Importance of High-Order N-Gram Models in Morph-Based Speech Recognition

Teemu Hirsimäki, Janne Pylkkönen, and Mikko Kurimo, *Senior Member, IEEE*

***Abstract*—Speech recognition systems trained for morphologically rich languages face the problem of vocabulary growth caused by prefixes, suffixes, inflections, and compound words. Solutions proposed in the literature include increasing the size of the vocabulary and segmenting words into morphs. However, in many cases, the methods have only been experimented with low-order n-gram models or compared to word-based models that do not have very large vocabularies. In this paper, we study the importance of using high-order variable-length n-gram models when the language models are trained over morphs instead of whole words. Language models trained on a very large vocabulary are compared with models based on different morph segmentations. Speech recognition experiments are carried out on two highly inflecting and agglutinative languages, Finnish and Estonian. The results suggest that high-order models can be essential in morph-based speech recognition, even when lattices are generated for two-pass recognition. The analysis of recognition errors reveal that the high-order morph language models improve especially the recognition of previously unseen words.**

***Index Terms*—Language modeling (LM), morphology, speech recognition, variable-length n-grams.**

I. Introduction

During recent years, there has been active research on improving language models (LMs) and speech recognition systems for agglutinative, inflecting, and compounding languages, such as Finnish, Turkish, and Estonian. In these languages, the prefixes and suffixes, numerous inflected forms, and compound words cause problems for traditional word-based language modeling approaches. A very large vocabulary is needed to cover the words of the language adequately, which causes n-gram language modeling to suffer from data sparseness, as many of the word n-grams occur only rarely. A natural solution to this problem is to segment the words into shorter units (or *morphs* as we call them) and train the n-gram language models over the morphs instead of whole words. This approach enables the model to create an infinite vocabulary by concatenating arbitrary morphs together. Thus, the model can assign a positive probability to words not seen in the training data, which alleviates the out-of-vocabulary (OOV) problem of the model.

This article addresses issues that are important when morph-based language models are used for modeling and recognizing agglutinative and compounding languages. We experiment how the high-order n-grams affect the error rates in different morph segmentations and language model sizes and analyze the errors the models produce when compared to a word-based approach. We also study how fixed-length and variable-length n-gram models perform in one-pass decoding and lattice generation. All experiments are performed using the Finnish SpeechDat telephone speech corpus and for comparison, some are also duplicated using the Estonian SpeechDat. We hope the results explain some of the discouraging results in the literature and help others to apply morph-based language models successfully in future.

The paper is organized as follows. In Section II, earlier work on recognizing morphologically rich languages is reviewed. Section III describes the algorithms that are used to train the recognition system for the experiments. The experiments are presented in Section IV with discussion, followed by conclusions in Section V.

II. Related Work

Speech recognition using morph-based language models or very large vocabularies have been experimented on many languages that are agglutinative, inflecting, or compounding. Table I summarizes results from earlier work on several of those languages. First, there are experiments in which traditional word-based language models were used, but the size of the vocabulary was increased. For Arabic [1], [2], Czech [3], Finnish [4], and German [5], increasing the size of the vocabulary to 300 000–800 000 words improved error rates. We believe the same behavior is also expected for the other languages that have rich morphology.

The results of the work comparing morph and word-based approaches are varying, which is partly explained by diverse experimental settings. One observation is that it seems to be difficult to get good results with morphs if only 2-gram models are used. Improvements have been reported for Czech [6], Slovenian [7], and Turkish [8], but only against word vocabularies of less than 60 000 words. With 3-gram and higher models, morphs have given better results against word models, but again, the vocabularies of the word models have been only around 60 000 words in most cases. Very large word vocabularies have been compared to morph-based approaches only for Arabic [1], [2] and Finnish [4]. For Arabic, the word vocabularies of 300 000 and 800 000 already provide a comparable performance, but for

Manuscript received August 03, 2008; revised December 11, 2008. Current version published April 01, 2009. This work was supported in part by the Academy of Finland in the projects Adaptive Informatics and New Adaptive and Learning Methods in Speech Recognition, in part by the EMIME project and PASCAL2 Network of Excellence in the European Community Framework Program 7, and in part by the Graduate School of Language Technology, Finland. This paper reflects only the authors' views. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Tanja Schultz.

The authors are with the Adaptive Informatics Research Center, Helsinki University of Technology, 02150 Espoo, Finland. (e-mail: teemu.hirsimaki@tkk.fi).

Digital Object Identifier 10.1109/TASL.2008.2012323

1558-7916/$25.00 © 2009 IEEE

Authorized licensed use limited to: Teknillinen Korkeakoulu. Downloaded on April 7, 2009 at 11:11 from IEEE Xplore. Restrictions apply.

TABLE I
RELATED WORK ON MORPH-BASED SPEECH RECOGNITION

Language	N-gram	Vocabularies		Corpus	Comment	Reference
		Words	Morphs	[M words]		
Arabic (ECA)	4	18k	6k	0.16	Words were better.	Creutz [9]
Arabic (ECA)	3	54k?	49k	0.15	Morphs were slightly better.	Kirchhoff [10]
Arabic (IRA)	3	98k	58k	2.8	Morphs were better; combining with words improved further.	Sarikaya [11]
Arabic (MSA)	3, 7	64k, 800k	64k, 800k	14 (utt.)[a]	64k: morphs were better; 800k: no difference.	Choueiter [1]
Arabic (MSA)	3	64k–300k	64k	400	Morphs were better than 64k words, but equal to 300k words.	Xiang [2]
Czech	2	20k	10k	33	Morphs gave better results.	Byrne [6]
Czech	2, 3	60k	25k	33	No difference.	Byrne [12]
Czech	2	64k–300k	—	360	Large vocabularies improve results. No comparison to morphs.	Nouza [3]
Dutch	3	40k	28k	35	Automatic splitting of compound words better than whole words.	Laureys [13]
Estonian	3	60k	60k	76	Morphs were better.	Alumäe [14]
Estonian	Var.	60k	37k	55	Morphs were better.	Kurimo [15]
Estonian	Var.	5k, 60k	5k, 60k	43	Morphs were better.	Puurula [16]
Finnish	3	64k	64k	30	Morphs were better.	Siivola [17]
Finnish	3–5	69k, 410k	26k, 65k	40	Morphs were better.	Hirsimäki [4]
Finnish	Var.	400k	25k	150	Morphs were better.	Kurimo [15]
German	2, 3	3k	2k	0.12	No difference.	Geutner [18]
German	4	20k–500k	—	315	Large vocabularies improve results. No comparison to morphs.	McTait [5]
Hungarian	3	20k	5k–8k	0.2	Morphs were slightly better.	Mihajlik [19]
Hungarian	2, 3	—	25k	39	Morph-based approach. No comparison to word models.	Szarvas [20]
Slovenian	2, 3	20k, 60k	20k, 60k	105	20k: Morphs were better, 60k: morphs slightly better.	Rotovnik [7]
Slovenian	2–4	20k	8k	59	Morphs were better.	Maučec [21]
Turkish	3	60k	15k–48k	2	Morphs were better, some recombination improved further.	Hacioglu [22]
Turkish	2, 3	30k	2k–23k	81	Half-words were better. Authors mention that the stems from the test set were added to all models.	Erdoğan [8]
Turkish	6	50k	34k	27	Morphs were better.	Kurimo [15]
Turkish	2	60k	60k	10	Words were better.	Arısoy [23]
Turkish	3-?	50k	34k	96	Morphs were better (highest n-gram order not mentioned).	Arısoy [24]

[a] the authors only report the number of utterances

Finnish the morph-based models have given better results. We believe that this relates to the higher degree of agglutinativity in Finnish.

Various methods have been used for segmenting words. For some languages, there exist morphological analyzers that can be used for segmenting words into morphs and to obtain other information such as stems and part-of-speech tags. Of the works shown in the table, morphological analyzers have been used in [4], [6], [11], [12], [14], [19], [20], and [22]–[24]. Syllables and other rule-based systems have been used in [8], [13], [17], and [21]. Data-driven algorithms have been used in [1], [4], [15]–[17], [19], and [22]–[24]. In the Morpho Challenge evaluation, different data-driven algorithms have been compared to each other and rule-based methods and reference analysis in speech recognition [25] and information retrieval [26].

In addition to directly applying morph-based n-grams, the morphological and lexical information can be combined and applied as, for example, Factored Language Model [10] or Joint Lexical–Morphological Language Model [11]. These models have so far been successfully applied in Arabic [10], [11] and Estonian [14].

III. TECHNIQUES

A. Variable-Length N-Gram Language Modeling

The goal of language modeling is to learn probabilities of the sentences in the target language or the target application. The probability of a word sequence $w_1^n = w_1, \dots, w_n$ is usually factored as follows:

$$P(w_1^n) = P(w_1)P(w_2|w_1)\dots P\left(w_n|w_1^{n-1}\right) \tag{1}$$

and in the case of n-gram modeling, the probability of the next word is conditioned only on $n-1$ preceding words, or the *history* denoted as $h = w_{t-n+1}^{t-1}$

$$P(w_t|w_1^{t-1}) \approx P_M(w_t|h). \tag{2}$$

Furthermore, in order to ensure nonzero probabilities for all possible word sequences, the model is usually represented in an interpolated form

$$P(w_t|h) = P_M(w_t|h) + \gamma(h)P_M(w_t|h') \tag{3}$$

where h' is the history h with the first word removed, $\gamma(h)$ is an interpolation coefficient, and $P_M(\cdot|\cdot)$ is the explicit estimate stored in the model. There are numerous smoothing methods for estimating P_M and γ from a text corpus, Kneser–Ney smoothing being the state-of-the-art [27].

Because of the interpolation, the model does not need to contain estimates for all possible word-history pairs (w, h) explicitly. Many of the explicit estimates can be removed without affecting the performance, if the lower-order estimates

Authorized licensed use limited to: Teknillinen Korkeakoulu. Downloaded on April 7, 2009 at 11:11 from IEEE Xplore. Restrictions apply.

provide reasonably close approximations. This leads to variable-length models where the length of the history is not fixed but varies depending on the context. Variable-length n-gram models can be created by creating first a full n-gram model and then pruning most redundant n-grams. Entropy-based pruning [28] can be used with most smoothing techniques, but Kneser–Ney smoothing needs a specialized pruning algorithm [29] that maintains the marginal properties of the lower-order distributions.

It is also possible to grow the variable-length model incrementally without creating a full fixed-length model first. In this paper, a growing and pruning algorithm [29] is used to create Kneser–Ney smoothed language models.

B. Morph-Based Language Modeling

In this paper, we use the unsupervised Morfessor Baseline [30] algorithm to split words into morpheme-like units. The main idea of the algorithm is to find a compact set of morphs that can be used to represent the words in the training corpus compactly by concatenating available morphs. A very compact morph set would consist of individual letters of the language, but then representing the corpus as individual letters would not be compact. The other extreme is to leave all words unsplit, which leads to more compact representation of the corpus. The morph set, however, becomes very large, and the optimal solution lies somewhere between these two extremes. Simplifying some details, the algorithm optimizes the following posterior probability (see [30] for details)

$$P(\text{lexicon}|\text{corpus}) \propto P(\text{lexicon})P(\text{corpus}|\text{lexicon}) \approx \prod_{\text{letters } \alpha} P(\alpha) \prod_{\text{morphs } \mu} P(\mu). \quad (4)$$

where α iterates over all morph strings in the lexicon, and μ iterates over all morphs in the training corpus.

Somewhat different segmentations are obtained if the model is trained on the list of words in the training corpus instead of the whole corpus. As in previous experiments [4], the model is trained on the list of most common words. If the size of the word list is decreased, morphs will be shorter, i.e., smaller morph sets are produced, which may be useful in generating lexica for speech recognition. In practice, the algorithm tends to select common substrings as morphs, which often resemble grammatical morphemes [25].

After the morph set is obtained, the LM training data is split into morphs. By using the Viterbi algorithm, the most probable morph sequence is obtained for each word. Adding individual letters as morphs with a small probability ensures that all words in the training data and even new words can be split. Also, a separate word boundary morph is inserted between words. Then standard n-gram modeling techniques can be used to create morph-based language models from the corpus.

The languages experimented in this paper, Finnish and Estonian, have a very regular grapheme-to-phoneme relation, so that inferring a pronunciation for any morph is simple. A more complicated grapheme-to-phoneme relation might pose additional restrictions to possible word segmentations, as it is desirable to select the morphs so that they have well defined context independent pronunciations.

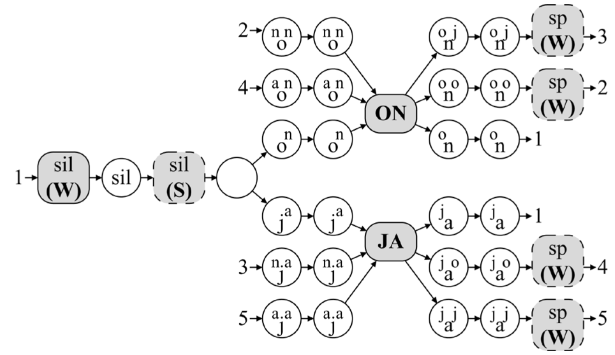


Fig. 1. Triphonic search network containing morphs “JA” and “ON” with word boundary “(W)’, and sentence boundary “(S)’. The phonemes are “j” “a”, “o”, “n”, “sp” (short pause), and “sil” (silence). Triphone contexts are shown in superscript. Dashed states are optional and can be skipped. Each numbered transition at right connects to the corresponding numbered transition at left. For clarity, only two states are shown for each triphone, and self-loops have been omitted.

C. Decoding

The search algorithm used in our decoder is based on a cyclic search network that contains the hidden Markov model (HMM) state sequences of the morphs (or words in case of word-based recognition) without any language model. Fig. 1 illustrates a network with two morphs: “JA” (phonemes “j” and “a”) and “ON” (phonemes “o”and “n”). Triphone contexts are applied across all boundaries. For readers familiar with recognition systems based on weighted finite-state transducers, the search network roughly corresponds to the deterministic composition $H \circ C \circ L$ (see [31], for example). The algorithm we use for building the network is described in [32].

The search is performed using a token pass algorithm [33]. Each hypothesis is represented as a moveable token, which contains the following information among other things: the path the token has traversed, cumulative acoustic and LM probabilities of the path, and the LM state index. The n-gram LM is represented as a separate finite-state automaton with special escape transitions for handling backoffing, and the maximum order of the LM is not restricted. At every frame, each token is propagated through the transitions leaving from its state, and the cumulative probabilities are updated. The LM probability can be updated when the token arrives in a state that defines the next morph (the shaded states in the figure). A state in the network can hold several tokens as long as the tokens have different LM state indices. Whenever two tokens with the same LM state index collide in the same search network state, only the one with a higher total cumulated probability is preserved. This ensures that two tokens are merged as soon as it is known that their order cannot change anymore, but not sooner. Standard beam and histogram pruning techniques are applied at the end of each frame to control the decoding speed.

To speed up the search, a 2-gram LM look-ahead [34] is used. At the branches of the search network, a separate 2-gram LM is used to compute temporary LM probabilities for the possible

Authorized licensed use limited to: Teknillinen Korkeakoulu. Downloaded on April 7, 2009 at 11:11 from IEEE Xplore. Restrictions apply.

TABLE II
TRAINING DATA STATISTICS

	Finnish		Estonian	
	Units[a]	Length[b]	Units[a]	Length[b]
Morph 2k	515 374 836	2.5	554 965 292	2.3
Morph 10k	427 629 417	3.0	476 038 512	2.7
Morph 50k	378 567 733	3.4	442 273 189	2.9
Word 500k	143 533 707[c]	7.9[c]	126 882 114[c]	6.6[c]

[a] includes word boundary morphs for the morph models
[b] average number of letters in a morph or word (word boundary morph counted as a one-letter morph for the morph models)
[c] includes also the OOV-words in the training data

morphs before the actual morph state is reached. Details are described in [32]. The decoder can also produce lattices using the word-pair approximation (see [35], for example), which corresponds to a morph-pair approximation in morph-based recognition.

IV. Experiments and Discussion

A. Finnish Data and Setup

For the Finnish speech recognition experiments we used the SpeechDat database,[1] which consists of 4000 speakers recorded over fixed telephone line. The corpus was partitioned as follows: 39 h from 3838 speakers for training, 46 min from 79 separate speakers for development and another similar set for evaluation. Only full sentences were used and sentences with severe noise or mispronunciations were removed. The acoustic features used Mel-frequency cepstral coefficients with first and second derivatives, 39-dimensions altogether, followed by a global maximum-likelihood linear transformation. The acoustic models were based on decision-tree-tied triphones with mixtures of Gaussians as state probability distributions. No adaptation was used in the task, only cepstral mean subtraction was used to compensate telephone channel differences.

The language models were trained on 150 million words from the Finnish Kielipankki corpus [36], which contains text from books, magazines, and newspapers. To study the importance of high-order n-gram LMs for both small and large lexicon size in morph-based language models, three morph lexica of different size were created. The smallest lexicon was obtained by taking the 6100 most frequent words in the training corpus as input to the Morfessor algorithm (Section III-B) and the two larger ones by using the list of 51 000 and 390 000 most frequent words. The Morfessor algorithm was then used to produce three corresponding segmentations for all the words in the training corpus to train the three morph-based language models. Additionally, a vocabulary of 500 000 most frequent words was created to train the word-based language model. Table II describes the segmentations.

B. Estonian Data

The Estonian speech recognition was experimented on an Estonian SpeechDat-like corpus [37]. The corpus contains 1335 speakers recorded over fixed telephone line as well as cellular network. Unlike in Finnish setup, also isolated words and phrases were used for training acoustic models, but only full sentences were used for the development and evaluation sets. No filtering of bad sentences was performed. The training set had 1266 speakers, totalling 110 h of speech. The development set had 15 speakers, and the evaluation set 50 speakers, eight sentences each. The acoustic modeling used similar techniques as the Finnish experiments, although a slightly older version of our speech recognition system was utilized. The language models of the Estonian experiments were trained on 127 million words from the Segakorpus.[2] As in Finnish experiments, three morph lexica were created using the Morfessor algorithm (Section III-B), as well as a word lexicon with a vocabulary of 500 000 most frequent words.

C. Language Model Comparisons

When language models based on different morph sets are compared, emphasis must be put on creating a fair comparison. If the order of the n-gram model is fixed, the setting favors longer morphs, because long morphs exploit more context in n-gram modeling. Another issue is that the size of the n-gram models may differ greatly, putting smaller models at a disadvantage.

We argue that a fair setting is to compare models that are equally large. Since the actual size of the LM data structure depends on the implementation, a reasonably fair approximation, in the context of n-gram modeling, is to count the n-grams in the model. Then the modeling problem for each morph set is to train the best possible model under the size constraint. By using methods for growing and pruning n-gram models, it is quite straightforward to train models that are nearly optimal for a given size.

In order to compare the performance of the different morph segmentations and the traditional word segmentation in both small and large language models, four target sizes were selected. A growing and pruning algorithm [29] was used to train Kneser–Ney smoothed variable-length n-gram models for both languages, each word segmentation and target size. For all models, n-grams that occur only once in the training data were pruned from orders 3 and above.

In Table I, the most common model was a 3-gram. In order to show the effect of using a fixed n-gram order for different segmentation approaches, we also trained models by limiting the growing to 3-grams and using the same pruning threshold as in the largest variable-length model. In practice, our 3-gram morph models are very close to full 3-gram morph models. On the contrary, our 3-gram word model is heavily pruned. The unpruned 3-gram word model would be enormous because of the huge vocabulary.

Table III shows how the n-grams are distributed in the Finnish 3-gram models and largest variable-length models for each lexicon. For example, in the Morph 2 k (Variable) model,

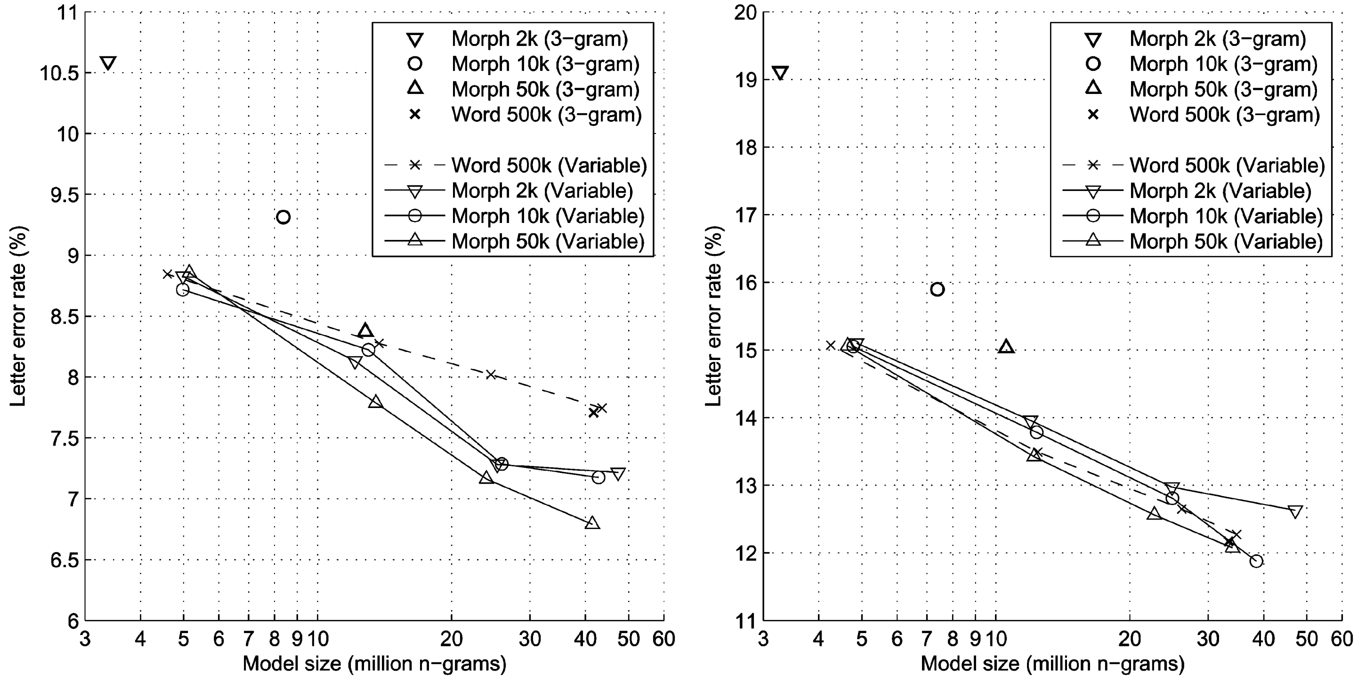
[1] http://www.speechdat.org/

[2] http://test.cl.ut.ee/korpused/segakorpus/

Authorized licensed use limited to: Teknillinen Korkeakoulu. Downloaded on April 7, 2009 at 11:11 from IEEE Xplore. Restrictions apply.

TABLE III
N-GRAM DISTRIBUTIONS OF THE FINNISH 3-GRAM MODELS AND LARGEST VARIABLE-LENGTH MODELS

	1-grams	2-grams	3-grams	4-grams	5-grams	6-grams	7-grams	8-grams	9-grams	10-grams	Total
						[×1000]					
Morph 2k (3-gram)	2	439	2940								3381
Morph 2k (Variable)	2	439	2727	13 230	15 681	9965	3799	1121	289	70	47 341
Morph 10k (3-gram)	10	1444	6920								8374
Morph 10k (Variable)	10	1444	6610	16 073	11 458	5385	1440	298	54	9	42 783
Morph 50k (3-gram)	50	2819	9932								12 801
Morph 50k (Variable)	50	2819	9406	14 691	9436	3748	1057	194	32	4	41 437
Word 500k (3-gram)	500	34 618	4745								39 863
Word 500k (10-gram)	500	34 665	4669	1280	355	103	30	9	3	1	41 615



(a) (b)

Fig. 2. Speech recognition results for the 3-gram and variable-length models. (a) Finnish SpeechDat. (b) Estonian SpeechDat.

the n-grams are concentrated on 5-grams, and for the other models the concentration is on the lower orders.

D. One-Pass Recognition Experiments

The decoder was run with rather loose pruning parameters to minimize search errors. In the Finnish experiments, the pruning parameters were set to achieve a real-time factor of 10 for all models. For Estonian an even slower setting was used resulting real-time factors of about 30. Apart from HMM/GMM acoustic models, the decoder also utilizes HMM state dependent duration models modeled as gamma distributions. The decoder parameters were tuned on the development data.

Due to compound words and suffixes, the words in Finnish and Estonian are rather long and may consist of several morphs (e.g., in the Finnish evaluation set the average is 7.75 letters/word). We therefore measure the speech recognition performance in letter error rate (LER) instead of word error rate (WER) to have a finer resolution for the results.

Fig. 2 shows the letter error rates for both languages, different word segmentations, and model sizes. First we note that the Estonian recognition task is clearly a more difficult one. The best letter error rate is 6.8% in Finnish and 11.9% in Estonian. One reason is that the noise and recording conditions in the Estonian data are more diverse.

The morph and word variable-length models improve roughly similarly when the model size is increased. In the Finnish case, the largest morph models perform better compared to the word models. In the Estonian task, however, the word models perform equally well as the high-order morph models.

In the Finnish experiments, the LER differences between the best morph 50 k model and the two other morph sets are statistically significant according to the Wilcoxon signed rank test at the 5% significance level. All these morph models are also statistically significantly better than the best word model. In Estonian, among the largest LMs of each lexicon, only the LER

Authorized licensed use limited to: Teknillinen Korkeakoulu. Downloaded on April 7, 2009 at 11:11 from IEEE Xplore. Restrictions apply.

TABLE IV
RESULTS OF THE FINNISH LATTICE RESCORING EXPERIMENT

Recognition method[a]	LER [%]	
Morph 50k (2-gram) lattice + rescore	7.8	
Morph 50k (3-gram) lattice + rescore	7.3	
Morph 50k (Variable) lattice + rescore	6.9	
Morph 50k (Variable) one-pass	6.8	(From Fig. 2)

[a] rescoring is done using the largest Morph 50k (Variable) model

difference between morph 10 k and morph 2 k is statistically significant. For completeness, the word error rates of the Finnish experiments using the largest morph models (2 k, 10 k, 50 k) and the word model are 22.4%, 21.6%, 21.7%, and 26.8%, respectively. For Estonian, the corresponding WERs are 34.6%, 33.1%, 33.9%, and 34.0%.

Looking at the performance of the 3-gram models, we clearly see the importance of high-order n-grams in the morph-based models. Preventing the morph models to grow beyond 3-grams hurts the models severely. The word model, on the other hand, is not hurt at all by the 3-gram restriction, which is quite natural in the light of Table III. The variable-length word model is practically a 3-gram model, since 95% of the model consists of 3-grams, 2-grams, and 1-grams.

It should be noted that the approach used for modeling word boundaries affects the context length for the n-gram models. Instead of using a separate word boundary morph, it is possible to append a word boundary character at the end of some morphs. Avoiding a separate word boundary morph allows the n-gram model to use longer context. On the other hand, this approach increases the size of the morph set, since two variants are required for the morphs that can have occurred both in the middle and in the end of a word. In Finnish, two variants would be needed for almost all morphs, so we have chosen to use a word boundary morph.

E. Lattice Rescoring Experiments

Since all speech recognition systems do not support one-pass recognition with high-order n-gram models directly, a common approach is to use 2-gram or 3-gram models to generate lattices and rescore them with higher-order models. In word-based English word recognition, this is a standard approach, but in morph-based recognition, a low-order model may be too weak if the model contains very short morphs. In order to study the issue, we ran a lattice rescoring experiment using the Finnish 50 k morph set. Lattices were first generated using 2-gram and 3-gram models and then rescored using the largest variable-length model (rescoring the LM probabilities only). For reference, we also generated lattices using the same variable-length model that was used in rescoring. The results are shown in Table IV.

Rescoring the lattices generated with the 2-gram model seems to give worse results than direct one-pass recognition with the high-order model. The 3-gram lattices give better results, but still the error rates do not reach the level of the one-pass recognition. When the lattices are generated with the high-order model, rescoring gives essentially the same result as the one-pass recognition.

One potential source of the degraded results could be the morph-pair approximation (word-pair approximation in word-based recognition) used in the lattice generation algorithm. The algorithm assumes that the best alignment of the morph depends only on the previous morph, and since the morphs can be very short, the assumption may not be valid. However, since using a large LM in lattice generation removes the problem, the source of the degradation should be elsewhere. A more probable reason is that the morph 2-gram and 3-gram are too weak models, and some essential hypotheses are pruned before they end up in the lattice. Generating the lattices with wider pruning beams might alleviate the problem, but then the first-pass recognition would already become slower than the one-pass recognition.

F. Recognition of Unseen Words

Since the word-based LM can correctly recognize the words included in the vocabulary only, and the morph-based LMs are, in theory, able to recognize unseen words, it is interesting to examine how well the models perform on different regions of the evaluation data. The words of the target language can be divided into three categories depending on whether they are present in the LM training data and the vocabulary of the word LM [9]:

1) *In*: Words that are present in the training data and in the vocabulary.
2) *Out*: Words present in the training data but not in the vocabulary.
3) *New*: Words unseen in the training data.

When the training data is used to estimate the model, the word LM utilizes only the words in the first category and ignores the two other categories. The morph LMs, on the other hand, utilize both the first and the second category directly, and depending on the morph structure, may also be able to model some of the words in the third category well.

In order to compute letter error rates for the categories, the words in the reference transcripts were partitioned to regions according to the above categories. The recognition outputs were aligned to the reference transcripts letter-wise, and the letter insertion, deletion, and substitution errors were assigned to different categories according to the letter alignment. Fig. 3 shows the error rates for the categories and, for reference, the total error rate (same as in Fig. 2). The number of words in different regions are shown in Table V.

The results are in line with the results reported earlier [9], even if the current experimental setup for Finnish is somewhat different from the previous one: the task is now acoustically more difficult (speaker-independent, telephone speech corpus), the decoder can fully handle triphones across morph and word boundaries, and higher-order variable-length n-gram models are used. The word model performs slightly better in the "In" category, but the morph models perform considerably better in the "Out" category. This is expected because the morph models are trained on all words of the training data, while the word model ignores the OOV words completely in training. The "New" category seems to be harder, but still the morphs give lower error rates compared to words.

In general, the results in the figure suggest that a morph model may be a better choice when it is expected that the language of the test material does not match the training data very well.

Authorized licensed use limited to: Teknillinen Korkeakoulu. Downloaded on April 7, 2009 at 11:11 from IEEE Xplore. Restrictions apply.

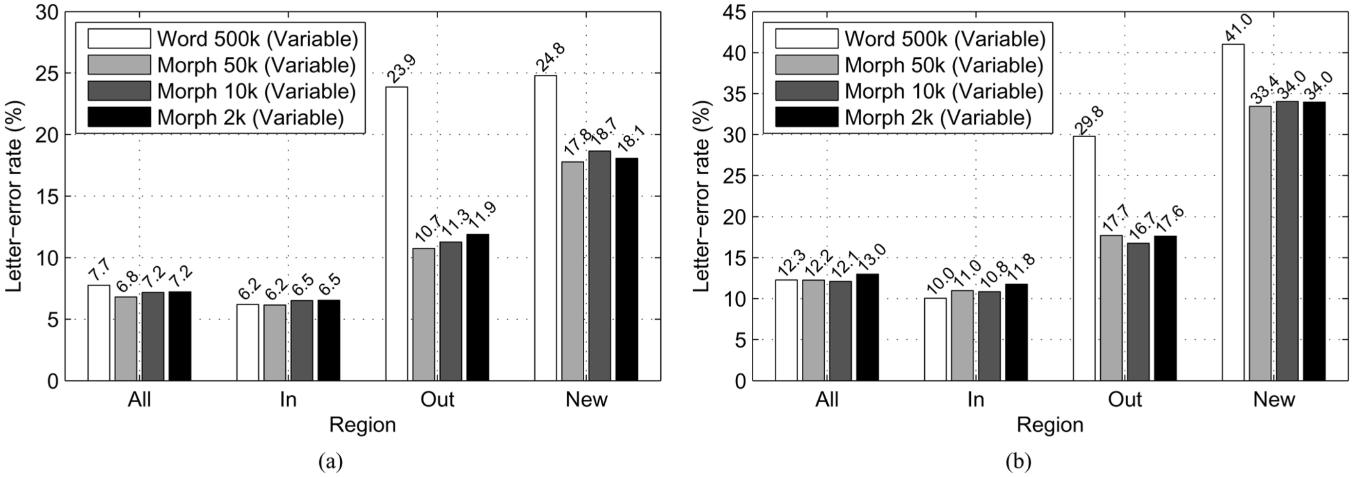


Fig. 3. Letter error rates in different regions of the evaluation data for Finnish (a) and Estonian (b). All: all words; In: words in the word vocabulary; Out: words in the language model training data but not in the vocabulary; New: words not in the training data. Note that majority of the test data belongs to "In" region (see Table V). The LMs used in this experiment are the largest models for each lexicon (from Fig. 2).

TABLE V
STATISTICS OF THE TEST DATA

	Words	In [%]	Out [%]	New [%]
Finnish	4686	94.6	3.2	2.1
Estonian	3718	94.4	2.9	2.8

The more dissimilar the materials are, the more words fall in the "Out" and "New" categories. Since the morph models outperform word models in these categories, the morph models are expected to perform better overall.

G. Short Units Versus Long Units

In many studies, it has been suggested that short morphs may cause acoustic confusability and degrade recognition results. The results shown in this article, however, suggest that the short morphs are not problematic if high-order n-gram models are used. On the other hand, if the context length of the model is restricted to a 3-gram, the performance of short morphs degrade considerably (see Fig. 2). After looking at the recognition outputs of the Finnish Morph 2 k models, it seems that the variable-length model almost never produces an OOV word when the correct word is from the training data. The 3-gram model produces slightly more errors in OOV words, but still the most of the additional mistakes the 3-gram model makes are just incorrect common words replacing other common words. This suggests that acoustic confusability is not the main problem, but that the 3-gram is just too weak a model for short morphs.

The optimal morph set for a specific task may depend on the decoder implementation. Short morphs require proper handling of across-unit phone contexts, and as demonstrated in the experiments, the use of high-order n-gram models already in the first pass is essential for best performance. The use of very large vocabularies may also be problematic due to large impact on the memory consumption of the recognition system. In our system, recognition with the word models takes up to 3–10 times more memory than recognition with the morph models. The overhead mainly comes from the larger recognition network and look-ahead cache tables.

V. CONCLUSION

The earlier research done on training speech recognition systems for agglutinative, inflecting and compounding languages has shown that dealing with rich morphology is not trivial. The proposed solutions include increasing the vocabulary size, segmenting words into smaller units, and utilizing part-of-speech tags, for example. Since the methods have been evaluated on different languages, and the experimental settings have been diverse, it can be hard to evaluate why a certain method works or does not work. The size of the vocabularies, the order of the n-gram models, the size of the training corpora, and the decoder implementation among other things affect the results.

The experiments carried out in this paper try to shed light on how important high-order language models are when recognition is based on morphs instead of words. The recognition experiments were performed on two highly inflective and agglutinative languages: Finnish and Estonian. Since it has earlier been shown that word-based recognition can be improved simply by increasing the size of the vocabulary, the word models on both languages utilized a very large vocabulary of 500 000 words. Also, in order to study what effect the lengths of morphs have in the recognition accuracy, models based on three different morph segmentations were created.

The main conclusions are the following. In the Finnish and Estonian tasks, high-order language models seem to be important if the recognition is based on morphs, and especially so, if the morphs are very short. The results also suggest that, even in two-pass recognition, the order of the first-pass language model may have a large effect. With short morphs, 2-gram and 3-gram models may be too weak for generating good lattices. Overall, in the Finnish task, the morph models seem to give better results than a word model using a very large vocabulary. In Estonian, the word model and the largest morph models performed equally.

Authorized licensed use limited to: Teknillinen Korkeakoulu. Downloaded on April 7, 2009 at 11:11 from IEEE Xplore. Restrictions apply.

The benefit of high-order morph models compared to word models was analyzed by dividing the evaluation set in the in-vocabulary and out-of-vocabulary word regions. For words that are in the vocabulary of the word models, the morph models provide no gains, but for the out-of-vocabulary words they perform much better. The morph models are superior also for the most difficult out-of-vocabulary words that do not exist in the language model training corpus, either.

References

[1] G. Choueiter, D. Povey, S. F. Chen, and G. Zweig, "Morpheme-based language modeling for Arabic LVCSR," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2006, pp. I-1053–I-1056.

[2] B. Xiang, K. Nguyen, L. Nguyen, R. Schwartz, and J. Makhoul, "Morphological decomposition for Arabic broadcast news transcription," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2006, pp. I-1089–I-1092.

[3] J. Nouza, J. Ždánský, P. David, P. Červa, J. Kolorenč, and D. Nejedlová, "Fully automated system for Czech spoken broadcast transcription with very large (300 k+) lexicon," in *Proc. Interspeech*, 2005, pp. 1681–1684.

[4] T. Hirsimäki, M. Creutz, V. Siivola, M. Kurimo, S. Virpioja, and J. Pylkkönen, "Unlimited vocabulary speech recognition with morph language models applied to Finnish," *Comput. Speech Lang.*, vol. 20, no. 4, pp. 515–541, Oct. 2006.

[5] K. McTait and M. Adda-Decker, "The 300 k LIMSI German broadcast news transcription system," in *Proc. Eurospeech*, 2003, pp. 213–216.

[6] W. J. Byrne, J. Hajič, P. Krbec, P. Ircing, and J. Psutka, "Morpheme based language models for speech recognition of Czech," in *Proc. 3rd Int. Workshop Text, Speech, Dialogue (TSD)*, 2000, pp. 211–216.

[7] T. Rotovnik, M. S. Maučec, and Z. Kačič, "Large vocabulary continuous speech recognition of an inflected language using stems and endings," *Speech Commun.*, vol. 49, no. 6, pp. 437–452, Jun. 2007.

[8] H. Erdogan, O. Büyük, and K. Oflazer, "Incorporating language constraints in sub-word based speech recognition," in *Proc. IEEE Workshop Automat. Speech Recognition Understanding (ASRU)*, 2005, pp. 98–103.

[9] M. Creutz, T. Hirsimäki, M. Kurimo, A. Puurula, J. Pylkkönen, V. Siivola, M. Varjokallio, E. Arısoy, M. Saraçlar, and A. Stolcke, "Morph-based speech recognition and modeling of out-of-vocabulary words across languages," *ACM Trans. Speech Lang. Process.*, vol. 5, no. 1, Dec. 2007.

[10] K. Kirchhoff, D. Vergyri, J. Bilmes, K. Duh, and A. Stolcke, "Morphology-based language modeling for conversational Arabic speech recognition," *Comput. Speech Lang.*, vol. 20, no. 4, pp. 589–608, Oct. 2006.

[11] R. Sarikaya, M. Afify, and Y. Gao, "Joint morphological-lexical language modeling (JMLLM) for Arabic," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 2007, pp. IV-181–IV-184.

[12] W. Byrne, J. Hajič, P. Ircing, F. Jelinek, S. Khudanpur, P. Krbec, and J. Psutka, "On large vocabulary continuous speech recognition of highly inflectional language—Czech," in *Proc. Eurospeech*, 2001, pp. 487–489.

[13] T. Laureys, V. Vandeghinste, and J. Duchateau, "A hybrid approach to compounds in LVCSR," in *Proc. Int. Conf. Spoken Lang. Process.*, 2002, pp. 697–700.

[14] T. Alumäe, "Methods for Estonian large vocabulary speech recognition," Ph.D. dissertation, Tallinn Univ. of Technology, Tallinn, Estonia, 2006.

[15] M. Kurimo, A. Puurula, E. Arısoy, V. Siivola, T. Hirsimäki, J. Pylkkönen, T. Alumäe, and M. Saraçlar, "Unlimited vocabulary speech recognition for agglutinative languages," in *Proc. Human Lang. Technol. Conf. North Amer. Chapter Assoc. Comput. Linguist.*, 2006, pp. 487–494.

[16] A. Puurula and M. Kurimo, "Vocabulary decomposition for Estonian open vocabulary speech recognition," in *Proc. 45th Annu. Meeting Assoc. Comput. Linguist.*, 2007, pp. 89–95.

[17] V. Siivola, T. Hirsimäki, M. Creutz, and M. Kurimo, "Unlimited vocabulary speech recognition based on morphs discovered in an unsupervised manner," in *Proc. Eurospeech*, 2003, pp. 2293–2296.

[18] P. Geutner, "Using morphology towards better large-vocabulary speech recognition systems," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 1995, pp. 445–448.

[19] P. Mihajlik, T. Fegyó, Z. Tüske, and P. Ircing, "A morpho-graphemic approach for the recognition of spontaneous speech in agglutinative languages—Like Hungarian," in *Proc. Interspeech*, 2007, pp. 1497–1500.

[20] M. Szarvas and S. Furui, "Evaluation of the stochastic morphosyntactic language model on a one million word Hungarian task," in *Proc. Eurospeech*, 2003, pp. 2297–2300.

[21] M. S. Maučec, T. Rotovnik, and M. Zemljak, "Modelling highly inflected Slovenian language," *Int. J. Speech Technol.*, vol. 6, no. 3, pp. 245–257, Jul. 2003.

[22] K. Hacioglu, B. Pellom, T. Ciloglu, O. Ozturk, M. Kurimo, and M. Creutz, "On lexicon creation for Turkish LVCSR," in *Proc. Eurospeech*, 2003, pp. 1165–1168.

[23] E. Arısoy, H. Dutağacı, and L. M. Arslan, "A unified language model for large vocabulary continuous speech recognition of Turkish," *Signal Process.*, vol. 86, no. 10, pp. 2844–2862, Oct. 2006.

[24] E. Arısoy, H. Sak, and M. Saraçlar, "Language modeling for automatic Turkish broadcast news transcription," in *Proc. Interspeech*, 2007, pp. 2381–2384.

[25] M. Kurimo, M. Creutz, M. Varjokallio, E. Arısoy, and M. Saraçlar, "Unsupervised segmentation of words into morphemes—Morpho challenge 2005: Application to automatic speech recognition," in *Proc. Int. Conf. Spoken Lang. Process.*, 2006, pp. 1021–1024.

[26] M. Kurimo, M. Creutz, and M. Varjokallio, "Morpho challenge evaluation using a linguistic gold standard," in *Proc. CLEF 2007 workshop*, 2008, vol. 5152. Springer, pp. 864–872.

[27] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," *Comput. Speech Lang.*, vol. 13, no. 4, pp. 359–393, Oct. 1999.

[28] A. Stolcke, "Entropy-based pruning of backoff language models," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, 1998, pp. 270–274.

[29] V. Siivola, T. Hirsimäki, and S. Virpioja, "On growing and pruning Kneser–Ney smoothed n-gram models," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 15, no. 5, pp. 1617–1624, Jul. 2007.

[30] M. Creutz and K. Lagus*, Unsupervised Morpheme Segmentation and Morphology Induction From Text Corpora Using Morfessor 1.0*. Espoo, Finland: Helsinki Univ. of Technology, Publications in Computer and Information Science A81, 2005.

[31] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," *Comput. Speech Lang.*, vol. 16, no. 1, pp. 69–88, Jan. 2002.

[32] J. Pylkkönen, "An efficient one-pass decoder for Finnish large vocabulary continuous speech recognition," in *Proc. 2nd Baltic Conf. Human Lang. Technol.*, 2005, pp. 167–172.

[33] S. J. Young, N. H. Russell, and J. H. S. Thornton, "Token passing: A simple conceptual model for connected speech recognition systems," Cambridge Univ. Eng. Dept., 1989, Technical report.

[34] S. Ortmanns, H. Ney, and A. Eiden, "Language-model look-ahead for large vocabulary speech recognition," in *Proc. Int. Conf. Spoken Lang. Process.*, 1996, pp. 2095–2098.

[35] X. Aubert and H. Ney, "Large vocabulary continuous speech recognition using word graphs," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, 1995, pp. 49–52.

[36] Finnish text collection 2004 [Online]. Available: http://www.csc.fi/kielipankki/

[37] E. Meister, J. Lasn, and L. Meister, "Development of the Estonian SpeechDat-like database," in *Proc. Eurospeech*, 2003, pp. 1601–1604.

Teemu Hirsimäki received the M.Sc. degree in computer science from the Helsinki University of Technology, Espoo, Finland, in 2002, where he is currently pursuing the Ph.D. degree.

Since 2000, he has worked in the Speech Group of Adaptive Informatics Research Center, Helsinki University of Technology. His research interest are language modeling and decoding in speech recognition.

Authorized licensed use limited to: Teknillinen Korkeakoulu. Downloaded on April 7, 2009 at 11:11 from IEEE Xplore. Restrictions apply.

Janne Pylkkönen received the M.Sc. degree in computer science from the Helsinki University of Technology, Espoo, Finland, in 2004.

He has been working in the Speech Group of Adaptive Informatics Research Center, Helsinki University of Technology, since 2003. His research interests are decoding and acoustic modeling in speech recognition.

Mikko Kurimo (SM'07) received the Dr.Sc. (Ph.D.) in technology degree in computer science from the Helsinki University of Technology, Espoo, Finland, in 1997.

He is currently Chief Research Scientist and Adjunct Professor in the Department of Information and Computer Science (previously Laboratory of Computer and Information Science), Helsinki University of Technology. His research interests are in machine learning, speech recognition, information retrieval, natural language processing, and multimodal interfaces.

Authorized licensed use limited to: Teknillinen Korkeakoulu. Downloaded on April 7, 2009 at 11:11 from IEEE Xplore. Restrictions apply.