1.1 The System in general

Since the MT-System of IBM Germany has already been described elsewhere [5] [6], we shall confine us to the aspects of the system which are substantial to our problem.

The system consists of two linguistically significant parts: a machine lexicon residing on a direct access device and a program package. The machine lexicon contains:

1. English words,
2. a code providing grammatical information about each word,
3. German equivalents.

In case of more than one possible German equivalent of an English word, there are - roughly spoken - as many separate lexicon entries in the machine lexicon as German translations. To put it in a more precise way: for an English homograph, redundant as it may be, there is a separate lexicon entry for each of its syntactic funtions, i.e. an English word like CHANGE has been entered in the lexicon as a noun CHANGE, with the German translation: ÄNDERUNG and also as a verb - CHANGE, with the translation VERÄNDERN. The order of entries is fixed.

While the machine lexicon provides each English word with one or more German equivalents, the program package operating on the lexicon develops the proper translation for the English input text.

The first step of processing is the lexical assignment, i.e. each English word is looked up in the machine lexicon and its first occurence in the lexicon is taken as the provisional translation [4] .

The following steps care for the proper choice of the German equivalents from those other alternatives provided by the lexicon, which remain accessible throughout the entire subsequent processing. In order to choose the correct equivalent, the programs rely on the original word order of the English input text and the codes which are primarily drawn from the lexicon.

Generally it can be said, that as the processing progresses, we have more and more information about the linguistic structures at our disposal, and we can make increasingly subtle decisions. It is obvious that to distinguish between

verbs and nouns is logically prior to the distinction between, say, abstract nouns and concrete nouns. (In praxi, however, the sequence of subroutines may not always be as simple as that.) Similarly, we can distinguish between transitive, reflexive and intransitive verbs, only after having established the category of verbs in general, i.e. after the solution of homographs like:

PLACE noun	PLACE verb
CHANGE noun	CHANGE verb

Although the disambiguation of syntactic homographs is by no means a trivial task, we shall not go into further details about its actual realisation.

For the following discussion we shall therefore presume, that occurring syntactic homographs have been solved.

2.0 The Problem

Consider the following sentence pairs:

(1a) *The company has increased the production.*
(1b) *The production has increased rapidly.*

(2a) *Register 3 returned the value unchanged.*
(2b) *The access arm returned to its original position.*

It is clear that the (a) and (b) sentences have to be translated in a different way:

(1a") *Die Gesellschaft hat die Produktion erhoeht* (1) .
(1b") *Die Produktion hat sich schnell erhoeht.*

(2a") *Register 3 brachte den Wert unveraendert zurueck.*
(2b") *Der Zugriffsarm kehrte zu seiner urspruenglichen Position zurueck.*

It is not difficult to see that the linguistically relevant criterion for the proper German translation is the presence or absence of a direct object-NP. For the German translation of (1b) the implicit object must be made explicit by the insertion of the German reflexive pronoun SICH. In the second example we have to select two different verbs to render both (2a) and (2b) correctly into German (2) .

2.1 A first approximation

As a first approximation the following scheme can be proposed (3) :

(R1') $\begin{bmatrix} V \\ [\pm \text{transitive}] \end{bmatrix} \Longrightarrow \begin{bmatrix} V \\ [+\text{transitive}] \end{bmatrix} / \_\_ \text{NP}$

(R2') $\begin{bmatrix} V \\ [\pm \text{transitive}] \end{bmatrix} \Longrightarrow \begin{bmatrix} V \\ [-\text{transitive}] \end{bmatrix}$

Note that (R1') and (R2') must follow each other immediately, otherwise the environment in (R2') must be made explicit.

Note also that verbs like INCREASE need not be mentioned at all: they can be marked in the lexicon as invariably [+ transitive] , to which the missing formal object should be inserted on the German side by an other rule:

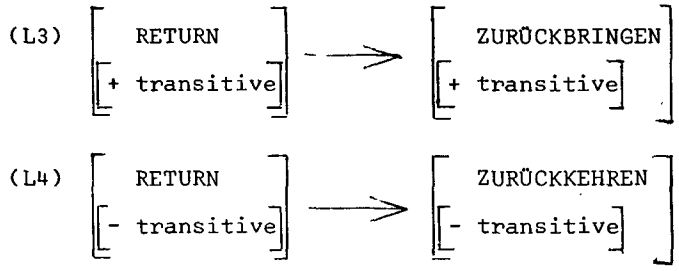
(R3') $\begin{bmatrix} V \\ [+ \text{transitive}] \end{bmatrix} \Longrightarrow \begin{bmatrix} V \\ [+ \text{transitive}] \end{bmatrix} \text{SICH} / \_\_ \text{X}$

Conditions: X contains no NP;
X may be 0.

As the final step, the correct German equivalent has to be determined by the English verb and the feature [+ transitive] or [- transitive] ; to secure the proper translation we have to provide appropriate lexicon entries for the rules:

(L1) $\begin{bmatrix} \text{MOUNT} \\ [+ \text{transitive}] \end{bmatrix} \longrightarrow \begin{bmatrix} \text{MONTIEREN} \\ [+ \text{transitive}] \end{bmatrix}$

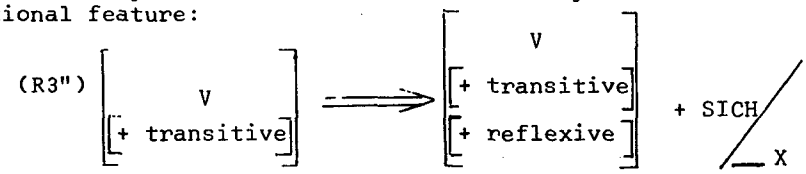
(L2) $\begin{bmatrix} \text{MOUNT} \\ [- \text{transitive}] \end{bmatrix} \longrightarrow \begin{bmatrix} \text{STEIGEN} \\ [- \text{transitive}] \end{bmatrix}$



In order to be able to cope with reflexives of variable verb stems like:

CHANGE — VERÄNDERN ⌣ SICH ÄNDERN
MOVE — ÜBERTRAGEN ⌣ SICH BEWEGEN

it is necessary to mark reflexives in (R3') by an additional feature:



Conditions: X contains no NP;
X may be 0.

2.2 Clause boundaries

This basic scheme has to be made somewhat more precise. Consider:

(3) *As data processing needs have increased, the basic card language remained the same.*

Without referring to the clause boundary after the word INCREASED, the following NP (= subject of the next clause) would be interpreted as an object belonging to the verb INCREASED.

Thus the environment in (R1') (and accordingly also in (R2') and (R3")) has to be further specified: in addition

to requirement, that the NP immediately follows the verb, we must also demand that the NP may not be separated from the verb by a sentence boundary. This means that the NP must belong to the same clause as the verb.

This can be done on the basis of the sentence analysis, which is motivated independently from the current problem, and the results of which remain accessible (4) . Thus the final set of rules would read as follows:

(R1) $\left[\begin{matrix} V \\ [\pm \text{ transitive}] \end{matrix}\right] \Longrightarrow \left[\begin{matrix} V \\ [+ \text{ transitive}] \end{matrix}\right] / \_\_ \text{NP} \quad X_S]$

(R2) $\left[\begin{matrix} V \\ [\pm \text{ transitive}] \end{matrix}\right] \Longrightarrow \left[\begin{matrix} V \\ [- \text{ transitive}] \end{matrix}\right]$

(R3) $\left[\begin{matrix} V \\ [+ \text{ transitive}] \end{matrix}\right] \Longrightarrow \left[\begin{matrix} V \\ [\pm \text{ transitive}] \\ [+ \text{ reflexive}] \end{matrix}\right] + \text{SICH} / \_\_ X_S]$

Conditions: X contains no NP;
X may be 0.

Symbol $X_S]$ means that we restrict the applicability of our rules to one and the same clause.

2.3 Passive-voice verbal constructions

Further more we should also be able to treat sentences like the following:

(4) *It has been returned to the initial position.*

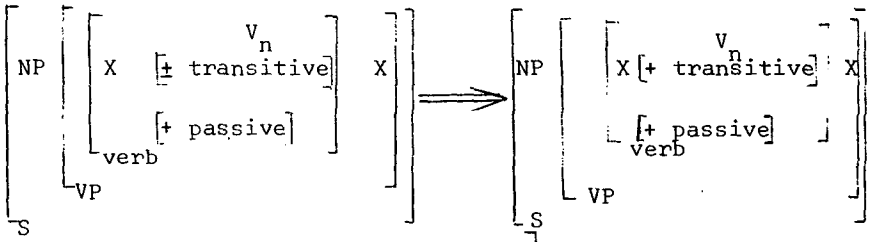
(4") *Es ist zu der urspruenglichen Position zurueckgebracht worden* (5) .

(5) *If the disk has been mounted, start the machine.*

(5") *Wenn die Platte montiert worden ist, starte die Maschine.*

In order to get the required feature [+ transitive] in spite of the physical lack of a direct object NP in the terminal string, we have to recognize passive-voice constructions as such.

The passive recognition should preceed (R1) and should discover the object of the main verb, otherwise (R2) treats sentences like (4) and (5) as intransitives, after which we have to eliminate the consequences of this misinterpretation. Presuming unique morphological marking of passive-voice the following recognition rule would be sufficient for the present purpose:

(R4) 

Conditions: $\begin{matrix} X \\ VP \end{matrix}$ contains no NP

X may be 0.

(R4) preceeds (R1).

For the further discussion note that the passive recognition unescapably involves hierarchical structures (6) .

3.1 Motion verbs

There are some subsidiary problems arising after the exchange of the transitive form of a multifunctional verb against its intransitive correspondence.

Consider:

(6) *The temperature has dropped rapidly.*

(6') *Die Temperatur hat fallen gelassen schnell.*

(6") *Die Temperatur hat gefallen schnell.*

Inspite of the correct translation of the main verb, sentence (6") is still false, because fallen - fiel - gefallen counts as a motion verb, and as such it takes the auxiliary verb sein in perfect tense. Therefore provision must be made to provide the proper auxiliaries for the motion verbs in perfext tense. The actual exchange of the auxiliary might be postponed to a later step, if the appropriate marking is provided by the lexicon.

E.G.

(L2') [MOUNT, [- transitive]] ⟶ [STEIGEN, [- transitive], [+ motion]]

(L4') [RETURN, [- transitive]] ⟶ [ZURÜCKKEHREN, [- transitive], [+ motion]]

Since the auxiliary selection has to be done for all motion verbs in perfect tense, i.e. also for the non-homographic type like

has arrived	ist angekommen
has gone	ist gegangen etc.,

it is practical to place the auxiliary selection behind (R1) - (R3) and even after the insertion of the improved, intransitive entries ((L2'), (L4') and the like):

(R5) [X HABEN X [V_n [- transitive] [+ motion]], verb] ⟹ [X SEIN X [V_n [- transitive] [+ motion]], verb]

Thus (R5) would automatically produce:

(6'") *Die Temperatur ist gefallen schnell.*

3.2 Order of composite verbs

Let us consider the order of primary elements of composite verbs like:

STOP —— STEHEN BLEIBEN

DROP —— FALLEN LASSEN

In order to be able to manipulate each unit separately it is expedient to set up two words on the German side:

STOP —— {STEHEN / BLEIBEN}

DROP —— {FALLEN / LASSEN}

At this point however we have to ask which of the two possibilities is the preferable order:

STEHEN or BLEIBEN ?
BLEIBEN STEHEN

When generating the final German word order, all composite verbs have to be inverted.

This inversion routine consists of a set of rules (7). The rule for subordinate clauses operates according to the following basic scheme:

(R6) SD: $\left[\begin{matrix} X \\ S \end{matrix} \right] - \left[\begin{matrix} X \\ verb \end{matrix} \right] - V_n - \left(V_{rest} \right) \Longrightarrow$

SC: 1 2 0 4 + 3

Conditions: S is a subordinate clause (i.e. it is dominated by an other symbol);

X may be an empty string;

V_n is the last part of the verb.

An inversion is necessary in any case independently of the present issue, also for sentences like (1) - (4), for which we have tacitly assumed the existence of a verb inversion routine [7].

We should like to translate correctly sentences like:

(7) *It causes the machine to stop.*

(7') *Es veranlasst die Maschine zu* {*stehen* / *bleiben*}

At the first sight it appears that sentence

(7a') *Es veranlasst die Maschine zu stehen bleiben.*

sounds better than:

(7b') *Es veranlasst die Maschine zu bleiben stehen.*

But if we make the verb inversion rule (R6) also for (7) applicable the verb will be automatically inverted. Thus (7b') is preferable to (7a'), since (R6) yields the correct translation out of (7b'), but not out of (7a'). All what we have to do is to treat the infinitive zu bleiben as a unit:

(7") *Es veranlasst die Maschine zu bleiben stehen.*

1st application: n=2 — $[_{S}\ X\ [\ X\ [_{verb}\ V_n\ ]\ ]\ ]$

2nd application: n=1 — $[_{S}\ X\ [\ V_n\ ][\ V_{rest}\ ]\ ]$

(7"') *Es veranlasst die Maschine stehen zu bleiben.*

(R6) applies first vacuously, i.e. it effects only the marking, but not the physical sequence of the constituents, since V_{rest} is an empty string. The variable zu bleiben is the remainder of the verb constituent after the separation of the last independent morphological unit. At the second application, there is a V_{rest} produced by step 1, (R6) places the now last - and only - member of the verb constituent after V_{rest} (8) .

These considerations show also the crucial importance of the ordering of rules and subroutines. We can solve the sequencing of the newly inserted compound verbs with minimal effort, if we have the verb inversion routines run after the disambiguation of the transitive/intransitive verbs.

4.0 Theoretical implications

Finally I should like to point out a theoretical implication of the present problem which might have bearings on linguistic analysis in general.

In a formal linguistic analysis it is unavoidable to set out from the information directly available in the terminal string. This primary information is organised necessarily in a linear fashion.

As the analysis proceeds the possibility emerges to organize the linguistic information in a non-linear, hierarchical way. The results of the analysis steps can be used to find and mark the boundaries of the clauses, to recognize the individual constituents within the larger units, i.e. to build up a tentative constituent structure.

As the disambiguation of verb with multiple meaning shows, we have to rely on hierarchically organized information in order to make the proper choice between transitive and intransitive usages even in this rather simple case. To solve the polysemy of the verbs in questions, we have to be able to find out that an NP is dominated directly by a VP and that this VP dominates directly a verb, which is in turn marked in an specific way.

Our transformation terminology, and the fact, that the systems lends itself readily for such a reinterpretation suggest our conclusion: We claim that the MT-system described above uses transformational rules.

This claim is supported

1. by the hierarchical organization of grammatical information, which is the computational representation of phrase markers (trees).

2. - and this is the main point - by our reinterpretation of the computational algorithms as mapping of phrase markers into phrase markers [2] [3] .

This implication seems interesting for the following reason. The MT-system of IBM Germany has grown out of the intention to create a translating tool for practical purposes. The system was developed in a basically empirical manner; problems were attacked at points, where we hoped to find the easiest way to the solution. A general frame of action has been invented, rules have been worked out and ordered coding considerations have been made, but it is only in retrospect that the theoretical evaluation of the system has become possible.

On the whole the system, heterogeneous as it is, resembles the fulcrum type of MT-scheme, operating on a different language pair as Garvin's system [1] . We have tried to show, however, that even the empirical approach felt the need for multidimensional structuring, and certain rules set up on empirical considerations - e.g. those treating verb with multiple meaning - turn out to be mappings of phrase markers into phrase markers in the transformational sense.

Summary

In the past years the Department of Basic Research IBM Germany developed an experimental Machine Translation System to translate technical texts on data processing and electronics from English into German. This MT-system displays some resemblances with the Fulcrum-type of systems, but at least some of its subroutines contain also transformational devices.

The treatment of verbs with multiple meaning of the type CHANGE - VERÄNDERN - SICH ÄNDERN is one of these cases. The correct recognition of the passive voice and direct objects presuppose hierarchically organised and permanently accessible syntactic markers.

While presenting this rather special case it has been attempted to show some further aspects as well as the general functioning of the system.

The MT-system represents a team work: the author of the present paper thanks especially Mrs. Brigitte Schirmer and Mr. Reinhart Herzog for reading and criticizing the first version of the manuscript.

Footnotes

(1) Our German translation may deviate occasionally, e.g. with regard to prepositions, from the common usage, since we draw the examples from our actual output.

(2) The following verbs display the same regularity:

Typ 1: transitive∼reflexive

CHANGE	VERÄNDERN	SICH ÄNDERN
REVOLVE	DREHEN	SICH DREHEN
INCREASE	ERHÖHEN	SICH ERHÖHEN
DECREASE	VERMINDERN	SICH VERMINDERN
MOVE	ÜBERTRAGEN	SICH BEWEGEN
etc.		

Typ 2: transitive∼intransitive

MOUNT	MONTIEREN	STEIGEN
RETURN	ZURÜCKBRINGEN	ZURÜCKKEHREN
CONTINUE	FORTSETZEN	FORTFAHREN
DROP	FALLEN LASSEN	FALLEN
etc.		

(3) In our notation symbols refer to code information, provided primarily by the word assignment routines, and - as we have already pointed out - improved by the syntactic component of the system. We differentiate category symbols from features by putting the latter in [ ]. The symbol X denotes strings having no relevance to the rules.

(4) Although the description of the voluminous analysis is beyond the scope of the present discussion, its concept influences also the treatment of verbs with multiple meaning. E.g. in (R1') we must know whether the constituent NP comprises initial adverbs on not. To dinstinguish an initial adverb dominated by an NP from an adverb merely preceeding on NP (but not dominated by it) is extremely complicated. If this distinction is not made, adverbs standing between verbs and NP-s must be jumped over.

(5) We cannot deal here with all problems arising, while translating the sentences. We shall return to the problem how to generate the correct German word order in 3.2.

(6) The actual machine realisation does not affect the basic properties of the passive recognition (R4); since in the actual system English verbs, which may be transitive and

intransitive, are provided uniformly with a transitive German correspondence in the first place, it is enough to recognize the passives and approve the given choice by taking no action. The actual recovery of the deep object of passive sentences seems to be redundant, since English passives correspond normally to passives in German.

(7) The rule appying to main clauses is more complicated. Therefore we have chosen the simpler rule for subordinate clauses, which suffices to illustrate the principle of verb conversion.

(8) A similar reasoning speaks for separating prefixes from main verbs and placing prefixes after the verb stems: BRINGEN ZURÜCK, KEHREN ZURÜCK usw. The entries in machine lexicon are actually in this sequence.

Bibliography

[1] Garvin, P.L.: The Place of Heuristics in the Fulcrum Approach to Machine Translation, in: LINGUA 21 (1968).

[2] Petrick, S.R.: A Recognition Procedure for Transformational Grammars, M.I.T. Doctoral Dissertation 1965.

[3] Rosenbaum, P.S.: English Grammar II, IBM Research, Yorktown Heights 1968 - RC 2070.

[4] Schäfer, G. - Wenzel, G.: Die Wortzuordnung - erste Stufe der mechanischen Übersetzung, in: Sprache im technischen Zeitalter, Heft 8 (1963).

[5] Schirmer, B. - Bâtori, I.: Machine Translation at Basic Research of IBM Germany, Proceedings - First International Symposium - Dec. 9 - 12, 1968, Tel-Aviv, Society of Technical Writers & Publishers Israel Chapter, Lod Airport.

[6] Schirmer, B.: Erfahrungen bei der Entwicklung eines maschinellen Übersetzungssystems, in: IBM Nachrichten Heft 194, Jg. 19 (1969).

[7] Schirmer, B.: Ausarbeitungen zur Ermittlung der englischen Satzstrukturen in Hinblick auf die Übersetzung ins Deutsche I - II., 1965 (unpublished paper).