Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

Low-Resource Active Learning of Morphological Segmentation

Stig-Arne Grönroos[1]
stig-arne.gronroos@aalto.fi

Katri Hiovain[2]
katri.hiovain@helsinki.fi

Peter Smit[1]
peter.smit@aalto.fi

Ilona Rauhala[2]
ilona.rauhala@helsinki.fi

Kristiina Jokinen[2]
kristiina.jokinen@helsinki.fi

Mikko Kurimo[1]
mikko.kurimo@aalto.fi

Sami Virpioja[3]
sami.virpioja@aalto.fi

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

October 10, 2016

Abstract

Many Uralic languages have a rich morphological structure, but lack morphological analysis tools 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 with a large unannotated corpus and a small amount of annotated word forms selected using an active learning approach. We apply the procedure to two Finno-Ugric languages: Finnish and North Sámi. The semi-supervised Morfessor FlatCat method is used for statistical learning. For Finnish, we set up a simulated scenario to test various active learning query strategies. The best performance is provided by a coverage-based strategy on word initial and final substrings. For North Sámi we collect a set of human-annotated data. With 300 words annotated with our active learning setup, we see a relative improvement in morph boundary F_1-score of 19% compared to unsupervised learning and 7.8% compared to random selection.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

1 Introduction

In morphologically rich languages, such as the Uralic languages, the number of observed word forms grows rapidly with increasing corpus size. For instance, in Finnish, nouns can have over 2000 different word forms due to case inflection and various clitics, while verbs can have about 12 000 different forms due to person, number, tempus, and modus inflection and especially to the abundance of infinitival and participle forms, the latter of which are inflected like nouns (Karlsson, 1982). Naturally not all valid combinations of suffixes are common in usage, but they are nevertheless not only theoretical possibilities but part of the living language.

This vocabulary growth can be problematic for natural language processing (NLP) applications, because it causes sparsity in the calculated statistics. Compared e.g. with English which has a small number of inflectional forms, Finnish does not easily lend itself to word form n-gram probability to be used as the basis of NLP tasks since not all possible word forms, not to mention their combinations, occur even in large corpora. Thus it is essential to model such languages on a sub-word level, using for example morphological analysis that allows word forms to be analyzed into parts of two types: the lexical meaning carrying part(s) and the various morphemes which carry grammatical information.

Despite the improvement of development tools and the increase of computational resources since the introduction of finite-state transducer (FST) based morphological analyzers in the 1980s (Koskenniemi, 1983), the bottleneck for the traditional method of building such analyzers is still the large amounts of manual labor and skill that are required (Koskenniemi, 2008). The strength of such analyzers is the potential to produce output of high quality and detailed 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 (Creutz et al., 2007), information retrieval (Kurimo et al., 2010), and machine translation (Virpioja et al., 2007; Fishel and Kirik, 2010; Grönroos et al., 2015b). 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.

A small amount of annotated data of this type can be used to augment a large amount of 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 compared to a linguistic gold standard (Kohonen et al., 2010). Adding more annotated data improves the results, with rapid improvement at least up to one thousand words. Ruokolainen et al.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

(2016) provide an empirical comparison of semi-supervised methods for morphological segmentation.

When gathering annotated training samples for a specific model, *active learning* may provide better results than selecting the samples randomly. A common objective is to reach adequate performance with a shorter annotator effort. In active learning, the annotated words are chosen according to some strategy, making use of information from the available data set, previously selected words, and models trained in previous iterations.

In this work, we use active learning for morphological segmentation of Finnish and North Sámi. This work extends the preliminary results in our previous work (Grönroos et al., 2015a). We extend the work by including experiments in a second language: Finnish. We explore several query strategies for selecting the words to annotate. The comparison to random selection is more rigorously performed.

2 Related work on North Sámi

There has been research effort into FST-based morphology for Sámi languages (Trosterud and Uibo, 2005; Lindén et al., 2009; Tyers et al., 2009). In particular, the Giellatekno research lab[1] provides rule-based morphological analyzers both for individual word forms and running text, in addition to miscellaneous other resources such as wordlists and translation tools. The morphological analyzer gives the morphological properties of a word in the form of tags. For example, given the word ***vaddjojuvvon*** ("cut", PASSIVE), the analyzer produces the following output:[2]

(1)
	vaddjojuvvon	vadjat+V+TV+Der/PassL+V+IV+Ind+Prs+Sg1
	vaddjojuvvon	vadjat+V+TV+Der/PassL+V+IV+Ind+Prt+ConNeg
	vaddjojuvvon	vadjat+V+TV+Der/PassL+V+IV+PrfPrc

Speech technology tools for North Sámi have been explored in the DigiSami project[3] (Jokinen, 2014), which is one of the projects in the Academy of Finland research framework aimed to increase and support digital viability of less-resourced Finno-Ugric languages with the help of speech and language technology. DigiSami focuses especially on North Sámi, and sets to collect data, provide tools, and develop technology to enable North Sámi speech-based applications to be developed (Jokinen and Wilcock, 2014a). Moreover, the project aims to encourage community effort for online content creation, and for this, Wikipedia-based applications are supported, such as WikiTalk (Jokinen and Wilcock, 2014b; Wilcock et al., 2016). This is a robot-application which allows the user to interact with a robot concerning information in the Wikipedia articles.

For speech recognition, a method for statistical segmentation may be preferred over rule-based morphological analyzers. A rule-based analyzer is limited in the vocabulary it recognizes, and non-standard spellings might not be analyzed at all. In addition, the tag set produced by the analyzer may be too rich. For instance, a morphological segmentation of the above example word, ***vaddj + ojuvvo + n***, consists of only 3 morphs, while the Giellatekno analyzer gives a lemma and 6 to 8 tags. Such abstract tags produced by a

[1] `http://giellatekno.uit.no/`
[2] For tag definitions, see `http://giellatekno.uit.no/doc/lang/sme/docu-sme-grammartags.html`
[3] `http://www.helsinki.fi/digisami/`

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

morphological analyzer are not directly applicable in speech recognition, which requires lexical units that can be concatenated into the surface form of the words (Hirsimäki et al., 2006). The work described in this paper directly supports development of the tools that can be used to develop speech technology for North Sámi or other less-resourced languages.

3 On North Sámi and Finnish Morphology

North Sámi (davvisámegiella) belongs to the Finno-Ugric languages and is related to Finnish and other Baltic-Finnic languages. It is one of the nine Sámi languages spoken in the northern parts of Norway, Sweden, Finland and Russia. North Sámi is the biggest of the Sámi languages, with around 30 000 speakers. As the Sámi language speakers do not necessarily understand each other, North Sámi functions as a lingua franca among the Sámi speakers. It is also widely used in newspapers and text books, and there are Sámi language TV and radio broadcasts.

Linguistically, North Sámi is characterized as an inflected language, with cases, numbers, persons, tense and mood. The inflectional system has seven categories: the nouns have four inflection categories (stems with a vowel or a consonant, the so-called contracting is-nouns, and alternating u-nouns), and the verbs have three conjugation categories (gradation, three syllabic, and two syllabic verbs). The only monosyllabic verbs are "leat" (to be) and the negation verb.[4] The verbs and pronouns have specific dual forms besides singular and plural forms, i.e. "we the two of us" and "we more than two".

North Sámi features a complicated although regular morphophonological variation. For instance, the inflected forms follow weak and strong grades which concern almost all consonants. North Sámi is also a fusional language and a single morph can stand for more than one morphological category. In a similar way as in Estonian, loss of certain suffixes has resulted in complicated morphophonological alternations or gradation patterns in the stem. This is especially true of the genitive-accusative form, e.g. *girji* ("book", SgNom) vs. *girjji* ("book", SgGen-Acc).

Adjectives typically have two forms: predicative (*duojár lea* ***čeahppi*** "the craftsman is skillful") and attributive (***čeahpes*** *duojár* "a skillful craftsman") (Sammallahti, 1998). Furthermore, for many adjectives the attributive form can take two alternative forms. For example *seavdnjat* ("dark") has the two attributive variants *sevdnjes* and *seavdnjadis*.

North Sámi has productive compound formation, and compounds are written together without an intermediary space. For example *nállošalbmái* ("into the eye of the needle"), could be segmented as *nállo + šalbmá + i*. North Sámi makes extensive use of derivation, both in verbs and in nouns. For example the adjective *muoŧtái* ("with many aunts") is derived in a regular manner from *muoŧta* ("aunt").

In order to show applicability of the proposed method to another language, we include experiments using Finnish. The choice of Finnish as the second language is motivated by its morphological similarity to North Sámi, making it reasonable to use the results of the Finnish experiment in designing the North Sámi experiment. In addition, we can take advantage of the wide availability of data and tools for Finnish. The Morpho

[4]Like Finno-Ugrian languages in general, also North Sámi forms negation by a particular negation verb which is inflected in person.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

Challenge data (Kurimo et al., 2007, 2010) processed by a morphological analyzer enable the experiment with a simulated annotator.

North Sámi and Finnish morphology share many similarities. Nouns are inflected by case and can have possessive suffixes attached, while verbs inflect by person, number, tempus and modus. Morphemes are typically ordered according to the same structure, such as in

(2) *kisso +i +lla +nne +kin*
stem PL. ADE. POSS. clitic
bussá +in +eattet +ge
also on your cats

There are also syntactic similarities, such as forming negations using a negation verb. Both languages have gradation of stems. There is even a large number of words with a shared origin, both through the shared origin of the languages and through loaning of words from Finnish to Sámi.

There are also some dissimilarities between the languages, including the dual form for pronouns and verbs in North Sámi, and the number of cases (6 in North Sámi, 15 in Finnish). Adjectives in Finnish do not have a separate attributive form.

Moreover, the morphophonology of the languages differs. North Sámi has neither vowel harmony nor final consonant gemination. North Sámi has 30 consonants, which is more than the 17 in Finnish, but less vowels (7 in North Sámi, 8 in Finnish) (Aikio, 2005; VISK, 2004). In North Sámi, gradation applies to almost all consonants, and thus there is more morphophonological alternation than in Finnish.

4 Annotation of North Sámi Segmentation

Most North Sámi words have an unambiguous segmentation agreeing both with intuition and with established linguistic interpretation. These words contain only easily separated suffixes: markers for case and person, and derivational endings. However, some words require the annotator to make choices on where to place the boundary. In this section, we will describe the challenges faced during annotation, and the decisions made in response.

As a general principle, we aimed to maximize the consistency of the annotations. For established linguistic interpretation we referred to the work by Aikio (2005); Álgutietokanta (2006); Nickel and Sammallahti (2011); Sammallahti (1998).

Inflectional morphology is typically more straightforward to analyze than derivational morphology. The optimal granularity on which to analyze derivations depends on the needs of the application. It appears that the Giellatekno analyzer does not return verbs derived from nouns to the originating noun, even though it does do so for verbs derived from other verbs. We have segmented both derivational and inflectional morphology, without marking the distinction in the segmentation. We deviate from the granularity preferred by Giellatekno by also segmenting derivational suffixes that convert nouns into verbs, if the boundary is distinct.

An exception was made in the case of certain 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.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

An example is *ráhkadi + t* ("make, produce"), rather than *ráhka + di + t*, compared to *ráhka + t* ("crack").

A related challenge was posed by certain lexicalized adverbial forms. These words appear to contain suffixes that could have been segmented, but these suffixes do not have their conventional function in the word. For example, the segmentations *davá + s* ("to the north") and *davvi + n* ("in the north") would appear to contain the singular locative and essive case marker, respectively, but would not have their conventional meanings. A decision was made to leave these forms unsegmented.

To remain consistent, it was rather important that the annotator(s) recognized the declensions of the words. This is because North Sámi has several declensions both in nouns and verbs, and the segmentations often vary depending on them when following grammatical interpretation.

A further 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. Even though language education classifies verbs into verb types according to the suffix (*-it, -at, -ut, ...*), we have segmented the infinitive marker as *-t*. The preceding vowel is seen as part of the stem, undergoing alternation for phonological and grammatical reasons. A similar decision was needed for the multifunctional derivational ending of verbs, *-d-* or *-di-*. Also, the corresponding literature shows some varying interpretations about these suffixes (Sammallahti, 1998; Nickel and Sammallahti, 2011). For example *boradit* could be segmented both as *bora + di + t* and *bora + d + it*. In this work we have used the former segmentation.

Exceptions include the passive derivational suffix, which is found as variants *-ojuvvo-*, *-juvvo-* and *-uvvo-*, depending on the inflectional category and stem type. The pleonastic derivational ending for actor occurs in the forms *-jeaddji-* and *-eaddji-*.

Observe that many of the segmented suffixes, such as *-i*, and *-t*, occur homonymously in different word classes. For example *-t* could act as a marker for nominative plural in nouns or a marker for present time Sg2 person in verbs, and can also have other functions.

5 Semi-supervised Morphological Segmentation

While unsupervised morphological segmentation has recently been an active topic of research (Hammarström and Borin, 2011), semi-supervised morphological segmentation has not received as much attention. Semi-supervised morphological segmentation can be approached in many ways. One approach is to seed the learning with a small amount of linguistic knowledge in addition to the unannotated corpus (Yarowsky and Wicentowski, 2000). Some semi-supervised methods where a part of the training corpus is supplied with correct outputs have also been presented, including generative (Kohonen et al., 2010; Sirts and Goldwater, 2013; Grönroos et al., 2014) and discriminative (Poon et al., 2009; Ruokolainen et al., 2014) methods.

5.1 Morfessor FlatCat

As a method for morphological segmentation of words, we use Morfessor FlatCat (Grönroos et al., 2014). It is the most recent addition to the Morfessor family of methods for learning morphological segmentations primarily from unannotated data.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

The method is based on a generative probabilistic model which generates the observed word forms by concatenating morphs. The model parameters $\boldsymbol{\theta}$ define a *morph lexicon*. The morph m_i is considered to be stored in the morph lexicon, if it has a non-zero probability $P(m_i \mid \boldsymbol{\theta})$ given the parameters.

Morfessor utilizes a prior distribution $P(\boldsymbol{\theta})$ over morph lexicons, derived from the Minimum Description Length principle (Rissanen, 1989). The prior favors lexicons that contain fewer, shorter morphs. The purpose is to find a balance between, on one hand, the size of the lexicon, and, on the other hand, the size of the corpus $\boldsymbol{D}$ when encoded using the lexicon $\boldsymbol{\theta}$. This balance can be expressed as finding the following Maximum a Posteriori (MAP) estimate:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} P(\boldsymbol{\theta} \mid \boldsymbol{D}) = \arg\min_{\boldsymbol{\theta}} \big(-\log P(\boldsymbol{\theta}) - \log P(\boldsymbol{D} \mid \boldsymbol{\theta}) \big). \tag{3}$$

In order to use the annotations produced in the active learning for training Morfessor, we employ the semi-supervised training approach by Kohonen et al. (2010). This involves replacing the MAP estimate (3) with the optimization

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \big(-\log P(\boldsymbol{\theta}) - \alpha \log P(\boldsymbol{D} \mid \boldsymbol{\theta}) - \beta \log P(\boldsymbol{A} \mid \boldsymbol{\theta}) \big), \tag{4}$$

where $\boldsymbol{A}$ is the annotated training corpus, and α and β are the weights for the likelihood of the unannotated corpus and annotated corpus, respectively. Both the hyper-parameters α and β affect the overall amount of segmentation predicted by the model. The β hyper-parameter also affects the relative importance of using the morphs present in the annotated corpus, compared to forming a segmentation from other morphs in the lexicon.

Morfessor FlatCat uses a flat lexicon, in contrast to the hierarchical lexicon in the Categories-MAP (Creutz and Lagus, 2005) (Cat-MAP) variant of Morfessor. In a hierarchical lexicon, morphs can be built using other morphs already in the lexicon, while in a flat lexicon each morph is represented directly as a string of letters. Each letter requires a certain number of bits to encode, making longer morphs more expensive to add to the lexicon.

A hierarchical lexicon has some benefits in the treatment of frequent strings that are not morphs, but it also presents challenges in model training. When using a flat lexicon, all morph references point from the corpus to the lexicon, making ML estimation of HMM parameters straightforward, and allowing the factorization required for the weighting of the cost function components seen in Equation 4. When using a hierarchical lexicon, also the references *within* the lexicon must be taken into account, making this approach to semi-supervised learning inapplicable.

Moreover, for a flat lexicon, the cost function divides into two parts that have opposing optima: the cost of the data (the likelihood) is optimal when there is minimal splitting and the lexicon consists of the words in the training data, whereas the cost of the model (the prior) is optimal when the lexicon is minimal and consists only of the letters. In consequence, the balance of precision and recall of the segmentation boundaries can be directly controlled by weighting the data likelihood using the hyper-parameters. Tuning these hyper-parameters is a very simple form of supervision, but it has drastic effects on the segmentation results (Kohonen et al., 2010). A direct control of the balance may also be useful for some applications: Grönroos et al. (2015b) used this method to tune segmentation for machine translation.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

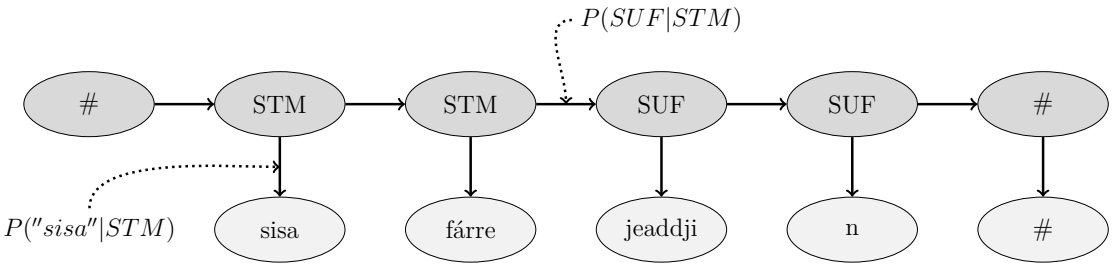

Figure 1: A graph representation of the Hidden Markov model morphotactics, applied to the example word *sisafárrejeaddjin*. The word boundary symbol is marked #. One transition and one emission probability are indicated.

Figure 1 illustrates the hidden Markov model (HMM) used for modeling word formation. The HMM has morph categories as hidden states and morphs as observations. Each morph token is categorized as prefix (PRE), stem (STM), or suffix (SUF). Internally to the algorithm, a non-morph (NON) category is used, intended to model frequent substrings that are not morphs but fragments of a morph. HMM morphotactics were previously used in the Categories-ML (Creutz and Lagus, 2004) and Cat-MAP variants of Morfessor, but Morfessor FlatCat is the first method to combine the approach with semi-supervised training.

In order to calculate the emission probability of a morph conditioned on the morph category, $P(m_i \,|\, c_i)$, the prior of Morfessor FlatCat 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. Morphs with unpredictable right or left contexts are more likely to be prefixes or suffixes, respectively. Longer morphs are more likely to be stems. The perplexities and length in characters are turned into probabilities, by applying a sigmoidal soft thresholding followed by normalization.

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, e.g. in **s + wing*. Modeling of morphotactics also improves the segmentation of compound words, by allowing the overall level of segmentation to be increased without increasing over-segmentation of stems. The presence of morph categories in the output makes it simple to use the method as a stemmer by removing affixes and retaining only stems. The main benefits of semi-supervised learning are in the modeling of suffixation. As the class of suffixes is closed and has high frequency, a good coverage can be achieved with a relatively small set of annotations, compared to the open morph classes such as compound parts. (Grönroos et al., 2014)

The model parameters $\boldsymbol{\theta}$ are optimized utilizing a greedy local search. In each step, a particular subset of the boundaries is reanalyzed and the model parameters updated.

Morfessor FlatCat is initialized using the segmentation from the 2.0 version (Virpioja et al., 2013) of Morfessor Baseline (Creutz and Lagus, 2002, 2007). It employs a morph lexicon $P(m \,|\, \boldsymbol{\theta})$ that is simply a categorical distribution over morphs m, in other words a unigram model.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

6 Active Learning

Data annotation is often performed for the specific goal of improving the performance of a particular system on a task. This gives the opportunity to carefully select the data that will be annotated, in order to maximize the effect and minimize the cost of annotations. This annotation process with systematic (active) data selection is called *active learning*. Many algorithms and methods exist for active learning, but they are not all equally suitable in every situation.

Active learning methods can be divided into three frameworks: pool-based active learning, (membership) query synthesis and stream-based selective sampling (Settles, 2009).

In *pool-based active learning* (Lewis and Gale, 1994), 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. Pool-based active learning can be performed either on-line by selecting one sample in each iteration, or as a batch algorithm by selecting a list of samples at once, before updating the information available to the learner. Pool-based active learning has been successfully applied in NLP (McCallumzy and Nigamy, 1998).

Pool-based active learning can be contrasted with *query synthesis*, in which the learner generates samples to annotate de novo, instead of selecting from a pool of candidates. These methods are difficult to apply to morphological segmentation, due to the challenge of generating valid surface forms.

The third category, *stream-based selective sampling*, is similar to pool-based active learning in that there is a pool of potential samples. In this framework, the samples come in one by one, and the learner has to decide in an on-line fashion whether to query an annotation for that sample or not.

In this work we apply pool-based active learning. Therefore we define the active learning procedure as follows:

In each iteration of active learning, a query strategy is applied for selecting the next samples to elicit and add to the annotated data. The query strategy has access to four sources of information that can be used for guiding the decision at time t:

1. the training pool $\mathcal{A}$,
2. the set of unannotated data $\boldsymbol{D}$,
3. the current set of annotated data $\boldsymbol{A}^{<1\ldots t>}$,
4. and the current best model trained with all training samples collected up to that point $\mathcal{M}^{<t>}$.

$$A^{<t+1>} = \textsc{Strategy}(\mathcal{A}, \boldsymbol{D}, \boldsymbol{A}^{<1\ldots t>}, \mathcal{M}^{<t>}) \tag{5}$$

In this work we make a distinction between the training pool $\mathcal{A}$, and the entire unannotated data $\boldsymbol{D}$, even though they are often chosen to be the same set.

More general reviews of active learning have been written by Settles (2009) and Guyon et al. (2011).

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

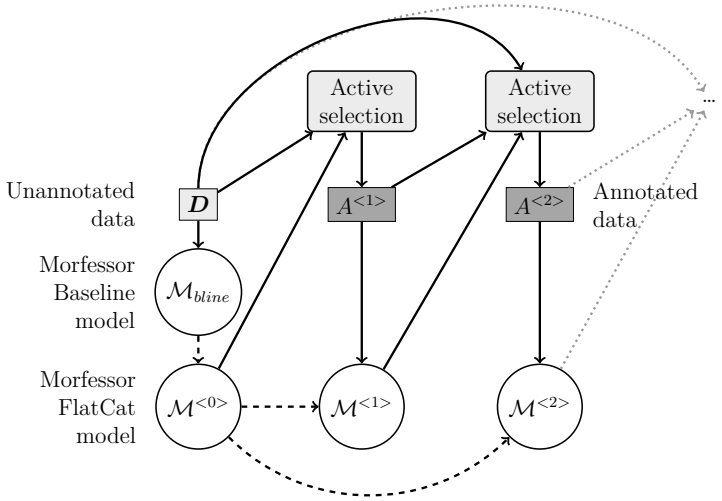

Figure 2: The two first iterations of the active learning procedure applied in this work. Dashed lines indicate the initialization of models. The dotted lines indicate that the procedure can be repeated for additional iterations.

6.1 Active Learning Applied to Morphological Segmentation

Active learning methods have been applied for constructing FST-based analyzers by eliciting new rules from a user with linguistic expertise (Oflazer et al., 2001; Bosch et al., 2008). These development efforts are fast for rule-based systems, but still require months of work.

In the case of morphological segmentation, we try to assess the value of adding the gold standard segmentation of new words into the annotated data set. The methods have access to a list of the n current best segmentations $Z^{<t>}_{i,(1)} \dots Z^{<t>}_{i,(n)}$ for each word w_i, together with their likelihoods given the current model.

Figure 2 shows our active learning procedure, which 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:

1. active selection of new words to annotate using the query strategy,
2. elicitation of annotations for the selected words, and
3. training of the new segmentation model using all available training data.

7 Query Strategies

Active learning requires a specific method for ranking the samples according to their informativeness. Finding the true informativeness of a sample would require looping over all samples in the pool, eliciting an annotation for the sample and training a new model

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

with only that sample added. As the cost of finding the true informativeness would completely negate the benefits, we need a surrogate objective function that is feasible to optimize. This surrogate objective together with the method for optimizing it is called the *query strategy*. The ranking of the training pool according to this query strategy can then be used for selecting data.

Query strategies fall into two broad categories: strategies that primarily use the previously trained model for the task at hand in order to estimate the objective function, and strategies that define the surrogate objective function separately, by directly modeling the properties of the training set.

In the following section we describe a set of query strategies that are applicable to active learning using Morfessor.

7.1 Uncertainty Sampling

Lewis and Gale (1994) introduced *uncertainty* sampling, which is one of the most commonly used methods (Settles, 2009; Guyon et al., 2011). It was used for the NLP tasks of document classification by Lewis and Catlett (1994), and parsing and information extraction by Thompson et al. (1999).

Uncertainty sampling uses the model's estimate of the uncertainty of the decision associated with a particular sample in order to select the additional samples to annotate. The certainty is given by the likelihood of the current best segmentation, compared to all alternative segmentations.

The next word to annotate $A^{<t+1>}$ at time step t is selected from $\mathcal{A}$ based on the uncertainty of the current best segmentation $Z^{<t>}_{i,(1)}$ for each word w_i

$$\begin{aligned} A^{<t+1>} &= \arg\max_{w_i \in \mathcal{A}} \left[1 - P(Z^{<t>}_{i,(1)} \mid w_i; \boldsymbol{\theta}^{<t>})\right] \\ &= \arg\min_{w_i \in \mathcal{A}} \frac{P(Z^{<t>}_{i,(1)}, w_i \mid \boldsymbol{\theta}^{<t>})}{P(w_i \mid \boldsymbol{\theta}^{<t>})}, \end{aligned} \tag{6}$$

where the likelihood of the word with the current best segmentation $P(Z^{<t>}_{i,(1)}, w_i \mid \boldsymbol{\theta}^{<t>})$ is given by the Viterbi algorithm (Viterbi, 1967) and the likelihood of the word with any segmentation $P(w_i \mid \boldsymbol{\theta}^{<t>})$ is given by the forward algorithm (Baum, 1972).

7.2 Margin Sampling

While uncertainty sampling compares the probability of the current best segmentation to all alternative segmentations, margin sampling (Scheffer et al., 2001) only compares to the second best alternative segmentation. The distance to the runner up is called the *margin*. If the margin is large, the model is certain about the segmentation. Therefore, the word with the smallest margin is selected.

$$\begin{aligned} A^{<t+1>} &= \arg\min_{w_i \in \mathcal{A}} \left[P(Z^{<t>}_{i,(1)} \mid w_i; \boldsymbol{\theta}^{<t>}) - P(Z^{<t>}_{i,(2)} \mid w_i; \boldsymbol{\theta}^{<t>})\right] \\ &= \arg\min_{w_i \in \mathcal{A}} \frac{P(Z^{<t>}_{i,(1)}, w_i \mid \boldsymbol{\theta}^{<t>}) - P(Z^{<t>}_{i,(2)}, w_i \mid \boldsymbol{\theta}^{<t>})}{P(w_i \mid \boldsymbol{\theta}^{<t>})} \end{aligned} \tag{7}$$

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

7.3 Query-by-Committee by Bracketing the Corpus Weight

In the *query-by-committee* (QBC) algorithm (Seung et al., 1992; Freund et al., 1997), a committee of predictors independently give their prediction for each sample. The samples that cause most disagreement among the committee members are considered most informative to annotate.

In this experiment the committee consists of two Morfessor FlatCat models, trained with the corpus coding weight hyper-parameter α set to values 10% above and below the optimal value. The reasoning is that the uncertainty about segmentations that are sensitive to a small shift in α is steering the hyper-parameter optimization. Annotating some of these words may allow the global benefits of a slightly different α without introducing errors in words containing the particular morphs in these annotations.

The algorithm filters the words in the training pool, leaving only the words that were segmented differently by the two models in the committee.

$$\mathcal{A}' = \{w_i \in \mathcal{A} : \mathcal{M}_1^{<t>}(w_i) \neq \mathcal{M}_2^{<t>}(w_i)\} \tag{8}$$

In order to select a particular word from the set of filtered words, we pick the one with largest sum of likelihoods given by the two models

$$A^{<t+1>} = \underset{w_i \in \mathcal{A}'}{\arg\max} \left[P(Z_{i,(1)}^{<t>} \mid \boldsymbol{\theta}_1^{<t>}) + P(Z_{i,(1)}^{<t>} \mid \boldsymbol{\theta}_2^{<t>}) \right]. \tag{9}$$

This selects a word that has high likelihood under both models, but that the models still disagree on.

7.4 Coverage of Initial/Final Substrings

The *Initial/final substrings* query strategy is inspired by the feature selection method called *coverage* by Druck et al. (2009), which aims to select features that are dissimilar from existing labeled features, increasing the labeled features' coverage of the feature space.

The method aims to select samples representative of the whole data distribution, instead of querying uncertain samples under the current model, which are likely to contain outliers and exceptional cases.

We apply the idea of coverage to selection of samples to annotate, by defining binary features for the words, and then selecting words so that the features present in them maximize coverage. Our active learning selection differs from the feature selection in that only one sample is needed to cover a feature, instead of labeling all samples with that feature.

We define the features to be substrings starting from the left edge (initial) or ending at the right edge (final) of the word. The length of substrings is limited to between 2 and 5 characters. Let $\Omega(w_i)$ be the set of such substring features in word w_i.

When ranking the words, points are awarded for each substring s present in the ranked word, unless that substring already occurs in the previously selected words. This can be written as the maximization

$$A^{<t+1>} = \underset{w_i \in \mathcal{A}}{\arg\max} \sum_{s \in \Omega(w_i)} \mathrm{I}\left(s \notin \Omega(A^{<j>}) \; \forall j \in \{1 \ldots t\}\right) \frac{\#(s)}{N_{|s|}} \tag{10}$$

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

where I is the indicator function, and $\#(s)$ the occurrence count of feature s. Dividing by

$$N_k = \frac{\sum_s \mathrm{I}(|s| = k)\,\#(s)}{\sum_s \mathrm{I}(|s| = k)} \tag{11}$$

normalizes the occurrence counts by the average occurrence count for substrings of the same length.

This query strategy differs from the other compared methods in that it does not use the Morfessor model when selecting words. However, it can be considered an active selection strategy, as it does define a surrogate objective to systematically take into account the available data $\mathcal{A}$ and the previous selections $\boldsymbol{A}^{<1\ldots t>}$. A benefit of this strategy is that the user does not have to interleave elicitation and Morfessor training. A large list of words can be selected in advance.

7.5 Words without Stem

No stem is a query strategy specific to Morfessor FlatCat. It uses the morph category tags in the current best analysis, to filter a smaller set of potential words from the pool.

Only words for which the current analysis does not contain any morph categorized as STM are kept. This finds stems that are not yet included in the lexicon, and therefore have been over-segmented into NON:s. This improves the coverage of the morph lexicon.

The uncertainty measure is used for selecting individual words from the filtered set.

7.6 Consequent Non-morphemes/Suffixes

Consequent NON/SUF is another strategy specific to Morfessor FlatCat. It is similar to the *No stem* strategy, filtering words to only the words with two or more consecutive morphs categorized as NON or SUF. This strategy is designed to improve suffix chains, in addition to finding over-segmented stems.

7.7 Representative Sampling

Xu et al. (2003) introduce *representative sampling* (RS), that selects samples which are dissimilar to each other, in order to give a good coverage of the dataset.

Selecting dissimilar samples is of particular importance when selection and training is done in batches instead of on-line. An on-line algorithm updates the uncertainty after each sample, making it less likely to select redundant words than a batch algorithm.

We apply representative sampling by clustering the 500 top ranked words for the *Uncertainty* and *QBC* strategies. We cluster the words using k-medoids, with k set to 50. Levenshtein distance (Levenshtein, 1966) is used as the string edit distance. The clustering is repeated 10 times, and the clustering with the smallest intra-cluster variation is selected. The final selection consists of the 50 cluster medoid words.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

8 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* (see, e.g., Virpioja et al., 2011). 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).

$$\text{Precision} = \frac{\#(\text{correct})}{\#(\text{proposed})}; \qquad \text{Recall} = \frac{\#(\text{correct})}{\#(\text{reference})} \tag{12}$$

Precision and recall are calculated using macro-averages over the words in the evaluation set. In the case that a 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.

In addition to the annotated data, we can consider the analysis produced by the North Sámi morphological analyzer from Giellatekno as a secondary gold standard. However, comparing a morphological segmentation to a morphological tagging is not trivial. First, tagging provides abundant information not present in the surface forms. Second, even for tags that have an approximately corresponding morph in the word form, the mapping between the tags and morphs is unknown and must be inferred.

Virpioja et al. (2011) describe several methods for the latter problem. We did preliminary tests with the CoMMA-B1 score that is based on the co-occurrence of the morphemes between the word forms. From the Giellatekno analyses, we split the word forms according to the marked compound boundaries and selected a subset of tags related to inflections and derivations. Then we ran CoMMA-B1 using the annotated test set words as predictions and the modified Giellatekno analyses as a gold standard. This provided precision 0.818, recall 0.155, and F_1-score 0.261. While the scores are also affected by the annotation decisions explained in Section 4, especially the low recall demonstrates that evaluating morphological segmentation based on morphological tagging is problematic.

9 Experiment I: Comparison of Query Strategies

For this experiment, we simulate an annotator using 500 000 segmented word types sampled from the Morpho Challenge 2007 (Kurimo et al., 2007) Finnish data set. This set was analyzed using the two-level morphology analyzer FINTWOL by Lingsoft, Inc., after which the analysis was mapped from the morpheme tags to surface forms of morphemes. This mapping is nontrivial due to the abundance of morphological tags with no surface representation, fusional morphemes, and allomorphy. The applied mapping is described by Creutz and Lindén (2004).

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

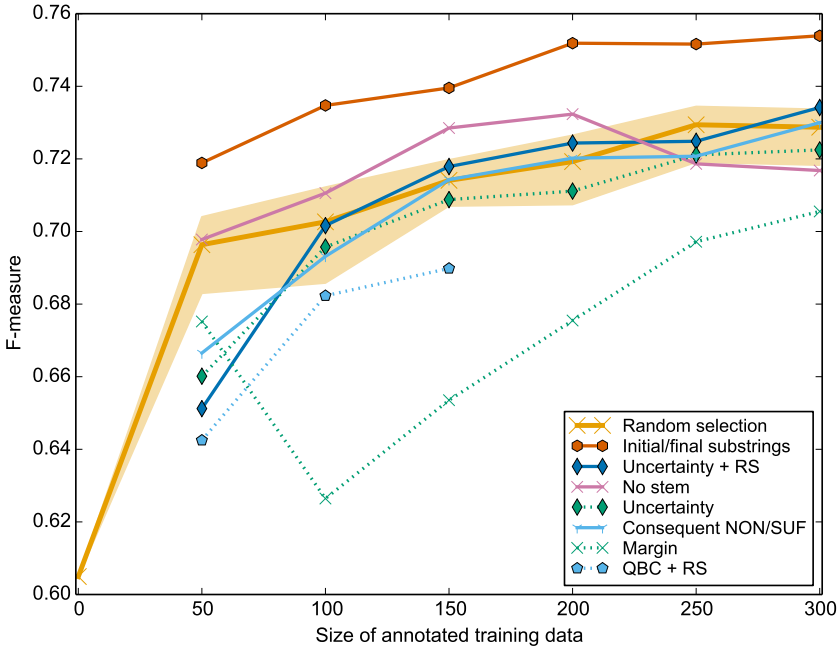

Figure 3: Comparison of different query strategies. The y-axis shows the performance evaluated using F_1-score for the Finnish test set, for models trained using varying amounts of annotated data selected using the query strategy. The thick orange crossed line shows the average of 5 random selections, while the shaded area shows the maximum and minimum.

For hyper-parameter optimization, we used the 835 words in the Morpho Challenge 2010 (Kurimo et al., 2010) development set, and for evaluation the 224 939 words in the corresponding test set.

We simulated an active learning setup using the large annotated data set, by applying the query strategies, and then constructing annotated training sets of the selected words with their annotations. The query strategies did not have access to the annotations of the complete data set.

Regarding the hyper-parameters of Morfessor FlatCat, the corpus likelihood weight α was set by grid search for each selection and iteration individually. In order to considerably decrease the amount of computation, the value for the annotation likelihood weight β was set using a heuristic formula optimized for Finnish:[5]

$$\log \beta = 1.9 + 0.8 \log |\boldsymbol{D}| - 0.6 \log |\boldsymbol{A}|, \tag{13}$$

where $|\boldsymbol{D}|$ and $|\boldsymbol{A}|$ are the numbers of word types in the unannotated and annotated training data sets, respectively. Although it is not guaranteed to be optimal, using the same heuristic value for all query strategies is not expected to favor any particular strategy. The perplexity threshold was set to 75.

[5]The formula is based on work currently being prepared for publication by the present authors.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

Table 1: Sizes of the unannotated corpora used in Experiment II, and the initial division into subsets.

Corpus	**Word tokens**	**Word types**
Den samiske tekstbanken	17 985 140	691 190
UIT-SME-TTS	42 150	8194
Development set	–	200
Evaluation pool	–	800
Training pool $\mathcal{A}$	–	7194

9.1 Results

Figure 3 shows the F_1-score for different query strategies with increasing amounts of annotations. The random selection baseline is averaged over 5 runs.

The only query strategy that consistently performs better than random selection is *Initial/final substrings*. It appears to plateau after 200 annotated words. Inspection of the selected words reveals an assortment of words with common suffixes and compound modifiers.

The *No stem* strategy initially shows strong performance, but falls below random selection when 250 or more words are annotated.

For the *Uncertainty* strategy, applying the *Representative sampling* improves performance, but it should be noted that when only 50 words have been selected, it performs worse than random selection. These first selected words appear to contain many outliers.

Margin sampling does not perform well when used with Morfessor FlatCat. Some selected words have a small margin due to small differences in the category tagging of morphs, which does not even affect the segmentation. Other words are outlier non-words, with several low-probability segmentation alternatives. Margin sampling would also benefit from applying the representative sampling, as it tends to select many words that are similar to each other.

For the Query-by-Committee (QBC) strategy, only the best results which included representative sampling (RS) are plotted. The method performed worse than random selection, and was discontinued after 3 iterations. The selections of this strategy consisted entirely of compound words, with much redundancy in compound parts despite the representative sampling.

Based on these results, *Initial/final substrings* was selected as the main query strategy for the North Sámi experiment. *Uncertainty+RS* was also included, due to its popularity in the literature, and receiving the second highest score at 300 annotated words.

10 Experiment II: Active Learning for North Sámi

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[6] was only used as source

[6]Provided by UiT, The Arctic University of Norway.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

for a word list, to use as the unannotated training data. It contains texts of six genres: administrative, bible, facta, fiction, laws and news.

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. Selecting sentences from *Tekstbanken* would have precluded release, as the restrictive license of the *Tekstbanken* corpus does not allow republication. 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 unavailable due to restrictive licensing. A similar scenario would be selection from a specific target domain corpus.

In contrast to our preliminary work (Grönroos et al., 2015a) we used Morfessor FlatCat during the entire experiment. We used Morfessor Baseline only as initialization method for the initial Morfessor FlatCat model. FlatCat models in later iterations were initialized from the unsupervised FlatCat model, as shown in Figure 2.

As prefixes are very rare in North Sámi, and none were seen in the annotations, we disabled the prefix category by setting an extremely high perplexity threshold for prefixes.

In contrast to the Finnish used in Experiment I, we did not have a heuristic formula for β similar to Equation 13 that would be suitable for North Sámi. However, as we had a smaller number of compared methods, we could set all three hyper-parameters (corpus likelihood weight α, annotation likelihood weight β, perplexity threshold for suffixes) by a grid search for each selection and iteration individually.

10.1 Elicitation of Annotations

In this section, we describe the tool used for elicitation during this experiment, and the resulting data set. For discussion on the challenges of annotating North Sámi for morphological segmentation and our responses to them, see Section 4.

There are no efficient on-line training algorithms for Morfessor FlatCat. Thus we used a batch procedure, by selecting a list of 50 new words to annotate with the query strategy being evaluated, and re-trained Morfessor once the whole list had been annotated.

As the *Initial/final substrings* query strategy does not depend on the Morfessor model during active selection, it was possible to evaluate in a single iteration selecting and annotating the full list of 300 words. Subsets were also evaluated, to show the effect of varying the size of annotations.

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[7] as a back-end. For words in the training pool, the interface shows the segmentation of the current model as a suggestion to the annotator. Words in the development and evaluation pools are shown unsegmented, in order not to bias the annotator.

The tool gives the option of providing a distinct segmentation for word tokens with the same surface form, depending on the sentence context. Even word forms belonging to

[7] `http://bottlepy.org/`

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

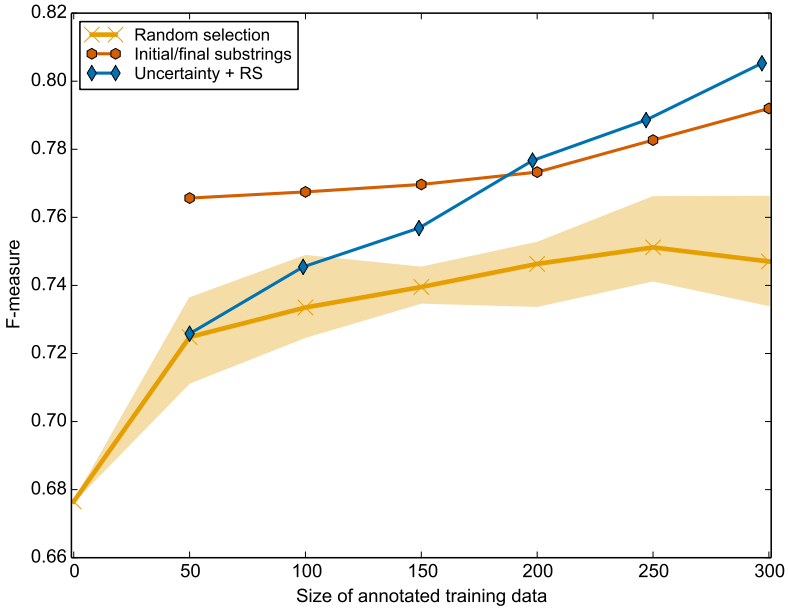

Figure 4: Evaluation using F_1-score for the North Sámi test set, for models trained using varying amounts of annotated data selected using the two selected query strategies. The thick orange crossed line shows the average of 5 random selections, while the shaded area shows the maximum and minimum.

different parts of speech could be homonymous through inflection, and therefore require phrasal context to disambiguate. For example *vearrái* would be unsegmented if it occurs as an adjective ("mean, evil"), but would be segmented *vearrá + i* if it occurs as the illative of the noun *vearri* ("mistake, wrongdoing").

In some rare cases there was no phrasal context provided with the word to be segmented, making it impossible to disambiguate between possible alternative segmentations. This could be caused by isolated words in the corpus, or by mistakes in the automatic tokenization. In these cases, the annotator had to make a judgment call on how to disambiguate the word token.

The annotations were produced by a single Sámi scholar, who is not a native speaker of Sámi. In total 2311 annotated words were collected, divided into 1493 randomly selected word types and 818 actively selected word types. The total time spent by the annotator was 32 hours.[8] A second non-native Sámi speaking linguist independently reannotated 815 of the same words. The principles for segmentation of ambiguous words were discussed prior to the reannotation, but the work itself was independent. Comparing the placement of morph boundaries in the annotations using Cohen's kappa (Cohen, 1960) results in an inter-annotator agreement of 0.82 ("almost perfect agreement").

[8]Includes time spent during the preliminary experiments. Breaks longer than 30 minutes are omitted.

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

Table 2: The model parameters, number of annotated words, and North Sámi test set BPR, for models trained in each iteration of Experiment II. For random selection the averages over all repetitions are shown. Note that the size of the annotated data set may be less than the number of selected words, if non-words were selected.

		Hyper-parameters			**Full test set**		
Model	$\|\boldsymbol{A}\|$	α	β	ppl-thresh	Pre	Rec	F_1
Unsupervised	0	0.4	–	20	0.726	0.633	0.677
Random selection	50	0.48	18000	40	0.725	0.725	0.725
Initial/final substrings	50	0.4	15000	40	**0.733**	**0.802**	**0.766**
Uncertainty + RS	50	0.3	21000	25	0.687	0.769	0.726
Random selection	100	1.02	18000	40	0.746	0.721	0.734
Initial/final substrings	100	1.5	23000	40	**0.765**	**0.769**	**0.767**
Uncertainty + RS	99	0.8	16000	30	0.732	0.760	0.745
Random selection	150	1.30	15000	40	0.754	0.726	0.740
Initial/final substrings	150	1.7	19000	40	**0.774**	**0.766**	**0.770**
Uncertainty + RS	149	1.4	15000	60	0.757	0.757	0.757
Random selection	200	1.32	16000	40	0.750	0.743	0.746
Initial/final substrings	200	1.9	18000	40	0.767	**0.780**	0.773
Uncertainty + RS	198	1.7	14000	70	**0.776**	0.778	**0.777**
Random selection	250	1.56	14000	40	0.760	0.743	0.751
Initial/final substrings	250	1.7	16000	50	0.766	0.800	0.783
Uncertainty + RS	247	1.5	15000	80	**0.768**	**0.811**	**0.789**
Random selection	300	1.68	10000	40	0.763	0.732	0.747
Initial/final substrings	300	1.4	14000	40	0.767	0.819	0.792
Uncertainty + RS	297	1.5	14000	80	**0.772**	**0.842**	**0.805**

10.2 Results

Figure 4 shows the improvement of the F_1-score as more annotations became available. The random selection baseline was averaged over 5 repetitions.

As in Experiment I, the *Initial/final substrings* strategy performs consistently better than random selection. In contrast to that experiment, its performance does not stagnate, but accelerates in the last two iterations.

The results for the *Uncertainty + Representative sampling* strategy differ in several ways from Experiment I. While performance at 50 words is again weak, it is at no iteration worse than random selection. Performance increases rapidly, with Uncertainty + RS surpassing Initial/final substrings when 200 words have been selected.

Table 2 shows the models trained in this experiment. For the full test set, we improve the F_1-score by 18.9% compared to unsupervised learning, with most of the improvement coming from an increase in recall. There is also a small increase in precision. Compared to random selection, the increase in F_1-score is 7.8%.

The values for the hyper-parameters are also shown in Table 2. The optimal value for the corpus likelihood weight α is different for unsupervised and semi-supervised training,

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

Table 3: Boundary precision (Pre), recall (Rec), and F_1-scores for different subsets of the evaluation data.

		STM	**STM+STM**			**STM+SUF**			**STM+SUF+SUF**		
Words in subset		228	55			335			65		
Model	$\|\boldsymbol{A}\|$	Pre	Pre	Rec	F_1	Pre	Rec	F_1	Pre	Rec	F_1
Unsupervised	0	.697	.897	.836	.866	.664	.427	.520	.715	.369	.487
Random selection	50	.648	.869	.848	.859	.696	.587	.637	.712	.433	.538
Initial/final substr.	50	.645	.827	**.909**	.866	.717	.716	.717	.733	.500	.595
Uncertainty + RS	50	.579	.820	.891	.854	.665	.654	.659	.751	.477	.583
Random selection	300	**.705**	**.899**	.807	.851	.739	.608	.667	.711	.477	.571
Initial/final substr.	300	.675	.842	.855	.848	.774	.743	.759	.715	.569	.634
Uncertainty + RS	297	.667	.867	.873	**.870**	**.777**	**.779**	**.778**	**.769**	**.638**	**.698**

with the change happening between 50 and 100 words. The same phenomenon could be seen in Experiment I. Different local optima of α seem to be dominant, depending on the influence of the annotations. Despite the decrease in overall segmentation caused by this increase in α, the semi-supervised models segment ca 15% more than the unsupervised model.

Statistical significance testing was performed using the Wilcoxon signed-rank test ($p < 0.01$). The difference between *Initial/final substrings* and random selection was shown to be statistically significant for all sizes of annotated data. The difference between *Uncertainty + RS* and random selection was only significant with 200 annotated words or more. The difference between the two active selection strategies was only significant at 50 annotated words.

Table 3 shows scores for different categories of words, defined using patterns of morph categories. The selected patterns include all patterns with two morphs or less. For these patterns, precision and recall have a straightforward interpretation. The STM+SUF+SUF pattern was included to shed light on the handling of the boundary between two suffixes. The selected patterns cover 86% of the words in the test set.

When comparing to unsupervised learning, all three forms of semi-supervised learning give better results for suffixation (STM+SUF and STM+SUF+SUF), already with just 50 annotated words. The score for words without internal structure (STM pattern) is only improved when selecting 300 words randomly. For both suffix patterns, active selection is superior to random selection, especially in recall. However, recall for the STM+SUF+SUF remains low for all compared systems. The boundary between two suffixes is the most difficult for Morfessor to place correctly (Ruokolainen et al., 2016).

Random selection gives the best precision for compound words (STM+STM), but has low recall also for this pattern.

After excluding the STM pattern, the best performing method is unambiguous for a particular number of annotations. If only 50 annotated words are used, the best performance for all remaining patterns is given by the *Initial/final substrings* strategy. With 300 annotated words, the best performing strategy is *Uncertainty + RS*.

Initial/final substrings assumes that one sample is enough to cover a feature. In other

Northern European Journal of Language Technology, 2016, Vol. 4, Article 4, pp 47–72
DOI 10.3384/nejlt.2000-1533.1644

words, it assumes that every word beginning or ending with a particular substring will equally well teach the model how to segment other words with the same substring. In practice this assumption does not hold, e.g. *seammaláhkái* ("by the same means") is segmented *seamma + láhkái*, while *govvadahkkái* ("to the picture maker") is segmented *govva + dahkká + i*. Both words end in *kái*, but it is segmented differently. The query strategy is to some extent able to compensate by using longer substrings, and in practice does not seem to make too many detrimental selections.

11 Conclusions

We have applied an active learning approach to modeling morphological segmentation of two Uralic languages: Finnish and North Sámi. The work was accomplished using open-source software.[9] We present the collected language resources for the use of the scientific community.[10]

We performed two experiments. In the first experiment, we compared seven different query strategies using Finnish gold standard segmentations to simulate an annotator. In the second experiment, we applied the active learning system to collect a set of human-annotated data for North Sámi.

In both of the experiments, the *Initial/final substrings* query strategy performed better than random selection regardless of the size of the annotated data set. In the Finnish language experiment, it is clearly the best method.

The performance of the segmentation model was shown to increase rapidly as the amount of human-annotated data was increased. With 300 annotated North Sámi words, collected using the *Uncertainty + Representative sampling* query strategy, F_1-score was improved by 19% (relative) compared to unsupervised learning and 7.8% (relative) compared to random selection. The increase was consistent over several sets of words with different morphological patterns. The largest benefit of the annotations was in the modeling of suffixation.

The results of the two experiments differ with regard to the performance of the *Uncertainty + RS* query strategy. In the last iterations of the North Sámi experiment, it outperforms Initial/final substrings, even though the difference is not statistically significant. The different outcomes may be caused by real differences between the morphology of the languages, or the properties of the data sets. However, the difference could also be an artifact caused by either the procedure of simulating an annotator or the heuristic hyper-parameter values used in Experiment I.

If the proposed method is applied to a new language, the minimum amount of training set words to annotate should be around 100, in addition to the development set needed for the hyper-parameter optimization. The transition of the value of the hyper-parameter α from the local optimum of unsupervised training to the optimum of semi-supervised training has not yet occurred at 50 annotated words. Additionally, with only 50 annotated words, *Uncertainty + RS* does not yet outperform random selection. If a very small number of words (100–200) are collected, we recommend using the *Initial/final substrings* query strategy. As the number of annotations grows larger, active selection is

[9]Morfessor is available at `http://www.cis.hut.fi/projects/morpho/`.
The annotation and active learning tool is available at `https://github.com/Waino/morphsegannot/`.

[10]The data is available at `http://research.spa.aalto.fi/speech/data_release/north_saami_active_learning/`.

still preferable over random selection, but the choice of specific query strategy may be less important.

The Initial/final substrings query strategy does not apply the current segmentation model when making selections, even though incorporating information also from this source might be useful. Hybrid strategies that combine or switch between multiple query strategies were not explored in this work. Another avenue for future work is the exploration of different string similarity metrics for the representative sampling, as the Levenshtein edit distance used in this work may not yield optimal clusters.

Acknowledgments

This research has been supported by EC's Seventh Framework Programme (grant n°287678, Simple4All) and the Academy of Finland under the Finnish Centre of Excellence Program 2012–2017 (grant n°251170, COIN), LASTU Programme (grants n°256887 and 259934) and the project Fenno-Ugric Digital Citizens (grant n°270082). Computer resources within the Aalto University School of Science "Science-IT" project were used.

References

Aikio, Ante. 2005. Pohjoissaamen alkeiskurssi. Lecture material.

Baum, Leonard E. 1972. An inequality and an associated maximization technique in statistical estimation of probabilistic functions of a Markov process. *Inequalities* 3(1):1–8.

Bosch, Sonja E, Laurette Pretorius, Kholisa Podile, and Axel Fleisch. 2008. Experimental fast-tracking of morphological analysers for Nguni languages. In *LREC*.

Cohen, Jacob. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* 20(1):37–46.

Creutz, Mathias, Teemu Hirsimäki, Mikko Kurimo, Antti Puurula, Janne Pylkkönen, Vesa Siivola, Matti Varjokallio, Ebru Arisoy, Murat Saraçlar, and Andreas Stolcke. 2007. Morph-based speech recognition and modeling of out-of-vocabulary words across languages. *ACM Transactions on Speech and Language Processing* 5(1):3:1–3:29.

Creutz, Mathias and Krista Lagus. 2002. Unsupervised discovery of morphemes. In *Proceedings of the Workshop on Morphological and Phonological Learning of ACL'02*, pages 21–30. Philadelphia, Pennsylvania, USA.

Creutz, Mathias and Krista Lagus. 2004. Induction of a simple morphology for highly-inflecting languages. In *Proc. 7th Meeting of the ACL Special Interest Group in Computational Phonology (SIGPHON)*, pages 43–51. Barcelona.

Creutz, Mathias and Krista Lagus. 2005. Inducing the morphological lexicon of a natural language from unannotated text. In *Proceedings of the AKRR'05*. Espoo, Finland.

Creutz, Mathias and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing* 4(1).

Creutz, Mathias and Krister Lindén. 2004. Morpheme segmentation gold standards for Finnish and English. Tech. Rep. A77, Publications in Computer and Information Science, Helsinki University of Technology.

Druck, Gregory, Burr Settles, and Andrew McCallum. 2009. Active learning by labeling features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 81–90. Association for Computational Linguistics.

Fishel, Mark and Harri Kirik. 2010. Linguistically motivated unsupervised segmentation for machine translation. In *LREC*.

Freund, Yoav, H Sebastian Seung, Eli Shamir, and Naftali Tishby. 1997. Selective sampling using the query by committee algorithm. *Machine learning* 28(2-3):133–168.

Grönroos, Stig-Arne, Kristiina Jokinen, Katri Hiovain, Mikko Kurimo, and Sami Virpioja. 2015a. Low-resource active learning of North Sámi morphological segmentation. In *Proceedings of 1st International Workshop in Computational Linguistics for Uralic Languages*, pages 20–33.

Grönroos, Stig-Arne, Sami Virpioja, and Mikko Kurimo. 2015b. Tuning phrase-based segmented translation for a morphologically complex target language. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Lisbon, Portugal: Association for Computational Linguistics.

Grönroos, Stig-Arne, Sami Virpioja, Peter Smit, and Mikko Kurimo. 2014. Morfessor FlatCat: An HMM-based method for unsupervised and semi-supervised learning of morphology. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics*, pages 1177–1185. ACL.

Guyon, Isabelle, Gavin C. Cawley, Gideon Dror, and Vincent Lemaire. 2011. Results of the active learning challenge. In *Active Learning and Experimental Design workshop, In conjunction with AISTATS 2010, Sardinia, Italy, May 16, 2010*, pages 19–45.

Hammarström, Harald and Lars Borin. 2011. Unsupervised learning of morphology. *Computational Linguistics* 37(2):309–350.

Hirsimäki, Teemu, Mathias Creutz, Vesa Siivola, Mikko Kurimo, Sami Virpioja, and Janne Pylkkönen. 2006. Unlimited vocabulary speech recognition with morph language models applied to Finnish. *Computer Speech and Language* 20(4):515–541.

Jokinen, Kristiina. 2014. Open-domain interaction and online content in the sami language. In *Language Resources and Evaluation Conference*, pages 517–522.

Jokinen, Kristiina and Graham Wilcock. 2014a. Community-based resource building and data collection. In *The 4th International Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU'14)*.

Jokinen, Kristiina and Graham Wilcock. 2014b. Multimodal open-domain conversations with the Nao robot. In J. Mariani, S. Rosset, M. Garnier-Rizet, and L. Devillers, eds., *Natural Interaction with Robots, Knowbots and Smartphones: Putting Spoken Dialogue Systems into Practice*, pages 213–224. Springer.

Karlsson, Fred. 1982. *Suomen kielen äänne- ja muotorakenne.* Helsinki: WSOY.

Kohonen, Oskar, Sami Virpioja, and Krista Lagus. 2010. Semi-supervised learning of concatenative morphology. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 78–86. Uppsala, Sweden: Association for Computational Linguistics.

Koskenniemi, Kimmo. 1983. *Two-level morphology: A general computational model for word-form recognition and production.* Ph.D. thesis, University of Helsinki.

Koskenniemi, Kimmo. 2008. How to build an open source morphological parser now. In *Resourceful Language Technology–Festschrift in Honor of Anna Sågvall Hein*, page 86.

Kurimo, Mikko, Mathias Creutz, and Ville Turunen. 2007. Unsupervised morpheme analysis evaluation by IR experiments – Morpho Challenge 2007. In A. Nardi and C. Peters, eds., *Working Notes for the CLEF 2007 Workshop.* CLEF. Invited paper.

Kurimo, Mikko, Sami Virpioja, and Ville T. Turunen. 2010. Overview and results of Morpho Challenge 2010. In *Proceedings of the Morpho Challenge 2010 Workshop*, pages 7–24. Espoo, Finland: Aalto University School of Science and Technology, Department of Information and Computer Science. Technical Report TKK-ICS-R37.

Levenshtein, Vladimir I. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady* 10(8):707–710.

Lewis, David D and Jason Catlett. 1994. Heterogeneous uncertainty sampling for supervised learning. In *Proceedings of the eleventh international conference on machine learning*, pages 148–156.

Lewis, David D and William A Gale. 1994. A sequential algorithm for training text classifiers. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 3–12. Springer-Verlag New York, Inc.

Lindén, Krister, Miikka Silfverberg, and Tommi Pirinen. 2009. Hfst tools for morphology–an efficient open-source package for construction of morphological analyzers. In *State of the Art in Computational Morphology*, pages 28–47. Springer.

McCallumzy, Andrew Kachites and Kamal Nigamy. 1998. Employing EM and pool-based active learning for text classification. In *Machine Learning: Proceedings of the Fifteenth International Conference, ICML.* Citeseer.

Nickel, Klaus Peter and Pekka Sammallahti. 2011. *Nordsamisk grammatikk.* Kárášjohka: Davvi Girji.

Oflazer, Kemal, Sergei Nirenburg, and Marjorie McShane. 2001. Bootstrapping morphological analyzers by combining human elicitation and machine learning. *Computational Linguistics* 27(1):59–85.

Poon, Hoifung, Colin Cherry, and Kristina Toutanova. 2009. Unsupervised morphological segmentation with log-linear models. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 209–217. Association for Computational Linguistics.

Rissanen, Jorma. 1989. *Stochastic Complexity in Statistical Inquiry*, vol. 15. Singapore: World Scientific Series in Computer Science.

Ruokolainen, Teemu, Oskar Kohonen, Kairit Sirts, Stig-Arne Grönroos, Mikko Kurimo, and Sami Virpioja. 2016. A comparative study on minimally supervised morphological segmentation. *Computational Linguistics* .

Ruokolainen, Teemu, Oskar Kohonen, Sami Virpioja, and Mikko Kurimo. 2014. Painless semi-supervised morphological segmentation using conditional random fields. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, pages 84–89. Gothenburg, Sweden: Association for Computational Linguistics.

Sammallahti, Pekka. 1998. *The Saami Languages: An Introduction*. Kárášjohka: Davvi Girji.

Scheffer, Tobias, Christian Decomain, and Stefan Wrobel. 2001. Active hidden markov models for information extraction. In F. Hoffmann, D. Hand, N. Adams, D. Fisher, and G. Guimaraes, eds., *Advances in Intelligent Data Analysis*, vol. 2189 of *Lecture Notes in Computer Science*, pages 309–318. Springer Berlin Heidelberg. ISBN 978-3-540-42581-6.

Settles, Burr. 2009. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin, Madison.

Seung, H Sebastian, Manfred Opper, and Haim Sompolinsky. 1992. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294. ACM.

Sirts, Kairit and Sharon Goldwater. 2013. Minimally-supervised morphological segmentation using adaptor grammars. *TACL* 1:255–266.

Thompson, Cynthia A., Mary Elaine Califf, and Raymond J. Mooney. 1999. Active learning for natural language parsing and information extraction. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML-99)*, pages 406–414. Bled, Slovenia.

Trosterud, Trond and Heli Uibo. 2005. Consonant gradation in Estonian and Sámi: two-level solution. In *Inquiries into Words, Constraints and Contexts–Festschrift for Kimmo Koskenniemi on his 60th Birthday*, page 136. Citeseer.

Tyers, Francis M, Linda Wiechetek, and Trond Trosterud. 2009. Developing prototypes for machine translation between two Sámi languages. In *Proceedings of the 13th Annual Conf. of the EAMT*, pages 120–128.

Virpioja, Sami, Peter Smit, Stig-Arne Grönroos, and Mikko Kurimo. 2013. Morfessor 2.0: Python implementation and extensions for Morfessor Baseline. Report 25/2013 in Aalto University publication series SCIENCE + TECHNOLOGY, Department of Signal Processing and Acoustics, Aalto University, Helsinki, Finland.

Virpioja, Sami, Ville Turunen, Sebastian Spiegler, Oskar Kohonen, and Mikko Kurimo. 2011. Empirical comparison of evaluation methods for unsupervised learning of morphology. *Traitement Automatique des Langues* 52(2):45–90.

Virpioja, Sami, Jaakko J Väyrynen, Mathias Creutz, and Markus Sadeniemi. 2007. Morphology-aware statistical machine translation based on morphs induced in an unsupervised manner. *Machine Translation Summit XI* 2007:491–498.

VISK. 2004. Auli Hakulinen, Maria Vilkuna, Riitta Korhonen, Vesa Koivisto, Tarja Riitta Heinonen and Irja Alho. Iso suomen kielioppi. [Online database, `http://scripta.kotus.fi/visk` referenced 7.10.2016].

Viterbi, A. J. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory* 13(2):260–269.

Wilcock, G., N. Laxström, J. Leinonen, P. Smit, M. Kurimo, and K. Jokinen. 2016. Towards SamiTalk: a Sami-speaking robot linked to Sami Wikipedia. In K. Jokinen and G. Wilcock, eds., *Dialogues with Social Robots: Enablements Analyses, and Evaluation*, pages 301–309. Springer.

Xu, Zhao, Kai Yu, Volker Tresp, Xiaowei Xu, and Jizhi Wang. 2003. Representative sampling for text classification using support vector machines. In F. Sebastiani, ed., *Advances in Information Retrieval*, vol. 2633 of *Lecture Notes in Computer Science*, pages 393–407. Springer Berlin Heidelberg. ISBN 978-3-540-01274-0.

Yarowsky, David and Richard Wicentowski. 2000. Minimally supervised morphological analysis by multimodal alignment. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 207–216. Association for Computational Linguistics.

Álgu-tietokanta. 2006. Kotimaisten kielten tutkimuskeskus. Sámegielaid etymologaš diehtovuođđu. [Online database, `http://kaino.kotus.fi/algu/` referenced 15.8.2015].