Associative Model of Morphological Analysis: An Empirical Inquiry[1]

Harri Jäppinen[2] and Matti Ylilammi[2]

Helsinki University of Technology
Helsinki, Finland

This paper presents a computational model for the analysis of word forms of a highly inflectional, agglutinative language. We call the model "associative" as it directly links phonemic stimulus with its morphemic interpretation(s) under the guidance of a coherence constraint. The model has been fully implemented for Finnish. We discuss separately the abstract model and various algorithms to implement the model. We also demonstrate the implementation. The best features of the method are its efficiency and its capability of supporting open lexicons.

1 Introduction

In the search for computational models of language, syntactic analysis of sentences has so far received much greater attention than morphological analysis of word forms. For example, Winograd (1983) in his thorough book on syntax and parsing methods has allocated six or so pages to the morphological analysis. That makes about one percent of the whole text. This does not, of course, mean that the author rates morphology unimportant, but it does, we believe, reflect the general interest of the research community.

Such heavy emphasis on syntax on the one hand and almost total neglect of morphology on the other follows from the idiosyncracies of English. And due to the dominant role English has in the computational linguistic community, this somewhat unbalanced view permeates the computational linguistic literature.

The neglect of English morphology has obvious reasons. The basic rules of English word inflection are quite simple. There are some fusion phenomena that produce portmanteau morphs resistant to inflectional analysis, but their number is small. For syntactic analysis of sentences, one needs a lexicon anyway. Why not then take the easy way out and let the lexicon bear the burden of morphology and carry at least the hard word forms if not all word forms as separate lexemes?

For agglutinative inflectional languages, however, the economy of computation may shake hands with theoretical ambitions. In Finnish, to take an example close to our heart, a nominal may appear in a running text in thousands of different forms, and verbs have an even wider spectrum of forms (Karlsson 1983). The probability distribution of the word forms of a given lexeme is uneven, to be sure, but even the most conservative estimates render the brute-force method inappropriate in other than naive attempts for extremely limited purposes. Fortunately Finnish is an agglutinative language; portmanteau morphs are almost nonexistent. In order to analyze Finnish sentences computationally one *must* analyze word forms as well.

This paper describes a morphological model for Finnish. The model has been fully implemented, and our tests rate it efficient. A synopsis of the model appears in Jäppinen et al. (1983a). This paper describes the model in more detail.

The functional requirements set for the model were:

1. clean separation between linguistic knowledge and algorithms (we wanted to increment and modify the model without structural changes in the algorithm),
2. general and efficient analysis of inflectional, possessive, and cliticized morphs (all inflectional word forms should be analyzed on the basis of general linguistic knowledge; synthesis of word forms and the analysis of derivational forms were left out), and
3. support of an open lexicon (the model should recognize the occurrence of a new lexeme and support lexical update).

The model we came up with is associative. It is not generative in the sense that it does not utilize, backwards or forwards, formal rules designed to generate valid and only valid Finnish word forms. The associative rules connect strings of phonemes *directly* with morphemes without passing through intermediate syntagmatic cate-

0362-613X/86/040257-272$03.00

gories. The associative model is composed of two interrelated parts:

1. a collection of independent stimulus-result rules, each associating a (partial) phonemic stimulus with one or more morphemes, and
2. a coherence constraint that determines which of the fragmentary results proposed by the rules make up a coherent whole.

This model is the result of an empirical inquiry guided by pragmatic constraints. Therefore, the discussion places emphasis on computational issues rather than on theoretical linguistic argumentation. We first outline the problem, and then describe separately a model for its solution and various algorithms to realize the model. We also discuss and demonstrate an implementation, and contrast it with other known models.

2 The Problem Decomposed

Henceforth we call **lexical entry** or **lexeme** the basic word form that serves as a carrier of morpho-syntactic information about a word in a lexicon. **Word form** or simply **form** denotes an inflected lexeme. The grammatical representation of a word form is called **grammatical word** or **word.** The grammatical word of a lexeme in our model is singular nominative case for nominals and first infinitive for verbs. This is the practice of printed Finnish mono- and bilingual dictionaries and lexicons.

There is a one-to-many (one-to-thousands) mapping between lexemes and their valid forms. An effective inverse mapping is the analysis problem. Two distinct lexemes may, and frequently do, produce identical forms, homographs. Consequently, the analysis of a form is occasionally ambiguous, and it is paramount for the analysis to find all lexemes a given form represents.

Any native Finn is able to produce without effort the proper phonemic (or graphemic) word form of a given lexeme to fit a given context. When various dialects are reduced into standard written orthographic Finnish, the lexeme *takki* ('coat'), to take a random example, may appear in text in any of the following forms:

(1)

{takki,takit,takin,takkien,takkiesi,takkina,takkeihin, takeissasi,takeittako,takeissannehan,takkeinensakohan,...}

Similarly, we can take any verb lexeme, say *jakaa* ('deliver'), and list all its distinct forms:

(2)

{jakaa,jaan,jaat,jakavat,jaoin,jakanevat,jakakoot, jakaisimme,jaetaankohan,jaettaisiinkohan,...}

A native speaker, when browsing lists such as (1) or (2), would accept the elements as valid forms and, furthermore, easily assign each its basic form, its lexeme. For the ninth element in (1), for instance, an average adult Finn would spontaneously recognize it as a properly inflected form of *takki.* But if asked to interpret the form, he after some hesitation would probably not be able to extract and identify affixes attached.

The basic meaning-bearing unit in a word is a **morpheme.** The linear arrangements of morpheme classes in Finnish word forms are (Ikola 1977):

(3) Nominals:

STEM < [COMPAR] < NUMBER < CASE < [POSSESSIVE] < [CLITICS1..2]

Verbs:

STEM < VOICE < [TENSE/MOOD] < [PERSON] < [CLITICS1..2]

Verbal Nominals:

STEM < [VOICE] < [VERBNOM] < [COMPAR] < NUMBER < CASE < [POSSESSIVE] < [CLITICS1..2]

Brackets are used to distinguish optional morpheme classes. VERBNOM (verb nominal) comprises two participle and four infinitive morphemes. Some of the classes are cross-categorial. COMPAR (comparison) may appear only with adjectival stems or with participial forms. Past tense can be joined only with the indicative mood; tense and mood are therefore expressed in a common class. Two clitics may be attached in a row, on rare occasions even three. In speech a < b means b follows a in time; in writing b is to the right of a.

Once the morpheme classes and their ordering are made explicit, it is then easy for any given word form to isolate the representatives of the morpheme classes. A noun, say *takkeinensakohan* from (1), when matched against (3) is analyzed as *takke+i+ne+nsa+ko+han* corresponding to a stem, plural, comitative case, third person possessive, and two clitic morphemes. Similarly, a randomly picked complex verb form such as *jaettaisiinkohan* from (2) reads as *jae+tta+isi+:n+ko+han,* representing a stem, passive, conditional (and hence present tense) passive suffix (serving as a kind of "fourth" person), and two clitic morphemes.

Each class has a small number of elements, and each morpheme has at most a few allomorphs, phonemic realizations. From the strategic point of view the morpheme classes fall into two categories. Other morphemes than the stems have a closed set of allomorphs. They can be recognized with a finite set of rules. Stems constitute a large and unbounded set of morphemes as new lexemes may develop and enter in the vocabulary. Hence the allomorphic variants of stems cannot be directly used in a closed set of rules. To bound a set of rules, the rules for stems must recognize invariant *parts* of phoneme strings, not entire stem alternants.

The original problem was thus decomposed into two parts: The Morphotactic Problem segments word forms and solves the allomorph relation for morphemes other than stems; The Stem Alternation Problem solves the residual allomorph relation for stems.

3 General View of the Associative Model

An associative model for the analysis of word forms of Finnish consists of a triplet $<\{MR_i\}, <^*, \{SR_i\}>$, where $\{MR_i\}$ is a set of associative morphotactic rules, $\{SR_i\}$ is a

set of associative stem rules, and <* is a precedence relation in the set of the morphotactic rules. The rules associate phonemic stimulus with morphemic data. They obey the general form:

(4) [ml context] <pl context> key <pr context> [mr context]--> [m result]

A rule is applicable and fires whenever a phonemic string identical to **key** appears somewhere in an input form and the contextual conditions of the rule are satisfied. **pl** and **pr** stand for phonemic contexts and **ml** and **mr** designate morphemic contexts. A contextual phonemic string calls for substring identity, morphemic contexts require set inclusion. Infixes **l** and **r** denote left and right sensitivity, respectively. When a rule fires, it contributes one or more morphemes (**m result**) as a possible partial interpretation of the word form. There is not necessarily a one-to-one correspondence between keys and morphs. Keys may represent entire morphs or morphs that have been truncated by fusion processes.

The model is implemented in a sequential machine. Therefore the rule (4) has two slightly differing applications:

(5a) Morphotactic rules:
<pl context>allomorph[mr context] → [morphemes]

(5b) Stem rules:
<pl context>a ending[mr context] → [conc(ROOT,b ending)]

Morphotactic rules recognize and interpret all other morphs except stem alternants. As our algorithm is tuned to right-to-left sequential processing, these rules are invoked first. Only left phonemic and right morphemic contexts make sense in these rules. Stem rules recognize the allomorph relation of a potentially unlimited number of stems. They use alternant stem endings (**a ending**) as keys. A stem rule produces a hypothetical basic stem (lexeme) in which the recognized alternant ending is replaced (concatenated with the root) by the basic ending (**b ending**) shown. Only left phonemic and right morphemic contexts are meaningful in stem rules in the chosen strategy. We discuss the morphotactic part and the stem alternation part of the model in separate sections below.

4 Morphotactic Model

This section lists the morphemes of Finnish and presents a few outstanding problems in their allomorph relation. The discussion uses similar semiformal generative rules as Matthews (1972). Then an associative solution to the Morphotactic Problem is displayed.

4.1 THE MORPHEMES

Expression (3) arranges the morpheme classes of Finnish in three precedence orders, one for the nominal forms, one for the verbal forms, and one for the verbal nominal forms. These morpheme classes have grammatical functions shown in (6). (Clitics have such complex functions in Finnish that we do not attempt to mark them by morphemes but use instead their phonetic realizations in the discussion.)

(6) COMPAR = {**com**(parative),**sup**(erlative)}
NUMBER = **s**(in)**g**(ular),**pl**(ural)}
CASE = {**nom**(inative),**gen**(itive),
part(itive),**ess**(ive),
in(essive),**ela**(tive),
ill(ative),**ad**(essive),
abl(ative),**all**(ative),
ins(tructive),**com**(itative),
ab(essive),**tran**(slative)]
POSSESSIVE = {**1p**(erson)**s**(ingular),**2ps**,**3ps**,
1pp(lural),**2pp**,**3pp**}
CLITICS = {...}
VERBNOM = {**Ipart**(iciple),**IIpart**,**Iinf**(initive),
...,**IVinf**}
VOICE = {**pass**(ive),**act**(ive)}
MOOD = {**ind**(icative),**imp**(erative),
cond(itional),**pot**(ential)]
TENSE = {**pres**(ent),**past**}
PERSON = {**1ps**,**2ps**,**3ps**,**1pp**,**2pp**,**3pp**,**p**(assive)
suff(ix)}

4.2 SOME PROBLEMS IN THE ALLOMORPH RELATIONS

Morphemes of Finnish are complex. Some morphemes have no correlates on the phonemic level, some morphemes have more than one allomorphs, and some phoneme strings have the simplest explanation, it seems, if empty morphs are postulated. Further complexities are caused by certain fusion processes. We outline these phenomena below for non-Finnish readers.

The **pl** morpheme has the suppletive allomorphs 'i', 'j', or 't', and in some rare cases no phonemic string marks plural. The **sg** morpheme has no allomorphs. These are not the only morphemes without phonemic correlates. In order to faithfully pair off morphs and morphemes in a one-to-one correspondence, it is convenient to postulate a zero alternant in place of a missing morph. For the lexeme *kala* ('fish'), for instance, we then get among other derivations between the morphemic (ML) and phonemic levels (PL) the few possibilities shown in Figure 1. (The root of a stem is henceforth written in capital letters.)

The standard treatment assigns four allomorphs for the partitive case: 'a', 'ä', 'ta', and 'tä'. This variety decreases if one posits two allomorphs, 'a' and 'ä', for **part** and allows the existence of an empty morph 't' to be conditioned by stress. The partitive forms of *kala* ('fish') and *pasuuna* ('trombone') in Figure 2 illustrate the interplay between **pl** and **part** morphemes in this stipulation.

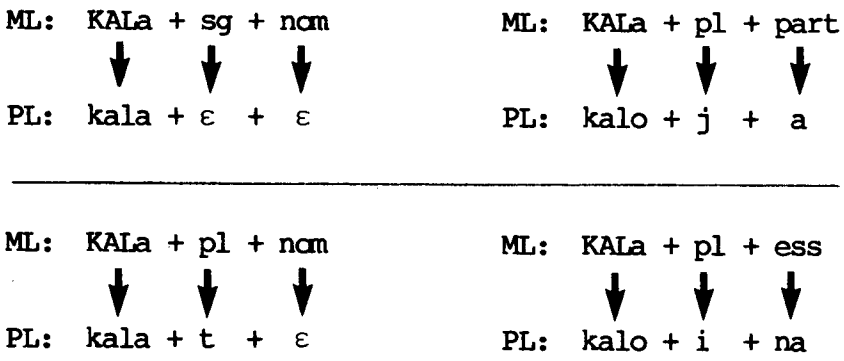

Figure 1. Examples of suppletive allomorphs and zero morphs.

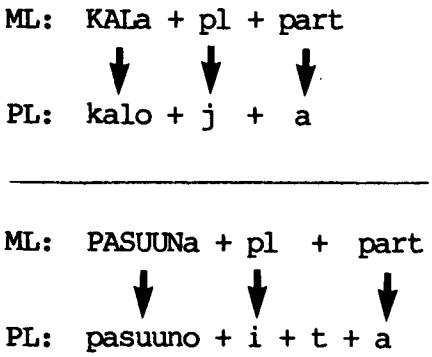

Figure 2. An example of empty morphs

Some alternants, when joined with neighboring affixes, exhibit regularities in behavior which can be captured conveniently by archiphonemes on the mediating morphophonemic level (MPL). The allomorphs of comparison are examples of such alternants, and so are some clitic segments. The use of archiphonemes captures nicely consonant gradation in the former and vowel harmony in the latter. The two part allomorphs discussed above can also be generated via a single archiphoneme 'A' on the morphophonemic level. It is realized as an 'a' or an 'ä' on the phonemic level as vowel harmony demands. In Figure 3 lexemes *suuri* ('big') and *jää* ('ice') exemplify how the use of archimorphemes reduces a set of generative rules. There are fusion processes that delete information. These phenomena are easily formulated in generative terms but are problematic for analysis. The leftmost consonant in the possessive morphs (**1ps**:'ni'; **2ps**:'si'; **3ps**:'nsa'; **1pp**:'mme'; **2pp**:'nne'; **3pp**:'nsa'), be it a nasal or a fricative, overlaps and dominates the preceding consonant. For the lexeme *kala* ('fish'), for instance, we get the derivations in Figure 4 in the singular and plural nominative and genitive cases when a possessive segment is present or absent, respectively.

Notice how the four forms are distinct when a possessive is absent *(kala, kalat, kalan, kalojen)* and become threefold ambiguous when the possessive segment is attached *(kalamme, kalamme, kalamme, kalojemme).* This is a general phenomenon. A nominal in Finnish always becomes grammatically ambiguous when a possessive suffix is attached to a singular nominative or genitive, or to a plural nominative form.

4.3 MORPHOTACTIC MODEL

An associative Morphotactic Model (MTModel) is a pair $<\{MR_i\},<^*>$, where $\{MR_i\}$ is a set of morphotactic rules (5a) and $<^*$ is a precedence relation in the set. $<^*$ is an irreflexive, antisymmetric, and nontransitive relation which imposes a coherence constraint on the rules. Each morphotactic rule associates a morphemic interpretation with a phonemic substring. The relation $<^*$ orders the rules in such a way that partial interpretations, when a word form is processed from right to left, contribute to valid total interpretations.

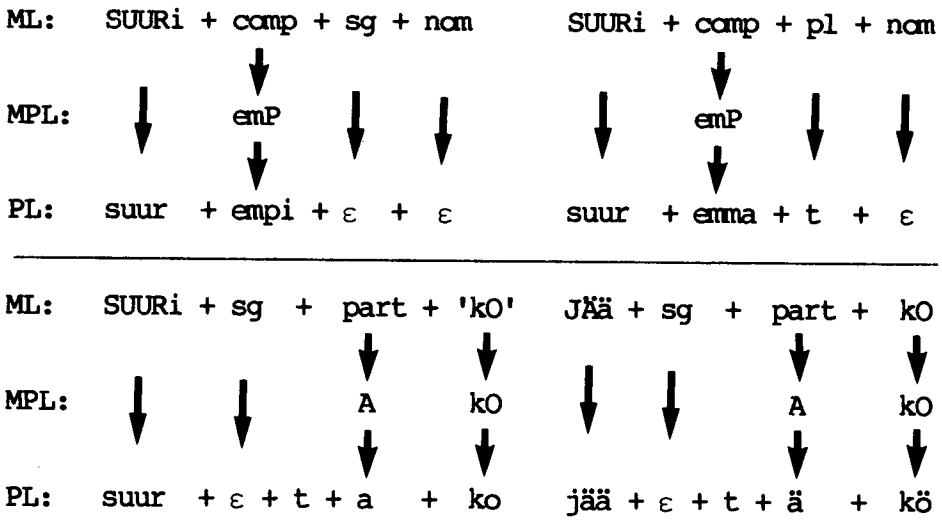

Figure 3. Examples of archiphonemes.

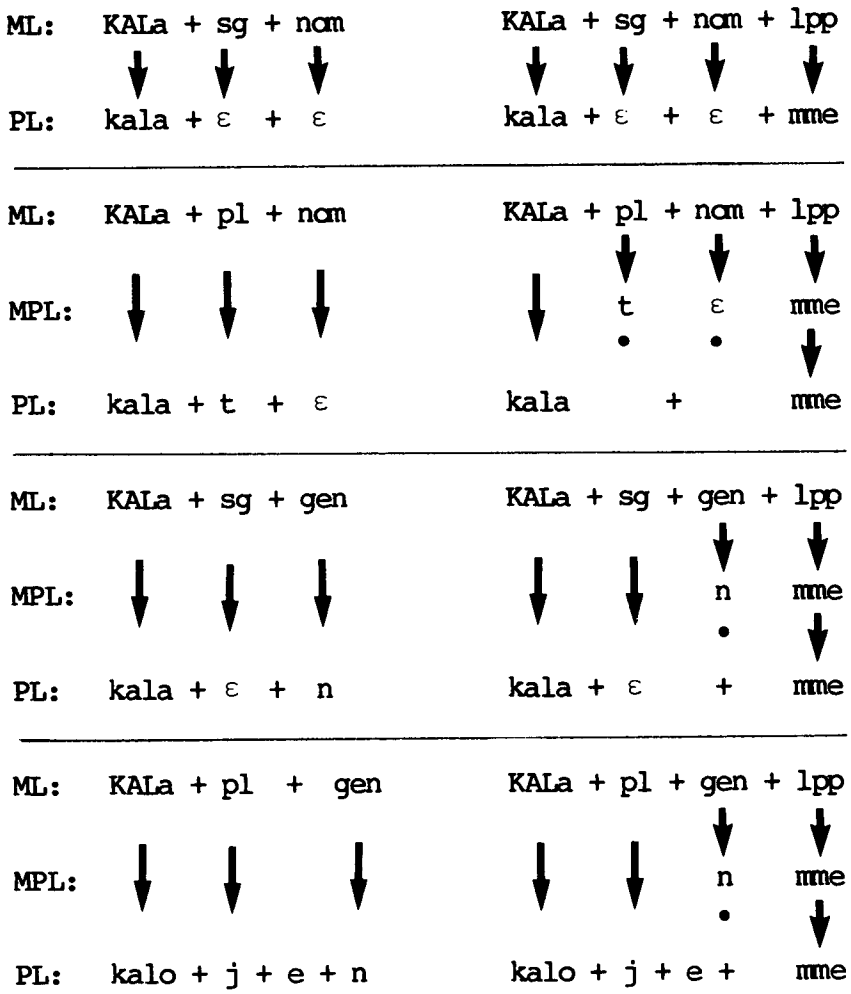

Figure 4. Examples of fusion processes.

$MR_i <^* MR_j$ iff MR_i can "immediately follow" MR_j. A rule can immediately follow another if the key of the former can be juxtaposed to the left of the latter on the phonemic level. The keys may not overlap, or be discontinuous, and their morphemic interpretations must obey the ordering (3).

For coherence, the model also needs boundary rules. Let ε denote a zero key for zero morphs, and α and β mark the zero keys for two special empty sets of morphemes. The "rightmost" morphotactic rule $MR_\alpha = \varepsilon \rightarrow [\ ]$ and the "leftmost" morphotactic rule $MR_\beta = \varepsilon \rightarrow [\ ]$ of the coherence constraint are defined below. The two boundary rules have obvious interpretations: MR_α signals the right end of a word form and MR_β indicates a stem boundary.

(7) ForAll (MR_i)[NOT($MR\alpha <^* MR_i$)]
ForAll (MR_i)[NOT($MR_i <^* MR\beta$)]

Brodda and Karlsson (1980) tried to find the most likely morphotactic segmentation for a given Finnish word form drawn from a running text. The algorithm does not use a lexicon, neither does it associate phonemic segments with their morphemic interpretations.

From that work we were able to extract and enumerate the valid phonemic keys for the morphotactic rules. The keys were then associated with their morphemic correlates and the rules were organized under the precedence relation $<^*$. The set in (8) lists a small subset of the rule set and a fragment of the coherence constraint. For the sake of brevity, only the key is shown in the left hand side of a rule. To compress the rules, archiphonemes, typed in upper case letters, are used in keys whenever possible. Figure 5 illustrates this part of the coherence constraint in graphic form.

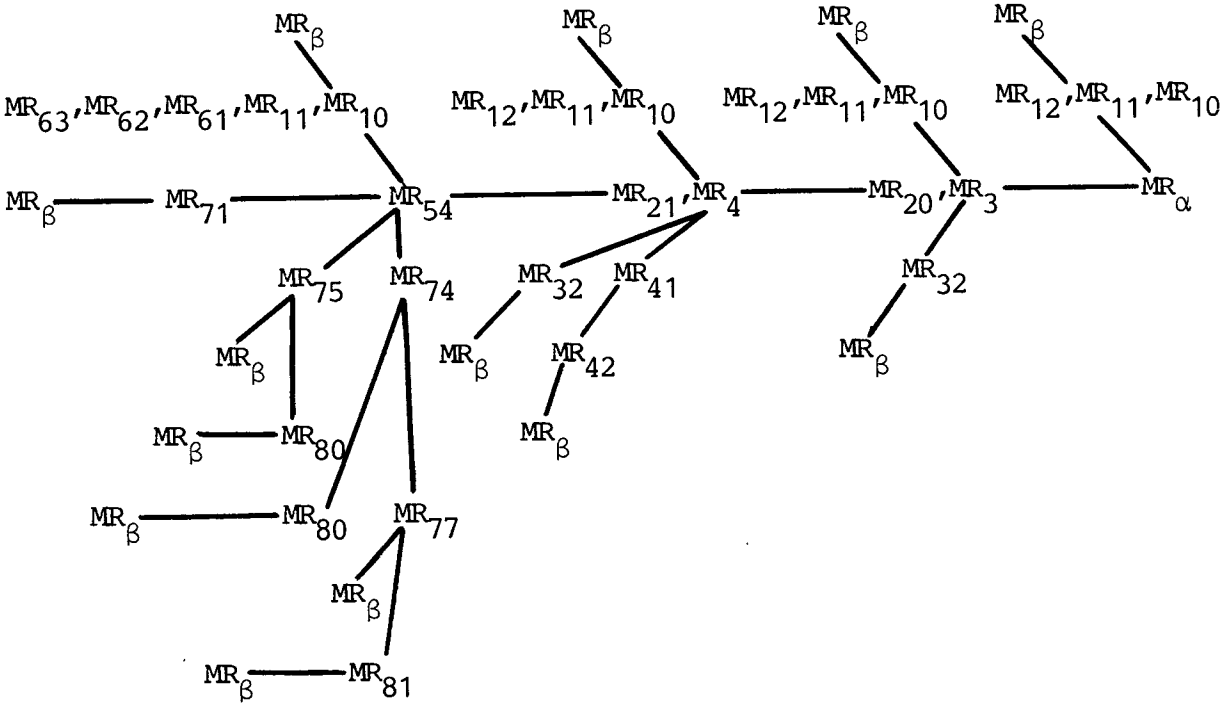

Figure 5. Partial coherence constraint of the Morphotactic Model

(8) MR_α = α → []
MR_β = β → []
MR_3 = ε → []
MR_4 = ε → []
MR_{10} = ε → [sg,nom]
MR_{11} = ε → [act,ind,pres,3ps]
MR_{12} = ε → [act,imp,pres,2ps]
MR_{20} = kO → ['kO']
MR_{21} = kin → ['kin']
MR_{32} = isi → [act,cond,pres,3ps]
MR_{41} = :n → [pres]
MR_{42} = lA → [pass,ind]
MR_{54} = mme → [1pp]
MR_{61} = ε → [pl,nom]
MR_{62} = ε → [sg,gen]
MR_{63} = ε → [act,ind,pres]
MR_{71} = mA → [act,IIIinf]
MR_{74} = ssA → [in]
MR_{75} = see → [sg,ill]
MR_{77} = i → [pl]
MR_{80} = nee → [act,IIpart]
MR_{81} = ne → [act,IIpart]

$R<^*$ = {$<MR_{10},MR_\alpha>$, $<MR_{11},MR_\alpha>$,
$<MR_{12},MR_\alpha>$, $<MR_{20},MR_\alpha>$,
$<MR_3,MR_\alpha>$, $<MR_\beta,MR_{10}>$,
$<MR_\beta,MR_{11}>$, $<MR_\beta,MR_{12}>$,
$<MR_{32},MR_{20}>$, $<MR_\beta,MR_{32}>$,
$<MR_4,MR_3>$, $<MR_{21},MR_3>$,
$<MR_4,MR_{20}>$, $<MR_{21},MR_{20}>$,
$<MR_{41},MR_4>$, $<MR_{41},MR_{21}>$,
$<MR_{42},MR_{41}>$, $<MR_\beta,MR_{42}>$,...}

The rule set and the coherence constraint represent the morphotactic part for morphological analysis. A phoneme string is a morphotactically valid form if there is a "path" between the "rightmost" rule, MR_α, and the "leftmost" rule, MR_β, in the coherence constraint. The interpretation of the form is the union of the morphemes associated with the rules along the path. For an ambiguous word form more than one path exists between the MR_α and the MR_β.

The fragmentary rule set and the constraint in (8) give, for instance, the following morphotactic interpretations for the ambiguous form *kalamme* shown in Figure 4:

(9)
kala+[sg,nom,1pp]
($MR_\beta <^* MR_{10} <^* MR_{54} <^* MR_4 <^* MR_3 <^* MR_\alpha$)
kala+[sg,gen,1pp]
($MR_\beta <^* MR_{62} <^* MR_{54} <^* MR_4 <^* MR_3 <^* MR_\alpha$)
kala+[pl,nom,1pp]
($MR_\beta <^* MR_{61} <^* MR_4 <^* MR_3 <^* MR_\alpha$)
kala+[act,ind,pr,1pp]
($MR_\beta <^* MR_{63} <^* MR_{54} <^* MR4 <^* MR3 <^* MR_\alpha$)

The first three are valid interpretations. The verbal interpretation, although morphotactically valid, does not result in an existing verb stem. That interpretation will be rejected by the Stem Alternation Model discussed below. That the verbal interpretation is indeed morphotactically plausible can be seen, for instance, with the form *palamme,* analyzed as *pala*+[act,ind,pr,1pp], which is a valid interpretation for the verb lexeme *palaa* ('burn').

MTModel for Finnish consists of 178 rules. It is not yet an algorithm. It does not state how analysis is being done, that is, how control is to proceed in an analysis. These are matters of an algorithm discussed in a later section. The previous discussion has committed the model from right to left processing, but reverse processing or some more advanced control schemes might be used as well.

5 Stem Alternation Model

For any given word form, MTModel resolves sets of morphemes that make up coherent wholes. MTModel also indicates stem alternant boundaries (MR_β) but leaves the alternants intact. The Stem Alternation Model (SAModel) discussed in this section finds for each postulated stem alternant its basic form(s), or rejects it. We first discuss the stem alternants in Finnish as they are customarily described in the Word and Paradigm Model. We then describe associative rules for the analysis of stem alternants.

5.1 The Stem Alternation Problem

The Standard Dictionary of Modern Finnish (*Nykysuomen sanakirja,* 1966) describes the behavior of Finnish word forms in terms of the Word and Paradigm Model. It classifies nominals into 82 and verbs into 45 equivalence classes – paradigms – based on variations in their stem alternants. For each paradigm the classification gives a theme word, to represent the class, and its stem alternants. Thus, for instance, the nominal paradigms 10 and 41, and the verb paradigm 25 are listed as in Figure 6. The theme words are *KALa* ('fish'), *TOSi* ("true"), and *TULla* ("come"), respectively. (We have slightly edited the entries for our purpose.) Upper case letters in Figure 6 indicate the roots and the stem-forming affixes; lower case letters are reserved strictly for the alternant stem endings.

The information conveyed by the paradigm tables can be compressed into two matrices below which show just the distributions of the stem endings. The rows of the matrices represent the paradigms and the columns morphemic contexts (not given here). Whenever allomorphs generate different stem endings, the endings are enclosed in parentheses. The vertical bars separate singular nominal stems from plural stems and active verbal stems from passive stems. The first column in both matrices represents the ending of the basic form, the lexeme. ε_v denotes a null ending in a vowel stem, ε a null ending in general. Upper case letters mark here archiphonemes.

Nominals:

	sg				pl		
nom	gen	part	ess	ill	gen	part	ill
...							
KALa 10	-aN	-aA	-aNA	-aAN	-oJEN -aIN	-oJA	-oIHIN
...							
TOsi 41	-deN	-tTA	-teNA	-teEN	-sIEN	-sIA	-sIIN
...							

Verbs:

	act							pass		
Iinf	ind		pot	cond	imp		IIpart	pres	past	IIpart
	pres 1ps	past 3ps	pres 3ps	pres 3ps	pres 2ps	pres 3ps				
...										
TULla 25	-eN	-I	-LEE	-ISI	-e	-KOON	-LUT	-LAAN	-TIIN	-TU
...										

Figure 6. Examples of the Finnish word paradigms.

(10) Nominal stem endings:

```
01: εv, εv, εv, εv, εv  |    εv, εv, εv
02: εv, εv, εv, εv, εv  |    εv, εv, εv
03: εv, εv, εv, εv, εv  |    εv, εv, εv
04:  i,  i,  i,  i,  i  |     i,  e,  e
05:  i,  i,  i,  i,  i  | (i,e),  e,  e
    ...
10:  A,  A,  A,  A,  A  | (O,A),  O,  O
    ...
41: si,de, t, te, te    |     s,  s,  s
    ...
```

Verbal stem endings:

```
01: εv, εv, εv, εv, εv, εv, εv, εv  | εv, εv, εv
02: A, A, ε, A, A, A, A, A          |  e,  e,  e,
03: tA,dA, εv, tA, tA,dA, tA, tA    | de, de, de
...
25: la, e, ε, ε, ε, e, ε, ε         |  ε,  ε,  ε
...
```

Each interpretation postulated by MTModel unambiguously chooses a column. The problem of stem alternation follows from the fact that the row of the stem is not known. Should SAModel know, say, that the postulated singular genitive stem *käde* in the form *käden* represents paradigm 41, simple substring replacement operation would produce the correct lexeme *KÄsi* rightaway (the singular genitive case occupies the second column in the nominal matrix above).

5.2 STEM ALTERNATION MODEL

Our associative SAModel consists of a set of stem rules $\{SR_i\}$, each of the form (5b) and retyped below:

(11) <pl context>a ending[mr context]
→ [conc(ROOT,b ending)]

'a ending' is an alternant stem ending. When a rule fires, its alternant ending is replaced with the basic stem ending ('b ending'). The operator 'conc' concatenates the new ending with the root, producing a hypothetical basic word form. The consonant gradation process in roots is not analyzed in SAModel. Weak and strong stems are dealt with as separate lexemes.

The paradigm tables (10) yield data for morphemic contexts ('mr context') and alternant and basic endings. Alternant endings are necessary but not sufficient phonemic data for rules. Stem rules without phonemic contexts are too productive.

Luckily, due to phonotactic reasons the orthographic distribution of roots (unvarying parts of stems) is uneven in various paradigms. A manageable number of short phoneme strings suffice to represent all roots of whole paradigms. *The Reverse Dictionary of Finnish* (Tuomi 1980) lists practically speaking all Finnish basic word forms (in reverse order), including some archaic ones and some of foreign origin. Each lexeme is tagged with its paradigm number and syntactic category. That dictionary

was a valuable source for the contextual phoneme strings for the stem rules.

For stem rules a well-formed phonemic context (WFPC) and its truth value is defined recursively as follows. Any lower case letter in the Finnish alphabet is a WFPC and the context is true if the last letter of a root is identical to that letter. If $\&_1$, $\&_2$, ..., $\&_n$ are WFPCs, then the following constructions are also WFPCs:

(12) (i) $\&_n \ldots \&_2 \&_1$
(ii) $<\&_1,\&_2,\ldots,\&_n>$

(i) is true if &1 and &2 and ... and &n are true, in that order. Testing continues from the point in a stem where the previous test left off. (ii) is true if &1 or &2 or ... or &n is true. The testing of $\&_i$'s halts if a recognition occurs. Each $\&_i$ starts its test afresh.

To enhance compact notation we stipulate that a single capital letter may represent a WFPC. Archiphonemes are conveniently expressed A for <a,ä>, O for <o,ö>, and U for <u,y>; the set of consonants and vowels appear compactly as K for <d,f,g,h,j,k,l,m,n, p,r,s,t,v> and V for <a,e,i,o,u,y,ä,ö>. But a WFPC of any complexity can be denoted by a single upper case letter.

The phonemic contexts vary in complexity in the rules in SAModel. Most of them have a fairly simple structure. Two paradigms are, however, without any phonemic contextual regularity. One is the nominal paradigm 08. The stem of the theme word *LOVi* ('notch') ends with an *i* in the basic form and with an *e* in singular genitive case *love+n* ('of a notch'). This paradigm represents an old form and the set of its lexemes is closed. All new nominals that end with an *i* in the basic form retain the *i* in the genitive case and in other singular cases. For example, the theme word for the paradigm 04 is *RISTi* ('cross') and its genitive form is *risti+n* ('of a cross'). The criterion for choosing between paradigms 04 and 08 is not phonotactic; it is diachronic. Therefore, no phonemic context short of a minilexicon would help us to resolve, say, that *suurin* (a valid *superlative* form for *suuri* ('big')) is not **SUURi+[sg,gen]**, as *muurin* (for *muuri* ('wall')) is **MUURi+[sg,gen]**. We solved the problem by using two kinds of *i*'s as the last letter of a lexeme.

SAModel consists of 280 rules. Added context sensitivity increased greatly the quality of stem rules. A stem alternant produces only a fraction over one basic forms on average. The stem rules augment the coherence constraint of MTModel with an obvious component: a morphotactically coherent word form passes the coherence test of SAModel only if at least one of the basic forms generated by the stem rules is a valid lexeme.

To illustrate the interplay of MTModel and SAModel, *käsissämmekö* will be analyzed **KÄsi+[pl,in,1pp,'kO']** in the way shown in Figure 7. The figure exhibits schematically only the stem rule responsible for the correct lexeme. The morphotactic rules in Figure 7 are from (8). The form gets other morphotactically coherent segmentations as well, but they are rejected by the stem rules and the lexicon.

6 Algorithm

One can think of various alternative algorithms to realize the model. A multiprocessor environment might make the blackboard strategy used in HearsayII (Erman et al. 1980) an attractive alternative. Our choice was a monoprocessor environment and right-to-left strategy: first all morphotactically coherent stem alternants are postulated, then stem rules and dictionary check are invoked in that order for each alternant. The algorithmic issues are briefly talked about in this section.

6.1 Morphotactics

First we decided to implement MTModel as a structured collection of interconnected "islands". Each island comprises the possible and mutually exclusive morphotactic rules at any given point of processing; the rules represent valid paths through the island. The coherence constraint provides "bridges" between the islands, a bridge indicating a valid continuation after a walk through an island. Computationally the islands were finite state transition automata.

There were 32 distinct automata: 3 for clitics, 1 for person, 5 for tense, 3 for case, 2 for number, 3 for passive, 5 for participle, 5 for comparation, and 5 for infinitive rules. To assist automatic compilation from the rules to the automata, the morphotactic rules were slightly modified to read as:

SR_{124} = <A>s[...,in(ssA),pl(i)] τ-> [conc(ROOT,si)]

SR_{124}	<*	MR_{β}	<*	MR_{77}	<*	MR_{74}	<*	MR_{54}	<*	MR_4	<*	MR_{20}	<*	MR_{α}
käs	+	β	+	i	+	ssä	+	mme	+	ε	+	kö	+	∝
KÄsi	+	[		pl,		in,		1pp,				'kO']		

Figure 7. The analysis of *käsissämmekö*.

(13) automaton: (pl2)(pl1)allomorph
→ ([morphemes],[next automata])

e.g.,

Rpp: (ALL-[i:,O:])(V+$\acute{V}$)n
→ ([verb,act,ind,1ps],[Tense1])

Left contexts of phonemes were confined into expressions of two optional sets: for phonemes next to the left and second to the left. The term **automaton** names the island the rule belongs to; **next automata** identifies valid continuations after this path. **mr context** in (5a) is represented implicitly in (13) as the path leading to this rule.

The example rule belongs to the person automaton. The rule recognizes the **1ps** suffix *n* for an active indicative verb if an *n* is found such that it has an ordinary or stressed vowel first to the left and any phoneme except a long *n,o,* or *ö* second to the left. Control proceeds to the automaton Tense1 to identify modal and temporal morphemes. In general more than one continuation automaton is possible.

The island approach worked quite well. However, it was redundant because identical transition paths existed for different automata. To save memory, we implemented another version of MTModel, this one as an orthographic tree of the keys (and rules). The islands were layered, so to speak, on top of each other. A pass through the constraint in (8), or a walk through an island, corresponds now to a traversal through the tree. Coherence is satisfied if, for each transition along a path from the $\mathbf{MR}_\alpha$ to a $\mathbf{MR}_\beta$, a successful walk through the orthographic tree can take place. Automatic compilation again transforms the rules of the form (13) into the orthographic tree.

The orthographic tree occupied only about one-tenth of the memory needed for the island approach. Using a novel key-and-lock construct we were able also to speed up the analysis. With each node in the tree a "lock" was associated as a union of the automata names ('next automata' in (13)) in its subtree. Each traversal through the tree provides a "key" as a set of possible continuations. During the next traversal the key is checked in the lock of each node along the path and only a match (nonempty intersection) permits continuation. This method aborts fruitless attempts through the tree early on. Morphotactic analysis in the orthographic tree with this lock-and-key approach takes about 40% of the time the original island approach took.

6.2 STEMS AND LEXICON

The control of the stem rules was first realized as an orthographic tree of "prolonged stem endings". A prolonged ending concatenates an alternant ending with its contextual strings. The 280 stem rules yield 420 distinct extended stem endings. Exit points were marked in the tree and morpheme contexts were attached to these nodes as exit conditions. Basic stem endings were also associated with the exit points. A stem alternant traversed the tree and produced basic forms along the path whenever the exit condition was satisfied in exit nodes.

The stem alternant tree wasted, however, memory to an extent that we implemented also a hash-coded version of the extended endings. This version saves memory considerably without a noticeable increase in the analysis time.

A word form is valid only if it has at least one coherent morphotactic interpretation and if at least one of the lexemes produced by the stem rules appears in the lexicon. Dictionary organization and its search procedure constitute therefore an integral part of the algorithm. In our implementation the dictionary is composed of three distinct parts. The main dictionary is preceded by a hash-coded lexicon that contains the function words.

The main dictionary consists of an open set of adjectives, nouns, and verbs (and also numerals). It is implemented as a backward-sorted orthographic tree. The unconventional ordering allows for iterative analysis of compound word forms. Lexemes whose roots participate in consonant-gradation process have two separate lexical entries: weak and strong.

7 IMPLEMENTATION AND DEMONSTRATION

The ultimate test of the model and the algorithm lies in its performance. We felt that the primary justification of our model is its capability of meeting certain functional requirements:

- clean separation between linguistic knowledge and algorithms,
- general analysis method,
- efficient analysis, and
- support of an open lexicon.

The model separates linguistic data from algorithms, as the discussion above has indicated. Due to the rule structure the model has proved to be easy to augment, and now it covers the entire Finnish inflectional morphology. This satisfies the second requirement. In this section we discuss efficiency, open lexicon, and other issues of implementation and demonstrate the implementation.

For the reasons of efficiency and portability we implemented the algorithm in PASCAL. Separate compiler procedures transform the associative rules into their internal representations, as discussed in the previous section. The orthographic morphotactic tree takes about 4kW and the hash coded extended stem endings 5kW of DEC2060 memory. The procedures that utilize these data structures take about 20kW. The two hash-coded front lexicons reside also in the main memory. They cover already the majority of function words in Finnish. Their data structures and code together occupy 21kW of DEC20 memory. There is also a version on VAX11 and one on IBM PC/XT. In the latter, MORFO, as we call the system, takes up 305kB of memory. That figure includes MS-DOS.

The main lexicon resides on disc. As of this writing the main lexicon contains over 30,000 of the most frequently used Finnish verbal and nominal lexemes taken from Saukkonen et al. (1979) and from running ordinary texts. Figure 8 shows a few sample analyses with the trace mode of the system switched on. *Alusta* is a highly ambiguous word form in Finnish. MTModel (JAOTIN) finds six coherent morphotactic interpretations for it. SAModel (MUOKKAIN) extracts two different basic word forms for the first interpretation, one for the second and the third, three for the fourth, and none for the fifth and the sixth. ('VA', 'HA', and 'NE' stand for strong (or neutral), weak (or neutral) and neutral grade, respectively. The numbers within angle brackets are identifiers of the stem rules.) The weak stem *alu* is accompanied by its strong partner *alku*. Of the seven postulated lexemes, five actually occur, found in the main lexicon (SANAKIRJAT). The presence of the affix *n* (**gen**, or **1pp**) greatly reduces ambiguity as Figure 8 further shows. The morph *n* is either **gen** or **1pp** person. This information disqualifies the cases **el** and **part**.

We have tested the system rather extensively. In addition to randomly picked word forms we typed in, a typist entered news reports and columns picked from various Finnish newspapers. The test texts also included, of course, function words and compound word forms. Over 300,000 forms have been thus introduced. The analysis of a word form takes about 20ms of DEC2060, 35ms of VAX11/780, and 50ms of VAX11/750 CPU-time on the average. Throughput on an IBM PC/XT is about 95 words forms per minute. These figures satisfy our functional requirement for an efficient analysis method.

As an example trace of the system at work, the first word forms of Genesis in the Finnish Bible are analyzed by MORFO in the way shown in Figure 9. (Our lexicons carry English equivalents for each lexeme.)

```
------------------------------------------------------------
Sane: ALUSTA

JAOTIN: 11.1 ms.
  1: ALUSTA=                          Noun SG Nom
  2: ALUSTA=                          Verb Act Imper Pr S 2P
  3: ALU=       STA=                  Noun SG El
  4: ALUS=       TA=                  Noun SG Part
  5: ALUS= TA=                        Verb Pass Ind Pr Neg
  6: ALUS=  TA=                       Verb Act Iinf SG Nom

MUOKKAIN: 16.2 ms.
  1: ALUST A (mon.part. -OJA)         VA   1, <1585>
  2: ALUST A                          VA   1, <1580>
  3: ALUST AA                         HA   2, < 710>
  4: ALU                              HA   3, <1810>
     ALKU
  5: ALU NEN                          NE   4, <2860>
  6: ALUS I (yks.gen. -EN)            VA   4, <2480>
  7: ALU S                            HA   4, <1750>

SANAKIRJAT: 75.4 ms.
  1: ALUSTA                           BASE  Noun SG Nom
  2: ALUSTAA                          INITIALIZE  Verb Act Imper Pr S 2P
  3: ALKU                             BEGINNING  Noun SG El
  4: ALUNEN                           BEDDING  Noun SG Part
  5: ALUS                             SHIP  Noun SG Part


> ALUSTAN
------------------------------------------------------------
Sane: ALUSTAN
ALUSTAA                         INITIALIZE  Verb Act Ind Pr S 1P
ALUSTA                          BASE  Noun SG Gen

> ?
------------------------------------------------------------
Sane: ALUSTAN

JAOTIN: 6.8 ms.
  1: ALUSTAN=                         Noun SG Nom
  2: ALUSTA=       N=                 Verb Act Ind Pr S 1P
  3: ALUSTA=      N=                  Noun SG Gen

MUOKKAIN: 13.5 ms.
  1: ALUST AA                         HA   2, < 710>
  2: ALUST A (mon.part. -OJA)         HA   3, <1855>
  3: ALUST A                          HA   3, < 185>

SANAKIRJAT: 46.8 ms.
  1: ALUSTAA                          INITIALIZE  Verb Act Ind Pr S 1P
  2: ALUSTA                           BASE  Noun SG Gen
```

Figure 8. Analysis of *alusta* and *alustan*.

```
> Alussa loi Jumala taivaan ja maan.
------------------------------------------------------------
SANE:  Alussa
ALKU                          BEGINNING   Noun SG In

------------------------------------------------------------
SANE:  loi
LUODA                         CREATE   Verb Act Ind Imp S 3P

------------------------------------------------------------
SANE:  Jumala
JUMALA                        GOD   Noun SG Nom

------------------------------------------------------------
SANE:  taivaan
TAIVAS                        HEAVEN   Noun SG Gen

------------------------------------------------------------
SANE:  ja
JA                            AND   Particle Conj

------------------------------------------------------------
SANE:  maan
MAA                           EARTH/COUNTRY   Noun SG Gen

> Ja maa oli autio ja tyhjä ja pimeys oli syvyyden päällä.
------------------------------------------------------------
SANE:  Ja
JA                            AND   Particle Conj

------------------------------------------------------------
SANE:  maa
MAA                           EARTH/COUNTRY   Noun SG Nom

------------------------------------------------------------
SANE:  oli
OLLA                          BE   Verb Act Ind Imp S 3P

------------------------------------------------------------
SANE:  autio
AUTIO                         DESERT   Adjective SG Nom

------------------------------------------------------------
SANE:  ja
JA                            AND   Particle Conj

------------------------------------------------------------
SANE:  tyhjä
TYHJÄ                         EMPTY   Adjective SG Nom

------------------------------------------------------------
SANE:  ja
JA                            AND   Particle Conj

------------------------------------------------------------
SANE:  pimeys
PIMEYS                        DARKNESS   Noun SG Nom

------------------------------------------------------------
SANE:  oli
OLLA                          BE   Verb Act Ind Imp S 3P

------------------------------------------------------------
SANE:  syvyyden
SYVYYS                        DEPTH    Noun SG Gen

------------------------------------------------------------
SANE:  päällä
PÄÄLLÄ                        UPON   Particle Adverb
PÄÄLLÄ                        ON   Particle Prep
PÄÄ                           HEAD   Noun SG Ad
```

Figure 9. Analysis of the first words in the Finnish Bible.

A randomly picked verbal lexeme, say *katua* ('repent'), to continue in the Biblical domain, has some of its various forms analyzed in Figure 10. Notice, by the way, how the verbal forms *katua* and *kadun* are homonymic with partitive and genitive forms of *katu* ('street'). (In Figure 10 'imp' stands for past, 'imper' for imperative; 's' for singular in verbs, 'sg' singular in nominals; 'p' for plural in verbs, 'pl' plural in nominals.)

The analysis of compound word forms is automatically invoked, if none of the basic forms postulated by the stem rules is found in the main lexicon. If this analysis also fails, control proceeds to the lexical acquisition mode. *Good Friday* is the compound *pitkäperjantai* in Finnish. (Its literal translation in English is *Long Friday.*) That compound belongs to a subclass of complex lexical items whose modifying part gets inflected in various cases in agreement with the head. Incidentally, this phenomenon holds also for numerals. Figure 11 shows example analyses of some forms of *pitkäperjantai.*

```
katua kadun katuvimmillaan katukaamme katumisessansa
-------------------------------------------------------------
Sane: KATUA
KATU                          STREET  Noun SG Part
KATUA                         REPENT  Verb Act Iinf SG Nom

-------------------------------------------------------------
Sane: KADUN
KATUA                         REPENT  Verb Act Ind Pr S 1P
KATU                          STREET  Noun SG Gen

-------------------------------------------------------------
Sane: KATUVIMMILLAAN
KATUA                         REPENT  Verb Act Ipartis Sup PL Ad 3P

-------------------------------------------------------------
Sane: KATUKAAMME
KATUA                         REPENT  Verb Act Imper Pr P 1P

-------------------------------------------------------------
Sane: KATUMISESSANSA
KATUA                         REPENT  Verb Act IVinf SG In 3P
```

Figure 10. Sample analyses of forms of *katua.*

```
> pitkäperjantai pitkänäperjantaina pitkäksiperjantaiksi
-------------------------------------------------------------
Sane: PITKÄ.. (pitkäperjantai)
PITKÄ                          LONG  Adjective SG Nom

-------------------------------------------------------------
Sane: ..PERJANTAI (pitkäperjantai)
PERJANTAI                      FRIDAY  Noun SG Nom

-------------------------------------------------------------
Sane: PITKÄNÄ.. (pitkänäperjantaina)
PITKÄ                          LONG  Adjective SG Ess

-------------------------------------------------------------
Sane: ..PERJANTAINA (pitkänäperjantaina)
PERJANTAI                      FRIDAY  Noun SG Ess

-------------------------------------------------------------
Sane: PITKÄKSI.. (pitkäksiperjantaiksi)
PITKÄ                          LONG  Adjective SG Transl

-------------------------------------------------------------
Sane: ..PERJANTAIKSI (pitkäksiperjantaiksi)
PERJANTAI                      FRIDAY  Noun SG Transl
```

Figure 11. Sample analyses of compound nouns.

```
> paholaisen
------------------------------------------------------------
Sane: PAHOLAISEN
  1: PAHOLAISEN                    HA Noun SG Nom
  2: PAHOLAIS TA                   VA Verb Act Ind Pr S 1P
  3: PAHOLAI NEN                   NE Noun SG Gen

Lisätään no. (0 = poistu,<esc> = peru,y = ysana,m = menu) : 3
Subst/Adj/Numer/Yosa : s
Semantiikka : devil

------------------------------------------------------------
Sane: paholaisen
PAHOLAINEN                     DEVIL  Noun SG Gen

Lisätään no. (0 = poistu,<esc> = peru,y = ysana,m = menu) :

> paholaistako
------------------------------------------------------------
Sane: PAHOLAISTAKO
PAHOLAINEN                     DEVIL  Noun SG Part ko

>
```

Figure 12. An example of lexical acquisition.

We may now state in more precise terms in what way our model is capable of supporting open lexicons. Maybe inadvertently, we had not inserted *paholainen* ('devil') in the lexicon. If we input one of its forms, say *paholaisen* ('devil's'), the failure of the analysis prompts the user to choose one of the postulated basic forms as shown in Figure 12. When the user has chosen the only valid option (3), has supplied syntactic category ('S' for substantive), and provided its English equivalent, the lexeme enters the lexicon, as the subsequent test proves in Figure 12. In this convenient manner, we have built up our lexicon to hold about 30,000 entries. We continuously augment the lexicon from running texts.

8 Discussion

This model for the analysis of word forms originated from certain functional requirements. The two most important ones were efficiency and an open lexicon. These two pragmatic considerations imposed quite naturally two specific strategic constraints which may be of theoretical interest.

Efficiency resulted in a computational strategy to use fully realized morphs as primitives in analysis rather than their abstract, morphophonemic representations. Computationally speaking, analysis amounts then to the ordered recognition of phoneme substrings (morphs) within a phoneme string (input word form). The result of an analysis is the union of the morphemes associated with the morphs. The model resulted in an efficient running system, as we have described.

The requirement of an open lexicon resulted, under the constraint of a sequential machine, in a right-to-left processing strategy. Only then are stem alternants of those lexemes not yet listed in the lexicon unambiguously recognized and analyzed. Compact expressions of phonemes suffice to represent the phonetic make-up of whole paradigms – unbounded sets of phonotactically possible stems. Right-to-left processing enables the stem rules using these expressions to handle all stem alternants, regardless of their occurrence in the lexicon. Native speakers have also the ability to analyze forms of non-existing lexemes.

Brodda and Karlsson (1980) reports on a program that aims at the most probable segmentation of Finnish word forms without using a lexicon. The program neither interprets segmentations nor finds all possibilities. Källgren (1983) describes a prototype system for the analysis and synthesis of Finnish nominals. Sagvall-Hein (1980) has studied the applicability of the chart parsing method for the morphological analysis of Finnish word forms. As this experiment did not result in a full-scale model, we do not discuss it further. Two other papers report more or less complete solutions for the analysis of inflected Finnish word forms.

Karttunen et al. (1981) reports on an implementation that "can recognize, in a fraction of a second, any inflected form of a word *it has stored in its lexicon* .... The present lexicon consists of about 100 roots.... It can analyze a short unambiguous word in less than 20 milliseconds [DEC-2060/Interlisp]. A long word or a compound that requires a lot of disambiguation can take ten times longer" (emphasis and comment added). The model utilizes phonetically realized morphs in analysis, as we do, but stores them in separate suffix lexicons. No explicit precedence relation links the morphs and, therefore, each suffix entry must carry a description of its environment. Processing is from left to right. A root lexicon first contributes a set of roots matching the input form. Each root entry lists constraints its valid forms must obey. As the residual input form is processed

phoneme by phoneme rightward, the interplay between the root constraints and retrieved suffix entries filters out the roots that match totally with the input form. A lot of run-time bookkeeping is involved. In this model as well as other models that use a left-to-right strategy, processing time is highly sensitive to the size of the root lexicon: the larger the lexicon the more there are matching roots.

Their root lexicon carries entries of such complexity that only a linguist can add them. Below appear lexical entry an ordinary bilingual dictionary entry and our lexical entry for "mato" ('worm'). (Our model uses two distinct entries for the gradated stem, but the system automatically generates the pair.)

(14) ma SUBST IntAlt t-d FinAlt o (Karttunen et al.)
mato SUBST (Ordinary dictionary)
mato SUBST VA (Our)
mado SUBST HA mato (Our)

Koskenniemi (1983) presents a "two-level model" and reports: "With a large lexicon it takes about 0.1 CPU seconds Burroughs B7800/PASCAL to analyze reasonably complicated word forms." This model also runs from left to right and first collects from the root lexicon the roots that match. Like Karttunen et al., he also stores morphs in separate suffix lexicons, and prunes from the initial root set those whose combinations with suffixes match with the input word form. The main difference is that Koskenniemi uses abstracted, morphophonemic representations of morphs, and the matching of a suffix is not performed by expert routines in the lexicons but by external rules, implemented as finite state automata. The morphophonemic representations of morphs and the processing rules capture linguistically appealing generalizations. The realization of rules at run time at least partly explains the slow speed reported. Lexical entries are abstracted roots and hence unnatural to ordinary users, although less so than in Karttunen et al. Below is his entry for *hakata* ('hack') contrasted to our entry.

(15) hakKa VERB (Koskenniemi)
hakata VERB (Ordinary dictionary)
haka ta VERB HA (Our)
hakka ta VERB VA hakata (Our)

Neither Karttunen et al. nor Koskenniemi seem to consider it important, as we do, that the form of lexical entries is "natural" to casual users. This issue certainly has at least practical import. Both of the discussed models process from left to right, unlike our model. It follows that they cannot analyze forms of lexemes not yet in the lexicon. For example, *vimpuloissa* is phonotactically well-formed form, and all native speakers would agree that it is plural inessive case for the meaningless word *vimpula*. As these models begin an analysis by finding first the matching roots, they are in deep trouble with a form whose root is missing. As our model proceeds from right to left and has all morphological knowledge embedded in rules, *vimpuloissa* will be analyzed into *vimpul a* + **[pl, in]**. This interpretation is rejected only because the lexeme does not exist.

One might argue that Koskenniemi could also augment his model to handle new forms by supplementing the root lexicon with a dummy auxiliary lexicon whose entries are skeletons that represent well-formedness constraints on Finnish stems. But lacking knowledge of the stem boundary for an unknown lexeme the system should invoke skeletons haphazardly and processing time would degrade.

9 Summary

We have described an associative model for the morphological analysis of word forms of Finnish, which is an agglutinative language. Such a model consists of sets of associative morphotactic and stem rules that directly link phonemic segments with their morphemic interpretations, and of a holistic coherence constraint which filters out the associations that make up coherent wholes. We have argued that such an associative model results in an efficient analysis and that it supports open lexicons. We then described a fragment of our fully defined associative model for Finnish word forms. We also discussed an algorithm and its various implementations.

The algorithm has been fully implemented in PASCAL. Our tests demonstrate that the model satisfies quite well the functional requirements we set for it. A clear separation exists between linguistic knowledge (associative rules, the coherence constraint, and the lexicons) on one hand and the algorithm on the other. The model provides a general analysis method of Finnish word forms, including compound word forms. (The model can easily be extended to analyze derivational word forms as well, and in fact it currently analyzes a few of the most commonly used derivational forms.) Analysis is efficient as it takes about 30 ms of VAX11/780 CPU-time on average to analyze word forms in a running text. The figure includes the analysis of compound word forms ordinary newspaper texts contain. Throughput in an IBM PC/XT is about 95 forms per minute. The model supports open lexicons, which is proved by the fact that the over 30,000 lexical entries we currently have have been added from ordinary word forms inputted to the system.

Acknowledgments

This research has been supported by SITRA Foundation. We greatly appreciate the help of Professor Tuomo Tuomi, who gave us a computer print-out of the Reverse Dictionary of Finnish. Aarno Lehtola and Esa Nelimarkka have contributed to the formation of the model. Juha Niemistö has been heavily involved in the implementation of various algorithms. In addition, Asko Hentunen, Esko Nuutila, Pentti Soini, Panu Viljamaa, and Vesa Yläjääski, all students at the Helsinki University of Technology, implemented various versions of the algorithm. We greatly appreciate their help.

References

Brodda, B. and Karlsson, F. 1980 An Experiment with Automatic Morphological Analysis of Finnish. Papers from the Institute of Linguistics, University of Stockholm, Stockholm.

Cercone, N.; Boates, J.; and Krause, M. 1983 A Semi-interactive System for Finding Perfect Hash Functions. Technical Report in Computing Science, Simon Fraser University, Burnaby, British Columbia, Canada.

Cercone, N. and Mercer, R. 1980 Design of Lexicons in some Natural Language Systems. *ALCC Journal* 1(2): 37-59.

Cichelli, R. 1980 Minimal Perfect Hash Functions Made Simple. *Comm ACM* 23(1): 17-19.

Erman, L., Hayes-Roth, F., Lesser, V., and Reddy, D. 1980 The Hearsay-II Speech-Understanding System: Integrating Knowledge to Resolve Uncertainty. *Computing Surveys* 12(2): 213-253.

Ikola, O. 1977 *Nykysuomen käsikirja.* Weiling & Göös, Espoo, Finland.

Jäppinen, H.; Lehtola, A.; Nelimarkka, E.; and Ylilammi, M. 1983a Knowledge Engineering Approach to Morphological Analysis. First Conference of the European Chapter of ACL, Pisa, Italy; 49-51.

Jäppinen, H.; Lehtola, A.; Nelimarkka, E.; and Ylilammi, M. 1983b Morphological Analysis of Finnish: A Heuristic Approach. Report B26, Helsinki University of Technology, Digital Systems Laboratory, Helsinki, Finland.

Karlsson, F. 1981 *Finsk Grammatik.* Suomalaisen Kirjallisuuden Seura, Helsinki, Finland.

Karlsson, F. 1983 *Suomen kielen ä änneja muotorakenne.* WSOY, Porvoo, Finland.

Karttunen, L.; Root, R.; and Uszkoreit, H. 1981 Morphological Analysis of Finnish by Computer. Proceedings of the 71st Ann. Meeting of the SASS. Albuquerque, New Mexico, USA.

Koskenniemi, K. 1983 Two-level Model for Morphological Analysis. IJCAI-83, Karlsruhe, West Germany; 683-685.

Källgren, G. 1983 Computerized analysis and synthesis of Finnish nominals. Papers from the Seventh Scandinavian Conference of Linguistics II, Helsinki, Finland; 433-444.

Matthews, P.H. 1972 *Inflectional Morphology.* Cambridge University Press.

Penttilä, A. 1957 *Suomen Kielioppi.* WSOY, Porvoo, Finland.

Sadeniemi, M., Ed. 1966 *Nykysuomen sanakirja.* WSOY, Porvoo, Finland.

Saukkonen, P.; Haipus, M.; Niemikorpi, A.; and Sulkala, H. 1979 *Suomen Kielen Taajuussanasto.* WSOY, Porvoo, Finland.

Sågvall-Hein, A. 1980 An Outline of a Computer Model of Finnish Word Recognition. Report 3, Fenno-Ugrica Suecana, Uppsala University, Center for Computational Linguistics, Uppsala, Sweden.

Tuomi, T. 1980 *The Reverse Dictionary of Finnish.* Suomalaisen Kirjallisuuden Seura, Hämeenlinna, Finland.

Wiik, K. 1967 Suomen Kielen Morfofonemiikkaa. Report 3, Publications of the Phonetics Department, University of Turku, Turku, Finland.

Winograd, T. 1983 *Language as a Cognitive Process. Volume I: Syntax.* Addison-Wesley.

Notes

1. This work was supported by SITRA Foundation (Finnish National Fund for Research and Development), Helsinki, Finland

2. Current address:
SITRA Foundation
P.O. Box 329
SF-00121 Helsinki, Finland