Low-Resource Active Learning of North Sámi Morphological Segmentation

Stig-Arne Grönroos¹
stig-arne.gronroos@aalto.fi

Kristiina Jokinen²
kristiina.jokinen@helsinki.fi

Katri Hiovain²
katri.hiovain@helsinki.fi

Mikko Kurimo¹
mikko.kurimo@aalto.fi

Sami Virpioja³
sami.virpioja@aalto.fi

¹Department of Signal Processing and Acoustics, Aalto University, Finland
²Institute of Behavioural Sciences, University of Helsinki, Finland
³Department of Information and Computer Science, Aalto University, Finland

December 15, 2014

Abstract

Many Uralic languages have a rich morphological structure, but lack tools of morphological analysis needed for efficient language processing. While creating a high-quality morphological analyzer requires a significant amount of expert labor, data-driven approaches may provide sufficient quality for many applications. We study how to create a statistical model for morphological segmentation of North Sámi language with a large unannotated corpus and a small amount of human-annotated word forms selected using an active learning approach. For statistical learning, we use the semi-supervised Morfessor Baseline and FlatCat methods. After annotating 237 words with our active learning setup, we improve morph boundary recall over 20% with no loss of precision.

This work is licensed under a Creative Commons Attribution–NoDerivatives 4.0 International Licence. Licence details: http://creativecommons.org/licenses/by-nd/4.0/
1 Introduction

In morphologically rich languages, such as the Uralic languages, the number of observed word forms grows rapidly with increasing corpus size. This vocabulary growth can be problematic for natural language processing (NLP) applications, because it causes sparsity in the calculated statistics. Thus it is essential to model such languages on a sub-word level, using for example morphological analysis.

Despite the improvement of development tools and increase of computational resources since the introduction of finite-state transducer (FST) based morphological analyzers in the 1980s [1], the bottleneck for the traditional method of building such analyzers is still the large amounts of manual labor and skill that are required [2]. The strength of such analyzers is the potential to produce output of high quality and richly informative morphological tags.

Morphological surface segmentation is a relaxed variant of morphological analysis, in which the surface form of a word is divided into segments that correspond to morphemes. The segments, called morphs, are not mapped onto underlying abstract morphemes as in FST-based analyzers, but concatenating the sequence of morphs results directly in the observed word form. Allomorphic variation is left unresolved.

Although unsupervised learning of morphological segmenters does not reach the detail and accuracy of hand-built analyzers, it has proven useful for many NLP applications, including speech recognition [3], information retrieval [4], and machine translation [5]. Unsupervised methods are especially valuable for low-resource languages, as they do not require any expensive resources produced by human experts.

While hand built morphological analyzers and large annotated corpora may be unavailable due to the expense, a small amount of linguistic expertise is easier to obtain. Given word forms embedded in sentence contexts, a well-informed native speaker of a language can mark the prefixes, stems, and suffixes of the words in question. A brief collection effort of this type will result in a very small set of annotated words.

Small annotated data of this type can be used to augment large unannotated data by using semi-supervised methods, which are able to learn from such mixed data. As little as one hundred manually segmented words have been shown to provide significant improvements to the quality of the output when comparing to a linguistic gold standard [6]. Adding more annotated data improves the results, with rapid improvement to one thousand words or beyond.

When gathering annotated training samples for a specific model, active learning may provide better results than selecting the samples randomly. In each iteration of active learning, the current best model, trained with all training samples collected up to that point, is used in selection of the new samples to annotate for the next iteration. In this work, we use active learning for morphological segmentation of North Sámi.
1.1 North Sámi

North Sámi (davvisámegiella) belongs to Finno-Ugrian languages and is related to Finnish and other Baltic-Finnic languages. It is one of the nine Sámi languages spoken in the Northern Polar Cap, and spread along Norway, Sweden, Finland and Russia. The speakers of the Sámi languages do not necessarily understand each other but the languages form a chain of adjacent groups. North Sámi is the most widely used Sámi language with around 20 000 speakers, functioning as a lingua franca among the Sámi speakers and used in text books, children’s books, newspapers, and broadcasts.

Linguistically North Sámi is characterized as an inflected language, with cases, numbers, persons, tense and mood. The inflectional system has seven cases. It is accompanied by complicated although regular morphophonological variation. The inflected forms follow weak and strong grades which concern almost all consonants. North Sámi is also fusional: a single word form can stand for more than one morphological category. The nouns have four inflection categories (stems with a vowel or a consonant, the so-called contracting is-nouns, and alternating u-nouns), while the verbs have three conjugation categories (gradation, three syllable, two syllable). The only one syllable verbs are “leat” (to be) and the negation verb. In syntax, the Sámi has separate dual forms for pronouns and verbs besides singular and plural forms.

1.2 Related work

While unsupervised morphological segmentation has recently been an active topic of research [7], semi-supervised morphological segmentation has not received as much attention. One approach is to seed the learning with a small amount of linguistic knowledge in addition to the unannotated corpus [8]. Some semi-supervised methods where a part of the training corpus is supplied with correct outputs have also been presented, including generative [6, 9, 10] and discriminative [11, 12] methods.

Active learning methods have been applied for constructing FST-based analyzers by eliciting new rules from a user with linguistic expertise [13, 14]. These development efforts are fast for rule-based systems, but still require months of work. There has been research effort into FST-based morphology for Sámi languages [15, 16, 17].

North Sámi is the focus of the DigiSami project, which attempts to increase the digital viability of minor Finno-Ugric languages by technology development, analysis, data collection (read and conversational speech), and encouragement of community effort in online content creation [18, 19]. This work directly supports the ultimate goal of the project, which is to produce tools and technology that would allow Sámi speech-based applications to be developed. Although North Sámi has various linguistic resources, there are not many related to speech technology.
2 Methods

As a method for morphological segmentation of words, we use Morfessor. It is a family of methods for learning morphological segmentations primarily from unannotated data. The methods are based on a generative probabilistic model which generates the observed word forms by concatenating morphs. The model parameters $\theta$ define a morph lexicon. The morph $m_i$ is considered to be stored in the morph lexicon, if it has a non-zero probability $P(m_i | \theta)$ given the parameters.

Morfessor utilizes a prior distribution $P(\theta)$ over morph lexicons, derived from the Minimum Description Length principle [20]. The prior favors lexicons that contain fewer, shorter morphs. The purpose is to balance the size of the lexicon, and the size of the corpus $D$ when encoded using the lexicon. This balance can be expressed as finding the following Maximum a Posteriori (MAP) estimate:

$$\hat{\theta} = \arg \max_{\theta} P(\theta | D) = \arg \min_{\theta} \left( - \log P(\theta) - \log P(D | \theta) \right).$$

(1)

In the Morfessor variants used in this work, the lexicon encodes the forms of the morphs directly as strings: each letter requires a certain number of bits to encode.

2.1 Morfessor Baseline

Morfessor Baseline [21, 22] employs a morph lexicon $P(m | \theta)$ that is simply a categorical distribution over morphs $m$, in other words a unigram model. The model parameters $\theta$ are optimized utilizing a greedy local search, in which one training word at a time is reanalyzed and the model parameters updated accordingly.

In order to use the annotations produced in the active learning for training Morfessor, we employ the semi-supervised extension to Morfessor Baseline [6]. This involves replacing the MAP estimate (1) with the optimization

$$\hat{\theta} = \arg \min_{\theta} \left( - \log P(\theta) - \log \alpha P(D | \theta) - \log \beta P(A | \theta) \right),$$

(2)

where $D$ is the unannotated and $A$ the annotated training corpus, $\alpha$ and $\beta$ are the weights for the likelihood of the unannotated corpus and annotated corpus, respectively. The hyper-parameters $\alpha$ and $\beta$ affect the overall amount of segmentation and the relative importance of using the morphs present in the annotated corpus.

2.2 Morfessor FlatCat

The most recent Morfessor variant is called Morfessor FlatCat [10]. The main difference between Morfessor Baseline and Morfessor FlatCat is the use of morph categories...
in the latter. Each morph token is categorized as PREFIX, STEM, or SUFFIX. Internally to the algorithm, a NON-MORPH category is used, intended to model frequent sub-strings that are not morphs but fragments of a morph. Word formation is modeled using a hidden Markov model (HMM) having morph categories as hidden states and morphs as observations. HMM morphotactics were previously used in the Categories-ML [23] and Categories-MAP [24] variants of Morfessor, but Morfessor FlatCat is the first method to combine the approach with semi-supervised training.

The benefit of the HMM morphotactics is increased context-sensitivity, which improves the precision of the segmentation. For example, in English, the model can prevent splitting a single s, a common suffix, from the beginning of a word. Modeling of morphotactics also improves the segmentation of compound words, by allowing the overall level of segmentation to be increased. The main benefits of semi-supervised learning are in the modeling of suffixation [10].

The prior of Morfessor FlatCat is otherwise the same as in Morfessor Baseline, but it also includes encoding of the right and left perplexity of the morph. The perplexity measures describe the predictability of the contexts in which the morph occurs. The perplexities, together with the length of the morph, are used to calculate the emission probability of a morph conditioned on the morph category, \( P(m \mid c) \).

2.3 Pool-based active learning

Pool-based active learning [25] has been successfully applied in NLP [26]. In pool-based active learning, the system has access to a pool of unlabeled data \( \mathcal{A} \) and can request from the annotator true labels for a certain number of samples in the pool.

A method for choosing which samples to annotate still needs to be defined. A well suited approach for generative models is to use the model’s estimate of the uncertainty of the decision associated with a particular sample in order to select the additional samples to annotate [27]. In the case of morphological segmentation, we use the uncertainty of a word’s current segmentation in order to assess its value as an additional annotation. The next word to annotate \( \mathbf{A}_{(t+1)} \) at time step \( t \) is selected from \( \mathcal{A} \) based on the uncertainty of the current best segmentation \( Z_i \)

\[
\mathbf{A}_{(t+1)} = \arg \min_{w_i \in \mathcal{A}} \frac{P(Z_i \mid \theta_t)}{P(w_i \mid \theta_t)},
\]

where the likelihood of the current segmentation \( P(Z_i \mid \theta_t) \) is given by the Viterbi algorithm [28] and the likelihood of the word with any segmentation \( P(w_i \mid \theta_t) \) is given by the forward algorithm [29].
Table 1: Sizes of the unannotated corpora and the initial division into subsets.

<table>
<thead>
<tr>
<th>Corpus</th>
<th>Word tokens</th>
<th>Word types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Den samiske tekstbanken</td>
<td>17,985,140</td>
<td>691,190</td>
</tr>
<tr>
<td>UIT-SME-TTS</td>
<td>42,150</td>
<td>8,194</td>
</tr>
<tr>
<td>Development set</td>
<td>–</td>
<td>100</td>
</tr>
<tr>
<td>Evaluation pool</td>
<td>–</td>
<td>900</td>
</tr>
<tr>
<td>Training pool A</td>
<td>–</td>
<td>7,194</td>
</tr>
</tbody>
</table>

3 Experiments

We used two different text corpora in our experiments. The sizes of the corpora are shown in Table 1. The larger *Den samiske tekstbanken* corpus was only used to construct a word list, to use as the unannotated training data. The smaller *UIT-SME-TTS* corpus was divided into separate pools from which evaluation and training words were drawn for annotation. The sentences in which the words occur were also extracted for use as contexts. To ensure that the evaluation words are unseen, the words in the evaluation pool were removed from the other subsets.

The use of two corpora enables the release of the annotations with their sentence contexts, which would have been precluded by the restrictive corpus of the *Tekstbanken* corpus. It also demonstrates the effectiveness of the system under the realistic scenario where a large general-domain word list for the language is available for use, even though the corpora themselves are restricted by licensing. A similar scenario would be selection from a specific target domain corpus.

Initially we use Morfessor Baseline, but towards the end of the experiment we switch the method to Morfessor FlatCat. As prefixes are very rare in North Sámi, and none were seen in the annotations, we disabled the prefix category.

3.1 Active learning

Our active learning procedure starts from nothing but an unannotated corpus collected for other purposes. An initial model is trained in an unsupervised fashion. The procedure then applies three components iteratively: (i) selection of new words to annotate using the current model, (ii) elicitation of annotations for the selected words, and (iii) training of the new segmentation model using all available training data.

For the elicitation step, we developed a web-based annotation interface. A javascript app using the jQuery framework was used as a front-end and a RESTful Python wsgi-app built on the bottle framework as a back-end. Screenshots of the annotation
interface are shown in Figure 1. For words in the training pool, the interface shows the segmentation of the current model as a suggestion to the annotator. Words in the evaluation pool are shown unsegmented, in order not to bias the annotator.

There are no efficient on-line training algorithms for Morfessor FlatCat. Thus we gather a list of 50 new words to annotate, by ranking the potential words according to (3), and re-train once the whole list has been annotated. Re-training includes hyper-parameter optimization (HPO) for $\alpha$ and $\beta$. Due to the very limited amount of training data, and a lack of previously collected annotated development set, we initially decided to use 3-fold cross-validation on the annotated training set for HPO. This initial approach was quickly shown to be flawed, as the values of the hyper-parameters did not begin to converge after multiple iterations. This divergence can be explained by HPO requiring the development set to be an unbiased sample of the data distribution. A subset biased towards maximally informative words is desired for use as training words, but using them for HPO introduces an undesired bias.

To remedy this situation, we constructed a development set of 100 randomly selected words with annotations. We then restarted the iterations, now using the development set for HPO. In this second approach, when a word that had already been annotated was reselected for annotation, the old annotation was used, making it unnecessary to re-elicit from the annotator. The training iterations and the respective hyper-parameter values are shown in Table 2.

### 3.2 Annotation details

The annotations were produced by a single trained linguist, who is not a native speaker of Sámi. In total 457 randomly selected word types and 346 actively selected word types were annotated under a time span of 17 days. The total time spent by the annotator was 19 hours (over 30 min breaks omitted).
### Table 2: The model parameters and number of annotations for the active learning iterations. U and S stand for unsupervised and semi-supervised training, C and D for setting hyper-parameters by cross-validation and development set, respectively.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Training</th>
<th>Hyper-parameters</th>
<th>Annotated words</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$\alpha$</td>
<td>$\beta$</td>
<td>Dev</td>
</tr>
<tr>
<td>C 0</td>
<td>U Baseline</td>
<td>0.42</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>C 1</td>
<td>U Baseline</td>
<td>1.3</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>C 2</td>
<td>S Baseline</td>
<td>2.4</td>
<td>1300</td>
<td>–</td>
</tr>
<tr>
<td>C 3</td>
<td>S Baseline</td>
<td>2.7</td>
<td>800</td>
<td>–</td>
</tr>
<tr>
<td>C 4</td>
<td>S Baseline</td>
<td>3.4</td>
<td>900</td>
<td>–</td>
</tr>
<tr>
<td>D 0</td>
<td>U Baseline</td>
<td>1.1</td>
<td>–</td>
<td>100</td>
</tr>
<tr>
<td>D 1</td>
<td>S Baseline</td>
<td>1.4</td>
<td>800</td>
<td>100</td>
</tr>
<tr>
<td>D 2</td>
<td>S Baseline</td>
<td>1.5</td>
<td>700</td>
<td>100</td>
</tr>
<tr>
<td>D 3</td>
<td>S FlatCat</td>
<td>0.2</td>
<td>1400</td>
<td>100</td>
</tr>
<tr>
<td>D 4</td>
<td>S FlatCat</td>
<td>0.5</td>
<td>2200</td>
<td>100</td>
</tr>
</tbody>
</table>

Most of the annotated word tokens had an unambiguous segmentation agreeing with established linguistic interpretation. These words contain only easily separated suffixes: markers for case and person, and derivational endings. However, some words required the annotator to make choices on where to place the boundary.

One challenge was posed by the extensive stem alternation and fusion in Sámi. To maximize consistency, the segmentation boundary was usually placed so that all of the morphophonological alternation remains in the stem. Exceptions include the passive derivational suffix, which is found as variants -ojuvvo- and -juvvo- depending on the inflectional category and stem type. Another challenge were lexicalized stems. These stems appear to end with a derivational suffix, but removal of the suffix does not yield a morpheme at all, or results in a morpheme with very weak semantic relation to the lexicalized stem. An example is ráhkadit (make, produce).

### 3.3 Evaluation

The word segmentations generated by the model are evaluated by comparison with annotated morph boundaries using boundary precision, boundary recall, and boundary $F_1$-score [30]. The boundary $F_1$-score equals the harmonic mean of precision (the percentage of correctly assigned boundaries with respect to all assigned boundaries) and recall (the percentage of correctly assigned boundaries with respect to the reference boundaries). Precision and recall are calculated using macro-averages over the
words in the evaluation set. In the case that the word has more than one annotated segmentation, we take the one that gives the highest score.

We also report the scores for subsets of words consisting of different morph category patterns found in the evaluation set. These categories are words that should not be segmented (STM), compound words consisting of exactly two stems (STM+STM), a stem followed by a single suffix (STM+SUF) and a stem and exactly two suffixes (STM+SUF+SUF). Only precision is reported for the STM pattern, as recall is not defined for an empty set of true boundaries.

3.4 Results

Figure 2 shows the improvement of the $F_1$-score as more annotations became available. Training a Morfessor FlatCat model after three iterations provided a large boost, even though the annotated words had so far been selected by Baseline models. In contrast, the words selected by the FlatCat model (3FC) for annotation did not benefit the Baseline model (4BL).

Table 3 shows scores for sets of words with different morphological patterns. For the full test set, we improve morph boundary recall over 20% (relative) with no loss of precision, when comparing the first model of the second approach (D0) to the last.
Table 3: Boundary precision (Pre), recall (Rec), and $F_1$-scores for different subsets of the evaluation data.

<table>
<thead>
<tr>
<th>Model</th>
<th>STM Pre</th>
<th>STM+STM Pre</th>
<th>STM+SUF Pre</th>
<th>STM+SUF+SU Pre</th>
<th>Full test set Pre</th>
<th>Full test set Rec</th>
<th>Full test set $F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>C0 Baseline</td>
<td>.57</td>
<td>.85</td>
<td>.88</td>
<td>.87</td>
<td>.65</td>
<td>.33</td>
<td>.44</td>
</tr>
<tr>
<td>D0 Baseline</td>
<td>.70</td>
<td>.89</td>
<td>.85</td>
<td>.87</td>
<td>.71</td>
<td>.31</td>
<td>.43</td>
</tr>
<tr>
<td>D4 FlatCat</td>
<td>.73</td>
<td>.85</td>
<td>.92</td>
<td>.89</td>
<td>.75</td>
<td>.38</td>
<td>.46</td>
</tr>
</tbody>
</table>

model (D4). The performance has improved for all morph patterns. The STM+SUF pattern has the largest increase, with improvements both in precision and recall. The recall scores of compound words (STM+STM) and suffix sequences (STM+SUF+SU) are also clearly improved.

4 Conclusions

We have applied an active learning approach to modeling morphological segmentation of North Sámi. The work was accomplished using open-source software ¹. We present the collected language resources for the use of the scientific community ².

The performance of the segmentation model was shown to increase rapidly as the amount of human-annotated data was increased. One of our findings is the importance of collecting an unbiased development set for optimization of hyper-parameters, even though this reduces the amount of human-labeled data available for training. Cross-validating using the selected samples is not an adequate compromise.

One avenue for future work is exploring other measures for selecting the words to annotate. These can include applying other language models, but can also be based on direct statistics of the language, e.g. frequencies and lengths of the words or substrings of the words. A thorough comparison to random selection should also be performed. Another question is how well this approach extends to other languages and corpora.

Acknowledgments

This research has been supported by EC’s Seventh Framework Programme under grant n°287678 and the Academy of Finland under the Finnish Centre of Excellence Program 2012–2017 (grant n°251170), LASTU Programme (grants n°256887 and 259934) and Fenno-Ugric Digital Citizens (grant n°270082). Computer resources within the Aalto University School of Science “Science-IT” project were used.

¹Available at http://www.cis.hut.fi/projects/morpho/.
²Available at http://research.spa.aalto.fi/speech/data_release/north_sami_active_learning/.
References


