It Would Be Much Easier If WENT Were GOED

Dan TUFIS
Institute for Computer Technique and Informatics
8-10, Miciurin Bd., 71316 Bucharest 1, Romania
Tel. 653390, Telex 11891-icpci-r

ABSTRACT

The paper proposes a paradigmatic approach to morphological knowledge acquisition. It addresses the problem of learning from examples rules for word-forms analysis and synthesis. These rules, established by generalizing the training data sets, are effectively used by a built-in interpreter which acts consequently as a morphological processor within the architecture of a natural language question-answering system. The PARADIGM system has no a priori knowledge which should restrict it to a particular natural language, but instead builds up the morphological rules based only on the examples provided, be they in Romanian, English, French, Russian, Slovak and tha like.

1. INTRODUCTION

For highly inflexional languages, encoding all word forms into a word lexicon (declarative morphology approach) appears to be a poor solution not only due to a great redundancy (which for some languages is prohibitive (Jappinen,1983)) but also with respect to some theoretical aspects (as for instance descriptional adequacy (Wehrli,1985)).

Within an inflexional morphology environment (Alshawi,1985), we propose a procedural approach based on automatically acquired flexioning paradigms. The paradigmatic model is incorporated into an experimental system called PARADIGM, which is intended to (partially) replace the acquisition and modelling part of our MORPHO lexicon management system (Tufis,1987a) incorporated by the IURES environment for building natural language applications (Tufis,1985).

The aim of our work is twofold: to obtain a sound linguistic tool for word-forms analysis and synthesis (which in case of highly inflexional languages is by no means a trivial task) and to provide for a psychologically motivated behaviour of such a system in dealing with unknown words. In the following, we shall dwell on the technical issues connected to the first of the above two mentioned motivations. With respect to the second one maybe it is worth saying that when PARADIGM lacks appropriate or complete knowledge it is supposed to act the same way a child or a foreign speak partner does. That is, for instance, to say "goed" or "womans" if the corresponding irregularity is unknown. The reason for such a decision stems mainly from our attempt to study in parallel with the implementation, the effectiveness of language learning based on informal examples. From the linguistic engineering point of view, the purpose of the system is stated very pragmatically, that is to ease and speed up as much as possible the building of language specific morphological knowledge bases without (too much) help from theoretical morphologists, experts on the language concerned. It is not always easy to find appropriate written material, not to speak about human experts, presenting in a rigorous manner (as imposed by computer applications) the rules and peculiarities of word structuring in different languages.

PARADIGM was conceived to overcome, at least partially, these difficulties and to provide a handy tool for immediate verification of specific rules validity. As the general situation is with learning systems, different copies of PARADIGM may be used in parallel and finally merge the individually developed knowledge bases. This is beneficialy not only with respect to the development time but also with respect to the linguistic coverage.

Architecturally, PARADIGM was influenced by DISCIPLE (Tecuci, 1988) in the sense that the behavioral dichotomy "apprentice-expert" was incorporated into its implementation. However, due to the more specific task, the technical solutions adopted in PARADIGM for knowledge acquisition are different, being problem oriented.

2. DEFINITIONS

2.1 MORPHOLOGICAL MODEL

We call a morphological model the tuple: MM = (C,SC,M,V,F1,F2,F3,P) where C is a set of categories: C = $\{c_1,...,c_i\}$; SC is a set of sub-categories of the categories in C: SC = $\{sc_1,...,sc_j\}$;

M is a set of features of the sub-categories in SC: $M = \{m_1,\ldots,m_k\}$; V is a set of values which features can take: $V = \{v_1, \ldots,v_m\}$; F1 is a function defined on C, taking values in the power set of SC: F1: C---> PS(SC); F2 is a function defined on SC, taking values in the power set of M: F2: SC---> PS(M); F3 is a function defined on M, taking values in the power set of V: F3: M---> PS(V); P is a subset of the Cartesian product $C \times SC \times P(M) \times P(V)$ so that $\forall p_i = (c_i,sc_i,M_i,V_i) \in P$ the following are true: $c_i \in C$ & $sc_i \in SC$ & $M_i \subset M$ & $V_i \subset V$ & $sc_i \in F1(c_i)$ & $M_i = F2(sc_i)$ & $M_i = \{m_{i1},\ldots,m_{ik}\}$ & $V_i = \{v_{i1},\ldots, v_{ik}\}$ & $\forall q \in [1,k]$ $v_{iq} \in F3(m_{iq})$. P is called the paradigmatic flexioning space of the morphological model MM. For instance a point of P in a certain MM might be:

(noun common-noun
(gender number case articulation)
(masculine plural genitive definite))

2.2. THEMATIC FAMILIES

We call a thematic family (TF) the set of all word-forms of a given (lemma) word, obtained by grammatical inflecting: $TF = \{W_1,\ldots,W_m\}$. Let us consider a TF to be always lexicographically sorted. Let $<X>$ denote an arbitrary string of characters and $<X><Y>$ the string obtained by concatenating the substrings $<X>$ and $<Y>$. We say that a TF is regular iff there is a q-letter substring $<R_q>$ called root, common to all the m words of TF so that:

1a)$\forall i <W_i> \in TF$, $<W_i> = <R_q><e_i>$

1b) $<R_q>$ is the longest substring with the property 1a)

1c) q > = low-limit (an integer varying from language to language)

1d) $\forall i \in [2,m\text{-}1]$ the subsets $\{<W_1>,\ldots,<W_i>\}$ and $\{<W_{i+1}>,\ldots,<W_m>\}$ give the roots $<R_{q1}>$ and $<R_{q2}>$ with $<R_{q1}>$ being a substring of $<R_{q2}>$ or at most equal to $<R_{q2}>$.

The remaining part of a word in TF after removing the root is called an ending (we use the term 'ending' to include both desinences and suffixes). The list of all endings obtained from a TF is called a paradigmatic endings family (PEF).

A thematic family is called partial regular if there is a partition of $TF = \{TF_1,TF_2,\ldots,TF_k\}$ so that:

2a) $\cup TF_i = TF$ & $\forall i,j(i \neq j)$ $TF_i \cap TF_j = \emptyset$

2b) $\forall i$ (TF_i is regular & $CARD(TF_i) > 1$).

According to the above definition, a partial-regular TF will be characterized by k roots. A thematic family which is neither regular nor partial-regular is called irregular.

In the following, in order to simplify notations, when referring to strings of characters, we use angular brackets only if we need to outline a composition/decomposition of a word-form.

A central notion of our approach is that of flexioning paradigm. Its meaning is similar to that used by most of the morphologists.

We define a flexioning paradigm Q as a list of pairs: $Q = \{(e_1\ p_1)(e_2\ p_2)\ldots(e_k\ p_k)\}$ where 'e_i' are endings extracted from a thematic family (irrespective of their regularity) and 'p_i' are appropriate points in P (the appropriateness will be revealed in the fourth chapter).

2.3 UNINTERPRETED LEXICON

Let LS be a set of words obtained from the union of K thematic families, called a lexical stock: $LS = TF_1 \cup TF_2 \cup \ldots \cup TF_k$. We call an uninterpreted lexicon of the word stock LS a set $UL = \{R_1,R_2,\ldots,R_p\}$ so that for any $i \in [1,p]$ R_i is a root of a certain TF_j in LS. The mapping
I: LS ---> PS(UL x P) is called an interpretation of an UL within a morphological model MM (recall that P is a paradigmatic flexioning space of a certain MM). Let us observe that I mapping allows a word to be ambiguously interpreted, which is quite natural at the level of isolated word-form analysis. Such a common ambiguity, for instance, is figured out by the Romanian word "modul", which may stand either for the unarticulated nominative/accusative form of "modul" (module) or for the articulated nominative/accusative form of "mod" (mode, manner). The I mapping abstracts the process of word-forms analysis. The abstraction of the reverse process, the generation of word-forms, is represented by the mapping G defined as follows: G: ULxP ---> LS. As opposed to I, G is a univoque function, that is for a given root and a specific point in the paradigmatic flexioning point P, a unique word-form will result.

3. BUILDING A MORPHOLOGICAL MODEL

To build a morphological model the designer starts by specifiying the categories of interest in his/her application. The traditional categories are NOUN, ADJECTIVE, VERB, PRONOUN and so forth, but by no means this categorial system is obligatory (for instance one might think of using semantically flavoured categories such as OBJECT, PROPERTY, ACTION, STATE, ANAPHOR etc).

For each defined category in C, the designer will be asked to provide the desired sub-categories (for instance COMMON-NOUN and PROPER-NOUN for NOUN). This activity is equivalent in the formal model to defining the SC subset and the F1 function. Further, for each sub-category in SC the system asks the designer to enter the specific features along which the inflexional behaviour of the words gets relevant. With Romanian language for instance, while number, case and enclitic articulation are relevant for COMMON-NOUN, for feminine PROPER-NOUN only the case is significant (but this is not always true: the feminine proper-nouns ending in a consonant, whatever their etimology, do not flexate at all). By entering all sub-category-features associations, the system is implicitly provided with the M set and F2 function. Finally, for each feature in M, the designer will be asked to define the possible values the current feature may take (e.g. 'singular' and 'plural' for the 'number' feature). When the list of features is exhausted, the system has already learnt the V set and F3 mapping. At this point, the activity of the designer is theoretically finished and it is the system itself which will generate, based on these definitions, the paradigmatic flexioning space (P), thus accomplishing the MM internal representation. From this internal representation, the system generates for each defined sub-category a graphic tabular menu (we call it an Acquisition Scenario AS) partially filled in. The only blank column in an AS is called WORD-FORM column and is accessible for writing in by the trainer (tutor) of the system. Each line in an AS is filled (except the last field corresponding to the WORD-FORM column) by the information uniquely identifying a point in P.

4. KNOWLEDGE ACQUISITION

When the tutor chooses a defined sub-category of a category in C to be exemplified, the system answers by displaying the associated acquisition scenario. What the tutor is asked to do is to fill in the blanks the WORD-FORM column with the inflected forms of the thematic word. Each word form must obey the restrictions imposed by the combination of the feature values displayed on the line which the tutor is writing in.

Once the WORD-FORM column of the current AS completely filled in, the root detection phase is activated. The word-forms are ordered lexicographically with the provision that the initial association: $((c_i\ sc_i\ M_i\ V_i)\ W_i)$ will be remembered. In the general case of a partially regular TF containing n word-forms the result of the root detection phase is represented as follows:

$LA = (((e_1 \ldots e_k) root_1) \ldots ((e_m \ldots e_n)\ root_i))$.

The n endings in the above list inherit the morphological features which are associated with the word forms which they were extracted from. That is if the word $W_i = root_i + e_i$ was associated with $p_j = (c_j\ sc_j\ M_j\ V_j)$ then e_i would also be associated with p_j. As a result, a possible new flexioning paradigm appears: $Q = ((e_1\ p_1)(e_2\ p_2) \ldots (e_n\ p_n))$. While p_i are all distinct, this is not obligatory the case for the endings. The Q paradigm is looked for in a list of already known paradigms and if not found there, is marked for interning. To interne a paradigm means to integrate it into an associative structure (a discrimination tree) appropriate for word-form morphological analysis (see further). With the generation of word forms, the above representation is very suitable (Tufis,1988). A paradigm is interned immediately after its marking only if it was learnt from a regular TF. Otherwise, this process is delayed until the roots of the TF are processed. The discrimination tree internally represents all the known endings and their morphological feature values. Its nodes are labelled with letters appearing in different endings. A proper ending is represented by the concatenation of the letters labelling the nodes along a certain path, starting from a terminal node towards the root of the tree. Due to the retrograde strategy used in our system, possible endings which are searched for in a word from right to the left, are checked in the discrimination tree from top to bottom. This explains the ordering of the label letters in the tree. A terminal node (T-node) is not obligatory a leaf node because of the possibility of inclusion of one ending into a longer one (the reverse is always true). All T-nodes are associated with the paradigmatic information specific to the ending which they stand for. This information is represented by a list of pairs: $((Q_1\ p_1)(Q_2\ p_2) \ldots (Q_m\ p_m))$ where Q_i are paradigm identifiers and p_i are (identifiers for) points in the paradigmatic flexioning space P. If a T-node (hence an ending) has associated more than a single pair (Q p) it is called extrinsically ambiguous. Another type of ambiguity is induced by the endings containing shorter embedded endings. We call such endings intrinsically ambiguous. Let us suppose two endings <X> and <Y> so that <X> may be written as <Z><Y>. In case of a word-form <A><Z><Y> without additional information one cannot definitely decide if the word should be segmented as <A><Z>+<Y>or as <A>+<Z><Y>. For both types of ambiguities there are sound methods of resolution if the decision procedure has access to the root lexicon or to some syntactic rules.

Anyway, for intrinsically ambiguous cases, our system has found out that for Romanian, in almost all cases the segmentation with the longest ending is the correct one. For extrinsically ambiguous endings, the system uses some statistics, extracted from the training data, which proved to be valuable. For instance, the system updates for each paradigm, a so-called local counter with each new thematic family behaving according to that paradigm. The value of this counter, specific for each paradigm is considered in sorting the interpretations of an ending :((Q_1 p_1)...(Q_r p_r)). According to this sorting, an interpretation (Q_i p_i) is considered more likely than another one (Q_j p_j), if in the lexicon there are more roots "belonging" to Q_i than to Q_j. This preference heuristics does not take into account the frequency of the words in running texts but only their paradigmatic classification. We plan to introduce the "dynamic counters" which are supposed to provide qualitative estimation based on word-forms frequences. It is clear that in order to provide valuable preferences, the values of the static/dynamic counters must result from large sets of examples and running texts. This requirement may be fulfilled by using in parallel many PARADIGM incarnations and finally by merging their outputs. It is important to note that the preference heuristics we talk about are intended only for getting a plausability ordering criterion for the possible interpretations of an ending or segmentations of an word-form. It means that no interpretation variant is rejected at this level, so that if a preferred (according to the preference heuristics) interpretation or segmentation was wrong, the correct one may still be found.

Roots processing and eventually paradigms modification or absorbtion (see further) are based on some similarity criteria. If no similarity is detected between the roots of a TF, the corresponding paradigm, if marked as new, is interned as it was initially constructed. But if the roots are similar, the system tries to reduce differences between them, either by modifying the inflexional paradigm or by inferring rules for root modification. The first approach is generally taken if the differences between roots appear at their boundary with the endings. The second method is usually tried in case of differences appearing inside the roots. The similarity criteria are declaratively specified, so that it is easy to modify, augment or adapt them to specific needs. The notion of similarity, as used in our approach, is very simple. We have developed a similarity description language in which one may describe the conditions under which two strings are to be considered similar. With the current version of the system, we use only three simple similarity rules:

s1) <X>?<Y> ≅ <X><Y>

s2) <X>! ≅ <X>?

s3) <X>?<Y>! ≅ <X>?<Y>?

In the above rules the metasymbol <X> stands for an arbitrary string, the question mark for zero or one letter, the exclamation mark for exactly one letter and ≅ for the similarity relation. Their readings are:

rs1) two strings are similar if they differ by at most one embedded letter (calculatoAr ≅ calculator);

rs2) two strings are similar if they are the same except the last letter of one or both of them (copiL ≅ copil);

rs3) two strings are similar if they differ by at most one embedded letter and by the last letter of one or both of them (fereAstrA ≅ ferestrE).

Actually, the similarity description language is more powerful than it is suggested. For instance, one may impose restrictions on an <X> construction such as minimal or exact number of characters in X, prosodic restrictions such as presence or absence of accent, a.s.o . If two roots are similar, the system attempts to generalize their similarity beyond the particular TF currently processed. The similarity between two roots is necessary but not sufficient for making a generalization. What is needed, is an explanation, in terms of morphological features, accounting for root modification. This explanation, if found, will be used as a precondition for the root modification rule to be synthesized. The explanation justifies the difference between the two roots (of the same TF), and consists of discriminant descriptions (in terms of morphological features) of the endings associated with them. In the current version of the system, it looks for the morphological features which have the same value for all the word-forms obtainable from the first root and another different value for all word-forms derived from the second root. For instance with the similar roots 'copil' and 'copii' (child), the system discovered that all the forms in singular are produced by the first root while the second generates all the plural forms.

Using this fact, the system built the following rule, entering only one root (copil) in the lexicon:

"If a root X behaves according to the paradigm Q39 and its last letter is 'l' then in all plural forms 'l' must be replaced by the letter''i'.

The "generative" flavour of this rule should not be misleading: that is, one must not infer that it is good only for generation. The same rule applies to analysis:

"If a root was discovered according to the paradigm Q39 and its last letter was 'i', the root may be recorded in the lexicon with its 'i' replaced by the letter 'l'".

As more data sets are provided the rules may be generalized further in order to cover the new cases.

We said before that the internalization of a marked paradigm was delayed until the roots of a partial TF were processed. As we shall see in the example below, the delay is justified by the possibility to alter the initial endings (hence the paradigm) in order to minimize the differences between the considered roots. A paradigm modification may appear if the last letter from each of the roots taken into account is transferred in front of all their corresponding endings (recall the LA list in the beginning of this chapter). If the system finds no feature-based justification for root modification and if the difference between the roots is given by their last letters, it decides to transfer these "faulty" letters into the appropriate endings, thus "regularizing" the TF. As a side-effect the initial paradigm is modified and in case the new one is already known the decision is considered sound and the older paradigm is forgotten. If the new paradigm is not known to the system then both paradigms (the initial and the modified ones) are kept until further evidence will allow the system to choose among them. If no such evidence is obtained in favour of one or another paradigm, it will be the task of the knowledge base designer to decide on the matter.

Let us follow on an example the process of learning a root modification rule. Consider that the trainer provided the thematic family for the thematic word "fereastra" (window). The root detection process will generate the following segmentations:

fereastra + ∅ (∅ stands for the null ending)
fereastra + ∅
ferestre + i
ferestre + ∅
ferestre + le
ferestre + ∅
ferestre + lor
ferestre + ∅

There are identified two roots: 'fereastra' and 'ferestre'. According to the rule s3) they are similar, with <X> and <Y> bound to 'fere' and 'str' respectively. The fault letters are associated with their appearance context: >e|a|s<, >r|a| and >r|e|. The notations are interpreted as follows:

">e" = = an 'e' preceded by some other letters;

"|a|" = = the 'a' fault letter;

"s<" = = an 's' followed by some other letters;

">r|a|" = = the final 'a' preceded by 'r';

">r|e|" = = the final 'e' preceded by 'r'.

The first decision made in order to minimize the differences between the two roots is to transfer the last character of them into endings, thus resulting the segmentations:

fereastr + a
fereastr + a
ferestr + ei
ferestr + e
ferestr + ele
ferestr + e
ferestr + elor
ferestr + e

A second step towards difference elimination is to consider the deletion of the 'a' letter between <fere> and <str>. But because this operation does not contribute to paradigm modification it must be generalized (if possible) as a rule for root modification. By inspecting the morphological features of the word-forms, the system finds out that the root 'fereastr' is characterized by the feature values: feminine, singular and nom-acc, while the root 'ferestr' is characterized in all its appearances only by the 'feminine' feature. Because 'feminine' value is common to all word-forms of the thematic family, it is considered irrelevant with respect to root modification. Moreover, no word-form derivable from the 'ferestr' root has attached the "singular + nom-acc" feature values combination. Therefore, this is taken as a possible condition for the faulty letter deletion and the synthesized rule is the following:

RMR1){<X> = >e|a|s<& PARADIGM = 'P00007' & NUMBER = 'singular' & CASE = 'nom-acc'} = = >{¬[NUMBER = 'singular' & CASE = 'nom-acc'] = = > >es< }

The reading of this rule is: "If a root of a word-form which flexions according to the P00007 paradigm, in singular and nom-acc, contains the embedded string "eas", then for all combinations of morphological features not containing both singular and nom-acc values, the 'eas' string is replaced by 'es'".

Let us notice that the rule is more specific than it should be, imposing that all eligible words behave according to the P00007 paradigm and requiring the letter 's' to follow the diphtong 'ea'. But the system cannot infer more from this single example. If provided with another example, let's say 'ceapa' (onion), with a similar behaviour the system synthesizes a rule very alike to RMR1:

RMR2){<X> = >e|a|p<& PARADIGM = 'P00007' & NUMBER = 'singular' & CASE = 'nom-acc'} = = >{¬[NUMBER = 'singular' & CASE = 'nom-acc'] = = > >ep< }

The only difference between RMR1 and RMR2 is the condition that the diphtong 'ea' must be followed by 's' and 'p' respectively. By considering this condition a particular one, the system drops it and obtains a more general rule subsuming both previous ones:

RMR3){<X> = >e|a|<& PARADIGM = 'P00007' & NUMBER = 'singular' & CASE = 'nom-acc'} = = >{¬[NUMBER = 'singular' & CASE = 'nom-acc'] = = > >e< }

The rule RMR3 is still too specific. The processing of the thematic family for the word 'seara' (evening) produces a further generalization of RMR3. Firstly, the system generates the following rule:

RMR4){<X> = >e|a|<& PARADIGM = 'P00008' & NUMBER = 'singular' & CASE = 'nom-acc'} = = >{¬[NUMBER = 'singular' & CASE = 'nom-acc'] = = > >e< }

The difference between RMR3 and RMR4 is made by the restriction that the flexioning paradigms are required to be P00007 instead of P00008. To generalize these rules, the system investigates the feature values of the two involved paradigms. Their common properties are SC = COMMON-NOUN, GENDER = FEMININE, so the system is able to propose a new rule subsuming the RMR3 and RMR4 rules:

RMR5){<X> = >e|a|<&SUB-CATEGORY = 'common-noun' & GENDER = 'feminine' & NUMBER = 'singular' & CASE = 'nom-acc'} = = >{¬[NUMBER = 'singular' & CASE = 'nom-acc'] = = > >es< }

Because generalization correctness over incomplete data cannot be guaranteed, each synthesized rule has two associated lists, one of them containing positive examples (initially only the prototype root which generated the rule) and the other one containing exceptions (initially empty). A similar point of view, that is attaching exception lists to general rules, may be found in (Bear,1988).

The roots are entered into the root lexicon. For partial regular thematic families, the two or more roots are linked together bidirectionally. The first of them, in lexicographic order, is attached to the relevant common morpho-lexical information: paradigm name and the label for the semantic description. This information is inherited by all linked roots. There is also root specific morphological information such as selectional restrictions and phonemic patterns. The selectional restrictions are contributed by the system and they refer to the constraints to be satisfied in order that a root be selected in a word-form generation. For the regular modifying roots, links to the rules they obey and the position(s) in the root where letter insertion or deletion is to be performed are also recorded in this field.

The lexicon building side-effect of the tutorial sessions is not the main interest of the research reported here (for this purpose we developed the MORPHO lexicon management system (Tufis,1987a)).This feature was implemented only for testing the PARADIGM system in learning and using learnt knowledge. Also, we were interested in experimenting some generation strategies at the level of morphology (for instance choosing the least ambiguous or the more common used root from a synonimy set - see (Tufis,1988)). It was possible, in this way, to test the functionality of PARADIGM without coupling it to MORPHO, operation which would have required a greater programming effort. The embedding of PARADIGM into MORPHO is planned for the immediate future.

At the end of the system's apprenticeship is activated a processing phase which we call the paradigmatic absorbtion. A paradigm Q_1 may be absorbed into another paradigm Q_2 iff:

ab1) they describe the same subcategory,

ab2) for each ending 'e_{1i}' from Q_1 and the corresponding ending 'e_{2i}' from Q_2 the following are true:

'e_{1i}' is a suffix of 'e_{2i}': $<e_{2i}> = <x> <e_{1i}>$ and the $<x>$ preffix in 'e_{2i}' exists as a suffix in all the roots in the lexicon which, from the flexioning point of view, behave according to Q_1.

The implementation of paradigms absorbtion is computationally motivated: firstly by decreasing the number of paradigms, the search space is narrowed and consequently word-form processing time improved; secondly, by lengthening the endings, they become more discriminating and therefore the ambiguity is reduced. In Romanian the case is that the longer an ending, the less ambiguous its interpretation. For instance the 'i' ending has 19 possible interpretations (in our model), while the ending 'ului' has only one. We think that this is a general property with inflexional languages and therefore we consider paradigmatic absorbtion not to be specific for Romanian.

The paradigmatic absorbtion limits both types of ambiguity discussed earlier: intrinsically (due to different possibilities of a word segmentation) and extrinsically (due to different interpretations an ending may have).

In order to obtain a complete morphological knowledge in a relatively short time, PARADIGM is accompanied by a merging utility program, (partially) able to unify two or more knowledge bases developed with different copies of the system.

5. FINAL REMARKS

One of our earlier goals, some years ago, was to establish, by manual procedures, a reasonable set of flexioning paradigms for Romanian, in order to implement a reliable morphological processor, general enough to cover the requirements of technical texts. The task was taken by seven colleagues with different backgrounds (linguists, logicians, engineers and mathematicians) and lasted for almost half an year (Cristea,1982). I used the examples from the then written material, in order to test the PARADIGM system. While differently organized, the equivalent (in linguistic coverage) knowledge base was obtained in a four-hour session. Moreover, the number of paradigms discovered by PARADIGM was smaller (97 paradigms versus 123). The rest were absorbed without any loss. By running test data on the manually discovered knowledge base and on the PARADIGM acquired knowledge base we noted up to 10% improvement in analysis time. In hypothesising the lexical status and morphological features of the unknown words, based only on endings analysis, the PARADIGM generated knowledge base was also better.

A morphological knowledge base for Russian and another one for Spanish are under development. Experiments have also been made with French, Slovak and Hungarian. In the near future, we plan to develop the system in two important directions:

- learning compound word-forms rules (proclitic articulation of nouns and adjectives, verb compound tenses, degrees of comparison for adjectives);

- learning lexical affixes (that is meaning modifying preffixes and suffixes (Tufis,1988)).

Related work is reported in (Wohtke,1986), (Trost,1986) but they are either concerned with English (not a very exciting language from the morphological point of view) or address generation or analysis only. The very popular two-level morphology model of Koskenniemi (1983) intended primarily to derivational morphology is, from our point of view, too expensive for a grammatical oriented processing.

Recent work reported in (Goertz, 1988), (Wothke, 1986), (Zock,1988) share some points with our approach.

We consider that the main contributions of our work stem from the following features:

- freedom in defining the categorial system for the model;

- independence of a specific natural language, provided it is within our "root + ending" approach;

- applicability of the synthesized rules both in analysing and generating word forms;

- possibility of rapid development of morphological knowledge bases, by merging the results of many parallel acquisition sessions;

- duality of system behaviour (apprentice - expert) which allows immediate check of the acquired knowledge;

- low level of linguistic competence required to the trainers.

ACKNOWLEDGEMENTS

The main part of the PARADIGM system, initially implemented in TC-LISP (Tufis,1987b) for PDP11-like computers, was transferred on IBM-PC under GCLISP, during a reseach stay at the International Basic Laboratory on Artificial Intelligence of the Institute of Technical Cybernetics of the Slovak Academy of Sciences in Bratislava. I would like to thank to all the people responsible for my research stay in the International Laboratory and especially to Dr.Josef Miklosko, Dr.Pavol Hrivik, Dr.Karol Richter and Dr.Rudolf Fiby. I would also like to thank to Leos Tovarek who kindly helped me on the Slovak language testing.

REFERENCES

Alshawi H., Boguraev B., Briscoe T. 1985; -Towards a Dictionary Support for Real Time Parsing. Proceedings of the 2-nd Conference of ECACL, Geneva,1985, 171-178.

Bear J., 1988; - Morphology with Two-Level Rules and Negative Rule Features. Proceedings of COLING'88, Budapest, 1988, 28-31.

Cristea D., Curteanu N., Mihaescu P., 1982; -Research in Natural Language Communication with Computers, Final Report to A.I.Cuza University - ICI Contract,1982

Gortz G., Paulius D.,1988; -A Finite State Approach to German Verb Morphology. Proceedings of COLING'89, Budapest, 1988.

Jappinen H, Lehtola A., Nelimarkka E., Yiilammi M., 1983; - Knowledge Engineering Approach to Morphological Analysis. Proceedings of the First Conference of ECACL, Pisa, 1983, 49-51.

Koskenniemi K., 1983; -Two Level Model for Morphological Analysis. Proceedings of IJCAI'83, Karlsruhe, 1983, 683-685.

Tecuci G., 1988; - DISCIPLE-1: A Theory Methodology and System for Learning Experts Knowledke. PhD Thesis, University of Paris-Sud, 1988.

Trost H., Buchberger E., 1986; - Towards the Automatic Acquisition of Lexical Data. Proceedings of COLING'86, Bonn, 1986, 387-389.

Tufis D., Cristea D., 1985;-IURES: A Human Engineering Approach to Natural Language Question Answering. In Bibel W., Petkoff B.(eds):Artificial Intelligence: Methodology, Systems, Applications. North Holland, Elsevier, 1985, 177-184.

Tufis D., Dumitrescu C., 1987; - MORPHO: A Lexicon Management System. Reference Manual, ITCI, 1987 (in Romanian).

Tufis D., Tecuci G.,Cristea D., 1987; - LISP (vol 2.):TC-LISP for Minis. AI Systems implemented in TC-LISP (IURES, QUERNAL, DISCIPLE), Technical Publishing House, Bucharest, 1987(in Romanian).

Tufis D., 1988; - Analysis and Generation of Words, Based on Automatically Acquired Morphological Knowledge. Research Report, International Basic Laboratory UTK, Bratislava,1988.

Wehrli E., 1985; - Design and Implementation of a Lexical Data Base. Proceedings of the 2-nd Conference of ECACL Geneva, 1985, 146-153.

Wothke K., 1986; - Machine Learning of Morphological Rules by Generalization and Analogy. Proceedings of COLING'86, Bonn, 1986, 283-289

Zernik U.,1988; - Language Acquisition: Coping with Lexical Gaps. Proceedings of COLING'88, Budapest, 1988, 796-800.

Zock M. ,Francopoulo G.,Laroni A., 1988; - Language Learning as Problem Solving. Proceedings of COLING'88, Budapest, 1988, 806-811.