus the main branch of the chart includes decisions which are considered less likely than the corresponding exit branch. The exit branches lead to, what might be considered sub-units of the routine each of which ends by an exit out of the routine. Thus the sub-unit may be changed completely later on the basis of further analysis. Also, where a decision is based on doubtful information a "dump" command is included to document the situation. In all that follows the terms "dump" or "message" are used interchangeably with the following significance. The "SMSG" macro executes a "dump" of the analysis area which reflects all action taken by the program up to that point. The term "dump" is used here in its usual assembly language programming meaning as covered for example in Chapin's book (Ref 2: 60-62). In the programs the message is given a number. This refers to the number of the message as it appears in Appendix K on "Messages". The appendix contains the message number, the message statement itself and the reasons for generating it.

Program The flow chart is then used to program the lexical routine. As mentioned earlier, the commands used are all taken from the SYSTRAN Macros. Since these Macros are not detailed here, the complete lexical program for this study is separated from the "Comment what more elaborate than those in the appendix to include explanation of some of the more or less arbitrary decisions that are required at this stage of development. In the appendix these are implied only, in the comment sheet and flow chart.

"Le, la, les, l'" Lexical Routine Example

The words "le, la, les, l'" are selected for discussion because, for one thing, they are the ones which occur most often according to Locke's study (Ref 4). The words "la, les" and "l'", being derivations of "le" for the gender, number and elision requirements of French, need not be carried in what follows, but it should be kept in mind that what is said for "le" applies to its derivatives also, with minor changes.

Comment Sheet for "Le, la, les, l'". "Le" appears in most dictionaries as a possible candidate for one of the syntactical functions: definite article, personal pronoun or neutral pronoun. As article, "le" translates to "THE" or is not translated in some cases; as a personal pronoun it can translate to "IT" or "HIM"; and as a neutral pronoun it may require translation to "SO" or "IT". These are some of the possible translations, by no means exhaustive but sufficient to indicate the need to develop a lexical routine. A study of context using this word soon indicates that it functions as a pronoun if it immediately precedes a verb or if it is hyphenated to a verb before. If none of these conditions is satisfied the word is assumed to function as an article for the present, and the routine sets its POS code and/or translation as such. If one is satisfied then the word is a pronoun and it remains to differentiate it as a personal or neutral pronoun. This requires a limited semantic analysis in the sense that, what the word "le" replaces as a pronoun must be discovered and established as a person or thing in order to recognize the pronoun as personal or neutral. This type of analysis is not included here and the decision in the routine to use the translation "IT" is based on the usual impersonal nature of technical writing. Further, to take care of the no-translation case, when the word acts as an article, requires the development of a program along similar lines as the article insertion one for Russian, except that in this case it would be to delete the article, and this also requires an additional code in the dictionary. The magnitude of the necessary effort needed to solve this last relatively simple situation obviously places its development outside the bounds of this investigation.

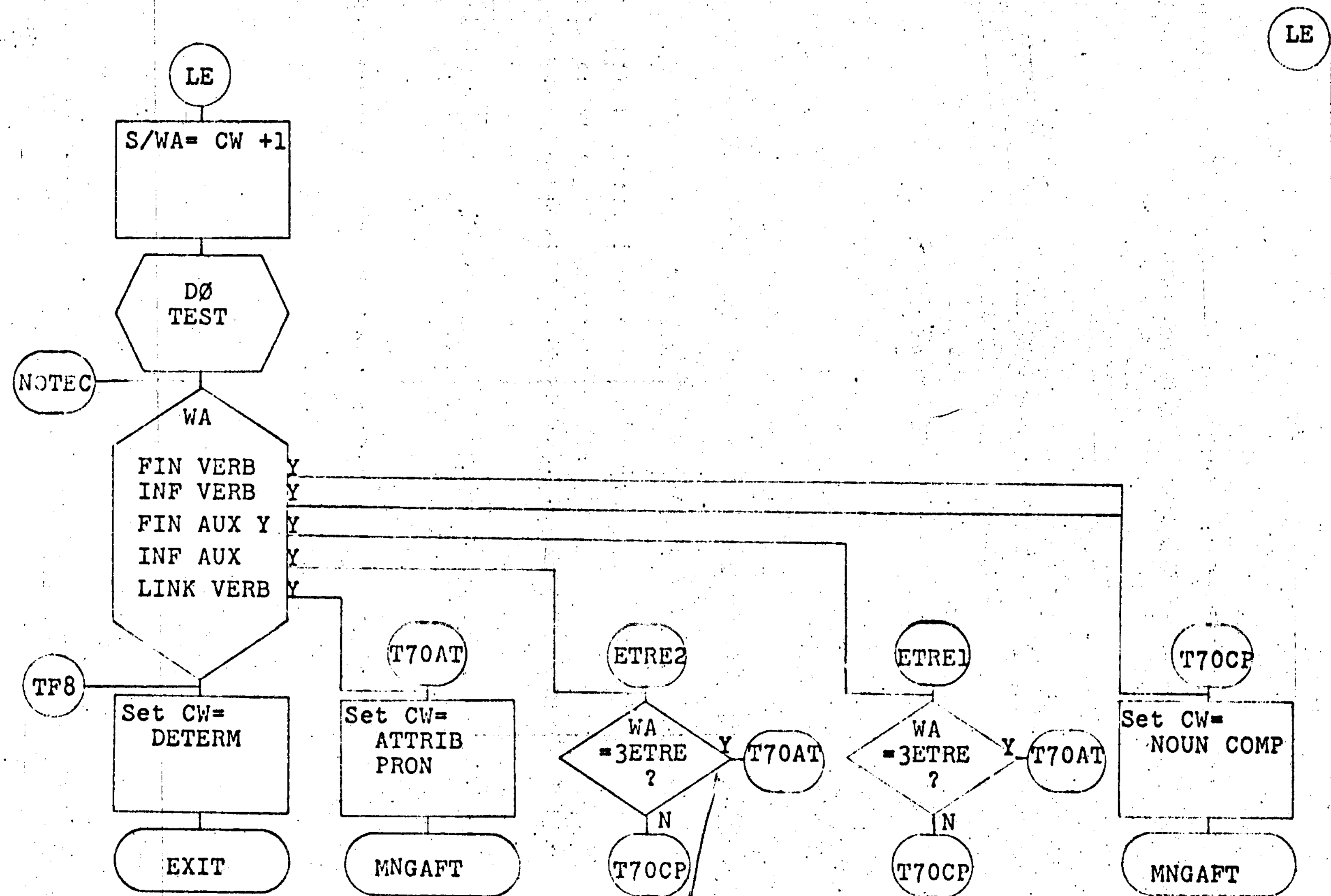

Figure 3. 'LE' LEXICAL Flow Chart.

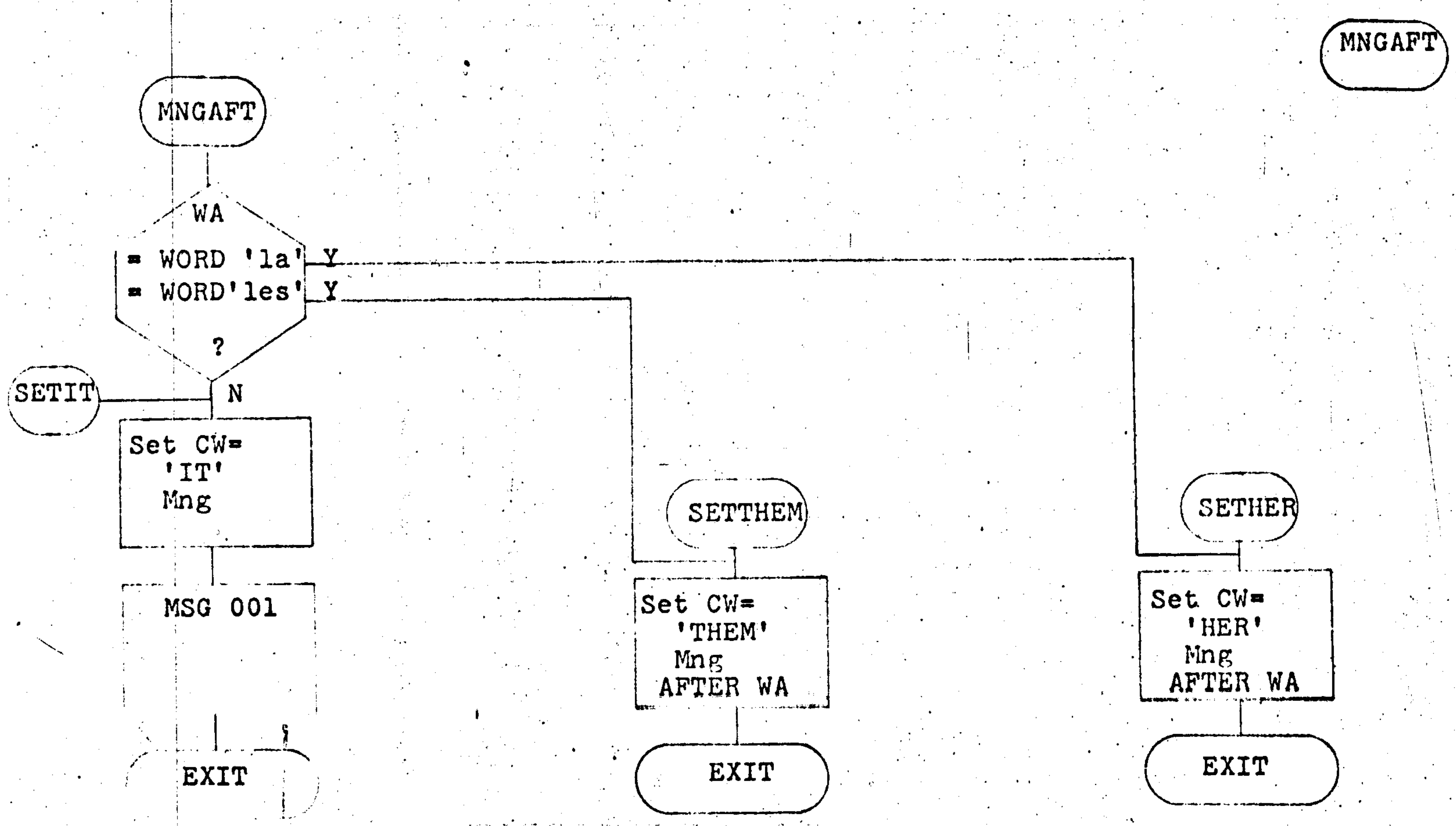

Figure 3. (Continued) 'LE' LEXICAL Flow chart.

Flow Chart for "Le, la, les, l'". The analysis of this particular word falls far short of its ultimate development, however, it is sufficient to serve the immediate purpose. More importantly a study of the flow chart, illustrated in Fig. 3, above, clearly indicates how any portions can be replaced without requiring major changes in the other portions.

Program for "Le, la, les, l'". Guided by the flow chart above the simple program for the 'LE' lexical routine is developed and a copy appears in Fig. 4 below.

Conclusion on Lexical Routines

With an understanding of the basic principle of the development of a lexical routine and of the detailed example, the interested reader can now read through the routines developed for this investigation as contained in Appendix G. The more observant reader will surely notice that some of the routines included are very minimal and only appear to serve the purpose of documentation. These are expressly included for that purpose, to accumulate data on words which are expected to require a routine in the near future. The lexical routines are seen to deal with particular words. A more general situation is encountered when a type of word may function as two different parts of speech. For example, in French, many adjectives can be used as nouns. The number of these makes it ludicrous to have one routine for each, therefore a noun-adjective homograph lexical routine is developed. This and others are also included, in a similar format to the lexical routines, in the appendix on the main program.

The previous chapter dealt with the coding of the words in the dictionary. This coding left open such ambiguities as the meaning dependency on the syntactical function of the word, or the more general case of the noun-adjective homograph which were handled in this chapter on the lexical routines. The way is now clear to enter into the more extensive structural analysis of the French sentence.

```
634*      **ENTRY WORD IS LE LA LES LZ POS=E8 BPQ=36**
614  LE1        SETWP CW,SB,-1
617  LE         SCANR CW,CW,(FOR,WD,LE,E,DR1),(FOR,WD,LA,E,
                DR1), (FOR,WD,LES,E,DR1),(FOR,WD,LZ,E,DR1)
632             GOTO PREP2A1
635  DR1        SETWP WA,CW,+1          SETS PTR ON WORD
639                                     -AFTER
640             TESTX WA,B01,CEC,E,NE12  WA POS=EC?
641             GOTO NOTEC
643             DO   TEST               ONLY VERB AMBIG
645*                                    NEED BE CHECKED
646*                                    HERE.
647*                                    TEST SUBR DUMMY
648*                                    AT PRESENT
649  NOTEC      TESTX WA,B01,C04,E-T70CP   IS WA VERB?
652             TESTX WA,B01,C48,E-T70CP   IS WA INF?
655             TESTX WA,B01,C40,E-ETRE1   IS WA AUX?
658             TESTX WA,B01,C44,E-ETRE2   IS WA INF AUX?
661             TESTX WA,B01,CF0,E-T70AT   IS WA LINK VERB?
664             GOTO TF8                   WA NOT VERB FORM
666*                                       SO CW = DETERM
668  ETRE1      TESTX WA,B08,C80,E-T70AT   AUX=3ETRE?
670             GOTO  T70CP
672  ETRE2      TESTX WA,B08,C4F,E-T70AT   INF AUX=3ETRE?
675             GOTO T70CP
677  T70AT      SETCH CW,B01,58            CW=ATTRIB?
679             GOTO MNGAFT
681  T70CP      SETCH CW,B01,70            SET CW=PRON
683             SETCH CW,B08,72                   PERSONAL
685  MNGAFT     CMPWD CW,LA,E-SETHER       CW=LA?
688             CMPWD CW,LES,E-SETTHEM     CW=LES?
691  SETIT      SETMA WA,IT                SET CW=IT AFT V
693             SMSG  'LE/LZ PERS/NEUTRAL PRON 001','LEXRTN'
712             GOTO  EXIT
714  SETHER     SETMA WA,HER               SET CW=HER AFT V
716             GOTO  EXIT
718  SETTHEM    SETMA WA,THEM              SET CW=THEM AF V
720             GOTO  EXIT
722  TF8        SETCH CW,B01,E8            SET CW-DETERM
724             SETCH CW,B08,36
726             GOTO LE
728  EXIT       SETTR CW
730             GOTO LE
```

Figure 4. 'LE' Lexical routine.

V. FRENSTRU

A change of source language from Russian to French required the development of the LEXRTN program in the previous chapter. The other program which remains to be developed is one to perform the structural analysis of the new source language. Numerous books have been written on the structure of different languages. In particular, for the French language, much effort and vast linguistic knowledge and insight were necessary to produce, for instance, such work as Lucien Tesnière's "Eléments de Syntaxe Structurale" (Ref 9), often quoted and used by workers in the Machine Translation and computational linguistic fields. Many others have contributed much to the study of French and a fairly extensive report would be required just to summarize some of the more pertinent ideas given there. This is not the purpose here, yet the fact needed mentioning lest the reader be tempted to accept what follows as anything but a small dent on the surface of the French language structure. The purpose of this study, it is recalled, is to attempt to adapt SYSTRAN to accept French as a source language. The only possibility of success of this project rests on making maximum use of SYSTRAN and changing the barest minimum of that system. English remaining the same, as the target language, dictates that the portions of the system dealing with that language not be changed. Thus it is that the 'FRENSTRU' program (French Structure) development, now introduced, is geared entirely to providing the necessary information for the target language programs to perform their task of synthesizing English sentences. It is not a structural analysis of French in the linguistic sense, although some computational linguists might discover some useful ideas in the approach. In any case the method is proved to work on the computer, is adapted to SYSTRAN, provides translation (admittedly in need of improvement) from French to English, and is open-ended for improvements.

The development of the 'FRENSTRU' program was greatly influenced by a chapter of K.V.H. Booth previously referenced (Ref 1). There she deals, in part, with the frequency of occurrences of the different parts of speech of the words of running French and English texts. The frequencies are compiled in groups of single occurrence (as, the number of times the noun part of speech occurs); of two occurrences (for example, how often a noun-adjective group is found), and so on up to groups of five occurrences. Her results led this writer to deal with the most often occurring parts of speech first and to proceed on this basis of frequency of occurrences hopefully to the end. However, if time required a cut-off then a stop would occur at a point where the remainder to be analyzed would not be of parts of speech of relative high frequency and so would cause the least damage. Another major advantage in directing the approach this way is that, the sequence of questions on a word is more likely to receive a "yes" answer sooner (causing an exit, in the program, out of a loop to a structural decision) if the question is biased toward an affirmative answer on the basis of what the word is most likely to be. The size of the text used in Booth's study introduces a shortcoming however when it comes to use it for guidance in the low-frequency occurrences. These low-frequency results were nevertheless used if for no other reason than that more extensive material was not available and also because the method developed on that basis admits, in any case, re-ordering when these new results become available. An increase in the number of words analyzed should not change drastically the order of occurrence of the high-frequency parts of speech. Thus her study points out that the first five most often occurring parts of speech in French are: noun, preposition, determiner, adjective and verb in that order. The noun governs the determiner and the adjective; the verb in the past or present participle form can act as an adjective; and the preposition generally introduces a noun, or a verb acting as a noun, complement. Because of the possibility of the verb acting as an adjective, which might affect the analysis of the noun, the verb is treated first resulting in a subprogram of 'FRENSTRU' called 'PREDIC'. The noun is then analyzed giving the subprogram 'NOMIN' and so on for the others. The 'FRENSTRU' program that is developed here is contained in Appendices I and J.

'PREDIC' Subprogram

As mentioned above the prime reason for dealing with the verb at this point is to eliminate the possibility of a past or present participle acting as an adjective. However, as progress is made through each sentence, for that purpose, the analysis takes advantages of other results as it goes and all the verbs are dealt with in 'PREDIC' and set as primary or secondary potential predicates. It will have been noticed in Appendix D, on coding, that the verb part of speech was subdivided into infinitives, past or present participles, auxiliaries and so on according to its specific function or its mode/tense. The principle used in 'PREDIC' is to seek the verbs in their particular form in the sentence, verify for and correct any person/mode ambiguities, and establish the verb as predicate, noun or adjective depending on its form or function. Just as in the case of the lexical routines a complete description of each step involved in the development of the subprogram would be quite lengthy and not too interesting to the reader not fully conversant with French, so the same format as used for the lexical routines will be employed here, as well as for the other subprograms below. That is, the main bulk of the development can be found in Appendix I, with "Comment Sheet" and "Flow Chart" on each sub-part of speech of the verb, and the actual subprogram can be found as part of 'FRENSTRU' in Appendix J. A detailed example here should suffice to make clear the approach used.

'PREDIC' Example

An interesting situation is when a finite verb is encountered and the ending look-up in the paradigmatic set table has resulted in a person/mode ambiguity. For the purpose of this example, assume the ambiguity is such that the verb appears in the analysis area with PØS = 04 (for finite verb), byte number 02 = A0 (for first and third person singular ambiguity) and byte number 03 = 98 (for indicative present, imperative and subjunctive present ambiguity). On the first page of the 'PREDIC' subprogram flow chart, in Appendix I, the reader can locate entry point "P4". Supposing all tests above this point have given negative answers then the next operation will look for and find the finite verb in question and branch out to PMA4. Turning to the flow chart titled "PMA4," the test for person/mode ambiguity is affirmative leading to the 'PERMODE' subroutine. Turning now to the 'PERMODE' subroutine flow chart, towards the end of the appendix, the reader will note that the PA0 branch will be taken on testing byte 02 for "A0" content. The next page has the flow chart for PA0 and indicates the procedure followed to resolve the person ambiguity which, when complete, returns control to the MODE entry in 'PERMODE' to test for mode ambiguity. The test here branches out to M98 which resolves the mode ambiguity either directly in the main branch or through the positive branches which introduces the further test M88. In either case the control is returned through EXIT to PMA4 at entry point PMA41 previously set up in preparation for the subroutine 'PERMODE'. Then, PMA4 sets the verb as primary predicate and returns control to 'PREDIC' at P4, so that more of the same can be sought. The procedure is the same for all the other tests of 'PREDIC' and the end result of this subprogram is that all types of verbs of the sentence are now set up as either potential predicates, nouns or adjectives. On running out of verbs, 'PREDIC' then passes control to 'NOMIN'.

'NOMIN' Subprogram

The purpose of this portion of the 'FRENSTRU' program is to examine the environments of the nouns to establish the governor-dependent indicators required by 'ESYNPRG'. The procedure here is the same as in 'PREDIC' except that now the establishment of relationships between words are more numerous and the build up of the synthesis array can be better appreciated.

The first step of the subprogram consist in locating a noun. This is done by scanning the sentence from left to right seeking for byte 01 containing hex. character "10_x", the code assigned to the noun part of speech. On locating a noun the 'NOMIN' subprogram can be considered to be engaged. The next step is to set a pointer on the word after the noun and to determine what function that word serves. The reader will notice on the flow chart page titled 'NOMIN', in Appendix I, that the series of tests, between AFTN3 and AFTN4, all lead to a series of branches named NA, NCMA, etc..., on an affirmative answer. These abbreviations indicate the type of sequence encountered, for example, NA means a Noun-Adjective sequence, similarly for the others. The order of these tests, once again, is based on Booth's double frequency occurrences so that an exit is achieved earliest. To continue with an interesting avenue the reader might assume that the word after the noun is an adjective leading to the branch named "NA". By going to the flow chart page titled "NA" the subsequent action can be appreciated. Here a gender and number check is made of the noun-adjective pair by calling a 'GENUMB' subroutine, included towards the end of the main program, and assuming the gender and number agreement is valid, an adjective-to-noun and noun-to-adjective relationship is indicated, by placing the word sequence number of the noun in byte 17 of the adjective and that of the adjective in byte 27 of the noun, two particular locations reserved for this right-to-left type of relationship. Once the adjective-noun relationship is established the pointer is moved over to the next word and the same procedure is repeated as if the adjective was no longer there. The noun stops looking at its right when it meets anyone of the following parts of speech: preposition, comma, coordinate conjunction, period, verb, another noun, relative pronoun, determiner, pronoun, adverb, subordinate conjunction or negation. On exit from the "after-noun" portion of the 'NOMIN' subprogram the noun starts looking on its left. More relationships are similarly established in this way with exit from this sequence on meeting one of: preposition, verb, no word, coordinate conjunction, adverb, subordinate conjunction or a comma. These types of words cause a stop to the analysis of the noun environment because, either no relationship can be established between them and the noun in the position they occupy, or the relationship is best established later in 'FRENSTRU'. For example, a noun followed by a determiner cannot be related to it, so the determiner is left to be picked up by the next noun. In addition to the relationships mentioned above, the 'NOMIN' subprogram also sets each noun as a "noun-phrase" (byte 100 = 01) and also indicates on all dependents that their function has been established (byte 39 = 01). These two additional settings are used later in 'FRENSTRU'.

Others

Of the five most frequent parts of speech, so far, four have been looked at leaving the preposition. Because of the particular function of the preposition it is relegated to a little later in 'FRENSTRU' and the adverb, pronoun, and one function of the coordinate conjunction parts of speech are handled at this point in the subprogram. These will not be detailed here as they amount to practically the same as before except for the linguistic differences which can be recognized in the flow chart of Appendix I. This brings this chapter to the 'PREPOS' subprogram. As realized the preposition is a very useful and necessary part of speech. This fact shows up clearly in Booth's study where, in both French and English, it ranks second in frequency. But, for translation purposes, whether human or by machine, it is also as troublesome as its frequency of occurrence is high. The SYSTRAN coding sheet for Russian indicates that the problem is not exclusive to French or English and that an elaborate system had to be developed to handle the situation there as well. An approach similar to the noun analysis was adopted at first to analyze the preposition. Unfortunately, time did not allow to perfect the analysis sufficiently to be included here and a relatively more simple subprogram is substituted for the time being. The method of development is the same for 'FRPREP' as for the others above and need not be detailed. At this point in the development of this study the cut-off mentioned earlier was reached. The remaining parts of speech could not be analyzed in detail so that the subprograms, part of 'FRENSTRU' in Appendix I, are the barest minimum to allow 'FRENSTRU' to run and to provide sufficient indicators in the synthesis array for the English synthesis program of SYSTRAN to function. Each of these latter subprograms can form the basis for a fairly extensive study by itself. The comment sheet attempts to indicate some of the avenues of investigation which could be taken in each case, but by no means exhaustively.

VI. Results and Conclusions

Results and Conclusions

The test translation consisted of four translation runs all made within two days. More tests were not possible because of the tight computer time schedule. The results obtained on the fourth run appear in Appendix L, where the input French sentences are each followed by two English translations. The second English translation usually undergoes a rearrangement sequence which is based on the left-to-right and right-to-left agreement pointers set up in the structural analysis program. At the time of this trial the REARRANG program of SYSTRAN was not functioning properly and had to be by-passed for this trial. This was unfortunate as it prevented the verification of some of the actions of the analysis programs such as the word-order rearrangement of the noun-adjective combination for example. The pluralization of SYSTRAN also failed for this test as is apparent from the translation. A brief study of the dumps obtained on each sentence also indicates that the two programs LEXRTN and FRENSTRU were either not called in at all or else did not function. The reasons for this are unknown at this time since a detailed analysis of the dumps would delay this report beyond the permissible dead-line, however a minor error in the supervisor program, which calls these analysis programs, is suspected.

Notwithstanding the above deficiencies, the purpose of this project, to show the adaptability of SYSTRAN to accept a different source language, was achieved to the extent possible within the tight bounds in time allotted for such a project as well as the very limited availability of computer time. Of significance here is the fact that the MACROS, the support programs and the pre-translation phase group of programs have performed well. The MACROS have compiled giving error messages which were corrected prior to this actual test. The support programs such as the idiom and dictionary edit programs have edited the SYSTRAN coding sheet format input cards and rejected illegal entries which were also corrected prior to this test. The idiom and dictionary update programs also produced the desired results of creating the idiom table and the dictionary. In the pre-translation phase category the preparation of the sentences for idiom and main dictionary look up was accomplished, the resorting of the input text in its original sequence and the establishment of the analysis area was done. Thus, although the effects of the two programs which were changed in SYSTRAN cannot be evaluated, the results obtained still provide a sound base from which to continue this study.

Recommendations

With the adaptability of SYSTRAN shown up to the actual translation phase, the following recommendations are made for immediate future activity in this area:

1. A detailed study of the dumps of the input text sentences to be made to discover the reason(s) for failure of the LEXRTN and FRENSTRU programs to be engaged.
2. Once the problem in 1 is resolved and the actions of the programs are felt each subprogram which gives unwanted or undesirable results can be applied independently to the same test corpus by removing all the other subprograms. In this way only the actions of the particular subprogram would be reflected in the dumps.
3. A new text can then be used as input for translation, the words not found (which are listed by BATR) can then be coded and the process as in 2 repeated.

4. When a sufficiently large dictionary is available then each subprogram, which reflects the function of individual part of speech, can form the basis for a detailed study of that particular part of speech.

For the more distant future the following possible avenues of investigation are submitted for consideration:

1. From the point of view of Machine Translation a considerable effort is required in coding a sufficiently extensive dictionary, idiom, topical glossary set to be of value in translating new text.
2. From the point of view of computational linguistics very minor adjustments can be made to the system to make it a valuable asset to research the source language and possibly also the target language by introducing dumping facility on that language.
3. From the point of view of modeling the cognitive aspects of the brain functioning in languages it is much too early to formulate any firm conclusions or directions. However, it appears that a detailed study of, and experimentation with, the dumps as provided by machine translation might prove useful to give indications of the underlying similarities between two seemingly divorced languages.

The study of one language from the structural or generative point of view is a highly interesting, necessary and useful area of activity. The study of two languages, as in machine translation, is certainly also necessary and useful as well as interesting. The compatibility between these two fields should be realized and accepted by the researchers in each, thereby providing a normalization of methods and a greater sharing and understanding of results. A system capable of translating two or more source languages into a same target language is bound to provide important clues about an underlying language, if such exists. SYSTRAN is a system, perhaps the only one, which appears to have the potential to become such a multilingual system.

Bibliography

1. Booth, K.V.H. "Machine Aided Translation with a Post-editor" in Machine Translation, edited by A. D. Booth. New York: Amsterdam, 1967.

2. Chapin, N. 360 Programing in Assembly Language, New York: McGraw-Hill Book, Co., 1968.

3. Lamb, S. M. and W. H. Jacobsen, Jr. "A High-speed Large-capacity Dictionary System", in Readings in Automatic Language Processing, edited by David G. Hays. New York: American Elsevier Pub. Co., 1966.

4. Locke, William, N. Scientific French. New York: John Wiley and Sons Inc., 1957.

5. Lufkin, J. M. "What Everybody Should Know About Translation". IEEE Trans., EWS-12:3-9 (May 1969).

6. Rondeau, G. (Translator). Initiation à la Programmation COMIT. Montreal: Carel, 1964.

7. Simmons, R. F. "Natural Language Question-Answering Systems: 1969", in Communications of the ACM, 13:28 (January 1970).

8. Tallman, O. H., II. The Classification of Visual Images by Spacial Filtering. Unpublished PhD Dissertation, Air University, Air Force Institute of Technology, Wright Patterson Air Force Base, Dayton, Ohio: June 1969.

9. Tesnière, Lucien. Eléments de Syntaxe Structurale. Paris: Klincksieck, 1969.

10. Toma, P. Development of Operational Russian-English System. Technical Report No. RADC-TR-68-331. Rome Air Development Center, October 1968. AD844168.

11. Yngve, V. H. COMIT, Programmer's Reference Manual. Cambridge, Mass.: The MIT Press, 1962.

12. -----. "MT at MIT, 1965", in Machine Translation, edited by A. D. Booth. New York: Amsterdam, 1967.

15. Yngve, V. "Gap Analysis and Syntax". IRE Trans. IT2-3:106-112 (1956).

André R. Gouin may be reached at : argo3119@yahoo.ca currently (August 2008).