Improved subword modeling for WFST-based speech recognition

Peter Smit, Sami Virpioja, Mikko Kurimo

Department of Signal Processing and Acoustics, Aalto University, Finland

firstname.lastname@aalto.fi

Abstract

Because in agglutinative languages the number of observed word forms is very high, subword units are often utilized in speech recognition. However, the proper use of subword units requires careful consideration of details such as silence modeling, position-dependent phones, and combination of the units. In this paper, we implement subword modeling in the Kaldi toolkit by creating modified lexicon by finite-state transducers to represent the subword units correctly. We experiment with multiple types of word boundary markers and achieve the best results by adding a marker to the left or right side of a subword unit whenever it is not preceded or followed by a word boundary, respectively. We also compare three different toolkits that provide data-driven subword segmentations. In our experiments on a variety of Finnish and Estonian datasets, the best subword models do outperform word-based models and naive subword implementations. The largest relative reduction in WER is a 23% over word-based models for a Finnish read speech dataset. The results are also better than any previously published ones for the same datasets, and the improvement on all datasets is more than 5%.

Index Terms: speech recognition, Kaldi, subword modeling, Finnish, Estonian

1. Introduction

In most large-vocabulary continuous speech recognition (LVCSR) systems for western languages, words are the basic units in building a lexicon and language models. However, in agglutinative languages such as Finnish or Estonian, the inflection, derivation and compounding is so frequent, that the language is hard to cover sufficiently without utilizing subword units [1]. Besides decreasing the out-of-vocabulary rate, the subword-based systems have the advantage that a smaller vocabulary reduces the model complexity significantly. While it has become possible to build n-gram language models to cover millions of words [2], it still requires special solutions for the state-of-the-art neural network language models [3] to train an output layer that has a vast number of possible output symbols. Thus, decreasing the vocabulary size is an important way to improve the efficiency of the models.

This work describes our research group's latest efforts to improve and implement subword models for a weighted finite state transducer (WFST) decoder in LVCSR. The contributions of the paper provided in the next sections include: techniques to model subwords effectively in a WFST decoder and evaluations of subword segmentation algorithms and ways to mark the word boundaries in subword sequences. Additionally, we report our latest performance status in the most commonly used Finnish and Estonian datasets. The new methods are provided[1] for the Kaldi speech recognition toolkit [4], but the general principles for lexicon modification are applicable to any WFST-based speech recognizer.

This work was financially supported by the Academy of Finland under the grant number 251170. PS would like to thank Lingsoft, Inc. and the Foundation for Aalto University Science and Technology. Computational resources were provided by the Aalto Science-IT project.

2. Subword modeling

For many parts of a speech processing pipeline there is no difference between subword and word systems. For example, whether the units are words, letters, or subwords, the n-gram modeling toolkits create a model that predicts the next token on based on the previous token; only the length of the n-grams may need to be increased. This section investigates the parts that do need to be changed in a speech recognition pipeline.

2.1. Boundary markers

The first step for a subword system is to define the subword unit. There are many choices for this; three segmentation methods are described in Section 4 and evaluated in Section 5.3. Regardless of the chosen units, it is important to be able to reconstruct words from the subwords to produce readable text. In previous work for Finnish [1], this was done by introducing a word boundary tag (<w>), which separated words, and normal spaces were used to separate subwords. There are however alternatives, as shown in Table 1. All these markings satisfy the requirement that the word text can be reconstructed in a trivial manner. The actual boundary tag or character can be changed without any loss of generalization.

Table 1: *Four methods of marking subword units so that the original word sequence 'two slippers' can be reconstructed*

Style (abbreviation)	Example
boundary tag (<w>)	<w> two <w> slipp er s <w>
left-marked (+m)	two slipp +er +s
right-marked (m+)	two slipp+ er+ s
left+right-marked (+m+)	two slipp+ +er+ +s

The other three styles beside <w> mark the subwords to indicate their location in a word. In left-marked style (+m), a subword is prefixed with a character to indicate that there was no word boundary directly preceding the subword. This style has been used for Turkish [5] and Hungarian [6, 7]. In [7], it was shown to outperform word boundary tags. In right-marked style (m+), a suffix marker is added to a subword if there is no word boundary after it. Finally, left+right-marked style (+m+) applies markers on both sides of the subwords.

The choice of the marking style is not just a matter of taste, because it affects the efficiency of the LVCSR system. For example, using <w> tags increases the number of tokens in a sen-

[1]The code belonging to this work will be published at http://github.com/aalto-speech/subword-kaldi

tence and requires a higher n-gram order in language modeling. The other style tags increase the vocabulary when compared to the <w> tags, but have less tokens in the segmented sentences. The different marking styles are compared in an end-to-end experiment in Section 5.2.

2.2. Lexicon generation

When a word is segmented into subwords, its pronunciation must be segmented as well. In many languages (e.g. English), this requires special attention. However, Finnish and Estonian have almost one-to-one letter-to-sound mapping. Thus, in this paper we have simply used the graphemes as phonemes.

2.3. Phone modeling

In some speech recognition systems subword phone modeling would be easy and straightforward, but in Kaldi systems there are two common phone modeling improvements that complicate the implementation of subword models.

First, silence is often modeled in such way that it is optionally allowed on word boundaries, but not in the middle of words. Therefore, a correct subword implementation needs to be able to indicate what transitions between tokens are actual word boundaries, or otherwise silences might be recognized on the wrong locations.

A second improvement is to use position-dependent-phones. Four separate phones are generated from every original phone, each labeled with its location in the word. This results in labels for the begin, end, internal and single phones, the last for words with a transcription of only one phone. If there is enough data for each of the labeled phones, they are modeled separately, otherwise they will be clustered together during the creation of the decision tree. In informal experiments this has shown to give good recognition improvements. For position-dependent-phones it has to be known if a subword is preceded or succeeded by a word-boundary, information which is not available at the moment the plain-text lexicon is created. It can however be encoded in the lexicon FST level as is described in Section 3.

2.4. Enforcing subword restrictions

With all different style subword markings there are restrictions on the possible output sequences the recognizer can generate. For the boundary tag style each sentence must start and end with a <w> tag. The +m and m+ style markings also have a restriction on the sentence boundaries; the sentence cannot start or end with a marked subword, respectively. The +m+ style marking has the highest degree of restriction. The first subword should be a starting subword and the last one an ending subword. Also each transition between subwords should be marked on both subwords the same as either in-word or between-word.

Systems that do not apply this restrictions will not be able to make an unambiguous decision how to reconstruct words from the subword sequence. In practice this does not have to be a problem, but the performance will likely be better if the correct restrictions are in place.

3. Subword lexicon modeling in a WFST

In WFST-based speech recognition, the search network for a decoder is composed out of four separate FSTs that each provide one part of the mapping from sounds to words [8]. The hidden Markov model FST ($\mathcal{H}$) maps emission distributions to context-dependent-phones. After that the context FST ($\mathcal{C}$) maps these context-dependent-phones to context independent phones. The third part is the lexicon FST ($\mathcal{L}$) which maps phone sequences to words and inserts appropriate silences on word boundaries. The last FST is actually more like an acceptor; the grammar FST ($\mathcal{G}$) gives appropriate probabilities to the word sequences.

To accommodate the desired properties of the subword recognition model, specifically the considerations regarding phone modeling and subword restrictions, the following modifications have been made to $\mathcal{L}$-FST.

3.1. Original word lexicon FST

A standard prototype $\mathcal{L}$-FST for a word based lexicon is shown in Figure 1. All standard arcs are marked with their input and output label. Weights have been omitted for brevity without a loss of generalization; in this and all later described FSTs both optional silences and word pronunciations can be weighted. Also self-loops are not shown; these loops allow to stay in the same state for multiple observations. Self-loops are the reason for the special disambiguation symbols (starting with #), these are necessary to keep the FST determinizable. On the end of the decoding process these symbols are mapped to ϵ meaning that they do not consume any actual input symbols.

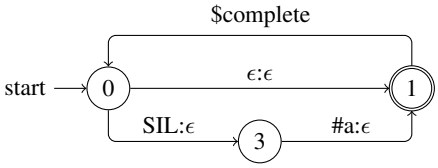

Figure 1: *Prototype $\mathcal{L}$-FST for a standard word lexicon. Each arc is labeled with its corresponding input:output symbols. The $\$x$-syntax indicates the location for replacement with a second FST named x.*

The arc marked as $complete is special. It is replaced with a linear FST that represents the actual lexicon. An example of this lexicon is given in Figure 2.

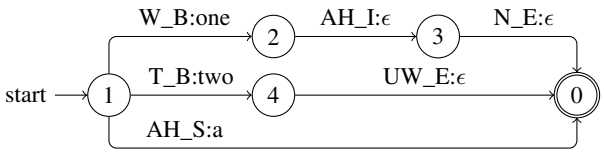

Figure 2: *Example of the $complete FST for the lexicon 'one' (W AH N), 'two' (T UW) and 'a' (AH), with position (_B: begin, _E: end, _I: internal, _S:single) marked.*

3.2. Proposed subword lexicon FST

Figure 3 shows a prototype $\mathcal{L}$ for subword recognition. Instead of only a single replacement it needs to be combined with four linear FSTs that represent the subwords that can appear on different locations in a word: begin, end, middle or complete. In the case of word boundary tags (<w>), all subwords can appear in any location, so they will added to each set of linear FSTs. Even though all subwords will appear in all sets of linear FSTs, the sets will not be the same because of position-dependent phones. In the 'prefix' set, each transcription will start with a _B (begin) phone and all other phones will be _I (internal) phones. Similar, for the 'suffix' set each transcription

will end with an _E (end) phone and in the 'infix' set there will be only _I phones. The 'complete' set will start with _B phones and end with _E phones, unless it is a single phone transcription, then it will be marked _S (single).

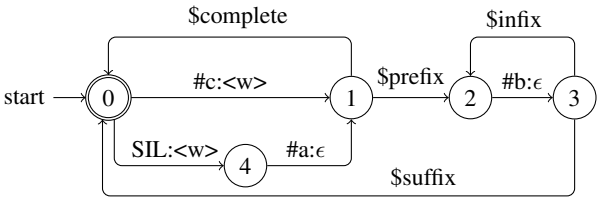

Figure 3: *Prototype subword $\mathcal{L}$-FST, specifically for the <w> style tags. For other markers the prototype is the same except that all <w> tags are replace by ϵ .*

For all other style markings the same prototype FST is used, except that the <w> tags in the output symbols are replaced by epsilons (ϵ). The exact configuration of the sub-FSTs depends on the marking. For a +m style model, the subwords that are not marked will appear in 'prefix' and 'complete', while the marked ones will be in 'suffix' and 'internal'. Similarly for the m+ style model, the unmarked subwords will be in 'suffix' and 'complete' and the marked ones in 'prefix' and 'complete'. In the +m+ style model, the subwords are completely marked and only appear once in their appropriate category.

4. Segmentation methods

The performance of the speech recognizer depends on the type of the subword units. In the Section 5.3, we compare three data-driven segmentation methods in our WFST-based system: Morfessor, greedy unigram, and byte-pair encoding.

4.1. Morfessor

Morfessor [9, 10] is family of statistical methods and tools for segmenting words based on the Minimum Description Length principle. Its primary goal is to find those units of language that resemble the surface forms of morphemes, the smallest information-bearing units of a language. However, the level of segmentation can be adjusted by changing the weight between the cost of encoding the lexicon (the parameters θ) and the cost of encoding the corpus part in the cost function:

$$L(\theta, D_W) = -\log P(\theta) - \alpha \log P(D_W \mid \theta). \qquad (1)$$

The Morfessor Baseline model has been a popular method for segmenting Finnish, Estonian and other agglutinative languages for speech recognition [11, 12]. In this work, we use the Morfessor 2.0 implementation [13].

4.2. Greedy Unigram

The Greedy Unigram (G1G) segmentation [14] was proposed as an alternative segmentation method specifically designed with speech recognition for agglutinative languages in mind. G1G encodes the corpus with high unigram likelihood via multigram expectation-maximization training and greedy likelihood-based pruning. We use the implementation from Factor Toolkit [15].

4.3. Byte pair encoding

Byte pair encoding (BPE) [16] is an algorithm that was designed for data compression and which was recently popularized for segmenting text in other natural language processing domains such as machine translation [17].

In BPE, iteratively the most frequent pair of symbols (initially characters) in a word list is replaced with a new symbol, representing a character n-gram, until a desired number of symbols is reached. We use the implementation provided with [17]. The implementation extends each word internally with an end-of-word character that signals a word break. We replace this by the different styles of marking used in this paper after segmentation.

5. Experiments

5.1. Setup

The experiments were done on five Finnish and Estonian LVCSR tasks. Table 2 describes the type of data and the available amount of audio for each set. The "News" data are authentic radio and television news programs. The "Read" data are selected sentences from a large text corpus read aloud by various volunteers, recorded either by a fixed telephone line or a lapel microphone in relatively quiet environments. All datasets have independent training, development and evaluation sets that do not overlap in speakers. For both languages a newspaper corpus is available for language modeling. The Finnish corpus is a subset of the Finnish Text Collection [18] and has 144 million words. For Estonian the corpus contains 83 million words.

Table 2: *Source and hours of data for each dataset. fi-news has no training set, therefore the fi-read training data is used*

Name		Type	train	dev	eval
et-bn-ak	[19]	News (TV)	164[2]	1.9	1.9
et-bn-er	[19]	News (Radio)	164[2]	2.0	2.0
fi-news		News (Radio)	fi-read	5.4	5.6
fi-phone	[20]	Read (Phone)	218.8	2.3	2.2
fi-read	[21]	Read	148.6	0.95	1.2

The acoustic models are all sequence-trained deep neural network models [23] trained with the Kaldi toolkit [4]. This particular type of discriminative training uses a phone language model, hence the actual units used during decoding time was irrelevant for the acoustic model training. Recognitions were done in a single pass using Minimum Bayes Risk decoding [24].

Language models are all n-gram models trained with the VariKN toolkit [25]. This toolkit was chosen as it has specific support for high-order n-gram contexts, which are beneficial when subword units are used. Specifically, when subword units are used a higher-order n-gram is needed to model the same amount of context as an equivalent word model. The size of the language models were tuned so that all models had approximately the same amount of n-gram contexts, 40 million, for all experiments in this paper.

5.2. Boundary marker comparison

To compare our subword implementation to word-based models we trained Morfessor segmentations for each dataset and evaluated them on the development sets. The Morfessor models are trained without any removal of low frequency words. The α parameter (which controls the vocabulary size) and the damp-

[2]The training set used was a combination of the training sets of et-bn-ak + et-bn-er and training data from a conversational corpus [22]

ening method were optimized using a grid search for the best recognition result.

In addition to the subword lexicon FSTs proposed in Section 3, we evaluate also a more naive subword modeling strategy, in which all subwords are simply treated as words. This naive strategy does allow silences on all subword boundaries and has a mismatch for the position-dependent phones.

For the word language models different frequency thresholds were experimented, but in all cases it was optimal to train a model on the full vocabulary.

Table 3 shows the resulting Word Error Rates (WER) for these experiments. The subword-based models with a naive lexicon implementation perform worse than the word-based models. However, with the proposed lexicon FSTs, the subword models outperform the word models in all experiments.

Table 3: *Recognition WER (%) on development sets using Morfessor segmentation for subword experiments.*

	et-		fi-		
	bn-ak	**bn-er**	**news**	**phone**	**read**
Word	17.46	9.03	23.73	14.45	8.60
Naive +m	17.52	9.22	24.11	15.48	9.70
Naive +m+	17.81	9.52	25.45	15.36	9.10
Proposed <w>	17.94	8.99	**22.89**	13.20	6.62
Proposed +m+	**17.12**	**8.47**	22.96	**13.13**	**6.55**
Proposed +m	17.46	9.14	23.47	13.27	7.12
Proposed m+	17.64	9.37	23.79	13.44	7.24

For all but one experiment the +m+ style marking of subwords is most effective, with the <w> style being the best in the fi-news experiment. The +m marking performs worse than +m+, and it only outperform the <w> tags in a single experiment. The m+ marking performs the worst in all experiments.

In order to compare the effect of the vocabulary size and marking style on the Word Error Rate, results for the fi-phone dataset are plot in Figure 4. Only those points are shown that use the default Morfessor parameters, except for parameter that controls the vocabulary size. The Figure shows that for small vocabulary sizes the +m+ style markings outperform the <w> tags and give the overall best result. For larger vocabularies, +m+ and <w> provide similar results.

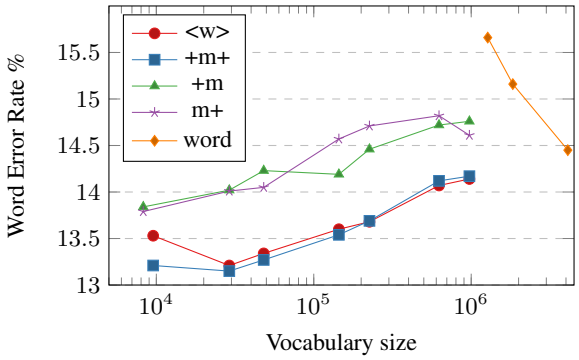

Figure 4: *Comparison of vocabulary size vs Word Error Rate for each style of marker.*

5.3. Subword type and segmentation comparison

We compare the effect of the segmentation type for the fi-news an fi-phone datasets. For each segmentation method we keep the default parameters, only optimizing the frequency threshold for words included in the training of the segmentation.

Table 4 shows that for both datasets the Morfessor segmentation works the best, but the differences are very small. Even the BPE method, which to our knowledge has not been used in speech recognition before, works relatively well and outperforms word-based recognition. We tried to extend the subword lexicon size above 15.000 words for all segmentation methods, but no better results were obtained.

Table 4: *Comparison of segmentation algorithms and segmentation vocabulary sizes. WER (%) on development sets.*

	fi-news			fi-phone		
	5k	**10k**	**15k**	**5k**	**10k**	**15k**
Morf.	23.02	22.82	**22.79**	13.13	**13.04**	13.17
G1G	23.06	22.93	23.02	13.19	13.11	13.17
BPE	23.18	23.17	23.17	13.15	13.30	13.37

5.4. Comparison with previous state-of-the-art

For all datasets, we selected the best models that performed best on the development data, and ran the recognition on the evaluation data. In Table 5, the accuracies for word and subword models are compared to the best previous results collected from literature. On all datasets we were able to outperform previous results by large margins.

Word-based model yields the best WER for the first Estonian dataset, but the subword model outperforms it in Letter Error Rate (LER). On all other datasets subword results outperform word based systems both in WER and LER.

Table 5: *Word Error Rates (WER) and Letter Error Rates (LER) on evaluation sets*

Set	**Word WER / LER**	**Subword WER / LER**	**Previous best WER / LER**
et-bn-ak	**17.48** / 9.56	18.28 / **9.36**	– / –
et-bn-er	8.36 / 1.77	**7.70** / **1.70**	8.2 [26] / –
fi-news	25.49 / 8.97	**24.98** / **8.92**	28.9 [27] / –
fi-phone	14.07 / 2.73	**12.79** / **2.47**	21.88 [14] / 7.18 [14]
fi-read	11.11 / 1.68	**9.44** / **1.44**	13.3 [26] / 2.81 [28]

6. Conclusions

We have implemented techniques to use subword modeling in the WFST-based framework in such way that silence modeling and position-dependent phonemes can be utilized in the same manner as in word-based models. On a variety of Finnish and Estonian datasets we have shown that subword models created with the proposed techniques outperform both word-based models and 'naive' subword modeling.

When comparing the different styles of subword boundary marking styles we have found that the +m+ style markings give the best performance for this setup.

Also we have compared three different segmentation toolkits, finding that the Morfessor toolkit works optimal for the tested datasets, but that the performances are in general close.

7. References

[1] T. Hirsimäki, J. Pylkkönen, and M. Kurimo, "Importance of high-order n-gram models in morph-based speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 724–732, May 2009.

[2] M. Varjokallio, M. Kurimo, and 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 2016*, ser. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), P. Král and C. Martín-Vide, Eds. Springer-Verlag, 2016, vol. 9918 LNCS, pp. 133–144.

[3] S. Enarvi and M. Kurimo, "Theanolm — an extensible toolkit for neural network language modeling," in *Interspeech 2016*, 2016, pp. 3052–3056. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2016-618

[4] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011, iEEE Catalog No.: CFP11SRW-USB.

[5] E. Arisoy, D. Can, S. Parlak, H. Sak, and M. Saraclar, "Turkish broadcast news transcription and retrieval," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 5, pp. 874–883, July 2009.

[6] B. Tarján, T. Fegyó, and P. Mihajlik, "A bilingual study on the prediction of morph-based improvement." in *SLTU*, 2014, pp. 131–138.

[7] Á. Varga, B. Tarján, Z. Tobler, G. Szaszák, T. Fegyó, C. Bordás, and P. Mihajlik, "Automatic close captioning for live hungarian television broadcast speech: A fast and resource-efficient approach," in *International Conference on Speech and Computer*. Springer International Publishing, 2015, pp. 105–112.

[8] M. Mohri, F. Pereira, and M. Riley, "Speech recognition with weighted finite-state transducers," in *Springer Handbook of Speech Processing*. Springer, 2008, pp. 559–584.

[9] M. Creutz and K. Lagus, "Unsupervised discovery of morphemes," in *Proceedings of the ACL 2002 Workshop on Morphological and Phonological Learning*, ser. MPL '02, vol. 6. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp. 21–30. [Online]. Available: http://www.aclweb.org/anthology/W/W02/W02-0603.pdf

[10] ——, "Unsupervised models for morpheme segmentation and morphology learning," *ACM Transactions on Speech and Language Processing*, vol. 4, no. 1, January 2007.

[11] M. Kurimo, A. Puurula, E. Arisoy, V. Siivola, T. Hirsimäki, J. Pylkkönen, T. Alumäe, and M. Saraclar, "Unlimited vocabulary speech recognition for agglutinative languages," in *Proceedings of the 2006 Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL)*, ser. HLT-NAACL '06. Stroudsburg, PA, USA: Association for Computational Linguistics, Jun. 2006, pp. 487–494. [Online]. Available: http://aclweb.org/anthology/N/N06/N06-1062.pdf

[12] M. Creutz, T. Hirsimäki, M. Kurimo, A. Puurula, J. Pylkkönen, V. Siivola, M. Varjokallio, E. Arisoy, M. Saraçlar, and A. Stolcke, "Morph-based speech recognition and modeling of out-of-vocabulary words across languages," *ACM Transactions on Speech and Language Processing (TSLP)*, vol. 5, no. 1, pp. 3:1–3:29, Dec. 2007. [Online]. Available: http://doi.acm.org/10.1145/1322391.1322394

[13] S. Virpioja, P. Smit, S.-A. Grönroos, and M. Kurimo, "Morfessor 2.0: Python implementation and extensions for Morfessor Baseline," Department of Signal Processing and Acoustics, Aalto University, Helsinki, Finland, Report 25/2013 in Aalto University publication series SCIENCE + TECHNOLOGY, 2013.

[14] M. Varjokallio, M. Kurimo, and S. Virpioja, "Learning a subword vocabulary based on unigram likelihood," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, Dec 2013, pp. 7–12.

[15] M. Varjokallio and M. Kurimo, "A toolkit for efficient learning of lexical units for speech recognition," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, N. C. C. Chair), K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, Eds. Reykjavik, Iceland: European Language Resources Association (ELRA), may 2014.

[16] P. Gage, "A new algorithm for data compression," *The C Users Journal*, vol. 12, no. 2, pp. 23–38, 1994.

[17] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, August 2016, pp. 1715–1725. [Online]. Available: http://www.aclweb.org/anthology/P16-1162

[18] CSC - IT Center for Science, "The Helsinki Korp Version of the Finnish Text Collection," 1998. [Online]. Available: http://urn.fi/urn:nbn:fi:lb-2016050207

[19] E. Meister, L. Meister, and R. Metsvahi, "New speech corpora at IoC," in *XXVII Fonetiikan päivät 2012 — Phonetics Symposium 2012: 17–18 February 2012, Tallinn, Estonia: Proceedings*, E. Meister, Ed. TUT Press, 2012, pp. 30–33.

[20] A. Rosti *et al.*, "Speechdat finnish database for the fixed telephone network," SpeechDat Technical Report, Tech. Rep., 1998.

[21] D. J. Iskra, B. Grosskopf, K. Marasek, H. van den Heuvel, F. Diehl, and A. Kiessling, "Speecon-speech databases for consumer devices: Database specification and validation." in *LREC*, 2002.

[22] P. Lippus, "Phonetic corpus of estonian spontaneous speech v1.0.3," 2016. [Online]. Available: https://doi.org/10.15155/1-00-0000-0000-0000-0012BL

[23] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi," in *Interspeech 2016*, 2016, pp. 2751–2755. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2016-595

[24] H. Xu, D. Povey, L. Mangu, and J. Zhu, "An improved consensus-like method for minimum bayes risk decoding and lattice combination," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2010, pp. 4938–4941.

[25] V. Siivola, T. Hirsimäki, and S. Virpioja, "On growing and pruning Kneser-Ney smoothed n-gram models," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 15, no. 5, pp. 1617–1624, 2007.

[26] T. Alumäe, "Neural network phone duration model for speech recognition." in *INTERSPEECH*, 2014, pp. 1204–1208.

[27] M. Varjokallio, M. Kurimo, and S. Virpioja, "Class and morphological category n-gram models for very large vocabulary speech recognition of finnish and estonian," *Computer Speech & Language*, in review.

[28] J. Pylkkönen and M. Kurimo, "Analysis of extended baum-welch and constrained optimization for discriminative training of hmms," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 9, pp. 2409–2419, Nov 2012.