Harri Jäppinen, Aarno Lehtola,
Esa Nelimarkka, and Matti Ylilammi
Helsinki University of Technology
Espoo, Finland

KNOWLEDGE ENGINEERING APPLIED TO MORPHOLOGICAL ANALYSIS[1]

Introduction

We are currently designing a data-base interface for queries in natural Finnish. As is well known, Finnish is a highly inflectional language. Consequently, in data base applications as well as in other natural Finnish processing systems morphological analysis of word forms constitutes a fundamental computational subproblem. This paper describes our solution to the problem.

Our model is intended for the analysis of Finnish word forms. The system performs all meaningful morphotactic segmentations for a given surface word form, transforms alternated stems into the basic form (sg nominative or 1st infinitive for nominals and verbs, respectively), and matches the stems against lexical entries in order to find the meaninful words. The present version of the system does not analyze compound word forms into their constituents, nor does it analyze derivational word forms. We are building a new version which will have some of these characteristics. Otherwise model is complete: it has been fully implemented, and tests indicate its correctness so far to lie in the neighborhood of 99.5 % (Jäppinen et al., 1983).

Other models have been reported. Brodda and Karlsson (1980) attempted to find the most probable morphotactic segmentations for Finnish word forms without a reference to a lexicon. They report close to 90 % accuracy. Sågvall-Hein (1978) reports an attempt to apply the Reversible Grammar System to a subclass of Finnish morphology. Karttunen et al. (1981) and Koskenniemi (1983) report two distinct and complete models. Both systems first search in a lexicon all words whose roots match with a given input word form and then prune the ones which could result in the input word form. The latter model is symmetric: it analyzes as well as generates Finnish word forms.

----

[1] This research is supported by SITRA (Finnish National Fund for Research and Development), PL 329, 00121 Helsinki 12, Finland

Knowledge Engineering Applied to Morphological Analysis
Harri Jäppinen, Aarno Lehtola, Esa Nelimarkka, Matti Ylilammi
Proceedings of NODALIDA 1983, pages 111-120

Due to the computational environment of our model, data-base interface, we set forth the following three design objectives for our analyser: 1) analysis should be efficient, 2) all valid interpretations of an input word form should be found (in the context of a given lexicon), and 3) the addition of new lexical entries should be easy. The last goal, a human engineering viewpoint, suggested us to minimize morphological information in lexical entries and, instead, store morphological data maximally into active knowledge sources.

Heuristic morphology

We approached the problem of morphological analysis from the vantage point of heuristic search. Heuristics has been used for many years in artificial intelligence problem solving situations. In more recent research heuristics has been expanded into multi-level search, and novel control strategies have been developed to govern the process. Examples are speech understanding (Erman et al., 1980) and many so called expert systems.

Morphology is a much more constrained task domain than speech understanding, mass spectrometry, medical diagnosis, and other typical expert system applications. We, however, decided to study how multi-dimensional heuristic search applies to morphology. We do not use heuristic search because algorithmic methods would not apply (they do, as Karttunen et al. (1981) and Koskenniemi (1983) have demonstrated). Our argument for using heuristic rules was to see if we could get a faster method which would distribute most morphological knowledge in active rules.

We knew that we must be on guard against risks involved: heuristics might sometimes erroneously generate wrong interpretations. Such dangers, however, did not materialize.

Finnish morphology briefly visited

A brief informal and incomplete statement of Finnish morphology is described below. We have presented a bit more detailed statement elsewhere (Jäppinen et al., 1983). An interested reader is adviced to consult Karlsson (1981) for a thorough exposure.

The surface form of a Finnish nominal is composed of the following constituents (parentheses denote optionality):

(1) root + $ + number + case + (possession) + (clitics).

Root denotes the unvarying head part of a word stem, and the stem ending ($) its alternating tail. The root may, however, vary under consonant gradation process. Finnish nominals appear in 14 cases: nominative, genetive, partitive, essive, translative, instructive, abessive, ablative, and allative. Genetive, partitive and illative are more irregular than the others in that they realize in more than one allomorphs. Plural is indicated by a 'i', 'j', 't' or a null string (ø) depending on a context.

To visualize, the Finnish word forms 'takissaniko' (= in my coat?) and 'takkeihisihan' (= into your coats!) are segmented as (notice consonant gradation k - kk):

```
(2)     tak  + i + ø + ssa + ni + ko
        takk + e + i + hi   + si + han
```

The constituent structure for <u>verbs</u> is

(3) root + $ + conjugation + person + (clitics) .

Verbs in Finnish have nominal forms: 5 infinitives and 2 participles in the active voice, and 1 infinitive and 2 participles in the passive voice. These nominal verb forms may receive some but not all cases, and the 1st participle may participate in the adjective comparation process. Most nominal forms may receive a person and a clitic segment in the standard way.

A comparative adjective in Finnish is indicated by the suffix 'mPA' and superlative by 'in' or 'imPA' where 'P' participates in the consonant gradation process (P -> p or P -> m) and 'A' may realize as an 'a', 'ä', 'i' or a null string depending on number and case. The stem ending between a root and a comparation segment is identical to that of the singular genetive case ($\$_{gen}$) for the comparative forms and the plural essive case ($\$_{ESS}$) for the superlative forms.

(4) root + $\$_{gen}$ + mPA + number + case + ...

root + $\$_{ESS}$ + $\substack{\text{imPA}\\ \text{in}}$ + number + case + ...

The heuristic model

The basic control structure of MORFIN, as we call the model, employs the hypothesis-and-test paradigm as follows. A global data base is divided into four levels: surface word form (SWF), morphotactic (MT), basic word form (BWT) and confirmation (C) levels. Between these levels active knowledge sources, production rules, progressively generate and test hypotheses.

Between the surface word form level and the morphotactic level morphotactic productions (MPs) produce hypotheses of possible segmentations and interpretations of an input word. They leave alternated stems untouched. Stem productions (SPs) produce inverse transformations of the variant stems into canonical basic word forms. For nominals we use the singular nominative case and for verbs the 1st infinitive as the basic form. Some stems get rejected in this second phase as impossible alternations. Dictionary look-up finally tests the proliferated hypothetical basic word forms against the existing lexicon. Morphologically ambiguous words result in multiple confirmation level entries, if the words exist in the lexicon.

Figure 1 portrays the levels and an example analysis of an ambiguous surface word form. The numbers in parentheses are pointers to the previous level. '*' indicates a confirmed hypothesis; 'VA' and 'HA' are postulates for the strong (or neutral) and weak (or neutral) grades, respectively.

SWF-level:		VOIMINA	
MT-level:		1. VOIMINA=	N SG nom
		2. VOIMINA=	V akt imper pr y 2 p
		3. VOIMI= NA=	N SG ess
		4. VOIM= I=NA=	N PL ess
		5. VOI= M= I=NA=	N akt III inf PL ess
BWF-level:		1. VOIMIN A	VA (1)
		2. VOIMIN AA	HA (2)
		3. VOIM I	VA (3)
	*	4. VOIM A	VA (4)
		5. VOI N	VA (4)
		6. VOI MI	VA (4)
		7. VO IDA	VA (5)
	*	8. VOI DA	VA (5)
C-level:		VOIMA - TR FORCE, N PL ess	
		VOIDA - TR CAN, N akt III inf PL ess	

Figure 1. The analysis of "voimina".

Morphotactic knowledge

Morphotactic knowledge in MORFIN is embedded in the MPs. The morpheme productions recognize legal morphological surface-segment configurations in a word and slice the word accordingly. The recognition proceeds from right to left up to the stem boundary as in Brodda and Karlsson (1981). That is, for nominals, verbs and adjectival comparation forms morphotactic segmentations are done from clitics up to (but excluding) the stem ending.

We use directly the allomorphic variants of the morphemes. Since possible segment configurations overlap, several mutually exclusive hypotheses are usually produced on the morphotactic level.

We dressed the morphotactical knowledge of MORFIN into context-sensitive rules. The condition part in a production recognizes a single valid morphotactic segment. To prune search we attached up to two left contextual graphemes in a condition:

(5) name: (&$_{i2}$)(&$_{i1}$) segment$_{i}$
--> POSTULATE([interpretation$_{i}$],[next])

Here &$_{i1}$ and &$_{i2}$ describe sets of allowed graphemes in a word to the left of the segment. A recognition leads into the removal of the segment. The (partial) interpretation is recorded and control proceeds to look for the subsequent consistent segmentations (indicated by 'next').

As an example the production

(6) Rpp14: (ALL-[i:,O:])(V+V́)n
--> POSTULATE([verb,active,ind,sg1,no_clitics],[ten1]),

recognizes the 1st person singular verb suffix if a word ends with an 'n', has an ordinary or stressed vowel next to the left and any letter except a long 'i', 'o' or 'ö' second to the left. The partial interpretation of a 1st person singular verb in active voice with no clitics is recorded. Control proceeds then to the collection of rules named 'ten1' which recognizes modal and temporal morphemes.

In specifying MPs we found Brodda and Karlsson (1981) a useful source. The morphotactic knowledge comprises currently 201 distinct MPs. To facilitate efficient processing we compiled the MPs into 32 distinct state transition 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 segments). 'next' in (5) hence indicates legal successor automata for the separated morpheme. Only segmentations and partial interpretations which comprise consistent total interpretations, which end up with an expected stem boundary, get postulated on the MT-level.

To define an abstract morphotactic automata (MTA) let {...}, [...] and <...> denote <u>context</u>, <u>continuation</u> and <u>interpretation</u> formulas, respectively. A concatenated expression {...} [..] <...> is a <u>termination</u> formula. Let § stand for a termination formula or sequence of them. A sequence of termination symbols, preceded by an optional word from the alphabet £={a, b, ... , A, B, ...} is a valid morphotactic automaton $:

```
(7)     $ --> §
        $ --> £+§     .
```

Complex MTAs are generated by the dotted pair

```
(8)     $ --> ($ . $)  .
```

To illustrate MTAs below is the automaton PP for possessive and person in list notation (including (6)):

```
(9)     (n{(V+V́)}(ALL-[i:,O:]) [TEN1<VERB,ACT,IND,S1P>
          (V{(V-[U:])(d,k,l,n,r,s,t,)}[obl,TEN3,PAS2]<VERB,ACT,IND,S3P>
             {(O:)(k)}[obl,TEN3]<VERB,ACT,IND,S3P>
             {(A:)(ALL}[obl,CA3]<3P>)
         t{(V+V́)(ALL-[i:,O:])}[TEN1]<VERB,ACT,IND,S2P>
          (A(v{(V+V́)(ALL-[i:,O:])}[TEN1]<VERB,ACT,IND,P3P>
           V{(O:)(k)}[obl,TEN3]<VERB,ACT,IMPER,P3P>)
         V{(V:-[e:])(ALL)}[]<VERB,ACT,IND,S3P>
         A(s(n{(V+V́)(ALL)}[PAR1,PAR4,INF3,INF4,KOMP2]<GENOM,3P>
              {(V+V́)(ALL)}[obl,CA1]<3P>))
         e(m(m{(V+V́)(ALL)}[obl,CA1]<P1P>
              {(V+V́)(ALL)}[PAR1,PAR4,INF3,INF4,KOMP2]<GENOM,P1P>
              {(V+V́)(ALL-[i:,O:])}[TEN1,TEN4]<VERB,ACT,IND,P1P>)
           n(n{(V+V́)(ALL)}[obl,CA1]<P2P>
              {(V+V́)(ALL)}[PAR1,PAR4,INF3,INF4,KOMP2]<GENOM,P2P>)
           t(t{(V+V́)(ALL)-[i:,O:])}[TEN1]<VERB,ACT,IND,P2P>))
         i(n{(V+V́)(ALL)}[obl,CA1]<S1P>
  {(V+V́)(ALL)}[PAR1,PAR4,INF3,INF4,KOMP2]<GENOM,S1P)
 s{(V+V́)(ALL)}[obl,CA1]<S2P>
  {(V+V́)(ALL)}[PAR1,PAR4,INF3,INF4,KOMP2]<GENOM,S2P>))
```

'V' stands for vowels, 'V́' for the stressed vowels, ':' a long vowel, 'A' for an 'a' or 'ä', 'O' for an 'o' or 'ö', 'U' for a 'u' or 'y', and 'ALL' for any letter.

Knowledge of stem behavior

Stem productions, SPs, describe inverse transformations of stems to their canonical basic forms (the nominative singular case or the 1st infinitive). Stem productions are case-, number-, genus-, mood- and tense-specific heuristic rules which postulate canonical basic forms which in the context of the given morphotactic segmentations might have resulted in the observed variant stem forms. The rules may reject a candidate stem form as an impossible transformation, or produce one or more basic form hypotheses.

Heuristic knowledge of stem behavior is dressed into a set of productions of the following form:

(10) condition --> POSTULATE(cut,string,shift,grade)

If the condition in a production is satisfied, a hypothesis in the canonical form is postulated on the BWF-level by cutting the variant stem ('cut'), adding a new string ('string') as a canonical ending (separated by a blank), and possibly shifting the blank ('shift'). 'Grade' postulates gradation for the stem: 'HA' signals the weak (or neutral) and 'VA' the strong (or neutral) grade.

A well-formed condition (WFC) and its truth value is defined recursively as follows. Any lower case letter in the Finnish alphabet is a WFC and such a condition is true, if the last letter of a stem is identical to that letter. If $\&_1$, $\&_2,\ldots,\&_n$ are WFCs, then the following constructions are also WFCs:

(11) (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, under the stipulation that the recognized letters in a stem are consumed (the next condition tests the next letter).

(ii) is true if $\&_1$ or $\&_2$ or ... or $\&_n$ is true. The testing of $\&_i$'s proceeds from left to right and halts if a recognition occurs. Each $\&_i$ starts afresh; only the recognizing $\&_i$ consumes letters.

To enhance compact notation we stipulate that a capital letter can be used as a macro name for a WFC. As an example production, consider the following:

(12)   <Ka,y>hde --> POSTULATE(3,'ksi',0,HA)

('K' is an abbreviation for <d, f, g, h, k,...>, i.e. the set of Finnish consonants). This production recognizes, among others, the genetive stems 'kahde' (=of two) or 'yhde' (=of one) and generates basic word form hypotheses 'ka ksi' (=two) and 'y ksi' (=one), respectively.

The SPs were collected into 11 distinct sets of productions for nominals and 6 sets for verbs. On average a set has about 25 rules. These sets were compiled into state transition automata to yield efficient processing.

Confirmation of postulates

In addition to an input word itself and its partial interpretations the lexicon is the only static data structure in MORFIN; all other morphological knowledge is dressed in active rules. For example, the word 'pur| si^{42}' is stored in a single entry as

(13)   pur si S NE <..semantic information..>

'takk| i^{4}' is stored in two separate entries as

(14)

takk i S VA <..semantic information..>
tak i   S HA <..semantic information..>

where 'S' stands for a noun, 'VA' for strong, 'HA' for weak, and 'NE' for neutral grade.

The confirmation of a basic word form hypothesis corresponds to a match against lexical entries. An entry matches with a hypothesis if the words match and the grades are not of the opposite strength. If the hypothesis is an adjective comparation form, the lexical entry must furthermore be marked as an adjective.

Performance of the model

MORFIN covers the whole Finnish morphology with the provision below. The singular instructive and the old plural genetive IN-cases are exceedingly rare and are left out for the sake of efficiency. They could be easily added. Currently compound nouns are not analyzed into their constituent parts but we are in the process of designing a new version of MORFIN which will analyze also compound nouns.

There is a precompiled version of MORFIN written in standard Pascal. It has been tested both in DEC 20 and VAX 11/780 configurations.

The analysis of a random word in a newspaper text takes between 4 to 60 ms of DEC 2060 CPU time, about 15 ms on average. On average about 4 postulates were generated on the basic word form level.

The basic approach of MORFIN applies well, we believe, also to analysis of derivational word forms. All one has to do is to add proper MPs and (sometimes) SPs (if needed). Thus, for instance, to account 'iloinen' --> 'iloisesti' derivation we have to add only a single MP to recognize 'sti' and can use the already implemented SPs: the stem alternation is similar to the singular elative case: 'iloinen' (=glad) --> 'iloisesta' (=from glad).

One of the main objectives in our design was to store minimal amount of morphological data in a lexicon and dress it maximally in active rules. We feel success in this regard. The only morphological knowledge words in a lexicon carry is the boundary between the root and the stem ending of a word and the grade of the stem. entries. Consequently, MORFIN has a convenient functional feature: words not existing in a lexicon get analyzed as well as those that do exist. A user can add new entries simply by indicating which of the postulated forms are right. It seems that the introduction of new lexical entries is not as straightforward for a casual user in the other systems.

It seems implausible to us that a native speaker of an inflectional language tags morphological data in individual words. If we take a grammatical but meaningless (non-existent) word, say, 'ventukoissa' and test native Finns, they probably all would agree that it represents the plural inessive form of a non-existent word 'ventukka'. Our model covers such phenomena. 'Ventukoissa' is analyzed as well as meaningful words. Only the dictionary look-up process rejects the word as meaningless.

Conclusion

We have designed and implemented a system for the computational morphological analysis of Finnish word forms. Our analysis is based on multi-level heuristic search in which modular active knowledge sources postulate and evaluate partial morphological interpretations. On the first level morphotactic productions postulate and interprete morphotactic segmentations of an input word. The second phase converts the postulated variant stems into their basic forms. The third phase matches the proliferated basic word form hypotheses against a lexicon.

All morphological knowledge other than the root boundary and the grade of a lexeme is dressed in procedural form, which yields efficient analysis. Grammatical but meaningless word forms become analyzed in the model as well as meaningful ones.

References

Brodda, B., Karlsson, F., An experiment with automatic morphological analysis of Finnish, Papers from the Institute of Linguistics, University of Stockholm, Stockholm 1980.

Erman, L.D. et al., The Hearsay-II speech-understanding system: integrating knowledge to resolve uncertainty. Computing Surveys (June, 1980), 213-253.

Jäppinen, H., Lehtola, A., Nelimarkka, E. and Ylilammi, M., Morphological Analysis of Finnish - A heuristic approach. Helsingi University of Technology, Digital Systems Laboratory, Report B 26, 1983.

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

Karttunen, L., Root, R., and Uszkoreit, Hl, TEXFIN: Morphological analysis of Finnish by computer. Proc. of the 71st Ann. Meeting of the SASS. Albuquerque, 1981.

Koskenniemi, K., Two-level model for morphological analysis. IJCAI-83, 1983, 683-685.

Sågvall-Hein, A-L., Finnish morphological analysis in the reversible grammar system, COLING 78, 1978.