TECHNIQUE

LETTERS WITH VARIABLE VALUES AND THE MECHANICAL INFLECTION OF RUMANIAN WORDS

Minerva Bocşa
University of Timişoara
Romania

The generation by computer of written Rumanian words faces two difficult problems: to produce automatically the numerous alternations which modify the stem and to add the inflectional endings, building a rich set of classes and subclasses. The mechanical morphological analysis is also complicated because of the stem's phonetic alternations.

For example, the Rumanian words

UNIVERSITATE / UNIVERSITĂŢI	(university)
SERIOS / SERIOŞI / SERIOAŞA	(serious)
PUTEA / POT/POŢI / POATE	(may)
VEDEA / VĂD/VEZI / VĂZUI / VADA	(to see)

present the alternations

(1) A/Ă, T/Ţ, S/S, O/OA, U/O/OA, E/Ă/A, D/Z

Phonetic rules describing the occurrence of these stem modifications have several exceptions and must include the presence or absence of stress, which is not marked in ordinary experiments in mechanical translation from English into Rumanian [16] and so on. Phonetic alternation in Rumanian has been investigated by Lombard [11], Felix [7], Juilland and Edwards [10], Augerot [1], and others.

The preparatory work for our automatic linguistic task has several stages:

Examine the inflection of each word.

Establish the set of phonetic alternations.

Attach a specific variable letter to each alternation. In our conception [4] these are different from those of [9, 14, 15].

Design a binary code for the variable letters, taking into account the possibilities of the IRIS 50.

Detach morphological parameters.

Code each word.

Punch a deck of cards.

The card file is the Morphological Dictionary. It is exploited by the programs in various ways. Here the working principles of a program to produce the paradigm (set of inflected forms) of each word in the Morphological Dictionary are presented.

In this process the computer writes the inflected forms in the p positions of the paradigm P The stem allomorphs constitute a set A with n elements. The different distributions of the allomorphs of A in P are described by a set G of *grouping functions* spelling. Nevertheless, the words with nonconstant stem are too numerous to be considered irregular. The method of storing the several allomorphs of the stem for automatic inflection misses the natural unity of the word.

We have constructed a mechanical *Morphological Dictionary*, containing 2058 written Rumanian words with a *synthetic representation* of all these phonetic alternations. An algorithm based on this representation generates the inflectional noncompound forms of these words. They are Rumanian nouns, adjectives, and verbs, the main part belonging to the *basic word stock* [8, 17]. About 45 percent of them present stem alternations.[1]

The algorithm whose logic was given in [3] is the background of a set of programs written in the programming language ASSIRIS for the French computer IRIS 50 and its Rumanian counterpart FELIX C-256. The programs were recently run at the Territorial Electronic Calculus Center of Timisoara, verifying the algorithm.

The synthetic representation uses G. C. Moisil's notion of *letters with variable values* [14, 15], which V. Gutu Romalo developed [9]. The setting of our research is Marcus's theory of mathematical linguistics [12, 13], Diaconescu's study of word segmentation and the degree of regularity [5, 6], Domonkos's

[1] It seems that in Rumanian only 20 percent or even less of the total number of words have these phonetic alternations, but in our dictionary reference is made generally to the most frequently used words, with relative frequency above 0.22‰ [17].

identified by numerals. Thus grouping function 00 associates allomorph *a* in *A* with positions 1, 2, 5, 6, ... in *P*, allomorph *b* in *A* with positions 3, 8, ... in *P* , etc. The different *partitions* of *A* are called *allomorph configurations* and symbolized by a/b (with *n* = 2), ab/c, a/bc, a/b/c, ... (with *n* = 3), etc. A *variable letter* maps the elements of the partition into the Rumanian alphabet A, A, A, B, ..., Z, Ø (here Ø represents the empty letter). Thus the variable letter T/C with the configuration ac/bd has the *realization* T in allomorphs *a* and *c*, and another realization C in allomorphs *b* and *d*. Not all of the theoretically possible variable letters exist in Rumanian; we found 85.

The set of fixed, variable, and empty letters is called the generalized Rumanian alphabet. A version of it is given in [2]. Words can be represented in this alphabet in either external or internal code.

The program operates in several steps which are described and then illustrated.

Input. In the Morphological Dictionary, the fixed letters are punched in accordance with the standard card code. Each variable letter is punched as a numerical prefix of one or two decimal digits followed by a letter. Part of speech, number of allomorphs, word length, stem length, etc. appear as parameters.

1. Recoding. The computer reads the word on the punched card and recodes it into an internal code; each letter is one byte. A fixed letter has zone E or F (leading four bits 1110 or 1111); variable letters have other zones. The recoding instruction in IRIS 50 is TRTR (translate and test).

2. Realization. The program reads the word byte by byte. If the zone is E or F, it writes the byte into the allomorph registers. If the zone is less than E, the program constructs a realization for each allomorph and stores it in the allomorph register.

The principles that govern the decoding of a variable letter into realizations are given in [3]. As an example, take the rule for regular variable letters (zone 0, 1 ... 7). Each regular variable letter has two realizations, and in the internal code the zone of each realization is F. The numeric of one realization is identical with the numeric of the regular variable letter, and the numeric of the other realization is greater by 1. The method of encoding partitions for regular variable letters is explained on the next frame.

The next program stage is on frame 43.

ALLOMORPH CONFIGURATIONS FOR REGULAR VARIABLE LETTERS

Eight zones (0, 1, ..., 7) encode regular variable letters. Each stem has two, three, or four allomorphs. Each partition of the paradigm has two members for a regular variable letter; the numeric of the variable letter is copied into the allomorphs of the first member of the partition, and incremented by 1 into those of the second member.

	Number of Allomorphs		
Zone	2	3	4
0	a/b		ac/bd
1	a/b		a/bcd
2			ab/cd
3	a/b		ac/bd
4	a/b	a/bc	ad/bc
5	a/b	a/bc	a/bcd
6		ac/b	acd/b
7		ab/c	ab/cd

3. Recoding. The program recodes the allomorphs into EBCDIC by another TRTR instruction.

4. Distribution. The program distributes the allomorphs to their locations in another region. The word's grouping function controls the process.

5. Inflection. The program adds the inflectional endings to the right of the stem allomorph in conformity with the class and subclass noted on the punched card.

6. Printing. The program condenses the empty letter and prints the inflected forms.

We illustrate concisely these phases for two words from our Morphological Dictionary, the verbs A PUTEA (may), and A VEDEA (to see). They have, respectively, four and five different allomorphs of the stem.

Input. The content of the card is

PUTEA	P8U19A8TEA	V4	100403
VEDEA	V9E9DEA	V5	070300

8U, 19A, 8T, 9E, and 9D are variable letters in the external code.

Some morphological parameters are

V	verb; part of speech
4 5	number of allomorphs
10 07	word length
04 03	stem length
03 00	grouping function

1. After translation into the internal code the words are represented in storage as

EA 84 A9 86 F2 FO

E6 92 93 F2 FO

EA, F2, FO, and E6 represent the fixed letters P, E, A, and V. 84, A9, 86, 92, and 93 represent the variable letters U/O, ∅/A, T/T., E/Ă/A, and D/Z. The symbol ∅ will be replaced by blank.

2. The four or three stem letters, specified by 04 or 03 on the punched card, give the following four or five allomorphs.

a	EA F5 FF FA	*a*	E6 F2 FC
b	EA F6 FF FA	*b*	E6 F1 FC
c	EA F6 FF FB	*c*	E6 F2 FD
d	EA F6 FO FA	*d*	E6 F1 FD
		e	E6 FO FC

The program decodes the irregular variable letter 84 and produces the realizations U and O (bytes F5, F6) in the allomorphs *a* (U) and *b*, *c*, *d* (O), in accordance with a translation table. (3) The allomorphs are translated into EBCDIC.

4. The allomorphs are placed in new registers as specified by the grouping functions 03 and 00.

PU T, PU T, PO T, PO T, POAT, PU T, PU T, PO T, ...

VED , VED , VAD , VEZ , VED , VED , VED , VĂD , ...

5. The inflectional endings are added.

PU TEA, PU TERE, PO T, POŢI, POATE, PU TEM, PU TEŢI, PO T, ...
VEDEA, VEDERE, VĂD, VEZI, VEDE, VEDEM, VEDETI, VĂD, ...

6. The computer condenses the empty letter in A PUTEA and prints theinflected forms.

The variable-letter method has the advantage of keeping the unity of the word in the Morphological Dictionary and producing the inflected forms correctly. At the same time it regularizes the greatest part of the irregular words. The only irregular verbs that still remain are A AVEA (to have), A DA (to give), A FI (to be), A LUA (to take), A STA (to stand). The other so-called irregular verbs A BEA (to drink), A MINCA (to eat), A RELUA (to retake), A USCA (to dry), A VREA (to want), and all the other semiregular verbs belonging to the third conjugation [5, 14] are regular for our algorithm, and so are the irregular nouns SORĂ-SURORI (sister), NORA-NURORI (daughter-in-law), OM-OAMENI (man), etc.

The program contains 1455 ASSIRIS statements and generates the inflected forms for all the 2058 words included in the Morphological Dictionary in 1 minute 39 seconds. It represents an experimental verification of our algorithm and may be extended without essential modifications to all other Rumanian words, coded in the same way.

Another program meant for users receives a word from the punched card without its special external code or grammatical parameters, looks for it in the Morphological Dictionary file now stored on the magnetic disk, and, if it is found, produces the paradigm of the word. Examples of its output appear on the next two frames.

Subsequent frames exhibit the complete internal and external codes.

The variable-letter method enables us to form an easy algorithm for morphological analysis, as indicated in [2].

TRANSCRIBED OUTPUT

```
          Cuvîntul cerut : PUTEA     Forma flexionară : Paradigma

          Răspunsul ordinatorului :

                1. PARADIGMA VERBULUI  A PUTEA
                ______________________________

________________________________________________________________
Nr.prs. Prezent   Imperfect Perfect Mai mult ca Prezent    Impe-
        indicativ           simplu  perfect     conjunctiv rativ
________________________________________________________________
Sg.  I  POT       PUTEAM    PUTUT   PUTUSEM     POT
    II  POȚI      PUTEAI    PUTUȘI  PUTUSEȘI    POȚI       POȚI
   III  POATE     PUTEA     PUTU    PUTUSE      POATE
Pl.  I  PUTEM     PUTEAM    PUTURĂM PUTUSERĂM   PUTEM
    II  PUTEȚI    PUTEAȚI   PUTURĂȚIPUTUSERAȚI  PUTEȚI
   III  POT       PUTEAU    PUTURĂ  PUTUSERĂ    POATE

Modurile
nepersonale : Infinitiv   PUTEA PUTERE
              Participiu  PUTUT
              Gerunziu    PUTÎND
________________________________________________________________
```

TRANSCRIBED OUTPUT

```
          Cuvîntul cerut : VEDEA     Forma flexionară : Paradigma

          Răspunsul ordinatorulus :

                   1. PARADIGMA VERBULUI  A VEDEA
                   ______________________________

____________________________________________________________________

Nr. pers. Prezent   Imperfect Perfect Mai mult ca Prezent ... Impe-
          indicativ           simplu  perfect     conjunctiv  rativ
____________________________________________________________________

Sg.   I   VĂD       VEDEAM    VĂZUI   VĂZUSEM     VĂD
     II   VEZI      VEDEAI    VĂZUŞI  VĂZUSEŞI    VEZI        VEZI
    III   VEDE      VEDEA     VĂZU    VĂZUSE      VADĂ
Pl.   I   VEDEM     VEDEAM    VĂŻURĂM VĂZUSERĂM   VEDEM
     II   VLDEŢI    VEDEATI   VĂZURĂTIVĂZUSERĂŢI  VEDEŢI      VEDEŢI
    III   VĂD       VEDEAU    VĂZURĂ  VĂZUSERĂ    VĂDĂ

Modurile nepersonale : Infinitiv: VEDEA VEDERE
                       Participiu: VAZUT Gerunziu : VĂZÎND
____________________________________________________________________
```

GENERALIZED RUMANIAN ALPHABET (EXTERNAL AND INTERNAL CODE)

Zone Num.	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	E	F
0 0000	ØA Ø/A	1A A/Ø			4A A/Ă	5A Ă/A		7A A/Ă	Â	A
1 0001	ØĂ Ø/A				4Ă Ă/E	5Ă E/Ă	6Ă Ă/E	7Ă Ă/E	F	Ă
2 0010		1E E/Ø							K	E
3 0011	ØÎ Ø/Î		2Î Î/Ă a/b	3Î I/Î	4Î Î/I				M	Î
4 0100	ØI Ø/İ	1I I/Ø							Q	I
5 0101	ØU Ø/U	1U U/Ø			4U U/O	5U O/U			R	U
6 0110		1O O/Ø							V	O
7 0111				3S Ş/S	4S S/Ş	5S Ş/S	6S S/Ş	7S S/Ş	W	S
8 1000	ØS Ø/S								X	Ş
9 1001	ØC Ø/C	1Ç C/Ø	2C C/Ø	3C T/C	4C C/T	5C T/C	6C C/Ţ	7C C/T	Y	C
A 1010				3T Ţ/T	4T T/Ţ	5T Ţ/T	6T T/Ţ	7T T/Ţ	P	T
B 1011		1G G/Ø	2G G/Ø						G	Ţ
C 1100	ØN Ø/N	1N N/Ø	2N N/Ø		4D D/Z	5D Z/D	6D D/Z	7D D/Z	N	D
D 1101	ØH Ø/H				4Z Z/J		6Z Z/J		H	Z
E 1110			2B B/Ø						B	J
F 1111		1L L/Ø							L	Bl

GENERALIZED RUMANIAN ALPHABET (EXTERNAL AND INTERNAL CODE)

Zone Num.	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111
0 0000	8A A/Ø ad/bc	9A A/Ø ae/bcd	10A Ă/E/A a/bd/c				Â	A
1 0001	8G G/P/Ø a/b/cd	9Ă A/Ă abe/cdf	11A E/A ac/bd				F	A
2 0010	8E E/I a/bc	9E E/Ă/A ac/bd/e	12A A/E/Ø a/b/cd				K	E
3 0011	8O O/U abc/d	9D D/Z abe/cd	13A Î/A/Ø a/b/cd				M	Î
4 0100	8U U/O a/bcd	9Z D/Z/Ø ab/c/de	14A E/A ab/cd				Q	I
5 0101	8S S/Ş abc/d	9J D/Z/Ø ac/bf/de	15A Î/Ă/A ab/c/d				R	U
6 0110	8T T/Ţ abd/c	9T T/Ţ/Ø a/b/cd	16A Ă/A ad/bc				V	O
7 0111	8D D/Z/Ø a/b/cd	9Z T/Ţ/Ø ab/c/de	17A Ă/A ab/cd				W	S
8 1000	8L L/Ø ac/bd	9S S/Ş abd/c	18A ØA abd/c				X	Ş
9 1001	8C C/P ab/cd	9G G/Ø ac/bd	19A Ø/A abc/d				Y	C
A 1010							P	T
B 1011							G	Ţ
C 1100							N	D
D 1101							H	Z
E 1110							B	J
F 1111							L	Bl

REFERENCES

1. Augerot, James E. *A study of Rumanian morphophonology.* Dissertation, University of Washington, 1968.

2. Bocşa, Minerva. Codage de l'alphabet généralisé du Roumain pour l'ordinateur IRIS 50 (FELIX C-256). *Cahiers de linguistique théorique et appliquée,* No. 10, Fasc. 2, 1973.

3. Bocşa, Minerva. Algoritm pentru generarea cuvintelor limbii române in ASSIRIS. *Revista de analiză numerică şi teoria approximaţiei,* vol. 3, fasc. 1, Cluj, 1974.

4. Bocşa, Minerva. *Dicţionar morfologic automat al limbii române pentru ordinatorul IRIS 50 (FELIX C-256).* Dissertation, Universitatea din Timişoara, 1974.

5. Diaconescu, Paula. Contribuţii la definirea şi clasificarea verbelor regulate în limba română. *Studii si cercetări lingvistice, no. 2, an XI,* Bucureşti, 1960.

6. Diaconescu, Paula. *Structură şi evolutie în morfologia substantivului românesc.* Editura Academiei Republicii Socialiste România, Bucureşti, 1970.

7. Felix, Jiři. Asupra alternanţelor fonologice din flexiunea verbala romaneaască. *Studii şi cercetări lingvistice, no. 6, an XVI.* Bucureşti, 1965.

8. Graur, Alexandru. *Incercare asupra fondului principal de cuvinte.* Bucureşti, 1954.

9. Guţu Romalo, Valeria. *Morfologie structurală a limbii române.* Editura Academiei Republicii Socialiste România, Bucureşti, 1968.

10. Juilland, Alphonse, and P. M. H. Edwards. *The Rumanian verb system.* Mouton, The Hague, 1971.

11. Lombard, Alf. *Le verbe roumain.* Lund, 1954.

12. Marcus, Solomon. *Lingvistica matematică.* Editura didactică şi pedagogică, Bucureşti, 1966.

13. Marcus, Solomon, *Algebraic linguistics, Analytical models.* Academic Press, New York and London, 1967.

14. Moisil, Grigore C. Probleme puse de traducerea automata, Conjugarea verbelor în limba romana scrisa. *Studii şi cercetari lingvistice,* no. 1, Bucureşti, 1960.

15. Moisil, Grigore C. Problèmes posés par la traduction automatique. La déclinaison en roumain écrit. *Cahier de linguistique theorique et appliquée,* no. 1, Bucureşti, 1962.

16. Nistor Domonkos, Erica. *Algoritm de traducere automată din limba engleza in limba romana.* Editura Didactică şi Pedagogica, Bucureşti, 1966.

17. Suteu, Valeriu. Observaţii asupra frecvenţei cuvintelor în operele unor scriitori romani. *Studii si Cercetari Lingvistice* no. 3, Bucureşti, 1959.