MORPHOLOGICAL ANALYSIS OF INFLECTED LANGUAGES AND FINITE AUTOMATA

Bach Iván, Farkas Ernő, Naszódi Mátyás
Computer and Automation Institute of
the Hungarian Academy of Science

In the last years the computer analysis of natural languages became an important issue of computer applications. In case of inflected languages the effectiveness of the morphological analyser is or capital importance. One of the first results in this field was published by Kimmo Koskenniemi et altera [1] using two level finite automaton. Their method can be applied quite generally i.e. for any inflected language but its usage more precisely the description of the morphological rules proved to be tiresome.

In this paper a new more simple method is outlined which does not require sophisticated descriptions so it can be used easily by linguists as well.

From the morphological rules through a relatively long lasting algorithm a finite automaton - the morphological parser - is generated.

It will be described what is new in the presented idea and what advantages can be expected from this method.

Words, morphemes and their forms

The word and the word-form should be distinguished. The latter is a character-string which appears in the text to be analysed, the former is an abstract object having a semantical meaning. Hungarian is an agglutinative language i.e. the stem of a word may be followed by affixes and be preceded by prefixes. Composition of words is also a commonly used method in Hungarian to form new words.

The words therefore can be partitioned in functionally different parts, in morphemes (stems, derivative affixes, marks, case-endings, flectional endings). All these morphemes have their forms in which they appear in the text. In the Hungarian each word-form is a series of such morphemes.

These morphemes may not occur in random sequence, there are strict rules governing their presence or absence and their order.

The words belong to different classes depending upon their grammatical category and the type of their inflection. The derivative endings may change the grammatical category of the word. Unfortunately different words and different endings may have identical forms which may cause ambiguity in the analysis.

Word parsing

The goal of the word parsing is to partition the word-form into stem and prefixes resp. suffixes. For example the Hungarian word *legelemibb* (*the most elementary*) should be partitioned as *leg-elem-i-bb*. In languages like English where affixes are rarely - if any - used all possible word-forms may be incorporated in the lexicon. Such solution is impossible in case of a highly agglutinative language like Hungarian where an already inflected word may have additional suffixes. In this paper this word-form will be referred as relative (word) stem.

The sequence of the suffixes is highly predetermined and in addition the affixes can be divided into classes according to their role and according to the grammatical category of the relative word-stem where they might be applied. As it has been mentioned, different affixes may have identical form like *kutyá-d-nak (to the dog*) and *lát-nak* (*they see*).

Some suffixes may cause a change in the root (*ökör ox - ökrök oxen*) others assimilate the (relative) word-stem (so instead of *ház+val - with house* - it will be *házzal*).

The levels of morphology

Three levels of the morphology can be distinguished.

1. Morphosyntax
   Assimilation rules
2. Morphemes
3. Morpheme-forms

The morphosyntax determines the allowed sequences of the affixes. These rules can be described through a graph which can be easily transformed into a finite automaton. The states of this finite automaton correspond to the grammatical categories of the relative word-stems while the transition rules are determined by the morpheme-classes.

More detailed partition in the grammatical categories and morpheme-classes means more accurate and precise morpho-syntactical analysis.

There are languages for instance the Slavic languages where this finite automaton does not contain cycles. The logic of the Hungarian language requires cycles in the finite automaton al though the number of cycles is strongly limited.

A not detailed scheme of a possible finite automaton for the Hungarian language may look like as it follows. The analysis is performed backwards, i.e. from the end of each written word-form.

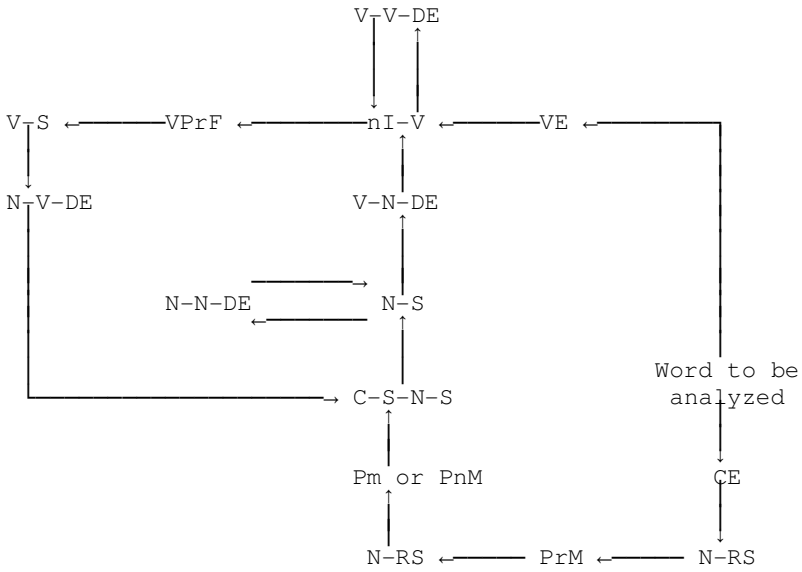
When separating the endings from the relative word-stem the assimilation rules of course must be taken into account.

To every transition branch of the automaton belongs a set of morphemes which are allowed at that particular position. Therefore the possibility of the transitions is analysed with subordinate finite automata.

Every morpheme may have more than one morpheme-form. As

an example the case-ending for the accusative is generally *-t* but its form can be *-at*, *-et*, *-ot*, *-öt* depending upon the word for which is it applied.

With the exception of the derivative endings each morpheme-class supposed to contain the empty affix, too.



Legends:		
	S	Stem
	N	Noun
	V	Verb
	VE	Verb Ending
	nI	not Inflected
	DE	Derivative Ending
	PrF	Prefix
	C-S	Comparative-Superlative
	C-S-As	Comparative-Superlative Affixes
	RS	Relative Stem
	CE	Case Ending
	PM	Plural Mark
	PrM	Possessor Mark
	Pnm	Possession Mark

Assimilation rules and morpheme-forms

From the description of the automaton it follows that during the morphosyntactic parsing process there is an underlying analysis taking assimilation and phonetic rules into account.

These rules can be generally characterized as neighbourhood-relations. Unfortunately these phonetic rules are quite sophisticated and often motivated by language-historical features. However some simple rules of thumb can be derived. Just to mention one such rule when the (relative) word-stem end with the letter *-a*, it will necessarily change when the stem gets a suffix. So the form of the word *kutya* (*dog*) when taken into the accusative case will be *kutyát*.

Both assimilation rules and the rules determining the choice among the possible morphemes can be described through a graph so realized as it has been mentioned – with the help of finite automata.

In Hungarian the governing phonetic rule is vowel harmony.

In some cases however there are difficulties in deciding the correct, phonetic class. Vowels even in identical word-forms may belong to different phonetic classes. In the verb *szív* (*to suck*) the sound *í* behaves as a back vowel (so *szívok – I suck*) while in the noun *szív* (*heart*) it is a front vowel (so *szívek – hearts*).

The sound *i* often used to cause problem since in the ancient. In Hungarian there were two *i*-s; a back and a front vowel. Nowadays not only in the written but also in the spoken Hungarian there is no difference between *i* and *I*, only when the word is inflected turns out the origin of the *i* in question.

In some cases it is practically impossible to follow the "beauty" of the language. In the word *cél* (*goal*) the definitely front-vowel *é* behaves as it were a back one. So the plural of the word is *cél-ok* instead of *cél-ek*. Of course there is always an explanation – the word is taken from the German, *Ziel* and the long *í* in the original word was taken as a back vowel;

Unfortunately all explanation cannot be included in a finite automaton so these irregularities must be taken into account in the lexicon.

About the practical solution

The morphological analyser is composed from two programs. The first one is an automaton-generator which produces the morphological parser a finite automaton. The input word-forms will be analyzed by the second program with the help of the automaton furnished by the first one.

The first program requires as data

1. The graph description of the morphosyntax. This is practically equivalent with the definition of the finite automaton.
2. The phonetic and assimilation rules given as neighbourhood relations.
3. The set of all possible morpheme-forms. To each morpheme-form, it must be given the class of the morpheme and the applicable phonetic and assimilation rules. The morpheme-class determines the branch in the automaton where this morpheme-form can be used for transition, while the phonetic rules define how "this transition can be performed.

Having both the states – from the graph description – and the transition rules the program produce minimal solution.

The first program is relatively slow but it must be run only in the case when some alteration in the rules i.e. in the grammar is inevitable.

In the experimental solution the lexicon is not realized through a finite automaton. Since the present restricted version contains only about 4000 word-stems the search using hash-code proved to be satisfactory.

Literature:

[1] Kimmo Koskenniemi:
Two-level Model for Morphological Analysis
Proceedings of the International Joint Conference
on Artificial Intelligence, 1983 Karlsruhe

[2] H.Jappinen; A.Lehtola, E.Nelimarkka, J.Niemistö
and M.Ylinammi
Morphological Analysis of Finnish Word Forms
Publications of the Kielikone Project 1987

[3] L.Karttunen, K.Koskenniemi and R.Kaplan
A Compiler for Two-level Phonological Rules
Report of the Center for the Study of Language and
Information, Stanford University 1987

[4] I.Bach and E.Farkas
Data Base Access through Hungarian
Advanced Dialog System and Natural Language Processing
Conference of Socialist Academies, Suwalki 1987

[5] E.Farkas and M.Naszódi
Hungarian Morphology and Syntax
Working Paper of the Computer and Automation Institute
of the Hungarian Academy of Sciences 1988