Morphologically motivated word classes for very large vocabulary speech recognition of Finnish and Estonian

Varjokallio, Matti

2021-03

Varjokallio , M , Virpioja , S & Kurimo , M 2021 , ' Morphologically motivated word classes for very large vocabulary speech recognition of Finnish and Estonian ' , Computer Speech and Language , vol. 66 , 101141 . https://doi.org/10.1016/j.csl.2020.101141

http://hdl.handle.net/10138/347659
https://doi.org/10.1016/j.csl.2020.101141

Morphologically motivated word classes for very large vocabulary speech recognition of Finnish and Estonian

Matti Varjokallio[a,*], Sami Virpioja[b], Mikko Kurimo[a]

[a] *Department of Signal Processing and Acoustics, School of Electrical Engineering, Aalto University, Espoo, Finland*
[b] *Department of Digital Humanities, Faculty of Arts, University of Helsinki, Finland*

Abstract

We study class-based n-gram and neural network language models for very large vocabulary speech recognition of two morphologically rich languages: Finnish and Estonian. Due to morphological processes such as derivation, inflection and compounding, the models need to be trained with vocabulary sizes of several millions of word types. Class-based language modelling is in this case a powerful approach to alleviate the data sparsity and reduce the computational load. For a very large vocabulary, bigram statistics may not be an optimal way to derive the classes. We thus study utilizing the output of a morphological analyzer to achieve efficient word classes. We show that efficient classes can be learned by refining the morphological classes to smaller equivalence classes using merging, splitting and exchange procedures with suitable constraints. This type of classification can improve the results, particularly when language model training data is not very large. We also extend the previous analyses by rescoring the hypotheses obtained from a very large vocabulary recognizer using class-based neural network language models. We show that despite the fixed vocabulary, carefully constructed classes for word-based language models can in some cases result in lower error rates than subword-based unlimited vocabulary language models.

Keywords: Language modelling, Class-based language models, Morphologically rich languages

*Corresponding author
Email addresses: matti.varjokallio@aalto.fi (Matti Varjokallio), sami.virpioja@helsinki.fi (Sami Virpioja), mikko.kurimo@aalto.fi (Mikko Kurimo)

1. Introduction

The conventional solution for language modelling in large vocabulary continuous speech recognition has for long time been a statistical n-gram model trained over words. The frequency estimates are smoothed to improve robustness and assign probabilities to word sequences that are not present in the training corpus [1]. Even though different neural network approaches for language modelling have been known already for some time [2, 3], they have only become commonplace in recent years. Modern large vocabulary language models need to be trained using large text corpora to achieve reasonable vocabulary coverage and modelling accuracy. The computational cost in training and applying the models has been inhibitory until the development of parallelization using graphical processing units (GPUs). Nevertheless, in most cases a statistical n-gram model is applied on the first recognition pass and neural network language models are used to rescore hypotheses stored in a n-best list or a recognition lattice. For a survey on applying recurrent neural networks (RNNs) to language modelling, see [4].

The problems in training language models are in many ways pronounced for morphologically rich languages. For example Finnish and Estonian, the languages studied in this work, are known to have 26 [5] and 28 [6] grammatical noun cases, respectively. However, due to clitic particles, irregularities and other phenomena, estimating the morphological generativeness of the languages is more complex. Already the non-inflected and non-compounded Finnish nouns can have 150 paradigmatic forms and appear in as many as 2000 different forms [7]. Also for Estonian, the possibility of over 400 noun patterns has been observed in some sources [8]. As forming compound words is common for both the languages, the vocabulary sizes are further increased. For a practical assessment of the morphological generativeness of the languages, we have estimated the type-to-token ratios for some agglutinative and other languages in Figure 1 for the Wikipedia corpus.

Even though the Wikipedia corpora for the different languages are, due to possible cultural and other factors, only approximately comparable, we may still observe that the agglutinative Uralic languages (Finnish, Estonian, Hungarian) and the major Dravidian languages (Tamil, Telugu, Malayalam, Kannada) are among the languages with the highest vocabulary growth rates. For example the Turkic languages and Arabic can also exhibit high vocabulary sizes, but at least as estimated from the Wikipedia corpus, the type-to-token ratios are somewhat lower than for the Uralic and Dravidian languages. We thus expect the models and evaluations in this work to be representative of at least the languages in the Uralic and Dravidian language families, but likely useful for many other languages as well.

For these languages, a very large vocabulary is thus required to achieve a sufficiently small ouf-of-vocabulary (OOV) rate, and even large text corpora are sparse for training accurate n-gram models. Training neural network language models over words is hard and computationally expensive thanks to the data sparsity and the normalization step in the softmax output layer. Approaches such as noise contrastive estimation [9] and hierarchical softmax [10] can be used to speed up training of word-based NNLMs. A recent study on Finnish and Estonian conversational ASR [11] compared these approaches to class-based NNLMs. In that work, the class-based NNLMs were mostly more accurate and faster to train, even though there may naturally be possibilities for improving the NCE and hierarchical softmax -approaches.

For agglutinative languages, building the language models over subword units such as

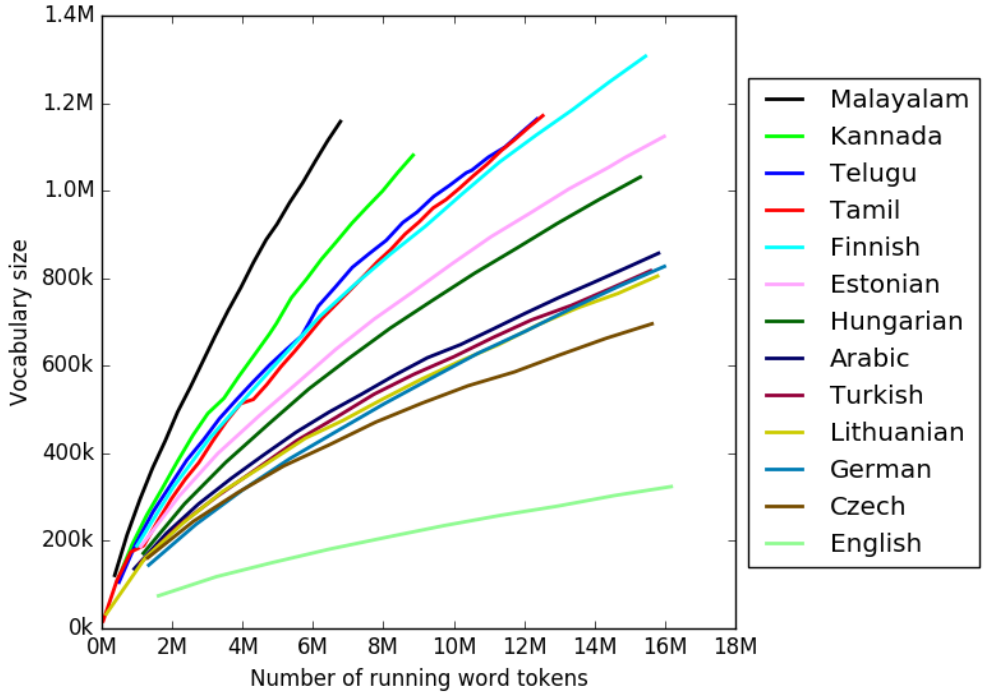

Figure 1: Vocabulary growth rates estimated from Wikipedia articles

statistical morphs has proven to be a solid choice [12, 13, 14]. Consequently, probabilities may be assigned to word forms which are not covered by the training corpus. With subword models, it is possible to opt either for unlimited vocabulary speech recognition [15] or a limited vocabulary that is easy to expand with new word forms [16]. In some cases, subwords also provide better n-gram estimates with the same vocabulary [17].

Studies of Hungarian [18] and Finnish [16] have shown that carefully implemented word-based n-grams can produce competitive error rates compared to the subword approach. This requires an ASR decoder that is capable of effectively handling a vocabulary of millions of word forms and large n-gram models. In addition, a large training corpus is needed for sufficient coverage of word forms and robust n-gram estimates.

The class n-gram model is a traditional approach for alleviating the data sparsity issues [19, 20]. In an early work [21], variable-length category n-grams over part-of-speech tags were trained and evaluated in English speech recognition. Using automatically derived classes and thus increasing the number of classes was found to give larger improvements when interpolated with word n-grams [22]. For languages with more generative morphology, class n-grams trained over automatically derived classes have been found to improve language modelling for Russian [23]. In a study on Lithuanian language modelling [24], up to 13% perplexity reductions were reached by a linear interpolation with a class n-gram using automatically derived classes. In another study on Czech and Slovak

language modelling [25], linear interpolation with morphological class n-grams improved perplexities by around 10% for a large corpus. Larger improvements were reached using a smaller training corpus. The size of the vocabulary has varied in these experiments, the largest being 430,000 words in the Russian language modelling experiments [23] and 1.2 million words in the Lithuanian language modelling experiments [24].

In this work, we study class-based language modelling for Finnish and Estonian speech recognition with very large vocabulary sizes. As the size of the vocabulary grows, the importance of the word clustering methods is expected to increase. Despite the potential of the class based models for the speech recognition of morphologically rich languages, there have been few studies of this topic.

Class n-gram models have been evaluated (for instance) in English speech recognition tasks in [26] and [27]. In the former work, a 2.2% relative improvement in word error rate was obtained in lattice rescoring experiments in a broadcast news task. The vocabulary size in this experiment was 65,000 words. In the latter, a 3.7% relative improvement was observed in the North American business news (NAB) task through lattice rescoring. The vocabulary size in this experiment was 20,000.

Some results on Lithuanian speech recognition have been mentioned in [28]. In these experiments, a larger vocabulary of 1.2 million words was used and the class n-gram interpolation improved the word-error rate by 5.2 percent relative using n-best list rescoring. In addition, the word-based recognizer was compared to a recognizer utilizing a particle (subword) recognition approach. The word-based recognizer outperformed the subword approach with a large margin of 8 percent absolute. This differs from the results reported for Finnish and Estonian speech recognition tasks, in which the subword recognizers have provided state-of-the-art accuracy [12, 29, 15]. Therefore, it is possible that the Lithuanian language is not particularly well suited to subword language modelling approaches in speech recognition.

More recent experiments used a class-based output layer for NNLMs for English language modelling and automatic speech recognition on the Switchboard conversational telephone speech corpus [30]. The word error rate was improved by around 2% absolute compared to the baseline result using Kneser-Ney smoothed 5-gram model. Neural network language models trained over classes have also been evaluated for conversational Finnish and Estonian speech recognition in [11]. In the context of language modelling using the so-called Model M [31], it was shown that optimizing the classes directly for the Model M criterion [32] improved the modelling accuracy in English speech recognition tasks.

In this work, we propose and evaluate a novel approach for training morphologically motivated word classes using open source morphological analyzers for both Finnish [33] and Estonian [34]. We use an expectation-maximization training procedure [35] for training morphological classes and we use the model for tagging words that are not covered by the analyzer lexicon. We further refine the resulting classes by a merging and splitting procedure, followed by exchanges with different morphological constraints. The use of merging and splitting of classes has previously been mentioned in [32] to overcome possible local maxima of the bigram objective function. We are not aware of earlier experiments where the morphological analyzer classes are refined using bigram statistics to achieve more powerful classifications for language modelling. However, the results show that at least some efficient solutions to the classification problem are closely related to further processed part-of-speech equivalence classes. We show that using classes with

morphological constraints improves the perplexities and word error rates, especially in settings with less training data.

Given that Finnish and Estonian are agglutinative languages, the most common language modelling approach for them has been to train the models over statistical morphs or other subword lexical units [12, 29, 15]. The perplexities of subword-based and word-based language models are not typically comparable because of the different OOV rates, so their performance needs to be evaluated in a speech recognition task. Due to our recent improvements in the decoder design [16], we are able to compare subword language models to word-based language models with a very large vocabulary size. An interpolation of a large word-based n-gram model and a class n-gram model is used as a language model in the first recognition pass. To ensure accurate decoding, we use a class bigram model for language model look-ahead during decoding [36].

We also evaluate the accuracy of the recognizers using NNLMs for rescoring n-best lists that were obtained from the decoder in the first recognition pass. For this purpose, either class-based or subword-based NNLMs could be utilized. In our experiments, the subword-based models appear to be more powerful of the two. However, in most cases, interpolating with both types of neural network models improves the results further. The type of word classifications can also have a significant impact on the recognition accuracy with class-based NNLMs. The morphologically motivated classes provide better accuracy compared to classes inferred purely using the exchange algorithm based on bigram statistics, especially with less training data. Compared to rescoring the output of an unlimited vocabulary recognizer, we show that in some cases better results are reached. In this work, the class-based modelling approach is used in all phases of a traditional ASR system: in a look-ahead language model during decoding, in interpolated language model component during the first recognition pass, and in a component model for rescoring the hypotheses with NNLMs.

2. Methods

In this section, we first define the class-based language models that are used in subsection 2.1. Inferring classes using bigram statistics and the exchange algorithm is described in subsection 2.2. The main methodological contribution of the work, a morphologically motivated class inference, is described in subsection 2.3. In the experiments, we compare the class-based models to subword-based language models, which are discussed in subsection 2.4. Decoding the very large vocabularies and language models is discussed in subsection 2.5.

2.1. Class-based Language Models

In this work, we use the following popular type of a class n-gram model [19, 20]:

$$P(w_i|w_{i-(n-1)}^{i-1}) = P(w_i|c_i) \times P(c_i|c_{i-(n-1)}^{i-1}), \quad (1)$$

where the words w are clustered into equivalence classes c. The word history is denoted by $w_{i-(n-1)}^{i-1}$ and the corresponding class history by $c_{i-(n-1)}^{i-1}$. After the classification, the class membership probabilities $P(w_i|c_i)$ and the class n-gram component $P(c_i|c_{i-(n-1)}^{i-1})$ are typically estimated as given by the maximum likelihood estimates:

$$P(w|c) = \frac{f(w)}{\sum_{v \in C(w)} f(v)} \tag{2}$$

$$P(c_i|c_{i-(n-1)}^{i-1}) = \frac{f(c_{i-(n-1)}, .., c_i)}{f(c_{i-(n-1)}, .., c_{i-1})}, \tag{3}$$

where $f(w)$ denotes the frequency of the word w, $C(w)$ the class of the word w, and $f(c_{i-(n-1)}, .., c_i)$ the frequency of a class sequence.

We also evaluate class-based neural network language models. These models utilize long short-term memory (LSTM) layers [2] and highway layers with tanh activations [37]. We train the NNLMs over class sequences and use a class-based output layer. The class membership probabilities are estimated by the formula 2 similarly to the class n-gram models.

2.2. Exchange Algorithm.

The so-called exchange algorithm for forming statistical word classes with bigram statistics was given in [19]:

Algorithm 1: Exchange algorithm

```
compute initial class mapping
sum initial class based counts
compute initial perplexity
repeat
    foreach word w of the vocabulary do
        remove word from its class
        foreach class k do
            tentatively move word w to class k
            compute perplexity for this exchange
        move word w to class k with minimum perplexity
until stopping criterion is met
```

The algorithm operates by iterating over all the words, evaluating all possible class exchanges for each word, and then choosing the exchange that provides the largest improvement for the likelihood. Later work discussed efficient implementations using the word-class and class-word statistics, as well as extension to trigram clustering [27]. While trigram statistics may provide improvements for a small number of classes, they often result in overlearning, and the best performance is normally obtained with bigram clustering [27, 38]. The evaluation step may be parallelized for each word [38].

2.3. Morphologically Motivated Classes

We extend our previous work on class language models [39] by more detailed experiments with morphologically motivated classes. For both of our target languages, there are open source morphological analyzers available. For Finnish, we used the Omorfi [33] package (version `0.4-20190511`) and for Estonian the Estnltk [34] package (version `1.4.1`). The Omorfi analyzer is able to analyze 82 percent of the word types found in

our training data. For Estnltk and our Estonian corpus the coverage was 77 percent. There are some considerations and challenges in utilizing the output of the morphological analyzer in forming word classes for language modelling. First, to prevent the increase of the OOV rate of the language model, word forms that are not recognized by the analyzer need to be tagged. Second, for some surface word forms, the output of the analyzer will contain multiple ambiguous analyses, and a decision needs to be made about whether only one or more lexical entries are included in the model. In many cases, there is a thin line between whether some analyses for a word are really distinct or if they are simply due to the very fine-grained output of the analyzer. In our initial experiments, training language models over fully disambiguated entries resulted in worse speech recognition accuracy compared to the standard approach of adding only one entry per surface word form. Third, the classifications given by the morphological analyzer are not optimal for use with a class-based language model as such. On the one hand, the output is sometimes very fine-grained, which leads to ambiguity and possible data sparsity. On the other hand, the largest classes consist of hundreds of thousands of words and for those the modelling accuracy inevitably suffers.

We thus suggest and evaluate an approach in which we use the morphological analyzer output to initialize a class-based model, which allows multiple classes per word [35, 21]. The model is trained using the expectation-maximization algorithm, as in [35], but we use the distribution marginalized over words, as in [21]. In the final training iterations, the words that are not covered in the morphological analyzer lexicon are tagged. To further refine the classes by a merging and splitting procedure, we also limit the number of classes per word to one. The training procedure makes it possible to "freeze" the words to the most likely class. This limitation does not incur a penalty, at least as evaluated by the model likelihood. This allows us to refine the classes using bigram statistics to reach more powerful classifications for language modelling.

2.3.1. Expectation-maximization Training

We use a generalization of the class n-gram model, which allows the words to belong to more than one class [35, 21]. The model has three types of parameters: class generation probabilities, the n-gram parameters and the class membership probabilities. In this work, we concentrate on the following model:

$$P(w_i|w_{i-(n-1)}^{i-1}) = \sum_j \left(P(w_i|c_{ji}) \times \sum_s \left(P(c_{ji}|s) \times \prod_{k=i-(n-1)}^{i-1} P(c_{sk}|w_k) \right) \right)$$

Here j denotes a possible class for the word w_i and s are the different class sequences generated by the word history $w_{i-(n-1)}^{i-1}$. This model assumes that the classes in the word history are generated independently of the other classes. We believe this to be a reasonable approximation considering that the n-gram term in the model is context-dependent. Moreover, the large vocabulary sizes that we apply in this work would pose challenges for more complex modelling of the class generation.

To train the model, we need a text corpus and a morphological analyzer that has a reasonable coverage for the corpus. The text corpus does not need to be tagged. Because many words have multiple possible analysis, the morphological disambiguation is modelled with alternate classes. A morphological analyzer is not required to disambiguate

the words because it is an inherent part in the class n-gram model training. In the later training iterations, the remaining unanalyzed words may be tagged to the classes. For this purpose, we used the n-gram probabilities to estimate the most probable classes in the sentence context.

In the following, we derive the expectation-maximization algorithm for training the model parameters. The EM-training approach was used in [35], but their task did not marginalize the model over word sequences. In the earlier work on the category n-gram models [21], a different approach was used for training the model. The notation that we use here follows the derivation of the expectation-maximization algorithm for the Hidden Markov models [40]. The EM algorithm starts with some initial selection for the model parameters, which are denoted by $\boldsymbol{\theta}^{old}$. In the E step, these parameter values are used to find the posterior distribution of the latent variables $p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})$. This posterior distribution is then used to evaluate the expectation of the logarithm of the complete-data likelihood function, as a function of the parameters $\boldsymbol{\theta}$, to give the function $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$ defined by:

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old}) \ln p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta}), \tag{4}$$

where $\boldsymbol{X}$ denotes the training observations and $\boldsymbol{Z}$ the latent variables (classes). The summation is over all of the sequences of the latent variables.

To some extent, the derivation of the expectation-maximization training scheme builds on the possibility of writing the log likelihood of one class sequence as a sum of three separate terms, where the model parameters appear separately. By a class sequence, we mean the class sequence for the whole training corpus, where sentence break markers are added between all sentences to cover the n-gram order. If the label sequence was known, then the parameters would have closed form solutions. We can thus deduce that the correct label sequence is the only hidden variable.

The joint probability of the latent variables and observations for the model under consideration may be written as:

$$\begin{aligned} p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta}) &= \prod_{t=1}^{N} p(\boldsymbol{z}_{t-(n-1)}^{t-1}|x_{t-(n-1)}^{t-1}, \boldsymbol{A}) p(\boldsymbol{z}_t|\boldsymbol{z}_{t-(n-1)}^{t-1}, \boldsymbol{B}) p(x_t|\boldsymbol{z}_t, \boldsymbol{C}), \\ &= \prod_{r=1}^{N} \prod_{u=r-(n-1)}^{r-1} p(\boldsymbol{z}_u|x_u, \boldsymbol{A}) \\ &\quad \prod_{s=1}^{N} p(\boldsymbol{z}_s|\boldsymbol{z}_{s-(n-1)}^{s-1}, \boldsymbol{B}) \\ &\quad \prod_{t=1}^{N} p(x_t|\boldsymbol{z}_t, \boldsymbol{C}), \end{aligned} \tag{5}$$

where n is the n-gram context length, N the number of words in the training corpus with special words appended between the sentences, $\boldsymbol{A}$ the class generation probabilities, $\boldsymbol{B}$ the class n-gram probabilities, $\boldsymbol{C}$ the class memberships, x_t the observation at position t and $\boldsymbol{z}_t$ corresponds to a binary state vector of the latent variables (classes) at position t.

Substituting this into formula 4 gives:

$$
\begin{aligned}
Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = & \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old}) \sum_{r=1}^{N} \sum_{u=r-(n-1)}^{r-1} \ln p(\boldsymbol{z}_u | x_u, \boldsymbol{A}) \\
& + \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old}) \sum_{s=1}^{N} \ln p(\boldsymbol{z}_s | \boldsymbol{z}_{s-(n-1)}^{s-1}, \boldsymbol{B}) \\
& + \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old}) \sum_{t=1}^{N} \ln p(x_t | \boldsymbol{z}_t, \boldsymbol{C})
\end{aligned} \tag{6}
$$

We notice that the three parameter types appear in separate terms, so each of the terms may be optimized independently. Introducing a suitable Lagrange multiplier and solving for the zero of the derivative will give these closed form solutions for the maximization step of each EM iteration:

$$
p(c|w) = \frac{\sum_{t=1}^{N} \gamma(z_{twc})}{\sum_{t=1}^{N} \sum_{k} \gamma(z_{twk})} \tag{7}
$$

$$
p(w|c) = \frac{\sum_{t=1}^{N} \gamma(z_{twc})}{\sum_{t=1}^{N} \sum_{v \in V} \gamma(z_{tvc})} \tag{8}
$$

$$
p(c_i | c_{i-(n-1)}, .., c_{i-1}) = \frac{\sum_{t=1}^{N} \xi(h, c)}{\sum_{t=1}^{N} \sum_{k} \xi(h, k)}, \tag{9}
$$

where γ and ξ are shorthand notations for the expected statistics collected in the E-step. For the definitions and the full derivation see the Appendix A.

In practice, sufficient statistics may be collected using a dynamic programming approach with tokens representing one sequence of classes in a training sentence. We applied a global likelihood beam for pruning unlikely class sequences and to further limit the number of tokens by histogram pruning. Standard ARPA-format backoff language models were used for the n-gram term. Witten-Bell smoothing [41] was applied in the training phase because it naturally supports fractional counts for different class sequence hypotheses and is implemented in the SRILM toolkit [42]. Kneser-Ney smoothing may also be generalized for fractional counts [43], although there are no publicly available implementations. We did not observe major differences compared to the unsmoothed estimates in the training phase, and the backoff approach was selected for practical reasons.

We used the following procedure to train the model. The word classes were initialized from the analysis output of the morphological analyzers. For words with multiple analyses, the class generation probabilities $p(c|w)$ were initialized evenly. The initial class membership probabilities $p(w|c)$ were obtained by applying the Bayes' formula as:

$$
p(w|c) = \frac{p(c|w) \times f(w)}{\sum_{v} p(c|v) \times f(v)} \tag{10}
$$

The training was started with unigram statistics and the n-gram order was increased to two and three in the subsequent iterations. The class generation and membership

probabilities were kept constant in the first iterations and were only updated when the n-gram order was increased to three. Tagging of new words was done in the later training iterations using the n-gram probabilities. In the final steps of the training, we observed that the number of classes per word could be reduced to one with only a minor loss in perplexity. This may at first appear counter-intuitive, but allowing multiple classes was more important during the training phase and to allow multiple hypotheses while tagging new words. By allowing only one class membership per word, the further processing of the classes by merging and splitting is simplified. These steps are explained in the next subsection.

2.3.2. Refining Classes by Merging and Splitting

The morphological word classes are not optimal for use with a class language model. For some words, the classes are too fine-grained, whereas the largest word classes consist of hundreds of thousands of words, leading to inaccurate modelling. To overcome these problems and to reach more efficient word classes, we further process the classes by a merging and splitting procedure. We first merged morphological classes with the smallest loss in training data bigram likelihood. A predefined number of randomly sampled class pairs were evaluated and the merger with the smallest loss in likelihood was performed. We constrained the class mergers to operate within the same major part-of-speech tag (i.e. nouns, adjectives, verbs etc.). As a result of this procedure, the classes typically contain words that have the same major part-of-speech tag and sharing similar morphological properties.

To improve the modelling accuracy, we proceeded by splitting the classes using bigram statistics. We evaluated the splitting of a predefined number of classes with the highest token count. For these classes, the words were split to two separate classes. One iteration of exchange algorithm was then run locally between these classes; that is, only for the words belonging to these classes and limiting the exchanges only to the other class. The split with the highest improvement in likelihood was performed.

After the splitting phase, the exchange algorithm was run to further improve the classifications. However, we evaluated different types of constraints for the exchanges that the algorithm was allowed to perform.

- In the unconstrained setting, the exchange algorithm was run normally as defined in Algorithm 1. The morphologically motivated classes acted thus as an initialization.
- In the second setting, the exchanges were constrained to classes, which shared the same major POS tag (i.e. nouns, adjectives, verbs and other classes).
- In the most constrained setting, the exchanges were constrained to the superclass created during the merge phase; that is, in addition to the major POS class constraint, the words are sharing other morphological traits

The number of classes after the merging phase was selected to optimize the likelihood after running the full training (i.e. merging, splitting and exchange procedures).

2.4. Subword Language Models

A popular approach for tackling the OOV and the data sparsity problems for agglutinative languages has been to train the statistical language models over morphs or

other subword units. By combining the subword units of the lexicon, it is possible to assign probabilities to word forms that do not occur in the training corpus. If the lexicon includes (for example) all individual letters or syllables of the language, then the resulting vocabulary of the recognizer is unlimited [15]. However, because some units are short, a high-order n-gram model is required to get the full benefit from the subword modelling [29].

Statistical approaches for learning the units have given good results on many languages [12, 18]. A popular method is Morfessor Baseline [44], which uses the minimum description length (MDL) criterion to find a balance between the cost of storing the model and encoding the training corpus with the model. Morfessor Baseline encodes the corpus with a unigram model. Another segmentation approach is the Greedy unigrams method [45], which also infers a unigram-based model but which does not include a cost for coding the subword lexicon in the optimization criterion. It has been shown to improve the accuracy in some cases compared to Morfessor Baseline, especially if well-matching and reasonably large training corpus is available. In this work, for each evaluated ASR corpus, we trained both one Morfessor Baseline model and one Greedy unigrams model. The Morfessor Baseline model had a lexicon size of 8000 optimized using word types, whereas the Greedy unigrams model had a vocabulary size of 10000 and was trained using the word counts. The better of the two was used for the unlimited vocabulary baseline result and as a component model in the interpolated word-based results. Morfessor Baseline was the better choice in the Finnish 10M word and Estonian 100M word conditions, whereas Greedy unigrams was better in Finnish 100M word and Estonian 10M word conditions. It is possible that the vocabulary was reasonably well-matching in the Estonian 10M word condition, but in the Estonian 100M condition the increased vocabulary size eased the modelling of some of the remaining OOV words in the case of Morfessor Baseline. In this case the results were quite close: whereas in the cases where Greedy unigrams provided better results, the marginal was larger.

In subword-based speech recognition, the word boundaries need to be modelled explicitly. The different possibilities for word boundary modelling were evaluated in WFST-based speech recognition in [46], where the dedicated word boundary symbol and modelling the boundaries redundantly both in the leftmost subword and rightmost subword provided the best results. We have also evaluated the word boundary modelling with mostly similar results. For Finnish, the dedicated word boundary symbol [47, 15] has so far been the most effective approach, whereas for Estonian, using the redundant approach has sometimes resulted in a small improvement. For the experiments in this work, the dedicated word boundary symbol was used because it provided the best or equal results in all cases.

2.5. Decoding

Speech recognition decoders can broadly be categorized into static and dynamic decoders [48]. In a static decoder, all data sources are included in the search network; whereas in a dynamic decoder, the language model probabilities are applied separately during the decoding. The most common type of a static decoder is based on the use of the weighted finite state transducers (WFST) [49]. The most typical dynamic decoder codes the recognition vocabulary using a lexical prefix tree [50] and performs the search using the token-passing procedure [51]. In this work, we follow the dynamic decoding approach. An important property for this work is that large and long-span n-gram models

may be efficiently applied with a dynamic decoder. The interpolation with the class n-gram models is also relatively straightforward to do in the first decoding pass. Standard techniques required for the decoding include the beam search, hypothesis recombination, language model look-ahead [52] and the cross-word modelling [53].

In this work, we use a modified version of the decoder in the AaltoASR package [54] [1]. The decoder was initially developed for the unlimited vocabulary morph-based recognition task [55, 56]. The recognition graph for the subword decoding needs a special construction to correctly handle the intra-word and the inter-word unit boundaries and to allow cross-word pronunciation modelling. The decoder is also able to handle long-span n-gram models [29]. According to an error analysis, only a small part of the recognition errors originate from the search [57].

The word-based recognition is potentially a simpler task than the unlimited vocabulary recognition because the possible word sequences are more constrained. There are, however, some practical challenges. Even if the graph is minimized by tying the suffixes, the graph size will be large, increasing different book-keeping costs. The look-ahead model is also very important for the recognition accuracy because the word labels are more unevenly located in the graph. Recent studies have shown that very large vocabularies may be efficiently decoded using large n-gram models [58, 16].

Because the perplexities for the word-based and the subword-based models are not directly comparable due to the different OOV rates, we compare their performance in a speech recognition task. The same recognizer implementation is applied for both models, but the recognition graph is constructed differently. Silence and cross-word modelling in the graphs are identical for the same word sequences. An important operation in the decoding is the so-called hypothesis recombination. If there are several tokens in the same graph node and in the same n-gram model state, then only the best token is kept and the rest discarded. The hypothesis recombination is extended for the class n-gram interpolation by applying the recombination on n-gram and class n-gram state tuples. This allows the class n-grams to be applied without additional approximations to the beam search. Following [36], we use a bigram look-ahead model with the subword n-grams and a class bigram look-ahead model with the word n-grams.

3. Experiments

3.1. Experimental Setup

To train the Finnish-language models, we used the CSC Kielipankki corpus [59]. The corpus contains text from Finnish newspapers, magazines and books. The size of the full corpus was 139M word tokens with 4.1M word types. The Estonian-language models were trained on a corpus of Estonian newspaper articles and news articles from the web [60]. The perplexity evaluations were performed using text both from the newspaper articles and the web articles. For the ASR evaluation, only the newspaper articles were used for language modeling because that provided a lower OOV rate and improved recognition accuracy. For domain adaptation experiments in the section 3.5, we used subtitles from the Opensubtitles [61] corpus.

[1] available in `https://github.com/aalto-speech/wdecoder`

For acoustic modelling, we used a speaker-independent deep neural network model [62] trained using the Tensorflow [63] package. Phonetic alignments and triphone state tying was obtained using the AaltoASR [54] package. We used a multi-layer perceptron (MLP) network with hidden layer size of 2000. As input features, seven neighboring frames of 39-dimensional Mel-frequency cepstral coefficients (MFCC) with delta and delta-delta features were spliced to form a 585-dimensional input vector. For Finnish, the model had six hidden layers and for Estonian it had five hidden layers. Stochastic gradient descent (SGD) optimizer was used to train the network. We increased the batch size during the training as mentioned in [62], which improved convergence in our experiments.

The earlier work [39] used speech from the Speecon database [64] for the acoustic model training. A 31-hour set of clean dictated wideband speech from 310 speakers was used for training from the Speecon corpus. We were able to significantly improve the recognition result by adding speech also from the Finnish parliament corpus [65, 66]. For training, we used 170 hours of parliament speech from 4 years with a total of 355 speakers. Adding more training data improved the recognition result compared to [39] by around 11% relative or 3% absolute. Switching from GMM to DNN backend improved the result by a further 4 % relative. The surprisingly small improvement from the DNN-based acoustic model may result from some specifics of the Finnish corpora or that the GMM modelling in the AaltoASR package has been developed mostly using Finnish corpora.

Estonian acoustic models were trained on a 30 hour set of broadcast news recordings [60]. Compared to the earlier result [39], switching from the GMM to DNN backend improved the result by 18% relative. Also, compared to [39], the training corpus for the language models is larger in the experiments with 100M word tokens.

The speech recognition experiments were performed in a broadcast news task for both Finnish and Estonian. For Finnish, the development set consisted of 5.38 hours of audio with 35 439 word tokens and the evaluation set 5.58 hours of audio with 37 169 word tokens. For Estonian, the development set consisted of 2.13 hours of audio with 15 691 word tokens and the evaluation set 2.03 hours of audio with 15 335 word tokens.

3.2. Language Models

To infer the word classes, we implemented optimized software [2] for the expectation-maximization algorithm over the classes, class merging with morphological restrictions, class splitting with local exchanges, and the standard exchange algorithm with the optional morphological constraints, as described in subsection 2.3.2. The implementation of the exchange algorithm used the word-class and class-word statistics [27] and multithreading [38]. The number of classes was in most perplexity evaluations 1000 for both Finnish and Estonian, with the exception of morphological classes after the expectation-maximization step and the merging phase. The number of classes after the merging phase was optimized by evaluating the accuracy after subsequent splitting and exchange phases. The perplexities for the class-based models would naturally improve by increasing the number of classes. However, we evaluated that for the interpolated word-based n-gram model and class-based n-gram model, the perplexity did not improve much by increasing the number of classes from 1000. Larger number of classes, 5000 for Finnish

[2] available in `https://github.com/aalto-speech/morphological-classes`

and 10000 for Estonian, were used for the rescoring using class NNLMs, as the result was slightly improved with the higher number of classes.

All of the evaluated n-gram language models used the modified Kneser-Ney smoothing [67] with three discounts per order [1]. The models were trained using the growing and pruning algorithm as implemented in the VariKN toolkit [68]. A development set of 17,000 sentences was used to optimize the discount parameters. The maximum n-gram order of the baseline word n-gram model was set to 3 because of the large size of the vocabulary. For the class-based n-gram models, a 4-gram model was used. The subword n-gram models were limited to 8-grams for Finnish and 6-grams for Estonian. These were the optimal n-gram orders as evaluated on the development set. In some cases, a minimal improvement in recognition accuracy could have been achieved by further increasing the n-gram order.

The neural network language models over class sequences were trained using the TheanoLM language modelling toolkit [69]. We used a configuration of a projection layer followed by a long short-term memory (LSTM) layer [2] and a highway layer with tanh activations [37]. We did not perform an exhaustive study on the NNLM architectures as our main goal was to compare the different classifications. This configuration has been successfully used also in experiments on conversational speech recognition in [69, 11], and it provides a good tradeoff between training time and accuracy. The size of the hidden layers was varied with the size of the used training corpus subset. The configuration for the smallest training corpus size was $250 - 500 - 500$ and the configuration for the largest training corpus sizes was $1000 - 2500 - 2500$. For training regularization, we used dropout with a rate of 0.2 for all the hidden layers.

All of the model combinations were done by a linear interpolation. The optimal interpolation weight for the perplexity evaluation was searched by modifying the weight in steps of 0.05. For the speech recognition experiments, the interpolation weight was optimized on the development set in steps of 0.05.

3.3. Class language model perplexities for different training corpus sizes

In this section, we analyze the perplexities for class n-gram and neural network models trained over the different classifications for different training corpus sizes for both Finnish and Estonian. The models included the morphological EM-trained classes and the models, where the morphological classes were further processed by the merging and splitting procedure. We also analyzed four different classifications for which the training utilized the exchange algorithm. This included a frequency initialized model, which did not use the morphological analyses. Three different models were initialized using the morphological classes, but different constraints were used when running the exchange algorithm. One model was not constrained in any way and used the morphologically motivated classes only for initialization. We experimented also with a classification, which constrained the exchanges to only within the same major part-of-speech class (i.e. nouns, verbs, adjectives etc.). In the most constrained model, the exchanges were restricted to the superclass created during the merge phase; that is, imposing additional constraints from the morphological classes in addition to the major part-of-speech class constraint.

3.3.1. Finnish

	Corpus size	1M	5M	10M	20M	50M
	OOV rate	13.59	7.19	5.38	4.0	2.68
	Model	Perplexity				
Morphological analyzer used	EM model	2980/2271	3555/2566	3711/2615	3805/2608	3887/2654
	EM/merge	3102/2339	3751/2800	3960/2970	4109/3073	4269/3176
	EM/merge/split	**2920**/1716	2903/1808	2905/1800	2900/1794	2776/1734
	Exchange[a], morph. init	2952/**1708**	3076/**1773**	2832/**1751**	2807/1711	2655/1666
	Exchange[b], morph. init	3226/1967	2930/1895	2850/1822	2762/1752	2580/1632
	Exchange, morph. init	3292/2166	**2886**/1905	**2773**/1777	**2659/1684**	**2477/1572**
	Exchange, freq. init[c]	3441/2661	3139/2138	2939/1960	2773/1816	2540/1636

Table 1: Class n-gram model and class-based neural network language model perplexities on the Kielipankki corpus for different training corpus sizes and class training approaches. The perplexity for a class n-gram model is on the left-hand side and for a class NNLM is on the right-hand side.

[a]Exchange algorithm with major part-of-speech tag and merge tree superclass constraint
[b]Exchange algorithm with major part-of-speech tag constraint
[c]Most common words initialized to own classes

The results for the Finnish corpus are given in Table 1 The size of the corpus subset varied for Finnish from 1 million words to 50 million words. The corresponding out-of-vocabulary rates varied from 13.59% to 2.68%. The results show that the frequency-trained classes are powerful compared to the morphological classifications after the expectation-maximization phase, mostly reaching much better perplexity values. However, in all cases the results could be improved by using the analyses provided by the Omorfi analyzer. For the corpus sizes $5M$ and upwards, initializing the exchange

classes with morphological analyses without further constraints in most cases resulted in the most efficient classes. For the corpus size of $1M$ words, constraining the exchanges to the superclass created in the merge phase improved the NNLM result. However, for the n-gram model, directly using the classes after the merge and split phases was the most effective approach. In this case, the difference in perplexity compared to the frequency-based classes was around 500 perplexity points for n-gram models and around 900 perplexity points for the NNLMs. Running the exchange algorithm without constraints degraded the results drastically. This is caused by the very sparse data compared with respect to the large vocabulary size. One may also observe that for the small corpus sizes, the perplexity increases when increasing the training corpus sizes. This is caused by the lower OOV-rate, and consequently more infrequent words being predicted in the perplexity computation.

3.3.2. Estonian

	Corpus size	1M	5M	10M	20M	50M
	OOV rate	9.15	4.53	3.28	2.35	1.47
	Model	Perplexity				
Morphological analyzer used	EM model	2583/2081	3237/2439	3203/2528	3336/2577	3330/2566
	EM/merge	2590/2094	3088/2471	3233/2577	3372/2647	3413/2642
	EM/merge/split	**1778**/1207	1678/1128	1553/1078	1405/1008	1041/956
	Exchange[a], morph. init	1779/**1190**	1630/**1061**	1487/**1001**	1322/933	981/865
	Exchange[b], morph. init	1843/1314	1633/1111	1471/1024	1313/933	970/864
	Exchange, morph. init	1862/1390	**1612**/1129	**1443**/1017	**1291**/**924**	**948**/**835**
	Exchange, freq. init.[c]	2030/1627	1699/1225	1502/1084	1339/970	999/872

Table 2: Class n-gram model and class-based neural network language model perplexities on the Estonian news corpus for different training corpus sizes and class training approaches. The perplexity for a class n-gram model is on the left-hand side and for a class NNLM is on the right-hand side.

[a]Exchange algorithm with major part-of-speech tag and merge tree superclass constraint
[b]Exchange algorithm with major part-of-speech tag constraint
[c]Most common words initialized to own classes

The results for the Estonian corpus are given in Table 2. The size of the corpus subset was varied similarly as for Finnish, upwards from 1 million words to 50 million words. For Estonian, the OOV rates varied from 9.15 % to 1.47 %. Similarly to Finnish, we see that even though the original morphological classifications were not so powerful as such: the results could be much improved by refining the morphological classes. The unconstrained but morphologically initialized exchange classification performed well for larger training corpus sizes, even though the difference to the frequency-initialized classes was not so large. The most constrained exchange classification, in which the exchanges were constrained to the superclass created in the merge phase gave good results for smaller training corpus sizes.

By interpolating the class language models trained using the frequency-based classes and classes utilizing a morphological analyzer, the perplexity can in most cases improved by over 10% relative. The word classifications thus tend to differ quite much between the frequency-based and morphologically motivated approaches for the class inference. For the morphological classes, the perplexity is especially improved for the rare words and in sentence contexts containing rare words.

3.4. Perplexities for interpolated word n-gram and class-based language model

In this section, we analyze perplexities for an interpolation of a word n-gram model and either a class n-gram model or class neural network model. The classifications and training corpora are the same as in the previous subsection 3.3.

3.4.1. Finnish

	Corpus size	1M	5M	10M	20M	50M
	OOV rate	13.59	7.19	5.38	4.0	2.68
	Model	Perplexity				
	Word n-gram	2887	2850	2633	2518	2015
Morphological analyzer used	+ EM model	**2185**/1770	2168/1723	2034/1610	1949/1528	1624/1306
	+ EM/merge	2213/1808	2200/1812	2070/1735	1984/1670	1657/1423
	+ EM/merge/split	2278/1498	**2114**/1431	1960/1340	1862/1276	1541/1088
	+ Exchange[a], morph. init	2294/**1488**	2147/**1414**	**1941**/**1320**	1835/**1242**	1515/1070
	+ Exchange[b], morph. init	2375/1684	2145/1513	1959/1375	1834/1285	1508/1069
	+ Exchange, morph. init	2399/1804	2138/1528	1942/1361	**1810**/1255	**1480**/**1057**
	+ Exchange, freq. init[c]	2601/2104	2250/1669	2010/1462	1849/1327	1500/1079

Table 3: Word n-gram and class n-gram model or class-based neural network language model interpolated perplexities on the Kielipankki corpus for different training corpus sizes and class training approaches. The perplexity for a class n-gram model interpolation is on the left-hand side and for a class NNLM interpolation is on the right-hand side.

[a]Exchange algorithm with major part-of-speech tag and merge tree superclass constraint
[b]Exchange algorithm with major part-of-speech tag constraint
[c]Most common words initialized to own classes

The interpolated perplexity results for the Finnish corpus are given in Table 3. The perplexity for a word-based n-gram model is much lower compared to the perplexities for the class-based models. For comparison, see Table 1. However, by linearly interpolating the word-based and class-based models, much lower perplexities were reached both in the case of a class n-gram and a class NNLM. For frequency initialized classes, the improvement in perplexity ranged from 9.9 % to 26.6 % for class n-grams and from 27.1 % to 47.3 % for class NNLMs. For the best morphologically motivated classes, the improvement in perplexity ranged from 24.3 % to 28.1 % for class n-grams and from 47.5 % to 50.7 % for class NNLMs. In the case of frequency initialized classes, the

improvement from using class-based models was relatively smaller with a smaller training corpora, whereas larger improvements were evaluated with a larger training corpora. The morphologically motivated classes improved results especially in the case of a smaller training corpus, helping to bridge this gap in the relative perplexity improvements. The perplexities could not be much improved by increasing the number of classes for class n-grams, where for class NNLMs the perplexities would still improve by increasing the number of classes. However, in this analysis, the main motivation was to study the relative differences between the different classifications. For the best morphologically motivated classes for each training corpus subset, the conclusions are fairly similar as the results in Table 1. For smaller training corpora, more constrained classes work better; whereas using the morphological classes as initialization is the best approach for larger training corpora.

3.4.2. Estonian

	Corpus size	1M	5M	10M	20M	50M
	OOV rate	9.15	4.53	3.28	2.35	1.47
	Model	Perplexity				
	Word n-gram	1679	1374	1112	930	521
Morphological analyzer used	+ EM model	1454/1271	1221/1088	1003/903	841/759	487/452
	+ EM/merge	1454/1278	1219/1092	1004/909	842/767	488/455
	+ EM/merge/split	**1396**/978	1122/782	919/655	746/543	426/345
	+ Exchange[a], morph. init	1402/**969**	**1103**/**757**	**902**/**631**	**730**/**524**	418/**330**
	+ Exchange[b], morph. init	1438/1052	1119/790	910/648	732/528	418/334
	+ Exchange, morph. init	1455/1099	1119/810	906/650	730/530	**416**/332
	+ Exchange, freq. init.[c]	1531/1239	1153/851	923/677	738/543	419/337

Table 4: Word n-gram and class n-gram model or class-based neural network language model interpolated perplexities on the Estonian news corpus for different training corpus sizes and class training approaches. The perplexity for a class n-gram model interpolation is on the left-hand side and for a class NNLM interpolation on the right-hand side.

[a]Exchange algorithm with major part-of-speech tag and merge tree superclass constraint
[b]Exchange algorithm with major part-of-speech tag constraint
[c]Most common words initialized to own classes

The interpolated perplexity results for the Estonian corpus are given in Table 4. For frequency initialized classes, the improvement in perplexity ranged from 8.8 % to 20.6 % for class n-grams and from 26.2 % to 41.6 % for class NNLMs. For the best morphologically motivated classes, the improvement in perplexity ranged from 16.8 % to 21.5% for class n-grams and from 36.7 % to 44.9 % for class NNLMs. As for Finnish, the more constrained morphological classes worked better for small training corpus. With the largest training corpus, the difference between the morphologically constrained classes and the morphologically initialized exchange algorithm -based classes was small.

3.5. Domain adaptation experiments

As evaluated in the previous subsections, the morphologically motivated classes reached improved perplexities, especially in the case of less training data. We thus evaluated the performance of the different class-based models in a domain adaptation experiment, where a small in-domain corpus was available. For the background language model, we utilized the models trained using the $50M$ word subset of the Kielipankki corpus for Finnish and the news corpus for Estonian. We adapted the models then for the Opensubtitles corpus [61] using a one million word corpus of subtitles from the corpus. The perplexities were evaluated on a held-out set of subtitles consisting of three million words. As vocabulary, we used all unique words from both the background model and the adaptation model. The OOV rates using only the one million word adaptation data would have been 11.19 % for Finnish and 7.86 % for Estonian. For the combined vocabulary with both the background model and the adaptation model, the OOV rates decreased to 3.46 % for Finnish and 2.19 % for Estonian.

Model combination	Perplexity
Word n-grams	626
Interpolated class n-grams without morphological analyzer	564
Interpolated class n-grams with morphological analyzer	543
Interpolated class NNLMs without morphological analyzer	486
Interpolated class NNLMs with morphological analyzer	448

Table 5: Domain adaptation experiments for Finnish

The results for the Finnish domain adaptation experiment are given in Table 5. The baseline perplexity for the interpolated background and adaptation word n-grams was 626. For comparison, the perplexity for a n-gram model trained from the combined background and adaptation data was 1122, which was clearly worse compared to all the results obtained by interpolation. For this baseline result, two language models were interpolated. For all of the other results, the perplexity was acquired as a combination of in total four language models; that is, word n-grams for both background and adaptation data and class-based models for both the background and the adaptation data. For the results with a morphological analyzer, we selected the best perplexity from all combinations using the morphological analyzer. Interpolating with a class n-gram model improved the perplexity by 9.9 % relative for the frequency-based classes and 13.3 % for the morphologically motivated classes. Interpolating with a class-based neural network language model improved the perpexity by 22.4 % relative for the frequency-based classes and 28.4 % for the morphologically motivated classes. According to paired sample t-test, the improvement in perplexity obtained by using morphological analyzer was for the class n-grams (from 564 to 543) statistically significant with the p-value $8e^{-11}$ and for the class NNLMs (from 486 to 448) statistically significant with the p-value $4e^{-13}$.

The results for the Estonian domain adaptation experiment are given in Table 6. The baseline perplexity for interpolated word n-grams was 392. Also for Estonian, combining

Model combination	Perplexity
Word n-grams	392
Interpolated class n-grams without morphological analyzer	360
Interpolated class n-grams with morphological analyzer	350
Interpolated class NNLMs without morphological analyzer	297
Interpolated class NNLMs with morphological analyzer	273

Table 6: Domain adaptation experiments for Estonian

the background and adaptation data to one corpus provided a significantly worse perplexity of 678. Interpolating with a class n-gram model improved the perplexity by 8.2 % relative for the frequency-based classes and 10.7 % for the morphologically motivated classes. Interpolating with a class-based neural network language model improved the perpexity by 24.2 % relative for the frequency-based classes and 30.4 % for the morphologically motivated classes. According to paired sample t-test, the improvement in perplexity obtained by using morphological analyzer was for the class n-grams (from 360 to 350) statistically significant with the p-value $5e^{-13}$ and for the class NNLMs (from 297 to 273) statistically significant with the p-value $8e^{-12}$. As in the experiments in subsections 3.3 and 3.4, the perplexity values were somewhat lower for Estonian than for Finnish. The relative improvements were slightly lower in the case of class n-grams, whereas in the case of class-based NNLMs, the relative improvements were even higher for Estonian than for Finnish.

3.6. Speech recognition results in a broadcast news task

We evaluated the different model combinations in broadcast news speech recognition tasks for Finnish and Estonian. The results for the experiments are given in Table 7 for two subsets—10 million and 100 million words—of the language model training corpus. As a baseline result, we use an unlimited vocabulary recognizer with subword based n-gram models. The n-gram language model used in the decoding was fairly large and the result was not improved by increasing the model size further. We use a subword-based bigram model as the look-ahead language model in the decoding.

In a second recognition pass, the result was rescored using an interpolation of the n-gram model and a subword-based neural network language model. For the second-pass recognition results with NNLMs, a N-best list with a maximum of 2000 hypotheses and the average ranging from 1200 to 1650 hypotheses per sentence, depending on the corpus subset, was generated for each sentence. For Finnish, the rescoring improved the result for 15 % relative for the subset of 10 million words and 17.2 % relative for the 100 million word corpus. The corresponding results for Estonian were 22.0 % relative for the subset of 10 million words and 20.9 % relative for the 100 million word corpus.

We experimented both with frequency-trained classes and morphologically motivated classes. The class-based recognition uses the interpolation of a large word n-gram and

Language	Finnish		Estonian	
Corpus size	10M	100M	10M	100M
OOV rate	5.36	2.55	2.48	1.18
Model	Word error rate			
Subword n-gram	28.65	25.59	13.44	10.95
+ Rescoring with interpolated subword n-gram and subword NNLM	24.35	21.18	10.48	8.66
Word n-gram	31.23	27.13	14.29	11.81
Classes without morphological analyzer				
+ Class n-gram	29.61	25.30	13.55	11.05
++ Subword n-gram	29.34	25.05	13.13	10.77
+++ Rescoring with interpolated word n-gram and class NNLM	27.21	22.24	11.93	9.39
+++ Rescoring with interpolated word n-gram and subword NNLM	26.58	21.36	10.90	8.60
+++ Rescoring with interpolated word n-gram, subword NNLM and class NNLM	26.30	**21.16**	10.64	**8.34**
Classes with morphological analyzer				
+ Class n-gram	29.39	25.16	13.32	11.07
++ Subword n-gram	29.36	24.95	13.01	10.78
+++ Rescoring with interpolated word n-gram and class NNLM	26.44	22.15	11.22	9.02
+++ Rescoring with interpolated word n-gram and subword NNLM	26.28	21.56	10.89	8.58
+++ Rescoring with interpolated word n-gram, subword NNLM and class NNLM	**25.79**	21.22	**10.38**	8.40

Table 7: Speech recognition results in a broadcast news task for Finnish and Estonian. The best result using a word-based recognizer is shown in bold for each training corpus.

a class n-gram model in the first recognition pass. A class bigram model is used as the look-ahead language model during the recognition [36] to ensure accurate decoding.

In the case of Finnish 10M word corpus, the baseline unlimited vocabulary recognizer was better in both the first and second recognition passes. This is mainly due to the high OOV rates, and thus the capability of modelling some of the OOV words with the unlimited vocabulary recognizer proved important. However, the difference compared to the word-based models was still below 1% absolute in the first pass and little above 1% absolute in the rescored results. In this case, it can be seen that using the morphologically motivated classes can improve the results reasonably well, especially in the class NNLM rescoring. Even though the improvement obtained by using morphological analyzer was smaller in the first recognition pass, according to the paired sample t-test the improvement (from 29.61 to 29.39 WER) was statistically significant with the significance level $p \leq 0.05$ with the p-value of 0.047 when evaluated block-wise.

In the experiments with Finnish 100M word corpus, slightly improved result could be reached in the first recognition pass using a word-based recognizer and a roughly equal result in the second pass. The OOV rate was quite much lowered by the increase in the training corpus size. In this case, using morphologically motivated classes improved the results only slightly and mostly in the first recognition pass. Consequently, the training corpus size was large enough to reach efficient classifications by the frequency-initialized exchange algorithm.

With the Estonian 10M word corpus, the result in the first recognition pass using a word-based recognizer improved over the baseline by 3.2% relative. However, this improvement was achieved only if the subword n-gram model is incorporated in the interpolated model and morphologically motivated classes were utilized. With the frequency-derived classes, the improvement was 2.3% relative. As in the case of Finnish, rescoring the results with NNLMs provides large improvements. In particular, the subword-based NNLM is powerful in both cases. As for Finnish, the result for the class-based NNLM was clearly better when the morphologically motivated classes were used. Considering the improvement obtained by using morphologically motivated classes in the first recognition pass, according to the paired sample t-test the improvement (from 13.55 to 13.32 WER) was statistically significant with the significance level $p \leq 0.05$ with the p-value of 0.014 when evaluated block-wise.

The Estonian task with the 100M word corpus was the only evaluated combination, in which we got improved results using a word-based recognizer both in the first and second recognition passes. The result with NNLM rescoring improved from the 8.66 WER to 8.34 WER for a relative improvement of 3.7 %. This improvement was, according to paired sample t-test, statistically significant with the significance level $p \leq 0.05$ with the p-value of 0.029 when evaluated utterance-wise. As for Finnish, with the larger corpus size, the frequency-trained classes provided almost as good results as the morphological classes.

4. Discussion

In this work, we studied class-based language models for the very large vocabulary speech recognition of Finnish and Estonian. For morphologically rich languages such as Finnish and Estonian, a very large vocabulary is required to achieve a reasonably low out-of-vocabulary rate in applications such as transcribing, dictation, or broadcast news recognition. This aggravates the data sparsity issues of the word-based n-gram models. As the vocabulary size grows, it is more likely that class-based language models and different word clustering schemes can provide improvements.

Our earlier experiments [39] showed that class-based n-gram models could be efficiently applied to the first recognition pass for morphologically rich languages, tackling some earlier challenges of scaling the class inference and decoding to a very large vocabulary. Compared to a baseline subword-based unlimited vocabulary recognizer, improved results were reached for Finnish and equal results were reached for Estonian. We have also recently studied the combination of word-based, class-based and subword-based n-gram models in the first recognition pass and introduced a class-based look-ahead language model approach that ensures high decoding accuracy with faster recognition times [36]. With these advances, the proposed system outperformed the baseline for Estonian, and further small improvements were achieved for Finnish.

In this work, we continued on similar themes for applying class-based and subword-based approaches for very large vocabulary speech recognition of morphologically rich languages. As a new contribution, we proposed and evaluated different approaches for inferring morphologically motivated classes. We used an expectation-maximization algorithm for training classes using the output from a morphological analyzer. The classes were further processed by a merging and splitting procedure, followed by an exchange algorithm with varying level of constraints. In the least constrained case, we utilized the inferred classes as an initialization for the exchange algorithm. In the most constrained case, the classes shared the POS tag and also more detailed morphological traits. The different constraints provided the necessary means to analyze the performance of different classifications with different datasets in more detail. The morphological constraints provided extra robustness to the class inference, especially in cases where less training data was available. The difference decreased in the case of larger training corpus. Thus, it seems that at least some efficient solutions to the class inference problem are closely related to part-of-speech classes, which have been further processed to smaller equivalence classes.

The improvement from using morphological classes can be attributed to at least three distinct reasons: 1) the degradation of bigram statistics in cases with little training data compared to the vocabulary size, 2) improved optimization of the bigram class likelihood criterion, and 3) different complementarity effects in the model combination with a word-count based n-gram model. It seems that the degradation of bigram statistics is by far the most important factor to consider, especially in the limited data scenarios.

Considering morphologically rich minority languages, such as Sami and the small Uralic languages, it is often the case that large text corpora for training language models are rare. However, the Giellatekno project [70] has been able to construct morphological dictionaries and analyzers for many dialects of the Sami language. The morphologically motivated classes that are evaluated in this work have the potential to improve language modelling in this type of case. As has been evaluated in [71], utilizing analogous analyzer entries can substantially speed up the expansion of the analyzer's coverage. In addition to the scenario with a less-resourced language, this approach is also relevant in language model adaptation for those cases where a limited corpus of in-domain data is available. We evaluated the approaches also in this type of a LM adaptation scenario, and obtained improved perplexities compared to classes inferred using only word frequencies.

The class-based models were evaluated in Finnish and Estonian ASR tasks. Interpolated class n-gram models were used in the first recognition pass and interpolated class NNLMs were applied through n-best list rescoring in the second pass. In the first recognition pass, the subword-based unlimited vocabulary approach was better in the Finnish 10M word corpus setting that has over 5% OOV rate. In all other settings, improved results could be achieved by using a word-based recognizer, especially when using both class and subword-based n-gram models. In the second pass, rescoring with neural network language models provided large improvements in word error rates in all cases. Subword-based NNLMs outperformed class-based NNLMs both when rescoring unlimited vocabulary recognition and word-based recognition. Nevertheless, class-based NNLM rescoring also improved the results in all cases, and the model could be used as another component in the rescoring phase, yielding additional improvements. Directly training the NNLMs over words would be prohibitive in terms of computational complexity and other requirements.

Even though the word-based recognizer was often better in the first recognition pass, it proved to be harder to improve over the unlimited vocabulary recognition rescored with a subword NNLM. It is possible that the hypotheses generated by the unlimited vocabulary recognizer are more suitable for the powerful subword-based NNLMs. As previously, the unlimited vocabulary approach yielded the lowest score in the Finnish 10M setting. In the Finnish 100M setting and Estonian 10M corpus setting, equal results could be achieved. In the Estonian 100M word corpus setting, where the OOV rate was the lowest, we were able to reach a 3.7 % relative improvement compared to the unlimited vocabulary baseline.

Achieving good results with word-based approaches is valuable because there are benefits in having a limited vocabulary. For instance, grammatically incorrect or very rare word forms can be avoided. In our experiments, we used existing corpora for training the models, and no active measures were taken to improve the OOV rates or training coverage. This was mainly to ensure level comparison to the unlimited vocabulary baseline and to avoid overly optimizing for the evaluation tasks. Despite fairly large corpora being available, it is likely that the modelling could still be improved, such as by crawling more data from the web. The lower OOV rate with the possibility of combining subword-based and class-based modelling approaches could potentially turn the results more favorable for the word-based recognizers. More accurate pruning of the vocabulary entries and improved pronunciation modelling for foreign proper names (FPNs) are other possibilities that could potentially improve the word-based recognition accuracy. Data selection and better modelling of FPNs are also possible in the case of unlimited vocabulary recognition, and further study would be required to see the final impact for the recognition accuracy in both cases.

5. Conclusion

In this work, we evaluated class-based language models for very large vocabulary speech recognition of Finnish and Estonian. We extended the analysis from our previous work by analyzing morphologically motivated classes initialized from the morphological tags obtained from a morphological analyzer. We further refined the resulting classes using a merging-and-splitting approach and running the exchange algorithm with different levels of morphological constraint. The morphologically motivated classes proved efficient, especially in cases where less training data is available. This is important in both less-resourced language modelling scenarios and in cases where class-based language models are adapted with a limited amount of in-domain training data. We utilized the class-based models in all phases of a traditional ASR system; that is, as a look-ahead language model during decoding, interpolated class n-gram model during the first recognition pass and class NNLMs for rescoring. Improved results compared to a subword-based unlimited vocabulary recognizer were obtained in the first recognition pass, with the exception of a low-resource setting with a high OOV rate. In the second recognition pass, it was harder to improve over the unlimited vocabulary recognizer. Significant improvements were only obtained in a case with a large training corpus and fairly low OOV rate. However, the word-based approach offers many practical possibilities for further improving the recognition result.

Acknowledgments

This work was supported by the Academy of Finland with the grant 251170. Aalto Science-IT project provided computational resources for the work. We would like to thank the anonymous reviewers for valuable comments, which helped to improve the article.

References

[1] S. F. Chen, J. T. Goodman, An Empirical Study of Smoothing Techniques for Language Modeling, Computer Speech & Language 13 (4) (1999) 359–394.
[2] S. Hochreiter, J. Schmidhuber, Long Short-Term Memory, Neural Computation 9 (8) (1997) 1735–1780.
[3] Y. Bengio, R. Ducharme, P. Vincent, C. Janvin, A Neural Probabilistic Language Model, Journal of Machine Learning Research 3 (2003) 1137–1155.
[4] W. De Mulder, S. Bethard, M.-F. Moens, A Survey on the Application of Recurrent Neural Networks to Statistical Language Modeling, Computer Speech & Language 30 (1) (2015) 61–98.
[5] F. Karlsson, Finnish: An Essential Grammar, Routledge Essential Grammars, Routledge, 1999.
[6] T.-R. Viitso, Estonian, in: D. Abondolo (Ed.), The Uralic Languages, Routledge, London, 115–148, 1997.
[7] F. Karlsson, K. Koskenniemi, A Process Model of Morphology and Lexicon, Folia Linguistica 19 (1–2) (1985) 207–232.
[8] P. Saagpakk, Estonian-English Dictionary — Eesti-Inglise Sonaraamat, Yale University Press, 1982.
[9] M. Gutmann, A. Hyvärinen, Noise-contrastive Estimation: A New Estimation Principle for Unnormalized Statistical Models, Journal of Machine Learning Research 13 (2010) 307–361.
[10] J. T. Goodman, Classes for Fast Maximum Entropy Training, in: Proceedings of the ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing, 561–564, 2001.
[11] S. Enarvi, P. Smit, S. Virpioja, M. Kurimo, Automatic Speech Recognition With Very Large Conversational Finnish and Estonian Vocabularies, IEEE/ACM Transactions on Audio, Speech, and Language Processing 25 (11) (2017) 2085–2097.
[12] M. Creutz, A. Stolcke, T. Hirsimäki, M. Kurimo, A. Puurula, J. Pylkkönen, V. Siivola, M. Varjokallio, E. Arisoy, M. Saraçlar, Morph-based Speech Recognition and Modeling of Out-of-vocabulary Words Across Languages, ACM Transactions on Speech and Language Processing 5 (1) (2007) 1–29.
[13] M. Kurimo, A. Puurula, E. Arisoy, V. Siivola, T. Hirsimäki, J. Pylkkönen, T. Alumäe, M. Saraçlar, Unlimited Vocabulary Speech Recognition for Agglutinative Languages, in: Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, 487–494, 2006.
[14] E. Arisoy, D. Can, S. Parlak, H. Sak, M. Saraçlar, Turkish Broadcast News Transcription and Retrieval, IEEE Transactions on Audio, Speech, and Language Processing 17 (5) (2009) 874–883.
[15] T. Hirsimäki, M. Creutz, V. Siivola, M. Kurimo, S. Virpioja, J. Pylkkönen, Unlimited Vocabulary Speech Recognition with Morph Language Models Applied to Finnish, Computer Speech & Language 20 (4) (2006) 515–541.
[16] M. Varjokallio, M. Kurimo, A Word-Level Token-Passing Decoder for Subword n-gram LVCSR, in: Proceedings of the IEEE Workshop on Spoken Language Technology, South Lake Tahoe, USA, 495–500, 2014.
[17] M. Kurimo, S. Enarvi, O. Tilk, M. Varjokallio, A. Mansikkaniemi, T. Alumäe, Modeling Under-Resourced Languages for Speech Recognition, Language Resources and Evaluation 51 (2017) 961–987.
[18] B. Tarján, T. Fegyó, P. Mihajlik, A Bilingual Study on the Prediction of Morph-Based Improvement, in: Proceedings of the Fourth International Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU-2014), 131–138, 2014.
[19] R. Kneser, H. Ney, Forming Word Classes by Statistical Clustering for Statistical Language Modelling, in: Proceedings of the First International Conference on Quantitative Linguistics (QUALICO), 221–226, 1991.
[20] P. F. Brown, P. V. deSouza, R. L. Mercer, V. J. D. Pietra, J. C. Lai, Class-based n-gram Models of Natural Language, Computational Linguistics 18 (4) (1992) 467–480.

[21] T. R. Niesler, P. C. Woodland, Variable-length Category n-gram Language Models, Computer Speech & Language 13 (1) (1999) 99–124.
[22] T. R. Niesler, E. W. D. Whittaker, P. C. Woodland, Comparison of Part-of-speech and Automatically Derived Category-based Language Models for Speech Recognition, in: Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, 177–180, 1998.
[23] E. W. D. Whittaker, P. C. Woodland, Language Modelling for Russian and English Using Words and Classes, Computer Speech & Language 17 (1) (2003) 87–104.
[24] A. Vaičiūnas, V. Kaminskas, G. Raškinis, Statistical Language Models of Lithuanian Based on Word Clustering and Morphological Decomposition, Informatica (Lithuanian Academy of Sciences) 15 (2004) 565–580.
[25] T. Brychcín, M. Konopík, Morphological Based Language Models for Inflectional Languages, in: Proceedings of the 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems, vol. 2, 560–563, 2011.
[26] E. W. D. Whittaker, P. C. Woodland, Efficient Class-based Language Modelling for Very Large Vocabularies, in: Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 1, 545–548, 2001.
[27] S. Martin, J. Liermann, H. Ney, Algorithms for Bigram and Trigram Word Clustering, Speech Communication 24 (1) (1998) 19–37.
[28] A. Vaičiūnas, Statistical Language Models of Lithuanian and Their Application to Very Large Vocabulary Speech Recognition, Ph.D. thesis, Vytautas Magnus University, Institute of Mathematics and Informatics, Kaunas, Lithuania, 2006.
[29] T. Hirsimäki, J. Pylkkönen, M. Kurimo, Importance of High-order n-gram Models in Morph-based Speech Recognition, IEEE Transactions on Audio, Speech and Language Processing 17 (4) (2009) 724–732.
[30] Y. Shi, W. Q. Zhang, J. Liu, M. T. Johnson, RNN Language Model with Word Clustering and Class-based Output Layer, Eurasip Journal on Audio, Speech, and Music Processing 2013.
[31] S. F. Chen, Performance Prediction for Exponential Language Models, in: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics on - NAACL '09, 450–458, 2009.
[32] S. F. Chen, S. M. Chu, Enhanced Word Classing for Model M, in: Proceedings of the 11th Annual Conference of the International Speech Communication Association, INTERSPEECH 2010, 1037–1040, 2010.
[33] T. A. Pirinen, Omorfi - Free and Open Source Morphological Lexical Database for Finnish, in: Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015), 313–315, 2015.
[34] S. Orasmaa, T. Petmanson, A. Tkachenko, S. Laur, H.-J. Kaalep, EstNLTK - NLP Toolkit for Estonian, in: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), 2460–2466, 2016.
[35] T. Kuhn, H. Niemann, E. Schukat-Talamazzini, Ergodic Hidden Markov Models and Polygrams for Language Modeling, in: Proceedings of ICASSP '94, IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, 357–360, 1994.
[36] M. Varjokallio, S. Virpioja, M. Kurimo, First-pass Techniques for Very Large Vocabulary Speech Recognition of Morphologically Rich Languages, in: Proceedings of the IEEE Workshop on Spoken Language Technology, Athens, Greece, 227–234, 2018.
[37] R. K. Srivastava, K. Greff, J. Schmidhuber, Training Very Deep Networks, in: Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS'15), vol. 2, 2377–2385, 2015.
[38] R. Botros, K. Irie, M. Sundermeyer, H. Ney, On Efficient Training of Word Classes and Their Application to Recurrent Neural Network Language Models, in: Proceedings of the INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, 1443–1447, 2015.
[39] M. Varjokallio, M. Kurimo, S. Virpioja, Class n-gram Models for Very Large Vocabulary Speech Recognition of Finnish and Estonian, in: Proceedings of the 4th International Conference on Statistical Language and Speech Processing, SLSP, Pilsen, Czech Republic, 133–144, 2016.
[40] C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer-Verlag, Berlin, Heidelberg, ISBN 0387310738, 2006.
[41] I. H. Witten, T. C. Bell, The Zero-Frequency Problem: Estimating the Probabilities of Novel Events in Adaptive Text Compression, IEEE Transactions on Information Theory 37 (4) (1991) 1085–1094.
[42] A. Stolcke, J. Zheng, W. Wang, V. Abrash, SRILM at Sixteen: Update and Outlook, in: Proceedings

of the IEEE Workshop on Automatic Speech Recognition and Understanding, 2011.
[43] Y.-C. Tam, T. Schultz, Correlated Bigram LSA for Unsupervised Language Model Adaptation, in: Proceedings of the 21st International Conference on Neural Information Processing Systems (NIPS'08), 1633–1640, 2008.
[44] M. Creutz, K. Lagus, Unsupervised Discovery of Morphemes, in: Proceedings of the ACL 2002 Workshop on Morphological and Phonological Learning, 21–30, 2002.
[45] M. Varjokallio, M. Kurimo, S. Virpioja, Learning a Subword Vocabulary Based on Unigram Likelihood, in: Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, 7–12, 2013.
[46] P. Smit, S. Virpioja, M. Kurimo, Improved Subword Modeling for WFST-based Speech Recognition, in: Proceedings of the INTERSPEECH 2017, 2551–2555, 2017.
[47] E. W. D. Whittaker, P. C. Woodland, Particle-Based Language Modelling, in: Proceedings of the Sixth International Conference on Spoken Language Processing (ICSLP 2000), 170–173, 2000.
[48] X. L. Aubert, An Overview of Decoding Techniques for Large Vocabulary Continuous Speech Recognition, Computer Speech & Language 16 (1) (2002) 89–114.
[49] M. Mohri, F. C. N. Pereira, M. Riley, Speech Recognition with Weighted Finite State Transducers, in: Handbook on Speech Processing and Speech Communication, Springer, Berlin, 1–31, 2008.
[50] H. Ney, S. Ortmanns, Progress in Dynamic Programming Search for LVCSR, Proceedings of the IEEE 88 (8) (2000) 1224–1240.
[51] S. Young, N. Russell, J. Thornton, Token Passing: A Simple Conceptual Model for Connected Speech Recognition System, Tech. Rep., Cambridge University Engineering Department, 1989.
[52] S. Ortmanns, H. Ney, Look-ahead Techniques for Fast Beam Search, Computer Speech & Language 14 (1) (2000) 15–32.
[53] A. Sixtus, H. Ney, From Within-word Model Search to Across-word Model Search in Large Vocabulary Continuous Speech Recognition, Computer Speech & Language 16 (2) (2002) 245–271.
[54] Aalto University, AaltoASR, `http://github.com/aalto-speech/AaltoASR/`, 2014.
[55] T. Hirsimäki, M. Kurimo, Decoder Issues in Unlimited Finnish Speech Recognition, in: Proceedings of the 6th Nordic Signal Processing Symposium (NORSIG 2004), 320–323, 2004.
[56] J. Pylkkönen, An Efficient One-pass Decoder for Finnish Large Vocabulary Continuous Speech Recognition, in: Proceedings of the 2nd Baltic Conference on Human Language Technologies, 167–172, 2005.
[57] T. Hirsimäki, M. Kurimo, Analysing Recognition Errors in Unlimited-Vocabulary Speech Recognition, in: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers, 193–196, 2009.
[58] H. Soltau, G. Saon, Dynamic Network Decoding Revisited, in: Proceedings of the 2009 IEEE Workshop on Automatic Speech Recognition and Understanding, 276–281, 2009.
[59] Gatherers: The Department of General Linguistics, University of Helsinki; The University of Eastern Finland; CSC - IT Center for Science Ltd. Available through the Language Bank of Finland, http://www.kielipankki.fi/, Kielipankki Corpus. An Electronic Document Collection of the Finnish Language, 2000.
[60] E. Meister, L. Meister, R. Metsvahi, New Speech Corpora at IoC, in: XXVII Fonetiikan päivät 2012 - Phonetics Symposium 2012, 30–33, 2012.
[61] P. Lison, J. Tiedemann, OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles, in: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), 923–929, 2016.
[62] D. Yu, L. Deng, Automatic Speech Recognition: A Deep Learning Approach, Springer Publishing Company, London, 2014.
[63] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, URL `http://tensorflow.org/`, software available from tensorflow.org, 2015.
[64] D. Iskra, B. Grosskopf, K. Marasek, H. van den Heuvel, F. Diehl, A. Kießling, SPEECON - Speech Databases for Consumer Devices: Database Specification and Validation, in: Proceedings of the Third International Language Resources and Evaluation Conference (LREC 2002), 329–333, 2002.
[65] The Parliament of Finland, Plenary Sessions of the Parliament of Finland, Downloadable Version,

URL http://urn.fi/urn:nbn:fi:lb-2017030901, 2017.
[66] A. Mansikkaniemi, P. Smit, M. Kurimo, Automatic Construction of the Finnish Parliament Speech Corpus, in: Proceedings of the Interspeech 2017, 3762–3766, 2017.
[67] R. Kneser, H. Ney, Improved Backing-off for M-gram Language Modeling, in: Proceedings of the 1995 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), vol. 1, 181–184, 1995.
[68] V. Siivola, T. Hirsimäki, S. Virpioja, On Growing and Pruning Kneser-Ney Smoothed N-gram Models, IEEE Transactions on Speech, Audio and Language Processing 15 (5) (2007) 1617–1624.
[69] S. Enarvi, M. Kurimo, TheanoLM - An Extensible Toolkit for Neural Network Language Modeling, in: Proceedings of the Interspeech 2016, 3052–3056, 2016.
[70] UiT The Arctic University of Norway ; The Divvun group at UiT The Arctic University of Norway, Giellatekno - Saami Language Technology, URL https://http://giellatekno.uit.no, 2020.
[71] K. Lindén, Entry Generation by Analogy — Encoding New Words for Morphological Lexicons, Northern European Journal of Language Technology 1 (2009) 1–25.

A. Derivation of the Expectation-Maximization Training Parameter Estimates

A.1. Class generation probabilities

$$\begin{aligned}\sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta}^{old}) \sum_{r=1}^{N} \sum_{u=r-(n-1)}^{r-1} \ln p(\boldsymbol{z}_u|x_u,\boldsymbol{A}) \\ = \sum_{w} \sum_{t=1}^{N} \sum_{r=t-(n-1)}^{t-1} \sum_{k=1}^{K} \gamma(z_{rwk}) \ln A_{wk}\end{aligned} \tag{11}$$

By recalling the constraint $\sum_w A_{wk} = 1$ for each w, we solve the parameters A_{wk} using Lagrange multipliers:

$$\frac{\partial}{\partial A_{wk}} \left[\sum_{w} \sum_{t=1}^{N} \sum_{r=t-(n-1)}^{t-1} \sum_{k=1}^{K} \gamma(z_{rwk}) \ln A_{wk} + \mu(\sum_{k} A_{wk} - 1) \right] = 0 \tag{12}$$

Taking the derivative:

$$\sum_{t=1}^{N} \sum_{r=t-(n-1)}^{t-1} \frac{\gamma(z_{rwk})}{A_{wk}} + \mu = 0 \tag{13}$$

Substituting in $\sum_k A_{wk} = 1$:

$$-\sum_{k} \sum_{t=1}^{N} \sum_{r=t-(n-1)}^{t-1} \frac{\gamma(z_{rwk})}{\mu} = 1 \tag{14}$$

solves μ:

$$\mu = -\sum_{k} \sum_{t=1}^{N} \sum_{r=t-(n-1)}^{t-1} \gamma(z_{rwk}) \tag{15}$$

and further A_{wk}:

$$A_{wk} = \frac{\sum_{t=1}^{N} \sum_{r=t-(n-1)}^{t-1} \gamma(z_{rwk})}{\sum_{k} \sum_{t=1}^{N} \sum_{r=t-(n-1)}^{t-1} \gamma(z_{rwk})} = \frac{\sum_{t=1}^{N} \sum_{r=t-(n-1)}^{t-1} \gamma(z_{rwk})}{\sum_{t=1}^{N} \sum_{r=t-(n-1)}^{t-1} \gamma(z_{rw})} \tag{16}$$

Here we assume that special symbols, which belong to an own class, are added between the training sentences to cover the n-gram order. The estimate thus simplifies:

$$A_{wk} = \frac{\sum_{t=1}^{N} \gamma(z_{twk})}{\sum_{t=1}^{N} \gamma(z_{tw})} \tag{17}$$

A.2. n-Gram probabilities

$$\begin{aligned}\sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta}^{old}) \sum_{t=1}^{N} \ln p(\boldsymbol{z}_t|\boldsymbol{z}_{t-(n-1)}^{t-1},\boldsymbol{B}) \\ = \sum_{t=1}^{N} \sum_{k=1}^{K} \sum_{h_{(t-1)}=1}^{K} \cdots \sum_{h_{(t-(n-1))}=1}^{K} \xi(h,k) \ln B_{hk},\end{aligned} \tag{18}$$

where $\xi(h,k)$ is a shorthand notation:

$$\xi(h,k) = \mathbb{E}[z_{tk}, z^{(t-1)(h_{-1})}_{(t-(n-1))(h_{t-(n-1)})}] = \sum_{\boldsymbol{z}} \gamma(\boldsymbol{z})[z_{tk}, z^{(t-1)(h_{-1})}_{(t-(n-1))(h_{t-(n-1)})}] \tag{19}$$

By recalling the constraint $\sum_k B_{hk} = 1$ for each h, we solve the parameters B_{hk} using Lagrange multipliers:

$$\frac{\partial}{\partial B_{hk}} \left[\sum_{t=1}^{N} \sum_{k=1}^{K} \sum_{h_{(t-1)}=1}^{K} \cdots \sum_{h_{(t-n+1)}=1}^{K} \xi(h,k) \ln B_{hk} + \mu(\sum_k B_{hk} - 1) \right] = 0 \tag{20}$$

Taking the derivative:

$$\sum_{t=1}^{N} \frac{\xi(h,k)}{B_{hk}} + \mu = 0 \tag{21}$$

Substituting in $\sum_k B_{hk} = 1$:

$$-\sum_k \sum_{t=1}^{N} \frac{\xi(h,k)}{\mu} = 1 \tag{22}$$

solves μ:

$$\mu = -\sum_k \sum_{t=1}^{N} \xi(h,k) \tag{23}$$

and further B_{hk}:

$$B_{hk} = \frac{\sum_{t=1}^{N} \xi(h,k)}{\sum_k \sum_{t=1}^{N} \xi(h,k)} \tag{24}$$

A.3. Class membership probabilities

$$\begin{aligned} \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old}) \sum_{t=1}^{N} \ln p(x_t|\boldsymbol{z}_t, \boldsymbol{C}), \\ = \sum_w \sum_{t=1}^{N} \sum_{k=1}^{K} \gamma(z_{tkw}) \ln C_{kw}, \end{aligned} \tag{25}$$

where index k is over all classes at position t. The γ term is a shorthand notation for the conditional probability $z_{tkw} = 1$:

$$\gamma(z_{tkw}) = \mathbb{E}[z_{tkw}] = \sum_{\boldsymbol{z}} \gamma(\boldsymbol{z}) z_{tkw}, \tag{26}$$

where $\gamma(\boldsymbol{z})$ is the posterior probability for one sequence of latent variable state vectors $\boldsymbol{z}$ and z_{tkw} the binary value of state k and word w at position t on that sequence.

By recalling the constraint $\sum_w C_{kw} = 1$ for each k, we solve the parameters C_{kw} using Lagrange multipliers:

$$\frac{\partial}{\partial C_{kw}} \left[\sum_w \sum_{t=1}^{N} \sum_{k=1}^{K} \gamma(z_{tkw}) \ln C_{kw} + \mu(\sum_w C_{kw} - 1) \right] = 0 \tag{27}$$

Taking the derivative:

$$\sum_{t=1}^{N} \frac{\gamma(z_{tkw})}{C_{kw}} + \mu = 0 \tag{28}$$

Substituting in $\sum_w C_{kw} = 1$:

$$-\sum_w \sum_{t=1}^{N} \frac{\gamma(z_{tkw})}{\mu} = 1 \tag{29}$$

solves μ:

$$\mu = -\sum_w \sum_{t=1}^{N} \gamma(z_{tkw}) \tag{30}$$

and further C_{kw}:

$$C_{kw} = \frac{\sum_{t=1}^{N} \gamma(z_{tkw})}{\sum_w \sum_{t=1}^{N} \gamma(z_{tkw})} = \frac{\sum_{t=1}^{N} \gamma(z_{tkw})}{\sum_{t=1}^{N} \gamma(z_{tk})} \tag{31}$$

B. Vitae

Matti Varjokallio received the M.Sc. degree in communications engineering from the Helsinki University of Technology, Espoo, Finland in 2007. He is pursuing doctoral studies in language technology at the Department of Signal Processing and Acoustics at Aalto University, Finland. He is currently working as a Speech Scientist at Speechly Oy, Helsinki, Finland.

Mikko Kurimo received the D.Sc. (Ph.D.) in technology degree in computer science from the Helsinki University of Technology, Espoo, Finland, in 1997. He is currently Associate Professor in the Department of Signal Processing and Acoustics at Aalto University, Finland. His research interests are in speech recognition, machine learning and natural language processing.

Sami Virpioja received the D.Sc. (Tech.) degree in computer and information science from Aalto University, Espoo, Finland, in 2012. Between 2013 and 2017 he has worked as a Research Scientist at Lingsoft and as a Postdoctoral Reasearch at Aalto University. He is currently a Lead Data Scientist at Utopia Analytics and a University Researcher in the Department of Digital Humanities at University of Helsinki, Finland. His research interests are in machine learning and natural language processing.