TASK QUARTERLY 4 No 2 (2000), 291–300

Invited Paper

A GRAMMAR FOR THE POLISH INFLECTION LEXICON

WIESŁAW LUBASZEWSKI

Computational Linguistics Unit,
Jagiellonian University,
and Computer Science Department, AGH Technical University,
Mickiewicza 30, 30-059 Cracow, Poland
lubaszew@uci.agh.edu.pl

(Received 20 January 2000)

Abstract: The inflection dictionary of Polish language availlable on the web: *www.icsr.agh.edu.pl/fleksbaz* was generated automatically. The text below is intended to explain this generation process.

Keywords: computational linguistics, machine dictionary, formal grammar, morphology

1. Linguistic evidence

Polish is a highly inflected language. Each word has a number of inflectional forms: verbs have 47 (if we exclude participles), adjectives 44, numerals up to 49, nouns and pronouns 14, and adverbs 3. These figures, and the fact that many words have irregular stem alternations, show that Polish inflection presents real problems for the computational linguist.

2. Generative phonology tradition[1]

If one asks how to inflect a particular Polish word the grammarian's answer is that one must first select the proper inflection ending and then must apply the proper stem alternation rule. For example, if one wants to produce the form *starzy*, that is the male nominative plural of the adjective *stary* 'old', one must select the nominative plural ending *–y* and then must apply the palatalization rule, which changes *r* to *rz* in inlection stem *star*. That is to say that to inflect a particular word from the generative point of view means to produce a particular grammatical form of a given word.

Knowing the above, one can construct *empirically motivated* formal grammar[2], which will perform a forementioned operation.

Let's assume that the lexical entry for the Polish word *stary* is the set:

{star + {Male_Nom_Sing, Male_Gen_Sing, ..., Male_Nom_Plur,
Male_Gen_Plur, ..., Female_Nom_Sing, Female_Gen_Sing, ...},

where *star* is the inflection stem, '+' is the boudry symbol and *Male_Nom_Sing, Male_Gen_Sing, ...*, stands for the grammatical cases of Polish adjectives.

Let's also assume that a particular inflection form, *i.e.* the grammar input string is derived from lexical entry by a certain algorithm during sentence production. We shall not discuss that algorithm. For our purposes is enough to say that the grammar input for the form *starzy* is the string of symbols:

star + Male_Nom_Plur

If one wants to follow the Chomskyan tradition, one must present the grammar in the form of replacement rules. Lets start with the nominative plural ending production rule. The rule may have a form:

Nom.Plur → y

and it replaces the case symbol *Male_Nom_Plur* by an inflection ending symbol *y*.

Polish adjective forms the nominative plural for male gender by two different endings, *i.e.* *–i* and *–y*. This means that one has to introduce into the grammar two different rules for the replacement of *Male_Nom_Plur* symbol. In such a situation the grammar must know, which ending is applicable to a particular word. This knowledge is introduced by ending marker *e_mark*, *i.e.* additional symbol which marks lexical entry and the proper replacement rule. As a result, marking changes the form of the input string:

e_mark, star + Male_Nom_Plur

and the form of the rule:

Male_Nom_Plur → y / e_mark

what means: replace (→) *Male_Nom_Plur* symbol by an inflection ending *y* only if the (/) input string and the rule have the same *e_mark* symbol.

Having selected the inflection ending one can form the rule, which will perform the final consonant palatalization. The rule, which follows the Chomskyan tradition may take the form:

r → rz / __+ y

which means: replace (→) *r* by *rz* only if (/) *r* is final stem consonant (__+) and it occurs before the ending *y*.

The Polish native speaker knows that inflection ending *–y* forms the noninative singular and nominatavive plural of male adjectives, but the inflectional stem of the adjective *stary* changes from *star–* to *starz–* before the ending *–y* only in the nominative plural. It makes clear that the global phonological rule which says that a front vowel causes consonant palatalization is not appropriate, and this means that the real palatalization rule should be biconditional:

r → rz / __+ y / Male_Nom_Plur

which means: replace (→) *r* by *rz* only if (/) *r* is final stem consonant (__+) and it occurs before the ending *y*, and only if (/) *y* is the male nominative plural ending.

Further linguistic evidence shows that the rule just introduced is too general. Let's compare the behaviour of the final stem consonant before the ending *–y*, which can also occur in the nominative plural of male nouns, *e.g. aktor–0 : aktorz–y* 'actor', *senior–0 : seniorz–y* 'senior', *amor–0 : amor–y* 'amor', *gbur–0 : gbur–y* 'boor', *traktor–0 : traktor–y* 'tractor'. Here, as in the nouns: *aktor : aktorz–y, senior : seniorz–y* the palatalization takes place before *–y* in the nominative plural, but *cf.* the co–existence of nouns: *amor : amor–y, gbur : gbur–y, traktor : traktor–y* in which palatalization does not occur.

This shows that there is a need for a new approach to the stem alternation process. To distinguish noun types one has to mark the proper lexical entries and the rule, which will operate on them. Let's introduce the alternation mark, *i.e. a_mark*, which will control the application of palatalization rule, which now takes a form:

r → rz / __+ y / Male_Nom_Plur / a_mark

which means: replace (→) *r* by *rz* only if (/) *r* is final stem consonant (__+) and it occurs before the ending *y*, and only if (/) *y* is Nominative Plural ending, and only if (/) the processed word is marked by the same *a_marker* as the rule.

Such a rule will perform palatalization in forms: *aktorz–y*, *seniorz–y* and will reject the palatalization process in the nominative plural of nouns like *amor–y, gbur–y, traktor–y.*

It is clear that marking will also change the input string, which will take the form:

e_mark, a_mark, star + Male_nom_Plur

Adding more linguistic evidence, *e.g.* the fact that the inflection ending *–y*, forms also the nominative plural of female nouns, *e.g. stor–a : stor–y* 'curtain' one has to introduce another rule:

r → rz / __+ y / Female_Nom_Plur /a_mark

Using marks grammar will correctly perform the inflection form generation process. But such a grammar will get confused during the processing of an unknown word, *i.e.* a word, which is not strored in the lexicon. In that point the formal grammar is outperformed by the natural grammar used by the Polish native speaker, which will correctly process any unknown word. In other words, the formal grammar as shown above does not describe that phenomenon, which we call the natural language creativity.

3. Pregenerative tradition

If one asks how to inflect a particular Polish word the grammarian's answer is that one must first learn the proper inflection pattern and then apply that pattern to a particular lexical item. This means that if we take for example the personal male noun *aktor* 'actor', and we apply the inflection pattern to that word we shall obtain

the following list of inflection forms with associated descriptions:

Nominative Singular	aktor 0
Genitive Singular	aktor a
Dative Singular	aktor em
Accusative Singular	aktor a
Instrumental Singular	aktor em
Locative Singular	aktorz e
Vocative Singular	aktorz e
Nominative Plular	aktorz y
Genitive Plular	aktor ów
Dative Plular	aktor om
Accusative Plular	aktor ów
Instrumental Plular	aktor ami
Locative Plular	aktor ach
Vocative Plular	aktorz y

In each form of the list, *e.g.* in the third row from the top we can distinguish: a grammatical case decription, *i.e. Dative Singular*, inflection stem, *i.e. aktor*, and an inflection ending, *i.e. em*. We can also see that the inflection stem changes from *aktor* to *aktorz*. The change *r* to *rz* is called an alternation.

Now we can say that the inflection pattern applied to the word *aktor* has a form of the matrix:

Nom._Sing	# : #	0
Gen_Sing	# : #	a
Dat_Sing	# : #	owi
Acc_Sing	# : #	a
Instr_Sing	# : #	em
Loc_Sing	r : rz	e
Voc_Sing	r : rz	e
Nom_Plur	r : rz	y
Gen_Plur	# : #	ów
Dat_Plur	# : #	om
Acc_Plur	# : #	ów
Instr_Plur	# : #	ami
Loc_Plur	# : #	ach
Voc_Plur	r : rz	y

where *Nom_Sing* stands for Nominative Singular, *Gen_Sing* stands for Genitive Singular, *etc. r : rz* is consonant alternation, # : # stands for an empty allternation, and *0, a, owi, a, ...* are inflection endings.

Now we can define an inflectional pattern, which is the key concept in linguistic tradition. We shall define it in terms of three related elements:

F — which is the unique set of inflectional endings, unique in that it differs from the rest of the F's at least by one ending, *e.g.* {0, a, owi, a, em, e, e, y, ów, om, ów, ami, ach, y};

O — which is the set of descriptions, one description for one ending, *e.g.* {Nom_Sing, Gen_Sing, Dat_Sing, Acc_Sing, Instr_Sing, Loc_Sing, Voc_Sing, Nom_Plur, Gen._Plur, Dat_Plur, Acc_Plur, Instr_Plur, Loc_Plur, Voc_Plur}. Since here we shall use abbreviated descriptions;

T — which is the set of stem alternation, *e.g.* {<# : #>, <r : rz>}.

If we compare inflection patterns of words like: *aktor* 'actor', *drapichrust* 'tramp', *obdartus* 'ragemuffin', we can find that they have the same F and O, but F's of each pattern have some different elements, *i.e. rz : rz*, *t : ci* and *s : si*. We argue that this difference is a superficial one. First, it can be easily seen that alternations in each pattern occurs in the same place, and second, that these alternations are of the same quality, *i.e.* each alternation changes a hard consonant, *i.e. r, t, s* to its soft equivalent, *i.e. rz, ci, si*. This means that we can distinguish an inflection pattern common for a certain group of words, in our example pattern for a certain subset of *male personal nouns*. The pattern takes the form:

F = {0, a, owi, a, em, e, e, y, ów, om, ów, ami, ach, y};

O = {Nom_Sing, Gen_Sing, Dat_Sing, Acc_Sing, Instr_Sing, Loc_Sing, Voc_Sing, Nom_Plur, Gen._Plur, Dat_Plur, Acc_Plur, Instr_Plur, Loc_Plur, Voc_Plur};

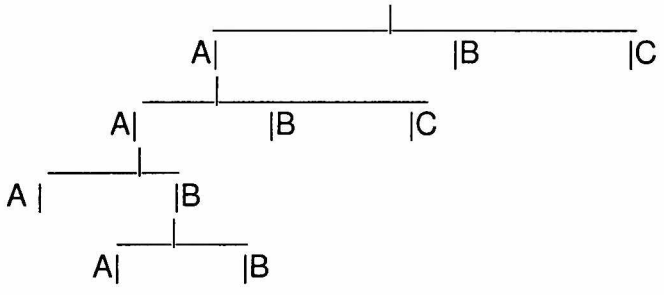
T = {<# : #>, <r : rz>, <t : ci>, <s : si>, ...}.

*

If one asks how to match a particular Polish word and its inflection pattern the grammarian's answer is that one must find the word in the dictionary and then read the word description, which contains the pointer to the proper inflection pattern. This is true. But this answer does not explain the phenomen that the Polish native speaker can properly inflect the word he never heard before, *i.e.* the word, which is not stored in his mental dictionary.

The hypothetical answer is that the native speaker can process a word's graphic or phonic shape as a pointer to the proper inflection pattern. One can describe that process as follows.

First we shall present the set of possible inflection patterns in the form of a labelled tree, in which each node subordinate to the root selects O and each terminal node selects F. Both O and F select T.



That is to say that if the node labelled A takes the value „noun", it will select O = {Nom_Sing, Gen_Sing, Dat_Sing, ..., Voc_Plur}. If then AA takes the value „male_personal", AAB takes the value „nominative_singular = 0", and AABA takes the value „nominative_plural = y" then the terminal node AABA selects F = {0, a, owi, a, em, e, e, y, ów, om, ów, ami, ach, y}. Both O and F select the T, *e.g.* {<# : #>, <r : rz>, <t : ci>, <s : si>, ...}. The path AABA selects the pattern proper, for example, for male personal nouns like: *aktor, drapichrust, obdartus.* The path AAA seletcs the pattern proper for male personal nouns like: *generał* 'general', *admirał* 'admiral'. The path AAA seletcs the pattern proper for male personal nouns like: *nauczyciel* 'teacher', *stręczyciel* 'procurer', and so on.

In the second step one must find the equivalence between a word graphic or phonic shape and the proper path in the tree. We argue that it is possible to find in the word shape right–bounded segment, which points to the path, *e.g. arz* = {Noun, Male, Mom_Sing = 0, Nom_Plur = e}, *ący* = {Adjective, Male, Mom_Sing = y, Nom_Plur = y}, *ować* = {Verb, First_Person_Singular_Present = ę, ...}, *etc.*

4. The computer model of Polish inflection

In our model the inflection paradigm is presented as local grammar, which consists of a grammar label, set of inflection endings and set of context sensitive bidirectional rules, each of which describes a particular stem alternation[3]. See *e.g.* local grammar, which describes the inflection of verbs, like *strzec* 'to watch', *siec* 'to cut', *etc.*

```
;
$BAFCAB
$TER=0, ę, esz, e, emy, ecie, ą, 0, #, my, cie, #, ąc, ący, łem, łeś, ł, łam,
łaś, ła, łom, łoś, ło, liśmy, liście, li, łyśmy, łyście, ły, łbym, łbyś, łby, łabym,
łabyś, łaby, łobym, łobyś, łoby, libyśmy, libyście, liby, łybyśmy, łybyście,
łyby, ono, ony, łszy,
trzec <=> trzeg / __+!0
trzeg <=> trzeż / __+e*, m*, c*, o*
trzec <=> trzeż / __+0 /"rozk
trzyc <=> trzyg / __+!0
trzyg <=> trzyż / __+e*, m*, c*, o*
trzyc <=> trzyż / __+0 /"rozk
rzec <–> rzek / __+ę, ą*, ł*, l*
rzec <–> rzecz / __+e*, m*, c*, o*
rzec <=> rzecz / __+0 /"rozk
iec <=> iek / __+!0
iek <=> iecz /__+e*, m*, c*, o*
iec <=> iecz / __+0 /"rozk
lec <=> lok / __+!0
lok <=> lecz / __+e*, m*, c*, o*
lok <=> lek / __+l*
lec <=> lecz / __+0 /"rozk
```

lok <=> lók / __+ł, łb*, łszy
óc <=> og / __+!0
og <=> oż / __+e*, o*
og <=> óż / __+m*, c*
óc <=> óż / __+0 /"rozk
og <=> óg / __+ł, łb*, łszy
uc <=> uk / __+!0
uk <=> ucz / __+e*, m*, c*, o*
uc <=> ucz / __+0 /"rozk
ąc <=> ęg / __+!0
ęg <=> ęż / __+e*, o*
ęg <=> ąż / __+m*, c*
ąc <=> ęż / __+0 /"rozk
ęg <=> ąg / __+ł, łe*, łb*, łszy
;

The local grammar can generate each inflection form, which belongs to a particular pattern. This means that grammar can also produce base forms of all four participles. It is too much, because the presence of a participle in the paradigm depends on the verb aspect, *i.e.* an imperfective verb has present participles ending in *–ąc* and *–ący* and does not have past participle ending in *–łszy, –wszy.* A perfective verb — on the contrary — has the a past participle and does not have present participles. Unfortunately, we are not able to associate verb aspect with its orthographic shape. Therefore, there is the need for a participle filter, which has the form:

;
absolutyzować : BAAA absolutyzując absolutyzujący absolutyzowany 0
absorbować : BAAA absorbując absorbujący absorbowany 0
absorbować się : BAAA absorbując absorbujący 0 0
abstrahować : BAAA abstrahując abstrahujący abstrahowany 0
adaptować : BAAA adaptując adaptujący adaptowany 0
adaptować : BAAA 0 0 0 adaptowawszy
adaptować się : BAAA adaptując adaptujący 0 0
adiustować : BAAA adiustując adiustujący adiustowany 0
administrować : BAAA administrując administrujący administrowany 0
admirować : BAAA admirując admirujący admirowany 0
adoptować : BAAA adoptując adoptujący adoptowany 0
adoptować : BAAA 0 0 0 adoptowawszy
adorować : BAAA adorując adorujący adorowany 0
adresować : BAAA adresując adresujący adresowany 0
adsorbować : BAAA adsorbując adsorbujący adsorbowany 0
adwokatować : BAAA adwokatując adwokatujący 0 0
afiliować : BAAA afiliując afiliujący afiliowany 0
afiliować : BAAA 0 0 0 afiliowawszy

We also need the filter for adjective grade forms. Each local grammar developed for an adjective and adverb can generate grade forms, but there are mamy adjectives and adverbs in Polish, which use only analytic grade forms, cf. eg. 'old': *stary* and grade form *starszy* in contrary to 'ill' *chory* and grade form *bardziej chory*, never *chorszy*!

*

To complete the model we need a set of rules, which would associate the graphic shape of the word with the proper local grammar. That is, we need the context sensitive rules of the form:

```
;**** sample — noun****
ulec => ACACBC
;
awiec => ACACBC
ywiec => AAACBA
iwiec => AAACBA, AAACAAAA
ywiec => AAACBA, AAACAAAA
ec => ABAAAB / sz __
ec => BAFAAB / rz __
ec => ACACBC
;
; **** sample — verb****
ać => BDA / w, i, n __
; ***** ną : n
nąć => BAEAB / a, i, y, o, u __
; imperative ending — 0
nąć => BAEAA
ąć => BAFCB / d, i, j, l, cz, ż __
;
ść => BAFAAA / a, ó __
ść => BAFCAA / ą, e, j __
ć => BAFB / u __
ć => BAFCAA / ź __
;
```

The string on the left side of the arrow, which we call the „segment", consits of a word formative morpheme and inflection ending. Unfortunately, some segments are ambiguous, *i.e.* they can associate a set of local grammars, see *e.g. ywiec, iwiec*. In addition, some words have an empty segment, *e.g.* verb *biec* 'to run', noun *koń* 'a horse', *etc.* Finally, some words with unambiguous segment can have a parallel inflection, *i.e.* they can use up to three different local grammars, *e.g.* the pronon *on* 'he'. In such cases there is the need for filters, which would preserve the proper association of word and local grammar. The filter takes the form:

; == verb filter for empty segment, eg. biec 'to run' ==
$BW piec BAFCAB ACACBC, ciec BAFAAB, ciec BAFCAB, dobiec BAFAAB, dociec BAFAAB, dociec BAFCAB, dopiec BAFCAB, dostrzec BAFCAB, dowlec BAFCAB, lec BAFAAB, nabiec BAFAAB, naciec BAFAAB, naciec BAFCAB, nadbiec BAFAAB, napiec BAFCAB, nawlec BAFCAB, obciec BAFAAB, obciec BAFCAB, obiec BAFAAB, oblec BAFAAB, oblec BAFCAB, ociec BAFAAB, ociec BAFCAB, odbiec BAFAAB, odciec BAFAAB, odciec BAFCAB, odwlec BAFCAB, opiec BAFCAB, ostrzec BAFCAB, pobiec BAFAAB, podbiec BAFAAB, podpiec BAFCAB, polec BAFAAB, ...
;

5. Conclusion

The system described above is called a lexical grammar of Polish. If the grammar operates on the list of headwords, it can generate full set of inflection forms for each word, that is to say that it produces the inflection dictionary. If the grammar operates on a text, it converts each text–form into a headword. In the second case one would associate the procedure, which will resolve the text–form ambiguity problem.

We belevie that going that way we can obtain some sort of self–expanding dictionary.

---

[1] I shall present just a general idea of the approach and I shall not refer to its higlhy sophisticated extension, which is called *two–level morphology*. I can only point out that two–level morphology, born in the begining of the 80's see R. Kaplan and M. Kay, *Phonological Rule and Finite–State Transducers*, Linguistics Society of America Meating Handbook, 1981, reached it's formal maturity in the mid 90's, see *e.g.* L. Kartunen, *The Replace Operator*, Xerox Research Report, Grenoble 1995, and at the same time faced a language with rich inflection, see J.P. Chanod, *Finite State Composition of French Verb Morphology*, Xerox Research Report, Grenoble 1995.

[2] Noam Chomsky, in the preface to the *Introduction to Formal Grammars* by M. Gross and A. Lentin, wrote: „I would like to stress again that there is still a significant gap between the mathematical and the empirical investigations that fall within the domain of what ultimately may become a mathematical theory of universal grammar. The schema for grammatical description that seems empirically motivated by the facts of particular languages specifies a class of systems that are for the moment, much too complex for fruitful and far–reaching mathematical investigation; furthermore, it must be born in mind that any proposals that can be made today concerning this universal schema are both highly tentative and also somewhat loose in important respects. [...] A mathematical theory of universal grammar is therefore a hope for the future rather then a present reality."

[3] Rule of the grammar uses generative–like notation, where '< = >' is bidirectional replace operator; '/' mens 'if'; '+' is the boundry symbol, which separates the stem and inflection ending; '__' symbolises the string to be replaced; '!' means 'not'; concatenation of quotation mark and the string, *e.g.* "rozk stands for grammatical category symbol; and ' * ' stands for any symbol string.