

LIMBAJUL DE DESCRIERE A MODELULUI MORFOLOGIC

ing. Cristian Dumitrescu

Institutul Național de Cercetare-Dezvoltare în Informatică – ICI, București

Rezumat: În lucrare se descrie sistemul MORPHO-2, sistem proiectat în scopul gestiunii lexicoanelor monolingve și a manipulării on-line a proceselor lexicale. Se asigură independența între limbaje (cu caracter preponderent flexionar) prelucrate și mediu work-bench, de dezvoltare a dicționarului.

Cuvinte cheie: SGBD, work-bench, monolingual dictionary.

1. Introducere

Sistemul MORPHO dispune de facilitățile necesare modelării morfologiei limbii române (în general, a limbilor cu caracter preponderent flexionar). În acest sens, lexicologul are la dispoziție un limbaj de definire a modelelor morfologice. Un model morfologic este alcătuit din mai multe secțiuni. Succesiunea și denumirile acestora sunt date de următoarea regulă de rescriere:

Model Morfologic:

- NumeModel Descriere Paradigmatică
- Specificări Caracteristici Implicite
- Asignări CâmpLema ParadigmeFlexionare.

Vom detalia, în continuare, fiecare secțiune din definirea unui model. În secțiunea NumeModel, se specifică numele modelului morfologic (nume utilizat și pentru denumirea lexiconului ce urmează a se crea după compilarea modelului morfologic). Sintaxa de descriere este următoarea:

NumeModel:- "MORPHOLOGICAL-MODEL-NAME" "=" IDENTIFICATOR ";"

unde IDENTIFICATOR este un nume valid de fișier MS-DOS (fără extensie).

Secțiunea Descrieri Paradigmatice este marcată de antetul "PARADIGMATIC - DESCRIPTIONS" și este alcătuită din specificații referitoare la caracteristicile coloanei cuvânt și la descrierile paradigmatică propiu-zise.

DescrieriParadigmatice:-
"PARADIGMATIC_DESCRIPTIONS"

CaracteristiciColoanăCuvânt
DescriereParadigmatică ". "

În urma compilării modelului morfologic, din descrierile paradigmatică se construiesc scenariile de achiziție morfologică. Ultima coloană din fiecare scenariu (numită și coloana cuvânt) este adăugată automat de către sistem și permite lexicografului vizualizarea formelor lexicale lemei curente.

Caracteristicile ultimei coloane dintr-un scenariu sunt specificate printr-o declarație de forma:

WORD_COLUMN = (" identificador", n)

unde:

identificator etichetează coloana cuvânt;
n este lungimea cuvântului maxim, ce va fi introdus în lexicon.

Facem observația că alocarea unui câmp de lungime mare, corespunzător coloanei cuvânt, nu înseamnă că memorarea cuvintelor în lexicon se va face automat, pe lungimea respectivă. Aceasta ar însemna o risipă inutilă și costisitoare a spațiului de memorare (a se compara, de exemplu, cuvintele ac și accesibilitate). Lungimea câmpului este semnificativă doar în faza de dialog cu lexicograful, stocarea efectivă a cuvântului (în fapt, a rădăcinii sale) se face exact pe lungimea necesară.

Descrierile paradigmatică au drept scop corelarea, într-o manieră ierarhică a categoriilor lexicale, a caracteristicilor și valorilor acestora. O astfel de ierarhie se construiește prin compilarea unei descrieri alcătuită din următoarele declarații speciale:

DescriereParadigmatică:-DescriereChoose
|DescriereForeach
|MeniuDeAchizițieMorfologică

Declarațiile de tip DescriereChoose sau de tip DescriereForeach înglobează criteriul de selecție a succesorilor, aplicat la traversarea unui nod neterminat din ierarhie.

Declarația de tip MeniuDeAchizițieMorfologică corespunde unui nod terminal și are ca efect construirea unui scenariu de achiziție morfologică.

În figura următoare, este prezentată parțial (sub forma unui arbore complet) descrierea ierarhică a categoriilor gramaticale pentru modelul morfologic.

```
+-----+
| PV |
+-----+
|SB|VB |
+--+--+
| • | |
+--+--+

+-----+ +-----+
| GEN | | TMP |
+-----+ +-----+
|MASC|FEM|NEUTRU| |PZ|IMPF|...|PART|
+-----+ +-----+
| | | | | | |
+-----+ +-----+
+--+ +--+
| NR | | NR | PER |
+--+ +--+
|SG|PL| |SG|PL| 1 | 2 | 3 |
```

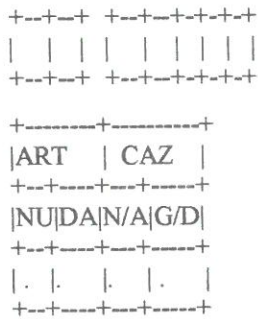


Figura 1. Descrierea ierarhica a categoriilor lexicale

Nodurile neterminale, construite prin declarații de tip Declarație Foreach sunt scoase în evidență prin trasarea unei linii curbe peste arcele divergente din nod.

Secvența declarațiilor, din secțiunea descrierilor paradigmatică, necesară construirii unui astfel de arbore ar fi următoarea:

```

PARADIGMATIC_DESCRIPTIONS
WORD_COLUMN = ("CUVÂNT", 28);
CHOOSE PV: (SB, VB) FROM
  SB: CHOOSE GEN: (MASC, FEM, NEUTRU)
      FROM
        MASC;
        FEM: FOREACH NR: (SG, PL) DO
          SG: SCENARIO
            ART: (NU, DA);
            CAZ: (N/A, G/D)
          END;
          PL: ...;
          END;
        NEUTRU: ...;
      END; { * SB * }
  VB: FOREACH TMP: (PZ, IMPF, ..., PART) DO
    PZ, IMPF, ...: SCENARIO
      NR: (SG, PL);
      PER: (1,2,3)
    END;
    ...
    PART: ...;
  END; { * VB * }
END { * PARADIGMATIC_DESCRIPTIONS * }.

```

Ca rezultat al compilării modelului morfologic se poate obține un raport lexicografic ce va conține, printre altele, și scenariile de achiziție morfologică. Declarațiile anterioare vor rezulta în descrieri paradigmatică corespunzătoare declinării substantivelor feminine și respectiv, a conjugării verbelor (figura 2)

ISA Borlandc este inclusă în descrierea de mai jos și permite pe baza descrierii ierarhice a morfologiei limbii române, lema să fie determinată de lexicolog și lexicograf "Lemma entries" și prin operația "+" de tip join în SGD cu structuri atribut-valoare (SGML) și anume sistemul de gestiune dicționare este extins la sistem de gestiune a lexiconului (cu caracteristici de SGBD) sistem de gestiune pentru baza de date, atribute valoare metode de indexare și arie selectate de adresa

pointer. Regăsirea inflexiunii și a paradigmei flexionare se face prin join "+" în SGBD. Este realizat, în continuare, un arbore ȘI-SAU cu frame-uri slot (regăsirea rădăcinilor și intrări de dicționar asemănătoare).

AQUISITION MENU:
PV = SB & GEN = MASC

PV = SB GEN = FEM
FEATURE SPECIFICATIONS DEFAULTS
CAZ: (N/A/G/D/V)
Lemma entry: 1
NR = SB

```

+-----+-----+
|ART|CAZ|CUVÂNT|
+-----+-----+
|NU |N/A|          |
+-----+-----+
|NU |G/D|          |
+-----+-----+
|DA |N/A|          |
+-----+-----+
|DA |G/D|          |
+-----+-----+

```

NR = PL
.....
PV = SB & GEN = NEUTRU

PV = VB
PER: (1,2,3)
Lemma entry: 36
TMP = PZ

```

+-----+-----+
|NR| PER| CUVÂNT|
+-----+-----+
|SG| 1  |          |
+-----+-----+
|SG| 2  |          |
+-----+-----+
|SG| 3  |          |
+-----+-----+
|PL| 1  |          |
+-----+-----+
|PL| 2  |          |
+-----+-----+
|PL| 3  |          |
+-----+-----+

```

TMP = IMPF
.....
TMP = PART
.....

Figura 2. Meniuri de achiziție morfo-lexicală

În continuare, este prezentată sintaxa de descriere a tipurilor de declarații utilizate.

a) Declarația de tip DescriereChoose
 DescriereChoose::-"CHOOSE"
 SpecificareCaracteristică "FROM"
 AlternativeChoose "END"
 SpecificareCaracteristică:: - IDENTIFICATOR ":"
 ListăValori
 ListăValori:: - "(" Listaid ")"
 Listaid:: - IDENTIFICATOR \ IDENTIFICATOR
 "," Listaid
 AlternativeChoose::-Selector
 DescriereAlternativaChoose
 RestAlternativeChoose
 Selector:: - Listaid ":"
 DescriereAlternativaChoose:: -
 DescriereParadigmatică
 | DescriereNulă
 DescriereNulă:: - "(" ")"
 RestAlternativeChoose:: - ":" AlternativeChoose|.

b) Declarație de tip DescriereForeach
 DescriereForeach::-"FOREACH"
 SpecificareCaracteristică "DO".
 AlternativeForeach "END"
 AlternativeForeach::-Selector
 DescriereAlternativaForeach
 RestAlternativeForeach
 RestAlternativeForeach:: - ":" AlternativeForeach
 DescriereAlternativaForeach:: -
 DescriereForeach
 |MeniuAchizițieMorfologică
 |DescriereNulă

c) Declarația de tip MeniuDeAchizițieMorfologică
 MeniuDeAchizițieMorfologică::-"SCENARIO"
 ListaCâmpuri "END"
 ListaCâmpuri:: Câmp |Câmp ":" ListaCâmpuri
 Câmp:: - Listaid ":" ListaValori
 După prezentarea completă a descrierilor paradigmatică, în dezvoltarea modelului morfologic urmează secțiunea în care se stabilesc specificările de caracteristici implicite atașate fiecărei descrieri.

Selectorilor de descrieri paradigmatică, care acceptă specificări de caracteristici implicite li se impun în corespondența perechilor (caracteristica: valoare*) ce sunt considerate moșteniri implicite pentru sloturile din spațiul paradigmatic, delimitate de selector. În exemplul nostru, sunt posibile următoarele asocieri:

FEATURE_SPECIFICATION_DEFAULTS
 [PV = VB] -> PER: (1,2,3)
 [PV = SB & GEN = MASC] -> CAZ: (N/A/G/D/V)
 [PV = SB & GEN = FEM] -> CAZ: (N/A/G/D/V)
 [PV = SB & GEN = NEUTRU] -> CAZ:
 (N/A/G/D/V)

Sintaxa de descriere referă următoarele reguli de rescriere:

SpecificăriCaracteristiciImplicite:: -

"FEATURE_SPECIFICATIONS_DEFAULTS"
 ListăSpecificări
 ListăSpecificări::-Asociere|Asociere
 ListăSpecificări.
 Asociere::- SelectorDescriereParadigmatică "->"
 SpecificareCaracteristică
 SelectorDescriereParadigmatică:: - "[" ListăPerechi
 "]"
 ListăPerechi::- Pereche | Pereche AMP ListăPerechi
 Pereche::-IDENTIFICATOR "=" IDENTIFICATOR
 În secțiunea AsignăriCâmpLemă, sunt descrise corespondențele (lema-intrare în descrierea paradigmatică). De fapt, lexiconul trebuie să specifice din spațiile paradigmatică ale descrierilor ce caracterizează câmpul lema din intrarea în lexicon. În acest mod, se asigură nivelul lexical solicitat de sistemele de traducere pentru realizarea transferului lexical.

Câmpului lema i se asociază, de regulă, formele standard ale unui cuvânt: nominal, singular, nearticulat, pentru substantive, infinitivul scurt pentru verbe etc.

În descrierile paradigmatică, din exemplul nostru, lemele pot fi asociate următoarelor intrări:

LEMMA_ENTRIES

[PV = VB & TMP = INF]
 [PV = SB & GEN = MASC & NR = SG & ART = NU]
 [PV = SB & GEN = FEM & NR = SG & ART = NU & CAZ = N/A]
 [PV = SB & GEN = NEUTRU & NR = SG & ART = NU]

Sintaxa de descriere a acestei secțiuni necesită respectarea următoarelor reguli de rescriere:

AsignăriCâmpLemă::-"LEMMA_ENTRIES"
 ListăAsignări

ListăAsignări:: - IntrareÎnDescriereaParadigmatică
 | IntrareaÎnDescriereaParadigmatică
 ListăAsignări

IntrareÎnDescriereParadigmatică:: -

SelectorDescriereParadigmatică

În ultima secțiune, din descrierea unui model morfologic, lexicologul are posibilitatea de a informa sistemul cum să construiască paradigmele flexionare și reguli de detecție a rădăcinilor.

Pentru fiecare descriere paradigmatică (identificată printr-un selector) lexicologul poate specifica mai multe paradigme flexionare. Fiecare paradigmă este etichetă cu un cod de maxim 8 caractere alfanumerice.

Asocierea paradigmele flexionare la descrierile paradigmatică din model se face respectând următoarele reguli de sintaxă:

```

DescriereParadigmeFlexionare::-
    "INFLECTIONAL_PARADIGMS"
    ListăAsocieri
ListăAsocieri::- SelectorDescriereaParadigmatică
    "[" ListăParadigmeFlexionare "]"
    RestListAsocieri
RestListAsocieri::- ListăAsocieri|.
ListăParadigmeFlexionare::-
    CodParadigmăFlexionară
    "[" ListăCuvinte "]"
    RestListăParadigmeFlexionare
RestListăParadigmeFlexionare::-
    ListăParadigmeFlexionare |.
CodParadigmăFlexionară::- IDENTIFICATOR
ListăCuvinte::- Cuvânt| Cuvânt ListăCuvinte
    Mai jos, sunt date ca exemplu câteva paradigme
    flexionare și reguli de detectare a rădăcinilor și de
    generare a cuvintelor, asociere dintr-un raport
    lexicografic, generat de sistem.
[PV=SB & GEN=MASC]
[
20 [ - UL ULUI I II ILOR]
....
24 [ A L LUI I II ILOR]
]
...
[PV = SB & GEN = NEUTRU]
[
38 [ - UL ULUI E ELE ELOR]
....
45 [ - UL ULUI URI URILE URILOR]
]
[PV = VB]
[
1. [- I A AM ATI A AM AI A AM ATI AU AI ASI
A ARAM ARATI ARA ASEM-ASESI ASE
ASERAM ASERATI AERA - I E AM ATI E IND
INDU A A ATI AT-ATA ATI ATE]
2. [- I E EM ETI - EAM EAI EA EAM EATI EAU
EI ESI U URAM URATI-URA USEM USESI USE
USERAM USERATI USERA - I A EM ETI A IND
INDU EA-I ETI UT UTA UTI UTE]
19 [-I E IM ITI - EAM EAI EA EAM EATI EAU I
ISI I IRAM IRATI IRA-ISEN ISESI ISE ISERAM
ISERATI ISERA - I A IM ITI A IND INDU I O-
ITI IT ITA ITI ITE]
]
ROOT DETECTION AND WORD-FORMS
SYNTHESIS RULES
A -> [PV = VB; 1;3 6 7 15 33 34][PV = VB ; 6; 3 6
9 34] [...]
...
[PV = VB; 5; 27 30]
...
[PV = VB & GEN = FEM;20;3]
...
ASCA -> [PV = VB; 9; 27 30]
EZ -> [PV = VB; 12; 1 25] [PV = VB; 17; 1 25]

```

2. Limbaj de descriere a intrărilor în lexicon

Reprezentarea adecvată a informației lexicale se obține dacă generalizările lingvistice sunt luate în considerare. Aceasta înseamnă că anumite informații să fie reprezentate grupat, cu posibilitatea neutralizării lor, în scopul eliminării redundanței. O astfel de abordare s-a utilizat pentru reprezentarea informației dintr-un model morfologic. Gruparea categoriilor, subcategoriilor, a caracteristicilor și a valorilor acestora, precum și reprezentarea lor într-o manieră ierarhică vor permite reutilizarea acestora la definirea de noi intrări în lexicon. În plus, această reprezentare creează premisele capturării unui alt tip de generalizare, referitoare la relația dintre regularități, regularități parțiale și neregularități. La proiectarea intrărilor din lexicon se au în vedere structura informațiilor lexicale și utilizarea unor mecanisme de acces, care să permită exprimarea generalizărilor lexicale, posibile la nivelul lexiconului.

O intrare în lexicon are următoarea structură formală:

```

<intrare_in_lexicon>::-
(
(<lemă>
(<selector_descriere_paradigmatică>
<paradigma_flexionară>
(<descriere_morfologică>
<rădăcină>)*
)*
)
Semnificația câmpurilor <lemă>,
<selector_descriere_paradigmatică>și
<paradigmă_flexionară> este cea deja stabilită.
Sintaxa de descriere (până în acest punct) a intrării
în lexicon necesită respectarea următoarelor reguli:
IntrareÎnLexicon::- "(" Lema ListaAsocieriLema
")"
ListaAsocieriLema::- "("
SelectorDescriereParadigmatică
EtichetaParadigmaFlexionară
("(" ListaDescMorRad")"
")" RestListăAsocieriLema
RestListăAsocieriLema::- ListăAsocieriLema|.
Lema::- CUVÂNT::-
SelectorDescriereParadigmatică "[" ListăPerechi "]"
ListăPerechi::-Pereche|Pereche AMP ListăPerechi
Pereche::-IDENTIFICATOR EG ListăIdentificatori
ListăIdentificatori::- IDENTIFICATOR
|IDENTIFICATOR SLASH
ListăIdentificatori
EtichetaParadigma::-IDENTIFICATOR

```

Modalitatea flexionară cea mai simplă, dar și cea mai eficientă, de reprezentare a rădăcinilor într-o descriere paradigmatică, constă în completarea directă a intrărilor în meniurile de achiziție morfologică

corespunzătoare descrieri. Redundanța poate fi înlăturată dacă se utilizează mecanismul de moștenire nonmonotonică în capturarea regularităților, subregularităților și neregularităților formelor flexate atașate temei curente [Evans și Gazdar, 1989]).

Câmpurile (<descriere morfologică><rădăcină>)* asociază rădăcinile specificate, în descrierea paradigmatică referită de selector. De fapt, asocierile sunt date sub forma unor reguli de forma:

```
[cale1] -> rădăcină1
[cale2] -> rădăcină2
...
[calen] -> rădăcinăn
```

în care fiecare cale pornește din rădăcina subarborelui dominat de selector. În [3], [4] a fost prezentată pe larg semnificația regulilor de asociere a rădăcinilor în descrierea paradigmatică, a mecanismului de moștenire nonmonotonică, precum și tehnica de manevrare a excepțiilor.

În lucrarea de față, insistăm asupra sintaxei de descriere a acestor reguli, și nu asupra semnificației lor. Spre exemplificare, vom considera totuși câteva intrări în lexicon. Cititorul interesat va trebui să coreleze regulile de asociere a rădăcinilor cu descrierea ierarhică a categoriilor lexicale.

```
OM
([CAT=NOUN&SCAT=COMMON&GEN=MASC]20
([NMB=SG]↔OM)
([NMB=PL]↔OAMENT)))
```

Sintaxa de specificare a rădăcinilor este următoarea:

```
ListaDescMorfRad::-Asociere ListaDescMorfRad|
Asociere:: - IntrareÎnDescriereaParadigmatică "<-
>" Rădăcină
Rădăcină:: - CUVÂNT
IntrareÎnDescriereaParadigmatică
:: - SelectorDescriereParadiagmatică
```

Intrarea de lexicon referă generalizarea de ierarhie cu moșteniri multiple cu parametrii, lexiconul fiind de tip compilativ. Trebuie să se facă verificarea codului paradigmei în raport cu selectorul paradigmatic, declarat în schema morfologică de lexicograf, și descrierea unei intrări lexicale având în vedere regăsirea din dicționarul curent (cu structuri statice), ca urmare a compilării lexiconului.

Lema se asociază în intrarea din lexicon prin regăsirea rădăcinii din dicționar a temei, stabilirea corectitudinii codului paradigmei flexionare, conform asocierii dintre selectorul descrierii paradigmatică cod paradigmă flexionară și flexiune. Rădăcinile sunt asociate cu simboluri și indică specificări de ierarhii cu moșteniri.

Lema este asociată cu rădăcina regăsită în dicționar, iar regăsirea în dicționar se face prin

intrarea de lexicon cu moșteniri specifice a informațiilor din intrarea de lexicon.

3. Mediul de dezvoltare a lexiconului

3.1. Interfața lexicologului

Pentru a construi modelul morfologic, lexicologul are la dispoziție un mediu de dezvoltare care permite compilarea, vizualizare și editarea descrierii modelului morfologic.

Interfața (de tip user-friendly) este orientată spre ferestre. Astfel, facilitățile oferite de către sistem (ferestre de afișare meniuri de tip pop-up pull-down, ferestre editare și vizualizare, interfața MOUSE) asigură proiectarea și dezvoltarea rapidă a lexiconului. Mai întâi, lexicologul selectează opțiunea CreateDictionary din meniul pop-up afișat de către sistem.

```
+-----+
|Create Dictionary|
+-----+
|Select Dictionary|
+-----+
|Exit           |
+-----+
```

Apoi, prin intermediul unei ferestre de editare, sistemul solicită numele fișierului care conține schema modelului morfologic.

```
+-----+-----+
|Schema File Name| |
+-----+-----+
```

Activitatea mediului de proiectare-dezvoltare este confirmată de apariția meniului pull-down:

```
+-----+-----+-----+-----+-----+
|Help |View| Edit |Options|Compile|Quit  |
+-----+-----+-----+-----+-----+
```

în partea superioară a ecranului.

Comenzile disponibile oferă următoarele facilități:

Help - informații privind comenzile și mesajele sistemului;
View - vizualizare fișier de lucru. În conformitate cu opțiunile de compilare selectate se obțin informații suplimentare (fișier listing de erori, raport lexicografic).

```
+-----+
| Schema File|
+-----+
|Listing File |
+-----+
|Report File  |
+-----+
```

3.2. Interfața lexicografului

Sesiunea de lucru a lexicografului începe prin selectarea opțiunii Select Dictionary din meniul afișat inițial de către sistem.

```
+-----+
|Create Dictionary|
+-----+
|Select Dictionary|
+-----+
|Exit             |
+-----+
```

Apoi, prin intermediul unei ferestre de editare, sistemul solicită numele lexiconului.

```
+-----+-----+
|Dictionary Name|  |
+-----+-----+
```

Extensiile pentru schema obiect (DIC) și fișierul de date (DAT) sunt adăugate automat de către sistem. Opțiunile disponibile lexicografului sunt următoarele:

- * compilează informațiile lexicale
- * vizualizează informațiile lexicale
- * sfârșit sesiune

```
+-----+-----+
|Load Lexical Entries|
+-----+-----+
|Show Lexical Information|
+-----+-----+
|Quit                 |
+-----+-----+
```

În continuare, Sesiunea de Selectare a intrărilor în lexicon este descrisă de următorul meniu pull-down, oferind facilitățile:

```
+-----+-----+-----+-----+
|Help|View|Edit|Options|Compile|Quit|
+-----+-----+-----+-----+
```

- Help - informații privind mesajele și comenzile de sistem;
- View - vizualizare fișier de lucru;
- Edit - editarea fișierului de lucru;
- Options - specificarea opțiunilor de compilare: opțiunea disponibilă se referă la acele intrări lexicale deja existente în lexicon;
- Compile - compilarea intrărilor lexicale;
- Lexicograful - asigură consistența datelor lexicale;
- Quit - încheie sesiunea de dezvoltare a intrărilor lexicale.

3.3. Vizualizarea informațiilor lexicale

Vizualizarea informațiilor lexicale:

```
+-----+
|Inflections  |
+-----+
```

```
|Inflectional Paradigms|
+-----+
|Roots                |
+-----+
|Lexical entries      |
+-----+
```

În continuare, este prezentată în detaliu fiecare comandă.

3.4. Inflexiuni

Vizualizarea inflexiunilor și a paradigmelor flexionare necesită, mai întâi, selectarea opțiunii de descriere paradigmatică. Dacă selectorul descrierii paradigmatică mai conține și alte specificații de categorie, pe lângă partea de vorbire, atunci este utilizat un meniu auxiliar.

Exemplu:

```
+-----+-----+
|PART OF SPEECH|  |
+-----+-----+
+-----+-----+
|CAT = NOUN & SCAT = COMMON & GEN=
MASC |
+-----+-----+
|CAT = NOUN & SCAT = COMMON & GEN =
FEM |
+-----+-----+
|CAT = NOUN & SCAT = COMMON & GEN =
NEUTER|
+-----+-----+
|CAT = NOUN & SCAT = PROPER & GEN =
MASC |
+-----+-----+
|CAT = NOUN & SCAT = PROPER & GEN =
FEM |
+-----+-----+
```

Raportul generat de către sistem pentru selectorul CAT = NOUN & SCAT = COMMON & GEN = MASC este descris mai jos. Numerele conținute în exemplu reprezintă coduri ale paradigmelor flexionare cărora aparțin inflexiunile.

Exemplu:

Selector: CAT = NOUM & SCAT = COMMON & GEN = MASC
A -> [24; NMB = SG & ART = INDEF]
B -> [22; NMB = SG & ART = INDEF]
ELE -> [22; NMB = SG & ART = DEF & CASE = N/A]
ELUI -> [22; NMB = SG & ART = DEF & CASE = G/D]
I -> [23; NMB = M & ART = INDEF] [24; NMB = PL & ART = INDEF]
[22; NMB = M & ART = INDEF] [21; NMB = PL & ART = INDEF]
[20; NMB = M & ART = INDEF]
II -> [23; NMB = PL & ART = DEF & CASE =


```

= N/A]
[24; NMB = PL & ART = DEF & CASE =
= N/A]
[22; NMB = PL & ART = DEF & CASE =
= N/A]
[21; NMB = PL & ART = DEF & CASE =
= N/A]
[20; NMB = PL & ART = DEF & CASE =
= N/A]
ILOR -> [23; NMB = PL & ART = DEF & CASE
= G/D]
[24; NMB = PL & ART = DEF & CASE
= G/D]
[22; NMB = PL & ART = DEF & CASE
= G/D]
[21; NMB = PL & ART = DEF & CASE
= G/D]
[20; NMB = PL & ART = DEF & CASE
= G/D]
L -> [23; NMB = SG & ART = INDEF]
[23; NMB = SG & ART = DEF & CASE =
= N/A]
LUI -> [24; NMB = SG & ART = DEF & CASE =
= G/D]
LUL -> [23; NMB = SG & ART = DEF & CASE =
= N/A]
LULUI -> [23; NMB = SG & ART = DEF &
CASE = G/D]
U -> [21; NMB = SG & ART = INDEF]
UL -> [21; NMB = SG & ART = DEF & CASE =
= N/A]
[20; NMB = SG & ART = DEF & CASE =
= N/A]
ULUI -> [21; NMB = SG & ART = DEF & CASE
= G/D]
[20; NMB = SG & ART = DEF & CASE =
= G/D]

```

3.5. Paradigme flexionare

Lexicograful poate solicita vizualizarea paradigmei flexionare selectând opțiunea Inflectional paradigms.

Pentru un selector specificat, se va obține un raport privind paradigmele flexionare (în forma simplă sau incluzând meniurile de achiziție morfologică).

Exemplu:

```

Selector: CAT = NOUN & SCAT = COMMON &
GEN = MASC
[20 L UL ULUI I II ILOR]
[21 U UL ULUI I II ILOR]
[22 E ELE ELUI I II ILOR]
[23 L LUL LULUI I II ILOR]
[24 A L LUI I II ILOR]
Show inflectional paradigm within screens [Y / N]
? N
Inflectional paradigm: 21

```

3.6. Intrările lexicale/Generare

Intrările lexicale prin analizor - sintactic YACC la nivel teoretic

Intrările lexicale pot fi listate în conformitate cu restricțiile de selecție, aplicate câmpurilor leme, și prin descriere paradigmatică.

Exemplu:

```

Lemma: CAPATA
Selector: CAT = VERB
Expanded form [Y / N] ? : N
Complete form[Y/N]?:Y
(CAPATA
([CAT = VERB] 1
([ <-> CAPAT
[FORM = FINITE & MOOD = IND & TENSE =
PS & NMB = SG & PER = 2] <-> CAPET
[FORM = FINITE & MOOD = SUBJC & NMB =
SG & PER = 2] <-> CAPET
[FORM = FINITE & MOOD = SUBJC & PER = 3]
<-> CAPET)))

```

3.7. Implementare

Proiectul MORPHO început în 1986, a avut drept prim rezultat, o versiune prototip pentru calculatoare compatibile PDP-11. A doua versiune a sistemului, cea descrisă aici, este în curs de dezvoltare pe calculatoare compatibile IBM-PC, în limbajul C++. Arhitectura intrării lexiconului, precum și tipurile de relații dintre câmpurile acesteia, pot fi definite pentru un lexicon manevrat de MORPHO-2 și sunt accesibile utilizatorului în scopul definirii lor. Metodele de acces pentru fiecare câmp al intrării, relațiile între câmpurile acesteia precum și între intrări diferite sunt controlate de către sistem și utilizator. Aceste restricții nu trebuie interpretate drept limitări ci drept o abordare disciplinată a procesului de construire a lexiconului. Structura unei intrări a lexiconului a impus utilizarea multilistelor precum și a înregistrărilor de lungime variabilă. Metodele de indexare a lexiconului utilizând arbori B+ virtuali prefixați precum și modul de grupare a datelor pentru o prelucrare morfo-lexicală optimă, au condus la un timp mediu de răspuns a procesului lexical, care este independent de dimensiunea lexiconului (pentru detaliu privind analiza performanțelor vezi [25]). Ulterior, s-au îmbunătățit performanțele arborilor B+ virtuali prefixați cu arbori binari SRD.

Bibliografie

1. **AHO, A., ULLMAN, L.:** The Theory of Parsing, Translation and Compilings Prentice-Hall, Englewood Cliffs, NJ.
2. **CARPENTER, B.:** The PROP User Manual, Cornege Mellon University, 1989.
3. **DUMITRESCU, C.:** Paradigmatic Morphology Modeling and Lexicon Management with MORPHO-2. În: Proc. of the 5th EURALEX International Congress, Tampere University, 1992(a).
4. **DUMITRESCU, C.:** MORPHO-2 Reference Manual, Research Institute for Informatics, Bucharest, 1992(b).
5. **DUMITRESCU, C., DAN, G.:** Dicționar de obiecte utilizabil în contextul gramaticilor de tip PART, Raport tehnic, ICI, București, 1993.
6. **DUMITRESCU, C., DIACONU, C.:** Algoritm de unificare a grafurilor aciclice directe pentru gramatici PART, Raport tehnic, ICI, București, 1993.
7. **DUMITRESCU, C., NICA, A.:** Gestiunea macrodefinițiilor sintactice - semantice de tip PART, Raport tehnic, ICI, București, 1993.
8. **ESTIVAL, D.:** The ELU User Manual, ISSCO, Geneva, 1990.
9. **EVANS, R., GAZDAR, G.:** Inference in DATR. În: The DATR Papers, February, 1990. Cognitive Science Paper, Sussex, UK, 1989.
10. **GAZDAR, G. and PULLMAN:** Computationally Relevant Properties of National Languages and their Grammars, New Generation, 1985, pp. 273-306.
11. **GAZDAR, G., MELLISH, C.:** Natural Language Processing in POP-11: An Introduction to Computational Linguistics, Addison Wesley, 1989.
12. **HIRSH, S.:** P-PATR, A Compiler for PATR Grammars, CSLI/SRI International, 1988.
13. **JOHNSON, C.:** YACC-Yot Another Compiler, Bell Laboratories, NJ, 1975.
14. **JOSHI, A.:** Tree Adjoining Grammars: How Much Context - Sensitivity is Required to Provide Reasonable Structural Descriptions? În: Natural Language Processing: Psycholinguistic, Computational and Theoretic Perspectives, Dowty, D, Karttunen, L. et al. (eds), Cambridge University Press, NY, 1995.
15. **MANASTER-RAMER, A.:** Topics in Mathematics of Language, ESSLI-92, Essex University, 1992.
16. **NAKAZAWA, T.:** An Extended LR Parsing Algorithm for Grammars Using Feature-Based Syntactic Categories, In EACL-91, Berlin, 1991, pp. 69-74.
17. **SALDER, L., ARNOLD, D.:** Unification and Machine Translation, Working Papers in Language Processing 34, Essex University, 1992.
18. **SERBĂNAȚI, L.:** Limbaje de programare și compilatoare, Editura Academiei, București.
19. **SHIEBER, S.:** Evidence Against the Context-Freeness of Natural Language, Linguistics and Philosophy, 8, 1988, pp. 333-343.
20. **SHIEBER, S.:** An Introduction to Unification - Based Approaches to Grammar, CSLI Lecture Notes No.4, Stanford University, 1986.
21. **SHIEBER, S.:** A uniform Architecture for Parsing and Generation, In COLING-88 614-619, Budapest, 1988.
22. **SIEKMANN, J.:** Unification Theory and Natural Language Processing, ESSLI-92, Essex University, 1992.
23. **TOMITA, M.:** Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems, Kluwer Academic Publishers, 1986.
24. **WINOGRAD, T.:** Language as a Cognitive Process, Addison Wesley, 1983.
25. **STROUSTRUP, B.:** The C++ Programming Language, Addison - Wesley Publishing Company The C++ Reference Manual, Borland International, 1986.
26. **TUFIȘ, D., DUMITRESCU, C.:** MORPHO - A Dictionary Management System. În: Proc. of the 13th Seminar on Data Base Management Systems, Mamaia, România, 1990.