Computer Science Journal of Moldova, vol.**14**, no.1(40), 2006

# The ascertainment of the inflexion models for Romanian*

Svetlana Cojocaru

**Abstract**

A method to increase the degree of inflexion process automatization for Romanian is proposed.

## 1 Introduction

The problem of the automation of words inflexion process in Romanian was investigated in [1], [2]. The obtained results permitted to construct a computational lexicon containing about 1 million of words: the lemmas and their word-forms. The inflexion process was based on two methods: a static and a dynamic ones. The first one is operating with a morphological dictionary[3], where the inflexion group is indicated explicitly; the second method tries to find the inflexion model analysing the word's structure, especially the affixes series. These series were determined by examination of vocabularies from different lexicographic sources. The dynamic method was implemented as an interactive program, which is able to inflect automatically about 80% [4] of words. The usage of this program shows that a user intervention is requested often to solve some ambiguities, although those cases could be solved automatically. In this paper we intend to improve the dynamic method in order to increase its degree of automation. In the first section we recall the definition of inflexional grammars with scattered context [1], in the next two sections the inflexion criteria are analyzed and an algorithm to determine the inflexion model for a given word is proposed.

©2006 by S. Cojocaru

*Supported by INTAS project Ref. Nr. 05-104-7633

## 2 Scattered context grammars for vocabulary generation

The starting point for this approach was the book [3], where main part of Romanian inflective words were classified according to the methods of the word-forms creating. There were 100 groups for masculine nouns, 273 - for verbs, etc in the book, and about 30000 words with their group numbers were listed. The classification was made from the linguistic point of view, and, for example, the accents were taken into account. In our case we can operate only with the graphical representation of the word, what equally simplifies and complicates the problem. Nevertheless, this classification was useful and have lead to the idea to introduce the special grammar to formalize word-forms producing.

**Definition 1** *The object $G = \{R, T, *\}$, where $R$ is the set of rules, $T$ is the (ordered) set of the list of endings, $*$ is a special symbol not contained in any words of the given language, is named an inflexion grammar.*

The grammar rules have the following form:

$$[/]^*[\#][N_1]a_1\overline{b_1}a_2\ldots a_{n-1}\overline{b_{n-1}}a_n \longrightarrow a_1^{'}\overline{b_1}a_2^{'}\ldots a_{n-1}^{'}\overline{b_{n-1}}a_n^{'}\ N_2,$$

where $a_i$ , $a_i^{'}$ are arbitrary words and either $b_i$ is nonempty word or the special symbol $*$ stands instead of $\overline{b_i}$. $N_j$ — endings set numbers.

The interpretation of this rule is as follows. Let $w$ be the word to produce word-forms (basic word-form). Every sign / indicates cutting the last letter from $w$. The obtained (after the deletions) word $v$ is considered as a root (if $N_1$ exists) and $N_1$ is its index in endings sets list $L$. In any case the word $v$ should have the form

$$f_0a_1f_1a_2f_2\ldots a_{n-1}f_{n-1}a_nf_n,$$

where every $f_i$ is arbitrary (possible empty) word, not containing (for $i = 1, 2, \ldots, n-1$) the veto subword $b_i$. If there exists more then one representation of this kind the first (scanning $v$ from left to the right

or vice versa if the sign # is present) should be selected. The special character $*$ instead of $\overline{b_i}$ admits arbitrary $f_i$.

After the evident substitution the word $f_0 a_1^{'} f_1 a_2^{'} \dots a_n^{'} f_n$ serves as a second (or first, if $N_1$ is absent) root and $N_2$ is its endings set number.

Veto for $b_i$ is conditioned by the necessity to determine the position of the subword $a_i$ to be substituted.

Using these grammar rules, we can formalize the process of creating of the decomposed vocabulary. According to the classification in [3], it is possible to build the grammar rules for every group. Sometimes more than two roots arise and more than one grammar rule is necessary.

The inflexion grammar for Romanian contains 866 rules and 320 endings sets. They were used to obtain a morphological dictionary with about 30000 basic lemmas.

## 3 Automatic inflexion criteria

The grammar rules define, in fact, the inflexion model on the algorithmic level: cutting a given number of symbols at the word ending, obtaining different roots by means of (parallel) substitutions (in order to produce vowel and consonant alternation), attaching the corresponding endings to the roots. But this method can be applied only to the case, when we know the inflexion group number. If this number is unknown the problem to ascertain the inflexion model having the graphical representation of the word arises. Is it possible to solve this problem algorithmically? The answer is a negative one. The first impediment is to determine the part of speech: there are a lot of homonymies denoting different parts of speech (Example: *abate* – a masculine noun and a verb. In the first case it means "abbot" and "to divert" in the second).

We can restrict the formulation of the problem: is it possible to determine the inflexion model (respecting the conditions mentioned above) if we know the part of speech? The answer is a negative one in this case also. For confirmation one can adduce a list of examples which prove that the ascertainment of the inflexion model is impossible if we don't invoke the phonetic or etymological information. Let us see

only one example of this kind: the feminine noun *masă*. Following the meaning "table" the plural will be formed as *mese*, using the model with vowel alternation "$a \rightarrow e$". But if we follow the meaning "mass" the plural *mase* will be obtained without any alternation. The origin of this phenomenon is an etymological one: in the first case the word derives from the Latin *mensa*, in the second case the French *masse* precedes it [5].

But the problem might be tackled in another way: to establish some criteria which permit after the analysing of the word structure to conclude about the possibility to determine the inflexion model and, if this is possible, to fix the specific model. Otherwise, we will try to formulate the criterion according to which one can affirm that the inflexion process can be performed automatically and denote the corresponding model.

Thus, let we have a word (a lemma) in its graphical representation. We know the part of speech, and the gender in the case of nouns. We will divide all words into three categories: irregular, absolute regular and partial regular.

For each part of speech the belonging to the group of the irregular words is determined by its belonging to the set of words, picked apriori. We will consider absolutely regular the words which admit the automatic inflexion. We will call partially regular the words which need some additional information (except the graphical representation) to be inflected. In the next section we will formulate the criteria of the belonging to the last two groups and establish the corresponding inflexion models.

### 3.1 The algorithm of the inflexion model ascertainment

Let $CG = \{M, F, N, A, V, P\}$ be the set of grammar categories which denote masculine, feminine and neuter nouns, adjective, verb and pronoun respectively. Let $c \in CG$ and $GF$ be an inflectional grammar. We will denote by $L_c$ the list of pairs $(\alpha, \mu)$, where $\alpha$ is a word of category $c$, and $\mu$ is its corresponding inflexion group number. Two inflexion groups $\mu_1$ and $\mu_2$ will be considered equivalent if they have the

same corresponding set of grammar rules from the inflectional grammar $GF$. To simplify the explanation the set of irregular words will be excluded from the examination; their presence or absence doesn't affect the generality of the algorithm.

Let us denote as $N_{max} = max \mid \alpha \mid$ the maximal length of the words $\alpha \in L_c$. Let $A_j = \{a_{1j}, a_{2j}, \ldots, a_{kj}\}$ be the set of endings with length $j$ of words $\alpha$ ($j \leq N_{max}$). We will denote by $n$ the length of the current ending. For each inflexion group $\mu$ we will put in correspondence a set $S_\mu$ of endings, which is initially empty. The equivalent groups will have the same corresponding set.

1. $n := 1$
2. $i := 1$
3. Select all the words containing the ending $a_{in} \in A_n$. For each of them we fix its inflexion group $\mu$.
4. If all the inflexion groups are equal or equivalent we include the ending $a_{in}$ into the set $S_\mu$, exclude from the list $L_c$ the words with ending $a_{in}$ and go to step 6.
5. If the selected words have different (nonequivalent) groups do the following verifications:
   - the ending $a_{in} = \alpha'$ and there are the pairs $(\alpha', \mu_1)$ and $(\alpha', \mu_2) \in L_c$. In this case the word $\alpha'$ is included into the partially regular category;
   - the ending $a_{in} = \alpha'$ and there are the pairs $(\alpha', \mu_1)$ and $(\alpha'', \mu_2)$, where $\alpha'' = \beta\alpha'$. In this case the word $\alpha'$ is included into the partially regular category.
6. Increment $i$ by 1 ($i \leq k$) and repeat the process from the step 3. If $i > k$ increment $n$ by 1 and follow step 2. The process will finish when $n > N_{max}$.
7. Construct the union of the sets having the same corresponding grammar rule.

The obtained result constitutes the set of automatic inflexion criteria.

### 3.2 Example of the algorithm application

We will illustrate the algorithm functioning applying it to the list of masculine nouns from [3]. The list contains about 5000 words. A fragment of it (where we added the corresponding English translations) looks as following:

| | | |
|---|---|---|
| abur | M1 | (steam) |
| leuşor | M2 | (little lion) |
| abonat | M3 | (subscriber) |
| watt | M4 | (watt) |
| brad | M5 | (fir) |
| urs | M6 | (bear) |
| boss | M7 | (boss) |

.....

We will operate with the inflexion grammar $GM$. A part of it is presented below:

M1 1;
M2 2 $u \longrightarrow$i 3;
M3 2 $t \longrightarrow$ ţ 3;
M4 2 $tt \longrightarrow$ ţ 3;
M5 2 $d \longrightarrow$ z 3;
M6 2 $s \longrightarrow$ ş 3;
M7 2 $ss \longrightarrow$ ş 3;
. . . . . .

The grammar rules are refering to the following paradigms:

```
1 [- - - ul ului ule i i i ii ilor ilor]

2 [- - - ul ului ule]

3 [i i i ii ilor ilor]

. . . . . .
```

The algorithm application generated the sets of endings, which ascertain the inflexion groups. We present here a part of them:

$a_f \in$ {b} $\cup$ {ic, ec, rac, mac, bac, âc, uc, dac, oc, nc, lac, zac, vac, rc, lc, geac, tac, lac, nac, pac, sac, jac, şac, cac} $\cup$ {fag, arag, ărag, bag, mag, ng, og, ug, ig, eg, rg, lg} $\cup$ {f} $\cup$ { h} $\cup$ {j} $\cup$ {ul, ol, ăl, ll, şial, cial, til, cil, mil, fil, ril, bil, vil, dil, xil, zil, nil, hil, upil, ral, tal, fal, şal, ibal, nal, lal, mal, pal, gal, dal, ual, val, sal, ghel, fel, udel} $\cup$ {m} $\cup$ {mn, en, in, on, ân, rn, un, vn, gan, can, zan, ban, nan, san, ran, tan, lan, van, han, pan, dan, ţan, uan, fan, aolean, oman, aman, rman, iman, esman, osman, hman, bman, şman, atman, lman, dman, rmăn, badian, radian} $\cup$ {ţap, up, ip, op, rp, mp, ep, cap, sap, rap, lap, nap} $\cup$ {ur, or, ir, âr, rr, ier, ger, mer, per, ler, her, fer, ber, xer, ner, ter, der, zer, jer, ţer, ier, ser, rer, ver, şer, găr, şafăr, tăr, hăr, căr, văr, bar, car, dar, far, ear, gar, har, iar, jar, mar, par, rar, sar, tar, oar, ţar, şar, var, zar, tuar, iuar, ouar, guar, zuar, onar, inar, unar, snar, enar, tnar, arnar, rnar, ânar, gnar, mnar, znar, olar, elar, ilar, glar, ular, blar, slar, plar, bălar, tănar, lănar, omănar, cănar, iermănar} $\cup$ {v} $\cup$ {ez, onz, lz, baz, ăz, âz, ruz, ăuz, moz, guz, tz, muz, suz, luz, iz, mz, anz, laz, uoz, tuz} $\cup$ {ş} $\cup$ {eţ, uţ, nţ, iţ, căţ}$\rightarrow M1$.

$a_f \in$ {it, ot, pt, ct, lt, ut, et, rt, şt, ft, ît, ent, ant, int, ont, unt, sânt, nat, tat, lat, bat, mat, zat, gat, pat, jat, rat, cat, sat, vat, eat, oat, ţat, fat, dat, şat, niat, liat, ciat, uiat, ţiat, miat, giat, diat, ariat, triat, priat } $\rightarrow M3$,

$a_f \in$ {tt} $\rightarrow M4$,

$a_f \in$ {d} $\rightarrow M5$,

$a_f \in$ {os, es, as, us, is, âs, ns, ps, rs, cs} $\rightarrow M6$,

$a_f \in$ {ss} $\rightarrow M7$

If the ending $a_f$ of the word $w$ belongs to one of the mentioned above sets, then it can be inflected according to the grammar rules which correspond to the indicated inflexion group.

The following endings point to partially regular nouns:

$p_f \in \{\text{osc}\} \to M17, M18;$
$p_f \in \{\text{iac}\} \to M13, M39;$
$p_f \in \{\text{drag}\} \to M14, M15;$
$p_f \in \{\text{gaci}\} \to M73, M98;$
$p_f \in \{\text{opil, cal, bel, ocel}\} \to M1, M12;$
$p_f \in \{\text{rial}\} \to M1, M43;$
$p_f \in \{\text{bouşor, cer}\} \to M1, M2;$
$p_f \in \{\text{leat}\} \to M3, M31;$
$p_f \in \{\text{lustru, leandru}\} \to M62, M63;$
$p_f \in \{\text{iandru}\} \to M62, M65;$
$p_f \in \{\text{roz}\} \to M1, M29.$

If an ending of a masculine noun belongs to the endings set $a_f$ one can affirm, that its declination can be performed according to the grammar rules which correspond to this set. If the ending belongs to a set $p_f$ than we can't indicate the unique model of inflexion and need some additional information to perform declination. For example, in the inflexion program [2] the user is asked to select one from the several possible word–forms of plural. This information is sufficient to fix the appropriate inflexion model. If the ending doesn't belong to any of the sets $a_f$ or $p_f$, and the word doesn't belong to the list of irregular words it remains to find other methods to produce the corresponding word-forms.

The obtained result was verified on the set of about 2000 masculine nouns from [5], which doesn't intersect the set of masculine nouns from [3]. We have seen the complete correctness in the cases when the ending belongs to the sets $a_f$ or $p_f$. At the same time we have found about 3% of nouns whose endings were not included into the sets generated by the described algorithm.

**Conclusions and results**

The automatization of the inflexion process is one of the problems which appear on computational lexicons constructing. It is especially difficult for high inflectional languages to which the Romanian one belongs as well. We have elaborated two methods to solve this problem:

a static and a dynamic one. The second one can be substantially improved applying the algorithm stated below. A computational lexicon for Romanian containing about 1 mln. words (obtained by inflexion of 60 000 lemmas) was constructed using these methods. The lexicon was used for different linguistic applications: the spelling checker for Romanian [6], the data base of linguistic resources [7], the search algorithm for web pages [8].

## References

[1] S.Cojocaru, M.Evstunin, V.Ufnarovski. Detecting and correcting spelling errors for Romanian language. Computer Science Journal of Moldova, Vol.1, N.1,1993, Kishinev, pp.3–22.

[2] E.Boian, A.Danilchenco, L.Topal. The automation of speech parts inflexion process. Computer Science Journal of Moldova. 1993, Vol. 1, N.2, pp.14–26

[3] A.Lombard, C.Gâdei. Dictionnaire morphologique de la langue roumaine. Bucuresti, Editura Academiei, 1981, 232 p.

[4] The inflexion regularities for the Romanian language.Computer Science Journal of Moldova, Vol.4, N.1, 1996, Kishinev, pp.40–58

[5] Dictionarul explicativ al limbii romane. Academia Romana, Institutul de Lingvistica "Iorgu Iordan", Editura Univers Enciclopedic, 1998, 1192 p.

[6] S. Cojocaru: Romanian Lexicon: Tools, Implementation, Usage. In: Dan Tufis and Poul Andersen (eds.), Recent Advances in Romanian Language Technology. Editura Academiei, 1997, pp. 107–114.

[7] E. Boian, C. Ciubotaru, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahova. Lexical resources for Romanian. Memoriile ştiinţifice ale Academiei Române, Bucharest, Romania, 2004, pp.267–278

[8] O.Burlaca, S.Cojocaru, C.Gaindric. A content management system for electronic theses.Proceedings of the 4rd International Conference on Microelectronics and Computer Science. Vol.II, 2005, pp.509–513.

S.Cojocaru, Ph.D.,

Received March 30, 2006

Institute of Mathematics and Computer Science,
Academy of Sciences, Moldova
5, Academiei str., Chişinău,
Moldova, MD 2028
e-mail: *sveta@math.md*