Weighted Finite-State Morphological Analysis of Finnish Inflection and Compounding

Krister Lindén
University of Helsinki
Helsinki, Finland
Krister.Linden@helsinki.fi

Tommi Pirinen
University of Helsinki
Helsinki, Finland
Tommi.Pirinen@helsinki.fi

Abstract

Finnish has a very productive compounding and a rich inflectional system, which causes ambiguity in the morphological segmentation of compounds made with finite state transducer methods. In order to disambiguate the compound segmentations, we compare three different strategies, which we cast in a probabilistic framework. We present a method for implementing the probabilistic framework as part of the building process of lexc-style morpheme sub-lexicons creating weighted lexical transducers. To implement the structurally disambiguating morphological analyzer, we use the HFST-LEXC tool which is part of the open source *Helsinki Finite-State Technology*. This is the first time all three principles are cast in a probabilistic framework and compared on the same corpus using one tool. On our Finnish test corpus, the best method succeeds with 99,98 % precision and recall. [1]

1 Introduction

In languages with productive multipart compounding, such as Finnish, German and Swedish, approximately 9-10 % of the word tokens in a corpus are compounds (Pirkola et al., 2001) and approximately 2/3 of the dictionary entries are compounds, cf. a publicly available Finnish dictionary (Kotimaisten kielten tutkimuskeskus, 2007).

There have been various attempts at curbing the potential combinatorial explosion of segmentations that a prolific compounding mechanism produces. Karlsson (1992) showed that for Swedish the most significant factor in disambiguating compounds was the counting of the number of parts in the analysis, where the analysis with the fewest parts almost always was the best candidate. This has later been corroborated by others. In particular, it was the main disambiguation criterion formulated by (Schiller, 2005) on German compounding. In addition, Schiller used frequency information for disambiguating between compounds with an equal number of parts. Schiller estimated her figures from compound part frequencies, which requires a considerable amount of manual labour in order to create the training corpora consisting of attested compound words and their correct segmentations.

We suggest two modifications to the strategies of Karlsson and Schiller. First we suggest that the word segment probabilities can be estimated from non-compound word frequencies in the corpus. The motivation for our approach is that compounds are formed in order to distinguish between instances of frequently occurring phenomena and therefore compounds are more often formed for more frequently discussed phenomena. We assume that the frequency by which phenomena are discussed is reflected in the non-compound word frequencies, i.e. high-frequency words should in general have more compounds.

In addition, we suggest that the special penalty suggested by Karlsson and maintained by Schiller is unnecessary when framing the problem in a probabilistic framework. This has also been suggested by others, see e.g. Marek (2006). However, this is the first time the disambiguation principles of Karlsson and of Schiller are compared with a fully probabilistic approach on the same corpus.

[1] This is author's draft; it may differ from print version. This paper was published in Proceedings of Nodalida 2009

Previously, there has been no publicly available general framework for conveniently integrating both a full-fledged morphological description and for representing probabilities for general morphological compound and inflectional analysis. Karlsson (1992) used applied a post-processing phase to count the parts, and Schiller (2005) used the proprietary weighted finite-state compiler of Xerox (**?**), which compiles regular expressions. We therefore introduce the open source software tool HFST-LEXC[2], which is similar to the Xerox lexc tool (Beesley and Karttunen, 2003). In addition to the fact that HFST-LEXC compiles lexc-style lexicons, it also has a mechanism for adding weights to compound parts and morphological analyses.

The remainder of the article is structured as follows. In Sections 2 and 3, we introduce a version of Finnish morphology for compounding. In Section 4, we introduce the probabilistic formulation of the methods for weighting the lexical entries. In Section 5, we briefly introduce the test and training corpora. In Section 6, we present the results. Finally, in Sections 7, 8 and 9, we give some notes on the implementation, discuss the results and draw the conclusions.

2 Inflection and Compounding in Finnish

In Finnish morphology, the inflection of typical nouns produces several thousands of forms for the productive inflection. Finnish compounding theoretically allows nominal compounds of arbitrary length to be created from initial parts of certain forms of nouns, and the final part inflects in all possible forms.

For example the compounds describing ancestors are compounded from zero or more of *isän* 'father SINGULAR GENITIVE' and *äidin* 'mother SINGULAR GENITIVE' and then one of any inflected forms of *isä* or *äiti*, creating forms such as *äidinisälle* 'grandfather (maternal) SINGULAR ALLATIVE' or *isänisänisänisä* 'great great grandfather SINGULAR NOMINATIVE'. As for the potential ambiguity, Finnish also has the noun *nisä* 'udder', which creates ambiguity for any paternal grandfather, e.g. *isän#isän#isän#isä*, *isän#isä#nisän#isä*, *isä#nisä#nisä#nisä*, ...

However, much of the ambiguity in Finnish compounds is aggravated by the ambiguity of the inflected forms of the head words. For example *isän*, has several possible analyses, e.g. ISÄ+SG+GEN, ISÄ+SG+ACC and ISÄ+SG+INS.

Finnish compounding also includes forms of compounding where all parts of words are inflected with same form, but this is limited to part of adjective initial compounds. Similarly some inflected verb forms may appear as parts of compounds. These both are more rare than nominal compounds (Hakulinen et al., 2008) and not considered in this paper.

3 Morphological analysis of Finnish

Pirinen (2008) presented an open source implementation of a finite state morphological analyzer for Finnish. We use that implementation as a baseline for the compounding analysis as Pirinen's analyzer has a fully productive compounding mechanism. Fully productive compounding means that it allows compounds of arbitrary length with any combination of nominative singulars, genitive singulars, or genitive plurals in the initial part and any inflected form of a noun as the final part.

The morphotactic combination of morphemes is achieved with sublexicon combinatorics as defined in (Beesley and Karttunen, 2003). We use the open source software called HFST-LEXC with a similar interface as the Xerox lexc tool. The HFST-LEXC tool includes preliminary support for weights on the lexical entries.

In this implementation, each lexical entry constitutes one full word form, i.e., we create a full form lexicon using the previously mentioned analyzer (Pirinen, 2008). This creates a text file of 22 GB for the purely inflectional morphology of approximately 40 000 non-compound lexical entries for Finnish, which were stored in a single CompoundFinalNoun lexicon as shown in Figure 1. The figure demonstrates an unweighted lexicon and also shows how we model the compounding by dividing the word forms into two categories: compound non-final (i.e., nominative singular, genitive singular, and genitive plural) and compound final forms allowing us to give weights to each form or compound part as needed.

Compounding implemented with the unweighted sublexicons in Figure 1 is equivalent to the original baseline analyzer. The root sublexicon specifies that we can have start directly from compound final noun forms, forming single part words, or start from compound initial forms, forming multiword compounds. The compound

[2] http://kitwiki.csc.fi/twiki/bin/view/KitWiki/HfstLexC

```
LEXICON Root
## CompoundNonFinalNoun ;
## CompoundFinalNoun ;

LEXICON Compound
#:0 CompoundNonFinalNound "weight: 0" ;
#:0 CompoundFinalNound "weight: 0" ;

LEXICON CompoundNonFinalNoun
isä   Compound  "weight: 0" ;
isän  Compound  "weight: 0" ;
äiti  Compound  "weight: 0" ;
äidin Compound  "weight: 0" ;

LEXICON CompoundFinalNoun
isä:isä+sg+nom     ## "weight: 0" ;
isän:isä+sg+gen    ## "weight: 0" ;
isälle:isä+sg+all  ## "weight: 0" ;

LEXICON ##
## # ;
```

Figure 1: Unweighted lexicon.

initial lexicon is a listing of all singular nominatives, singular genitives and plural genitives, which is followed by compound boundary marker on in separate sublexicon, and another word from either compound initial sublexicon or compound final sublexicon. The compound final sublexicon contains the long listing of all possible forms of all words, and their analyses,

4 Methodology

We define the weight of a token through its probability to occur in the corpus, i.e. we use the count,*c*, which is proportional to the frequency with which a token appears in a corpus divided by the corpus size, *cs*. The probability, *p(a)*, for a token, *a*, is defined by Equation 1.

$$\mathrm{p}(a) = \mathrm{c}(a)/\mathrm{cs} \tag{1}$$

Tokens known to the lexicon but unseen in the corpus need to be assigned a small probability mass different from 0, so they get *c(x) = 1*, i.e. we define the count of a token as its corpus frequency plus 1 as in Equation 2.

$$\mathrm{c}(a) = 1 + \mathrm{frequency}(a) \tag{2}$$

If a token, e.g. *isän*, has several possible analyses, e.g. ISÄ+SG+GEN and ISÄ+SG+ACC, the total count for *isän* will be divided among the analyses in a disambiguated training corpus. If the disambiguation result removes all readings ISÄ+SG+ACC from the disambiguated result, the count for this reading is still 1 according to Equation 2. We need the total probability mass of all the tokens in the lexicon to sum up to 1, so we define the corpus size as the number of all lexical token counts according to Equation 3.

$$\mathrm{cs} = \sum_x \mathrm{c}(x) \tag{3}$$

To use the probabilities as weights in the lexicon we implement them in the tropical semiring, which means that we use the negative log-probabilities as defined by Equation 4.

$$w(a) = -\log(p(a)) \tag{4}$$

For an illustration of how the weighting scheme is implemented in the lexicon, see Figure 2.

According to Karlsson (1992) and Schiller (2005), we may need to ensure that the weight of the compound segmentation *ab* of a word always is greater than the weight of a non-compound analysis *c* of the same word, so for compounds we use Equation 5, where *a* is the first part of the compound and *x* is the remaining part, which may be split in to additional parts applying the equation recursively.

$$w(ax) = w(a) + M + w(x) \tag{5}$$

In particular, it is true that $w(ab) > w(c)$ if *M* is defined as in Equation 6.

$$M = -\log(1/(\mathrm{cs} + 1)) \tag{6}$$

For an illustration of how a structure weighting scheme with compound penalties is implemented in the lexicon, see Figure 3.

In order to compare with the original principle suggested by Karlsson (1992), we create a third lexicon for which structural weights are placed on the compound borders only, so for compounds we use Equation 7.

$$w(ax) = M + w(x) \tag{7}$$

For an illustration of how a weighting scheme with the compound penalty suggested by Karlsson is implemented in the lexicon, see Figure 4.

```
LEXICON Root
## CompoundNonFinalNoun ;
## CompoundFinalNoun ;

LEXICON Compound
0:# CompoudNonFinalNoun "weight: 0" ;
0:# CompoudFinalNoun "weight: 0" ;

LEXICON CompoundNonFinalNoun
isä   Compound  "weight: -log(c(isä)/cs)" ;
isän  Compound  "weight: -log(c(isän)/cs)" ;
äiti  Compound  "weight: -log(c(äiti)/cs)" ;
äidin Compound  "weight: -log(c(äidin)/cs)" ;

LEXICON CompoundFinalNoun
isä:isä+sg+nom     ## "weight:-log(c(isä+sg+nom)/cs)" ;
isän:isä+sg+gen    ## "weight:-log(c(isä+sg+gen)/cs)" ;
isälle:isä+sg+all  ## "weight:-log(c(isä+sg+all)/cs)" ;
isin:isä+pl+ins    ## "weight:-log(c(isä+sg+all)/cs)" ;

LEXICON ##
## # ;
```

Figure 2: Structure weighting scheme using token penalties.

```
LEXICON Root
## CompoundNonFinalNoun ;
## CompoundFinalNoun ;

LEXICON Compound
0:# CompoundNonFinalNoun "weight: -log(1/(cs+1))" ;
0:# CompoundFinalNoun "weight: -log(1/(cs+1))" ;

LEXICON CompoundNonFinalNoun
isä   Compound  "weight: -log(c(isä)/cs)" ;
isän  Compound  "weight: -log(c(isän)/cs)" ;
äiti  Compound  "weight: -log(c(äiti)/cs)" ;
äidin Compound  "weight: -log(c(äidin)/cs)" ;

LEXICON CompoundFinalNoun
isä:isä+sg+nom     ## "weight:-log(c(isä+sg+nom)/cs)" ;
isän:isä+sg+gen    ## "weight:-log(c(isä+sg+gen)/cs)" ;
isälle:isä+sg+all  ## "weight:-log(c(isä+sg+all)/cs)" ;
isin:isä+pl+ins    ## "weight:-log(c(isä+sg+all)/cs)" ;

LEXICON ##
## # ;
```

Figure 3: Structure weighting scheme using token and compound penalties.

5 Training and Test Data

For training and testing purposes, we use a compilation of three years, 1995-1997, of daily issues of Helsingin Sanomat, which is the most wide-spread Finnish newspaper. This collection contained approximately 2.4 million different words, i.e. types. We disambiguated the corpus using Machinese for Finnish[3] which provided one reading in context for each word based on syntactic parsing.

To create the test material from the corpus, we selected all word forms with more than 20 characters for which our baseline analyzer (Pirinen, 2008) gave a compound analysis, i.e. 53 270 types. Of these, we selected the types which had a structural ambiguity and found 4 721 such words, i.e. approximately 8.9 % of all the compound words analyzed by our baseline analyzer. Of the remaining more than 20-character compounds 63.7 % contained no ambiguities or only inflectional ambiguities. At most, the combination of structural and inflectional ambiguities amounted to 30 readings in three different words which after all is a fairly moderate number.

[3]Machinese is available from Connexor Ltd., www.connexor.com

```
LEXICON Root
## CompoundNonFinalNoun ;
## CompoundFinalNoun ;

LEXICON Compound
0:# CompoundNonFinalNoun "weight: -log(1/(cs+1))" ;
0:# CompoundFinalNoun "weight: -log(1/(cs+1))" ;

LEXICON CompoundNonFinalNoun
isä   Compound  "weight: 0" ;
isän  Compound  "weight: 0" ;
äiti  Compound  "weight: 0" ;
äidin Compound  "weight: 0" ;

LEXICON CompoundFinalNoun
isä:isä+sg+nom     ## "weight:-log(c(isä+sg+nom)/cs)" ;
isän:isä+sg+gen    ## "weight:-log(c(isä+sg+gen)/cs)" ;
isälle:isä+sg+all  ## "weight:-log(c(isä+sg+all)/cs)" ;
isin:isä+pl+ins    ## "weight:-log(c(isä+sg+all)/cs)" ;

LEXICON ##
## # ;
```

Figure 4: Structure weighting scheme using compound penalties.

On the average, the structural and inflectional ambiguity amounts to 2.79 readings per word. Examples of structurally ambiguous words are *aktivointimahdollisuuksien* with the ambiguity *aktivointi#mahdollisuus* 'of the opportunities to activate' vs. *akti#vointi#mahdollisuus* 'of the opportunities to act health' and *hiihto-harjoittelupaikassa* with the ambiguity *hiihto#harjoittelu#paikka* 'in the ski training location' vs. *hiihto#harjoittelu#pai#kassa* 'ski training pie cashier'.

The characteristics of all the compounds in the corpus is presented in Table 1.

# of Characters			# of Segments		
Min.	Max.	Avg.	Min.	Max.	Avg.
2	44	15.34	2	6	2.19

Table 1: Evaluation of compounds, segments and readings.

Examples of six-part compounds are:

- *elo#kuva#teatteri#tuki#työ#ryhmä* 'movie theater support workgroup'
- *jatko#koulutus#yhteis#työ#toimi#kunta* 'higher education cooperation committee'
- *lähi#alue#yhtei#työ#määrä#raha* 'regional cooperation reserve'

The longest compounds found in the corpus is *liikenne#turvallisuus#asiain#neuvottelu#kunnassa* 'in the road safety issue negotiating committee'

6 Tests and Results

We estimate the probabilities for the non-compound words in the 1995 part of the corpus. Since we do not use the compounds for training we can test on the compounds of all three years.

We evaluated the weighting schemes described in Section 4, i.e. the probabilistic method without compound boundary weighting, the probabilistic method combined with compound weighting and the traditional pure compound weighting. The precision and recall is presented in Table 2. Since we only took the first of the best results, the precision is equal to recall.

Parameters	Precision
Only compound penalty	99.94 %
Compound penalty and prefix weights	99.98 %
No compound penalty and prefix weights	99.98 %

Table 2: Precision equals recall for the test results when we use only the first result.

7 Implementation Note

In HFST-LEXC, we use OpenFST (Allauzen et al., 2007) as the underlying finite-state software library for handling weighted finite-state transducers. The estimated probabilities are encoded as weights in the tropical semiring, see (Mohri, 1997). To extract the n-best results, we use a single-source n-best paths algorithm, see (Mohri and Riley, 2002).

8 Discussion and Further Research

Previous results for structural compound disambiguation for German using word probabilities and compound penalties (Schiller, 2005) or using only word probabilities (Marek, 2006) also achieved results with precision and recall in the region of 97-99 %. In German the ambiguities of long compounds may produce even 120 readings, but on the average the ambiguity in compounds is between 2-3 readings (Schiller, 2005), which is on par with the ambiguity of 2.8 readings found for long Finnish compounds. As pointed out initially (Pirkola et al., 2001), the amount of compounds occurring in Finnish, Swedish and German texts is also on a comparable level.

If a disambiguated corpus is not available for calculating the word probabilities, using only the structural penalties may still be an acceptable replacement in Finnish. However, we need to note, that a similar strategy in German, i.e. using only compound penalties on all compound prefixes, did not seem to perform as well (Schiller, 2005). This may be due to the fact that German contains a high number of very short one-syllable words which interfere with the compounding, whereas Finnish is more restricted in the number of short words. Scandinavian languages are similar to German in that they have a number of short one-syllable nouns. Using probabilistic approach with swedish compound disambiguation is demonstrated in (2004), which shows results of 86 % accuracy of compound segmenting when using compound component frequencies and 90 % for number of compound components. However, it is a question for further research whether a pure probabilistic approach could fare as well for Scandinavian languages.

9 Conclusions

For Finnish, weighting compound complexity gives excellent results around 99.9 % almost regardless of the approach. However, from a theoretical point of view, we can still verify the two hypotheses we postulated initially. Most importantly, there seems to be no need to extract the counts from lists of disambiguated compounds, i.e., it is quite feasible to use general word occurrence probabilities for structurally disambiguating compounds. In addition, we can also corroborate the observation that when using word probabilities, it is possible to forego a specific structural penalty and rely only on the word probabilities. From a practical point of view, we introduced the open source tool, HFST-LEXC, and demonstrated how it can be successfully used to encode various compound weighting schemes.

Acknowledgments

This research was funded by the Finnish Academy and the Finnish Ministry of Education. We are also grateful to the HFST-Helsinki Finite State Technology research team and to the anonymous reviewers.

References

[Allauzen et al.2007] Cyril Allauzen, Michael Riley, Johan Schalkwyk, Wojciech Skut, , and Mehryar Mohri. 2007. OpenFst: A general and efficient weighted finite-state transducer library. In *Lecture Notes in Computer Science*, volume 4783, pages 11–23. Springer.

[Beesley and Karttunen2003] Kenneth R Beesley and Lauri Karttunen. 2003. *Finite State Morphology*. CSLI publications.

[Hakulinen et al.2008] Auli Hakulinen, Maria Vilkuna, Riitta Korhonen, Vesa Koivisto, Tarja Riitta Heinonen, and Irja Alho. 2008. Iso suomen kielioppi. referred on 31.12.2008, available from `http://kaino.kotus.fi/visk`.

[Karlsson1992] Fred Karlsson. 1992. Swetwol: A comprehensive morphological analyser for swedish? in nordic journal of linguistics, 15, pp 1?45. In *Machines in Language Translation? Xerox PARC Working Paper, reprinted in Machine Translation*, pages 3–23.

[Kotimaisten kielten tutkimuskeskus2007] Kotimaisten kielten tutkimuskeskus. 2007. Nykysuomen sanalista. Available: `http://kaino.kotus.fi/`.

[Marek2006] Torsten Marek. 2006. Analysis of german compounds using weighted finite state transducers. Technical report.

[Mohri and Riley2002] Mehryar Mohri and Michael Riley. 2002. An efficient algorithm for the n-best-strings problem.

[Mohri1997] Mehryar Mohri. 1997. Finite-state transducers in language and speech processing. *Comp. Linguistics*, (23):269–311.

[Pirinen2008] Tommi Pirinen. 2008. Suomen kielen äärellistilainen automaattinen morfologinen analyysi avoimen lähdekoodin menetelmin. Master's thesis, Helsingin yliopisto.

[Pirkola et al.2001] A Pirkola, T Hedlund, and H Keskustalo. 2001. Dictionary-based cross-language information retrieval: problems, methods and research findings. *Inf. Retrieval*, (4):209–230.

[Schiller2005] Anne Schiller. 2005. German compound analysis with wfsc. pages 239–246.

[Sjöbergh and Kann2004] Jonas Sjöbergh and Viggo Kann. 2004. Finding the correct interpretation of Swedish compounds – a statistical approach. pages 899–902. Lisbon, Portugal.