Cristian Dumitrescu, Research Institute for Informatics, Bucharest

Paradigmatic Morphology Modeling and Lexicon Design with MORPHO-2

ABSTRACT: The paper describes the MORPHO-2 system, designed to handle monolingual lexicons and morpho-lexical processes (i.e. word-form analysis and synthesis). From the computational morphology point of view, MORPHO-2 can be characterized as being based on a paradigmatic approach to root or lemma lexicons.

To model paradigmatic morphology and design the lexicon entries the lexical information is properly structured and adequate access mechanisms are used to capture linguistic generalizations at lexicon level.

1 Introduction

Modern linguistic theories, irrespective of the approached component (morphology, syntax, semantics, pragmatics), attribute a very important part to the lexicon, from the practical but especially the theoretical point of view.

Researches in this direction lead to specific models and techniques which seek a lexical dimension for the linguistic generalizations. Consequently, the lexicon can no longer be viewed as a simple list of lexical entries.

Since in our approach the morphological processes obey a paradigmatic morphology (Tufis 1989), word-forms analysis and synthesis take into account only grammatical endings (which include both desinences and suffixes) and the lexicons handled by *MORPHO-2* system are root- or lemma-oriented.

The linguist may develop morphological models, following a paradigmatic approach, by means of a proper description language. We have represented morphological feature bundles as attribute value pairs organized in a hierarchical manner (Dumitrescu 1991).

When new lexicon entries are defined the hierarchy is referred to by the lexicographer. With regard to the word-forms the roots of which are specified for an entry the relations between the regularity, subregularity and irregularity may be established.

PATR conditions, parameterized macros and macro name overloading facilitate the syntactic description specification of a lexical entry.

2 Building the Morphological Model

In order to build the morphological model, an integrated environment which allows editing, viewing and compiling the morphological model description, is available to the linguist.

Defining the morphological model takes place in several steps, during which the linguist has to specify the following:

a) the categories, subcategories, features and their values, in a hierarchical manner
b) the paradigmatic descriptions
c) the feature specification defaults associated to each paradigmatic description
d) the lemma-entry correspondence, for each paradigmatic description
e) the inflectional paradigms and root detection rules.

The hierarchical description of features is achieved by correlating several feature specifications. A feature specification is given in the form of a (feature: value$^+$) pair, where feature and values are atomic. We call a paradigmatic description a hierarchical description built of several simple (feature: value) pairs.

Figure 1 partially presents, in the form of an incomplete tree, the hierarchical description of features from the morphological model for the Romanian language. By tree traversal, all paradigmatic descriptions of the model may by generated.

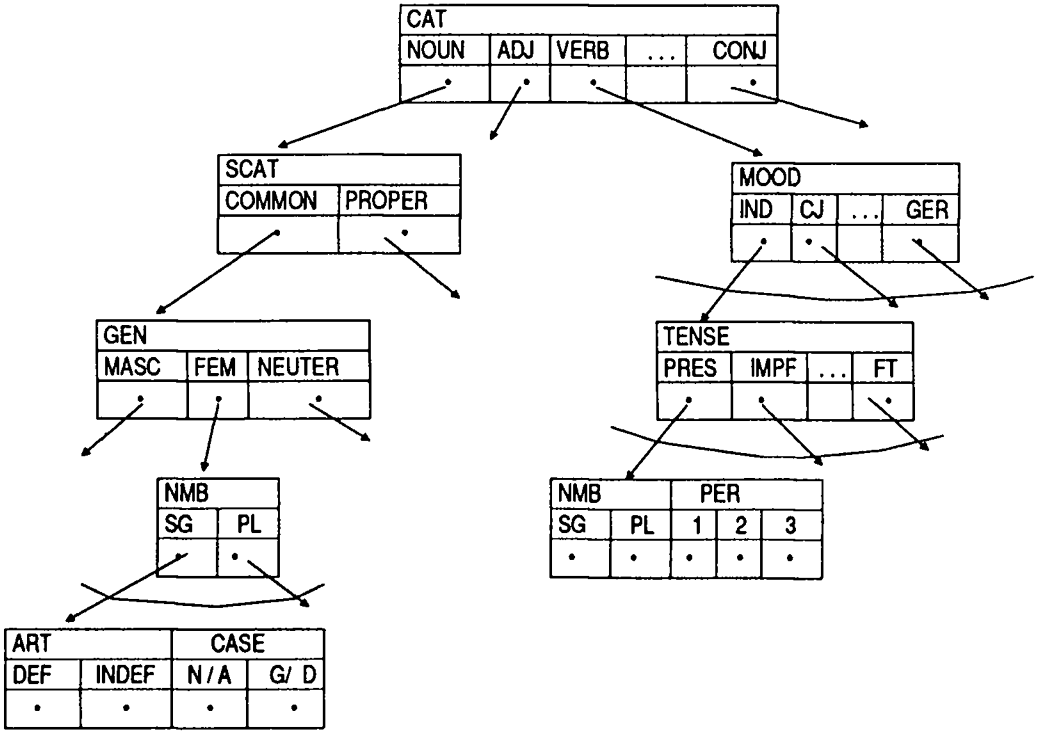

Figure 1 Hierarchical description of features

Each non-terminal node contains a single feature specification. The leaf nodes may contain one or more feature specifications. According to the successor selection criteria, which is applied when visiting a non-terminal node, we can distinguish CHOOSE nodes (when only one successor is selected) or FOREACH nodes (when the individual selection of each successor is required). In the figure, a FOREACH node is outlined by a curve drawn over the emerging edges. By traversing the tree across the longest path which starts from the root node, through CHOOSE nodes only, the selector of a paradigmatic description is obtained (e.g. CAT = NOUN & SCAT = COMMON & GEN = FEM, CAT = VERB).

The description attached to a leaf node is represented by means of a morpho-lexical acquisition scenario. A scenario entry (further on referred to as a slot) corresponds to a point of the paradigmatic description space.

For the specifications in Figure 1, if generating the descriptions as described before, the paradigmatic descriptions will be obtained, which stand for feminine common noun declension and verb conjugation respectively.

As a result of the morphological model compiling a lexicographer report may be obtained, which will also contain the morphological acquisition menus (Figure 2).

Selectors of those descriptions allowing feature specification defaults are attached with (feature: value$^+$) pairs which are default inheritances of the corresponding slots. In our example the following association is possible: (CAT=VB) —>(PER 1 2 3).

The area of the morphological model where the lemma - entry (from paradigmatic description) correspondences are described, consists in a specification of the points from the paradigmatic description spaces, which characterize the lemma field from the lexicon entry. This way, the lexical level required by the lexical transfer is ensured.

The last step in the morphological model description is to inform the system about how to build inflectional paradigms and root detection rules. For each paradigmatic description the linguist may specify more paradigmatic ending families from which the system then builds the inflectional paradigms. For the Romanian language, there have been identified 136 inflectional paradigms (Tufis 1989).

Based on the inflectional paradigms, the system will determine the rules for root detection and word-form generation.

Such a rule has the following form:

<inflexion>:=(<inflectional-paradigm><slot-number>)

with the following meanings:

a) if a word ends in <inflexion> then

- the root is what remains from the word after dropping the <inflexion>
- the root belongs to the <inflectional-paradigm>
- the contextual information corresponding to the current word is given by <slot-number>

b) if a root belongs to the <inflectional-paradigm> and it is used in the context given by <slot -number> then

- the word is obtained by concatenating the given root with the <inflexion>.

```
CAT = NOUN & SCAT = COMMON & GEN = FEM
      NMB = SG
```

ART	CASE	WORD_FORM
DEF	N / A	
DEF	G / D	
INDEF	N / A	
INDEF	G / D	

```
       NMB = PL
        . . .
CAT = VERB
       MOOD = IND
                TENSE = PRES
```

NMB	PER	WORD_FORM
SG	1	
SG	2	
SG	3	
PL	1	
PL	2	
PL	3	

```
                TENSE = IMPF
                 . . .
                TENSE = FT
                 . . .
        MOOD = CJ
            . . .
```

Figure 2 Morphological acquisition menus

Further on some examples are given which contain inflectional paradigms, root detection and word-forms synthesis rules, as they appear in a system generated lexicographic report.

```
  SELECTOR:  CAT  = NOUN & SCAT = COMMON & GEN = FEM
      [INFLPR25   A  I  A  II  I  ILE  ILOR]
      [INFLPR26   E  -  A  I  I  ILE  ILOR] ...
      [INFLPR37   I  I   EA  II  I  ILE  ILOR]
  ...
  SELECTOR:  CAT  = VB
      [INFLPR1   - I  A  AM  ATI  A  AM  AI  A  AM  ATI  AU  AI  ASI  A  ARAM  ARATI  ARA
      ASEM  ASESI  ASE  ASERAM  ASERATI ASERA  -  I  E  AM  ATI  E  IND  INDU
                 A  A  ATI  AT  ATA  ATI  ATE]  ...
      [INFLPR19  -  I  E  IM  ITI  -  EAM  EAI  EA  EAM  EATI  EAU  I  ISI  I  IRAM  IRATI  IRA  ISEM
                 ISESI  ISE  ISERAM  ISERATI  ISERA - I A  IM  ITI  A  IND  INDU  I  O  ITI  IT  ITA
                 ITI  ITE]
   ...
  A        <—>   [CAT  = VB;  INFLPR1;  3  6  9  15  33  34]  [CAT = VB;  INFLPR2;  27  30]  ...
                 [CAT  = NOUN & SCAT = PROPER & GEN = FEM;  INFLPR47;  1] ...
  ASCA     <—>   [CAT  = VB;  INFLPR9; 27  30] ...
```

The lexicographer's interface is strictly dependent on the specifications from the linguist's interface since a large part of the former is built automatically from the specifications of the latter.

3 The Lexicon Entry

MORPHO-2 lets the lexicographer define new entries in the lexicon by means of a user-friendly window oriented interface.

A lexicon entry has the following formal structure:

```
<entry>::= (<lemma>
            (<paradigmatic-description-selector>
             <inflectional-paradigm>
            (<morphologic-description><root>)*
            (<syntactic-description><semantic-description>*)*)*)
```

The fields <lemma>, <paradigmatic-description-selector> and <inflectional-paradigm> have the obvious meaning.

The straightforward way the roots are represented (but also the most inefficient), within a paradigmatic description, consists in simply filling in the corresponding slots. Redundancy can be reduced if the nonmonotonic inheritance mechanism is used for the inflected forms regularity, subregularity and irregularity (Gazdar 1988), (Evans and Gazdar 1989).

The fields (<morphologic-description><root>)* associate the current roots within the paradigmatic description referred by the selector.

In fact, the associations are given by rules of the following form:

[path$_1$] <—> root$_1$
[path$_2$] <—> root$_2$
. . .
[path$_n$] <—> root$_n$

where each path starts at the top of the subtree which defines the paradigmatic description.

Let us consider from the above given feature hierarchy the feminine common noun description (Figure 3).

For the established morphological model there has been previously specified the association:

(CAT = NOUN & SCAT = COMMON & GEN = FEM) —> (CASE N/A/G/D/V)

which will result in default inheritances for the feature CASE.

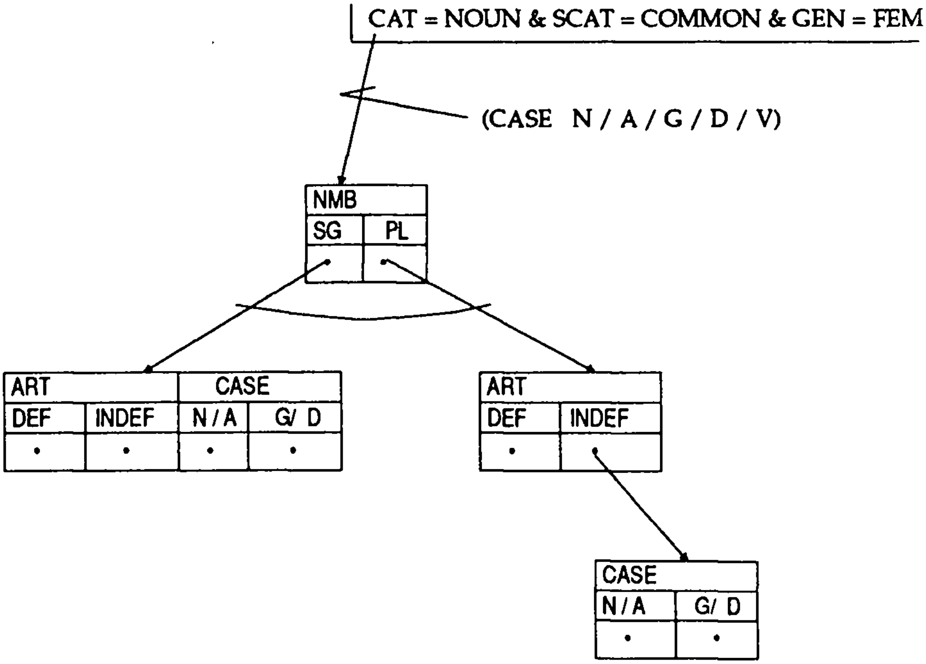

Figure 3 Hierarchical description for feminine common noun

Within this context the root association rules for the lemma FEMEIE (WOMAN) will be specified as shown below:

```
        (FEMEIE
          ( [CAT = NOUN & SCAT = COMMON & GEN = FEM]
            INFLPR26
            ( [NMB=SG] <—> FEMEI
              [NMB=PL] <—> FEME  )
    ... )
```

More precisely, a rule of the form:

[$path_i$] <—>$root_i$

has the following double meaning:

a) <u>if</u> $path_i$ has an associated $root_i$ <u>then</u>

- $root_i$ is the default inheritance for the slots reached through $path_i$

b) <u>if</u> $root_i$ is associated to $path_i$ <u>then</u>

- $root_i$ inherits morphological features bundled together by selector, feature specification defaults and $path_i$.

Applying such rules one can easily capture total or partial regularity. The same mechanism may be used to handle exceptions also.

Thus, if given two rules:

[$path_i$] <—> $root_i$
[$path_j$] <—> $root_j$

so that $path_i \subset path_j$ ($path_j$ is an extension of $path_i$) then $root_j$ overwrites $root_i$ in the $path_j$ slots.

The following examples use this technique.

```
(FATA
  ( [CAT =NOUN & SCAT =COMMON & GEN =FEM]
    INFLPR30
    ( [ ] <—> FET
      [NMB=SG & CASE =N/A] <—> FAT   )
 ... )

(FEMEI
  ( [CAT =NOUN & SCAT =COMMON & GEN =FEM]
    INFLPR26
    ( [ ] <—> FEME
      [NMB =SG] <—> FEMEI  )
 ... )
```

By syntactic-description we refer to restrictions on co-occurrence with other words (or phrases). In order to specify such restrictions, the category-valued features used in a PATR-like representation (Shieber 1986) have been enriched (Estival 1990) with extensions based on linguistic motivations for the Romanian language.

One extension concerns the PATR conditions with the special attribute *this* (here after '*') at the top of their path description, which refers to the lexicon entry itself of the current analysis context. What we really want is, given a word-form, to obtain the complete feature structure, by unifying the descriptions fetched from the corresponding lexicon entry, as a result of the lexical analysis.

As an example, if the lexicon entry for the lemma AJUNGE (to get to) contains the following PATR conditions:

```
<* HEAD AGREEMENT PER>  = <* PER>
<* HEAD AGREEMENT NMB> = <* NMB>     (1)
```

and if the morphological analysis of the word-form AJUNG leads to:

```
<* CAT>  = VERB
<* MOD> = IND
<* TENSE> = PRES                         (2)
<* NMB> = SG
<* PER> = 1
```

then we may unify (1) with (2) and thus enrich our feature structure.

Another extension, described below, allows atomic disjunctive values (e.g. A/G) and list values (e.g. [SUBJ]) to be specified.

```
    Macro InTrans:
            <* SUBCAT> = [SUBJ]
            InTran.

    Macro InTran:
            <SUBJ CAT> = NP
            <SUBJ CASE> = A/G
            <* HEAD  AGREEMENT  PER> = <* PER>
            <* HEAD  AGREEMENT  NMB> = <* NMB>
            <* HEAD  AGREEMENT> =  <SUBJ  HEAD  AGREEMENT>.
```

Using parameterized macros and macro name overloading, the valency models for the Romanian transitive verbs may be easily expressed as Trans(NP), Trans(PP), Trans(NP/PP/PPp), etc.

```
     Macro Trans(NP):
             <* SUBCAT> = [OBJ SUBJ]
             InTran
             <OBJ CAT >= NP
             <OBJ CASE> = A.

    Macro Trans(PP):
             <* SUBCAT> = [OBJ SUBJ]
            InTran
            <OBJ CAT> = PP
            <OBJ PREP> = pe/la.

            Macro Trans(PPp):
            <* SUBCAT> = [[OBJ1 OBJ2] SUBJ]
            InTran
            <OBJ1 CAT> = PP
            <OBJ1 PREP> = pe
             <OBJ2 CAT> = PPron
            <OBJ2 CASE> = A.
```

The last macro underlines a phenomenon, Romanian language specific, that of doubling a direct object. For instance, in the next sentence the direct object (**pe Ion**) is doubled by accusative, personal pronoun (**L-**):

> **L-am vazut pe Ion.**
> I have seen **John.**

but the "two" direct objects refer to the same object and therefore only one valency is required.

For each syntactic description, the lexicographer may provide one or more semantic descriptions. We consider that the semantic description for an analysis and generation lexicon (like the one presented here) should be a mediator between a given natural language and the meaning representation language. From our point of view the lexemes are the necessary primitives to work with.

Lexical ambiguity, marked by more than one lexeme for a lexical entry, is possible either due to category ambiguity (e.g. noun vs. verb) or to polysemy and homonymy. To solve the latter type of ambiguity a detailed meaning and contextual analysis is required. Consequently, additional mechanisms are needed.

Thus, the actual semantic descriptions are stored in a separate date area from the rest of the lexicon (Nirenburg 1987) and managed independently of *MORPHO-2*.

Further on an example is given which describes a complete lexicon entry. We should notice that the same verb may be transitive or intransitive, according to its meanings; for example A AJUNGE (to get to) is transitive and with the meanings A DEVENI (to become), A SOSI (to arrive) and A FI SUFICIENT (to be enough) is intransitive.

```
( AJUNGE
 ( [PV = VB]
  INFLPR15
  ([ ]  <—>AJUNG)
  ( ( [Intrans]   A_DEVENI  A_SOSI  A_FI_SUFICIENT)
    ( [Trans (NP/PP/PPp)]   A_PRINDE))))
```

As far as the linguist and lexicographer are concerned, to express the lexicon, the system offers a lexical representation language. By compiling the provided lexical information, structures will be generated which are optimal with respect to morpho-lexical processings. When needed, the lexicographer may modify and compile again some lexicon entries (Dumitrescu 1991).

For the target natural language processing system, which is the beneficiary of the morpho-lexical processes, *MORPHO-2* is a lexical information retrieval system (Dumitrescu 1992).

4 Implementation

The *MORPHO* project, started in 1986, has achieved as a first result, a prototype version now available on a PDP-11 compatible computer. The second version of the system, the one presented in this paper, is implemented in C and PROLOG on a IBM-PC compatible.

The lexicon entry architecture as well as the type of relation between its fields are the same for all lexicons handled by *MORPHO-2* and are not accessible to the user in order to be defined. The access methods for each entry field, relations among entry fields as well as those among different entries are directly controlled by the system.

These restrictions should not be interpreted as system limitations but as a disciplined approach of the lexicon building process.

The structure of the lexicon entry has imposed the use of multilists and variable length record handling. Lexicon indexing techniques by means of prefixed virtual B+tree, as well as optimal grouping data with regard to morpho-lexical processings, have led to an average response time of lexical processes, quite independent of the lexicon's size (for more details on performance analysis see (Tufis and Dumitrescu 1990)).

Bibliography

DUMITRESCU, C.(1992): MORPHO-2 Reference manual. I.C.I., Bucharest

DUMITRESCU, C.(1991): MORPHO - "Design and development environment for monolingual lexicons", Romanian Informatics and Control Engineering Review, Vol. 1, No. 2, Bucharest, pp. 23-27

ESTIVAL, D. (1990): ELU User Manual, ISSCO, Geneva

EVANS, R., GAZDAR, G. (1989): "Inference in DATR". In: Proceedings of the 4th Conference of ECACL, Manchester, pp.66-71.

GAZDAR, G. (1988): The organization of computational lexicons, Cognitive Science Research Paper, The University of Sussex, Brighton

NIRENBURG, S., RASKIN, V. (1987): "The subworld concept lexicon and the lexicon management system", In: Computational Linguistics, Vol.13, No.3-4, pp.270-289.

SHIEBER, S. (1986): "An introduction to unification-based approaches to grammar", CSLI / SRI International, Stanford

TUFIS, D.(1989): "It would be much easier if WENT were GOED". In: Proceedings of the 4th Conference of ECACL, Manchester, pp.145-152

TUFIS, D. and DUMITRESCU, C. (1990): "MORPHO - A dictionary management system". Proceedings of the 13th International Seminar on DBMS, Mamaia, pp. 174-182

KEYWORDS: paradigmatic morphology, monolingual lexicon, feature hierarchy, inflectional paradigms, nonmonotonic inheritance, PATR conditions, lexeme