Automatic Compound Word Reconstruction for Speech Recognition of Compounding Languages

Tanel Alumäe
Laboratory of Phonetics and Speech Technology
Institute of Cybernetics at Tallinn University of Technology
Estonia
tanel.alumae@phon.ioc.ee

Abstract

This paper compares two approaches to lexical compound word reconstruction from a speech recognizer output where compound words are decomposed. The first method has been proposed earlier and uses a dedicated language model that models compound tails in the context of the preceding words and compound heads only in the context of the tail. A novel approach models imaginable compound particle connectors as hidden events and predicts such events using a simple N-gram language model. Experiments on two Estonian speech recognition tasks show that the second approach performs consistently better and achieves high accuracy.

1 Introduction

In many languages, compound words can be formed by concatenating two or more word-like particles. In Estonian (but also in other languages, such as German), compound words occur abundantly and can even be built spontaneously. In a corpus of written Estonian consisting of roughly 70 million words, the number of different word types (including inflected words forms) is around 1.7 million and among those, around 1.1 million (68%) are compound words.

In large vocabulary continuous speech recognition (LVCSR) systems, an N-gram statistical language model is used to estimate prior word probabilities in various contexts. The language model vocabulary specifies which words are known to the system and therefore can be recognized. However, the large amount and spontaneous nature of compound words makes it difficult to design a language model that has a good coverage of the language. In addition, when vocabulary is increased, it becomes more difficult to robustly estimate language model probabilities for all words in different contexts. In order to decrease the lexical variety and the resulting out-of-vocabulary (OOV) rate, compound words can be split into separate particles and modeled as separate language modeling units. As a result however, the output of the recognizer consists of a stream of non-compound units that must later be reassembled into compound words where necessary.

In this paper, we compare the accuracy of two different methods for compound word reconstruction from recognizer output. The first model was proposed by Spies (1995) and is based on the assumption that a compound word can be decomposed into its first part(s) and the tail part. The predictive effect of the preceding context is only applied to the tail of the compound word. The head part, on the other hand, is assumed to be independent of the preceding context and its probability is calculated given only the tail. The second approach treats imaginable connectors between compound word particles as hidden events in the language model. Such a language model is typically used for sentence segmentation of conversational speech based on recognized words (Stolcke and Shriberg, 1996), but can be generalized for detecting other hidden events between recognized units. The latter approach is in essence similar to the method used in the morph-based speech recognition system described in (Siivola et al., 2003), except that they model word boundaries, not compound word connectors as seperate units, and do it already in the decoder.

Joakim Nivre, Heiki-Jaan Kaalep, Kadri Muischnek and Mare Koit (Eds.)
NODALIDA 2007 Conference Proceedings, pp. 5–12

The paper is organized as follows. In section 2 we describe the approach to statistical large vocabulary language modeling for Estonian. Section 3 describes the two approaches for compound word reconstruction in more detail. Results of a variety of experiments are reported in section 4. Some interesting error patterns are identified and analyzed. We end with a conclusion and some suggestions for future work.

2 Language modelling for Estonian

Estonian is an agglutinative and highly inflective language. One or many suffixes can be appended to verb and noun stems, depending on their syntactic and semantic role in the sentence.

Estonian is also a so-called compounding language, i.e. compound words can be formed from shorter particles to express complex concepts as single words. For example, the words *rahva* 'folk' and *muusika* 'music' can be combined to form a word *rahvamuusika* 'folk music' and this in turn can be combined with the word *ansambel* to form *rahvamuusikaansambel* 'folk music group'.

As a result, the lexical variety of Estonian is very high and it is not possible to achieve a good vocabulary coverage when using words as basic units for language modelling. Figure 1 compares the out-of-vocabulary (OOV) rates of three different vocabularies: words, words after decompounding, and after full morphological decomposition. The vocabularies are selected from a corpus described in section 4.1 and the OOV-rates are measured against a set of sentence transcripts used for speech recognition. The OOV-rate was measured using varying vocabulary sizes.

It is clear from the experiments that neither words nor decompounded words are suitable for language modelling using a conventionally sized vocabulary. The OOV-rate of the word-based vocabularies is much over what can be tolerated even when using a very large 800K size vocabulary. It can be seen that after splitting the compound words, the OOV-rate is roughly halved. Still, even when using a large 100K vocabulary, the OOV-rate is about 6% – too much to be used in large vocabulary speech recognition. However, the OOV-rates of morphemes is much lower and can be compared with the OOV-rates of English word-based vocabularies of similar sizes. The OOV-rate of the morpheme-based vocabulary reaches the 2% threshold already when using a 40K vocabulary.

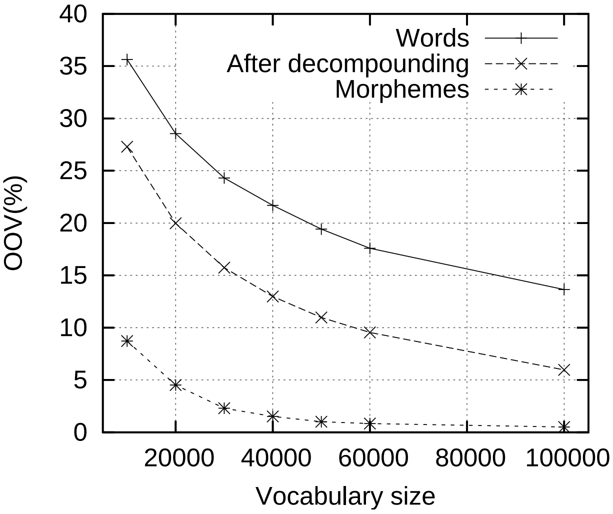

Figure 1: Out-of-vocabulary rate of different vocabularies.

When using morphemes as basic units for language modelling, the output of the decoder is a sequence of morphemes. The set of different suffix morphemes is rather small and thus the suffixes can be tagged in the vocabulary so that they can be concatenated to the previous stem after decoding. However, this approach can not be applied for reconstructing compound words as the set of stems and morphemes that take part in forming compound words is very large and sparse. The rest of the paper describes and compares two methods that attempt to reconstruct compound words from the sequence of morphemes.

3 Methods

This section describes two independent approaches to compound word reconstruction. Both of those methods enable us to compute a posterior probability of a compound word once its subsequent composing words have been recognized.

3.1 Compound word language model

The compound word language model proposed by Spies (1995) is based on the observation that the grammatically determining part of a compound word in many languages is the last particle. This is true for both German, for which the model was originally developed, as well as for Estonian. The head words of a compound may be considered as semantic modifiers of the last particle.

This observation suggests that when calculating language model scores for compound words, the predictive effect of the preceding context should be applied only to the tail part of the compound, while the probabilities of head words are computed given the tail. Let h_1 denote the first head of a compound, $h_2...h_n$ the (optional) remaining heads, t the tail of part of the compound and w_1w_2 the two preceding context words. Then, the total probability of a compound word $h_1..h_nt$ given the two preceding words w_1w_2 can be calculated as

$$P(h_1..h_nt|w_1w_2) =$$
$$P_h(h_1)\prod_{i=2}^{n}P_h(h_i|h_{i-1})\frac{P_{bw}(h_n|t)P_{tail}(t|w_1w_2)}{P(h_n)}$$

Here, $P_h(h_i|h_{i-1})$ is the within-head bigram probability, i.e., the probability that the compound head h_i occurs after the compound head h_{i-1}. $P_{bw}(h_n|t)$ is the backward bigram probability of compound head h_n followed by tail t, i.e., the probability of the last head given the tail. $P_{tail}(t|w_1w_2)$ is the distant trigram probability of the compound tail, i.e. the probability of a compound ending with the tail t given the last two context words. The given equation consists of two parts: the first part amounts to a simple bigram probability of the compound head sequence, independent of the observed context, while the last fraction expresses the distant trigram probability of the tail, multiplied by the gain in probability of the last head due to the observed tail. See the original proposal of this model (Spies, 1995) for more details about the derivation of this equation.

Given a sequence of recognized units (that are either true words or compound particles), the most probable reconstruction is found as follows:

1. Any unit can be regarded as a non-compound part. Unit probability is then calculated using the trigram distribution.
2. In case the unit has occurred as a compound head in the training corpus, a new compound branch is created. The compound branch continues as follows:
   (a) If the next word is again a head candidate, a new compound branch is created, and the processing in the new branch is continued as in step 2.
   (b) If the next word is a compound tail candidate, a new possible compound word has been found. The compound word probability is calculated according to the compound word model equation. Processing in this branch continues as in step 1.
   (c) If the next word is neither a head nor a tail candidate, the current branch is discarded.
3. The most probable reconstruction of a sentence is the one that corresponds to the path with the highest product score.

3.2 Hidden event language model

The hidden event language model (Stolcke and Shriberg, 1996) describes the joint distribution of words and events, $P_{LM}(W, E)$. In our case, words correspond to the recognized units and events to the imaginable interword compound particle connectors. Let W denote the recognized tokens $w_1, w_2, ..., w_n$ and E denote the sequence of interword events $e_1, e_2, ..., e_n$. The hidden event language model describes the joint distribution of words and events, $P(W, E) = P(w_1, e_1, w_2, e_2, ..., w_n, e_n)$.

For training such a hidden event language model, a training corpus is used such that the compound words are decomposed into separate units, and the compound connector event is represented by an additional nonword token (<CC>), for example:

```
gruusia rahva <CC> muusika <CC>
ansambel andis meelde <CC> jääva
kontserdi
```

'Georgian folk music group gave a memorable concert'.

The language model used for recognition is trained on the corpus where the compound connector tags are removed. The vocabulary of the compound reconstruction language model is the same as that of the main language model, with an additional token "<CC>". We do not explicitly model the "non-CC" event in order to make more effective use of the contextual information. During compound reconstruction, the Viterbi algorithm is used to find the most likely sequence of words and hidden tokens for the given input sequence. The word/event pairs correspond to states and the words to observations, and the transition probabilities are given by the the hidden event N-gram model.

4 Experiments

4.1 Training data

We tested the concepts and algorithms described here using two different Estonian speech databases, BABEL and SpeechDat.

The Estonian subset of the BABEL multi-language database (Eek and Meister, 1999) contains speech recordings made in an anechoic chamber, directly digitized using 16-bits and a sampling rate of 20 kHz. The textual content of the database consists of numbers, artificial CVC-constructs, 5-sentence mini-passages and isolated filler sentences. The isolated sentences were designed by phoneticians to be especially rich in phonologically interesting variations. The sentences are also designed to reflect the syntactic and semantic complexity and variability of the language. For training acoustic models, the mini-passage and isolated sentence recordings of 60 speakers were used, totalling in about 6 hours of audio data. For evaluation, 138 isolated sentence utterances by six different speakers were used.

The SpeechDat-like speech database project (Meister et al., 2002) was aimed to collect telephone speech from a large number of speakers for speech and speaker recognition purposes. The main technical characteristics of the database are as follows: sampling rate 8 kHz, 8-bit mono A-law encoding, calls from fixed and cellular phones as the signal source, calls from both home and office environments. Each recording session consists of a fixed set of utterance types, such as isolated and connected digits, numbers, money amounts, spelled words, time and date phrases, yes/no answers, proper names, application words and phrases, phonetically rich words and sentences. The database contains about 241.1 hours of audio data from 1332 different speakers. For recognition experiments, the database was divided into training, development and test set. The development and test sets were chosen by randomly assigning 40 different speakers to each of the sets. To avoid using the same speaker's data for both training and evaluation, those 80 speakers were chosen out of those contributors who only made one call session. Only the prompted sentence utterances were used in evaluations, thus both the development and test set contained 320 utterances.

For training language models, we used a the following subset of the Mixed Corpus of Estonian (Kaalep and Muischnek, 2005), compiled by the Working Group of Computational Linguistics at the University of Tartu: daily newspaper “Postimees” (33 million words), weekly newspaper “Eesti Ekspress” (7.5 million words), Estonian original prose from 1995 onwards (4.2 million words), academic journal “Akadeemia” (7 million words), transcripts of Estonian Parliament (13 million words), weekly magazine “Kroonika” (0.6 million words).

4.2 LVCSR system

The CMU Sphinx (Placeway et al., 1997) speech recognition system was used used for speech recognition experiments. The latest version of SphinxTrain was used for training and Sphinx 3.6.3 was used for decoding test utterances. For acoustic features, MFCC coefficients were used, extracted from a window of 0.0256 seconds with a frame rate of 100 frames/second. All acoustic units are modeled by continuous left-to-right HMMs with three emitting states and no skip transitions. The output vectors are 39-dimensional and are composed of 13 cepstral coefficients, delta and double delta coefficients. Data-driven decision trees were used for creating tied-state triphone models. Each state is modeled by 8 Gaussian mixture components. The BABEL-based acoustic models use a sample rate of 16 kHz, the number of senones is fixed to 3000. The SpeechDat-based models use a sample rate of 8000 Hz, a frequency band of 130 Hz - 3400 Hz, and the number of senones was fixed to 6000. Models were created for 25 phonemes, silence, and five filler/noise types (the latter only for the SpeechDat-based system). Long phonemes as well as diphthongs are modelled by sequences of two corresponding phone units. The only exception in the handling of short and long phonemes lies in the modelling of plosives since the realization of long plosives is clearly different from concatenation of two short plosives. Therefore, we model short and long plosives using separate units. Pairs of palatalised and unpalatalised phonemes are merged into one acoustic unit.

The SRILM toolkit (Stolcke, 2002) was used for selecting language model vocabulary and compiling the language model. The language model was created by first processing the text corpora using the Estonian morphological analyzer and disambiguator (Kaalep and Vaino, 2001). Using the information from morphological analysis, it is possible to split compounds words into particles and separate morphological suffixes from preceding stems. Language model vocabulary was created by selecting the most likely 60 000 units from the mixture of the corpora, using sentences in the SpeechDat training set as heldout text for optimization. The resulting vocabulary has a OOV-rate of 2.05% against the sentences in the BABEL test set and 2.20% against the sentences in the SpeechDat test set. Using the vocabulary of 60 000 particles, a trigram language model was estimated for each training corpus subset. The cutoff value was 1 for both bigrams and trigrams, i.e. singleton n-grams were included in the models. A modified version of Kneser-Ney smoothing as implemented in SRILM was applied. Finally, a single LM was built by merging the six models, using interpolation coefficients optimized on the sentences in the SpeechDat training set.

Since Estonian is almost a phonetic language, a simple rule-based grapheme-to-phoneme algorithm described in (Alumäe, 2006) could be used for generating pronunciations for both training data as well as for the words in the language model used for decoding. The pronunciation of foreign proper names deviates obviously from rule-based pronunciation but since our test set did not contain many proper names, we limited the amount of proper names in the vocabulary to most frequent 500, which were mostly of Estonian origin. No manual correction of the pronunciation lexicon was done.

4.3 Training models for compound word reconstruction

The models for compound word reconstructions were estimated using the morphologically analyzed corpora, that is, words were split into morphemes and compound word connector symbols marked places where compound words are formed.

The compound word language model consists of three sub-models: the distant trigram model, inner-compound head bigram model and head-given-tail bigram model. All given models were trained over the union of the text corpora as follows: for training the distant trigram model, all head compound particles were removed from the texts and a trigram language model was estimated; for training the inner-compound head bigram, all compound head sequences were extracted from the corpus, and a bigram language model was estimated; for training the head-given-tail bigram model, all compounds were extracted from the corpus, all but the last head and tail were removed from the compound words, the remaining word pairs were reversed and a bigram language model was estimated. In all cases, modified Kneser-Ney smoothing using a cutoff value of 2 was applied.

For training the hidden event language model, we took the same vocabulary as was used for training the main language model, added the compound connector symbol to it, and estimated a trigram model over the union of the subcorpora, using a cutoff value of 2 and Kneser-Ney smoothing.

4.4 Evaluation metrics

We tested both compound word models on two kinds of test data:

- reference transcripts, split into morphemes. This corresponds to perfect recognizer output;
- actual recognizer output, consisting of recognized morphemes.

To evaluate the accuracy of reconstructing compound words in reference transcripts, the reconstructed sentences were simply compared with original sentences. However, it is not obvious what to use as reference when evaluating reconstruction of recognizer output. We chose to use dynamic programming for inserting compound word connectors in the recognizer output by aligning the recognized units with reference units and inserting compound word connectors according to their location in reference transcripts. This approach however sometimes inserts compound word connectors in places where they are linguistically not legitimate. For example, consider a reference sentence

```
.. pälvis suure tähele <CC> panu
```

and the recognized token stream

```
... pälvis suure tähele PANNA
```

Test set	Model	Inserted tags	Precision	Recall	F measure	WER
BABEL	Compound word LM	154	0.64	0.83	0.72	8.2
	Hidden event LM	122	0.82	0.85	0.83	4.4
Speechdat	Compound word LM	395	0.84	0.89	0.86	6.5
	Hidden event LM	352	0.89	0.94	0.91	4.2

Table 1: Compound word connector tagging accuracies and the resulting would-be word error rate resulting from incorrect tagging, given perfect morpheme output by the decoder.

According to the alignment, the token *tähele* and the misrecognized token *panna* should be recomposed, although in reality, those two words often occur together and are never written as a compound word (as opposed to *tähele* and *panu* which are always written as a compound word).

For measuring compound reconstruction accuracy, we calculated compound connector insertion precision and recall. Precision is defined as a measure of the proportion of tags that the automatic procedure inserted correctly:

$$P = \frac{t_p}{t_p + f_p}$$

where t_p is the number of correctly inserted tags (true positives) and f_p the number of incorrectly inserted tags (false positives). Recall is defined as the proportion of actual compound word connector tags that the system found:

$$R = \frac{t_p}{t_p + f_n}$$

where f_n is the number of tags that the system failed to insert (false negatives).

Precision and recall can be combined into a single measure of overall performance by using the F measure which is defined as follows:

$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} \overset{[\alpha=0.5]}{=} \frac{2PR}{P+R}$$

where α is a factor which determines the relative importance of precision versus recall.

Another measure we used was the word error rate, calculated after compound word reconstruction, after alignment with the original reference transcripts. Word error rate is calculated as usual:

$$WER = \frac{S+D+I}{N}$$

where S is the number of substitution errors, D the number of deletion errors, I the number of insertion errors and N the number of words in the reference.

4.5 Results

As the first test, the method was tested on the reference transcripts from the BABEL and SpeechDat speech databases. The input consists of morphemes where compound word connectors are deleted. Results are shown in table 1.

As can be seen, the hidden event language model does better than the compound word language model. The latter seems to have a big problem with overgenerating compound words which lowers the precision figures.

The second test analyzed compound word reconstruction, given the recognized hypotheses from the decoder. Results are listed in table 2. The table also gives the “oracle” WER for each test set, that is, the WER given the perfect compound word reconstruction based on alignment with reference sentences.

The precision and recall of the models is much lower than when using reference sentences as input. This is expected, as often one particle of a compound word is misrecognized which “confuses” the models and gives them no reason to suggest a compound word.

For both test sets, the hidden event language model performed better in terms of both precision/recall as well as the final WER. The relative improvement in WER of the hidden event language model over the compound word language model was 5.0% for the BABEL test set and 4.7% for the SpeechDat test set.

4.6 Analysis

Table 3 lists some sentences from the SpeechDat test set that contain mistakenly compounded or uncom-

Test set	Model	Inserted tags	Precision	Recall	F measure	WER	Oracle WER
BABEL	Compound word LM	160	0.54	0.73	0.62	31.7	28.9
	Hidden event LM	123	0.67	0.70	0.68	30.2	
Speechdat	Compound word LM	378	0.66	0.67	0.66	44.2	40.0
	Hidden event LM	338	0.74	0.67	0.70	42.2	

Table 2: Compound word connector tagging accuracies and the resulting word error rate compared to the "oracle" word error rate, given the actual recognized hypotheses from the decoder.

pounded words, using the hidden event LM. The errors are written in upper case and the correct words are written in the right column. Quick investigation reveals at least three common patterns where compound recomposition errors occur:

1. a compound word is not recognized when both of the compound word particles are very infrequent: the result is that there is not enough occurrences of the pair, nor occurrences where the head word is a head in a compound, neither where the tail word is a tail in a compound; as a result, the statistical model has no reason to insert a compound connector between them (e.g. *piirde-tross*, *traks-tunkedes*, *ainu-autorsusest*, *broiler-küülik*)
2. two words are mistakenly recognized as a compound word when the first word is often a head word in compound words, and/or the second word is often a tail word in compound words, although their pair may actually never occur as a compound, and it also does not occur as an uncompounded pair often enough (e.g. *suur laud / suur-laud*, *kuue meetri / kuue-meetri*)
3. in some cases, words are mistakenly recomposed into a compound word when the fact that the words should be written separately comes from the surrounding context (e.g. *laulu looja / laulu-looja*, *kunsti tekke põhjuseks / tekke-põhjuseks*, *eri värvi osadest / värvi-osadest*). Those errors are probably the hardest to handle since the correct behavior would often require understanding of the discourse. Often, it is arguable whether the words should be written as a compound or not (e.g. *tekke-põhjuseks*, *taime-seemnetes*.

Manual analysis of the compounding errors of the SpeechDat reference texts shows that the majority of errors (around 60%) were of type 1. About 30% of the errors could be classified as context errors (type 3) and the rest (around 10%) were of type 2.

5 Conclusion

We tested two separate methods for reconstructing compound words from a stream of recognized morphemes, using only linguistic information. The first method, using a special compound word language model, relies on the assumption that the head part of a compound word is independent of the preceding context and its probability is calculated given only the tail. Probability of the tail, on the other hand, is calculated given the preceding context words. As an alternative approach, we proposed to use a trigram language model for locations of hidden compound word connector symbols between compound particles. Experiments with two test sets showed that the method based on hidden event language model performs consistently better than the compound word language model based approach.

The proposed compound word reconstruction technique could be improved. The analysis of reconstruction errors revealed two kinds of problems caused by data sparseness issues. Some of such issues could probably be eliminated by using a class-based language model. An added area for further study is to combine acoustic and prosodic cues, such as pause length, phone duration and pitch around the boundary between possible compound particles, with the linguistic model, as has been done for automatic sentence segmentation (Stolcke et al., 1998).

Acknowledgments

This research was partly funded by the Estonian Information Technology Foundation as part of the Tiigriülikool program and by the Estonian Associ-

Recognized	Actual
ühevärviline kostüüm pikendab teie figuuri samas kui eri VÄRVIOSADEST lühendab	.. VÄRVI OSADEST ..
üles pannakse uued liiklusmärgid PIIRDE TROSS tõmmatakse pingule	.. PIIRDETROSS ..
üheks kunsti TEKKEPÕHJUSEKS peetakse inimese tarvet ilu ja loomisrõõmu järele	.. TEKKE PÕHJUSEKS ..
väikeses ja pimedas kambris oli näha vaid voodi ja SUURLAUD	.. SUUR LAUD ..
väga soodsalt mõjuvad organismile tsitrused küüslauk ja TAIME SEEMNETES leiduvad ained	.. TAIMESEEMNETES ..
viis miljonit aastat tagasi VÄLJA SURNUD hiire fossiil oli üllatavalt hästi säilinud	.. VÄLJASURNUD ..
vaikne ja ennast ise kütusega varustav liikur on KUUEMEETRI pikkune silindriline puur	.. KUUE MEETRI ..
vaguniuksel istub taburetil õlistes TRAKS TUNKEDES naine	.. TRAKSTUNKEDES ..
vaesed MAA INIMESED said aru et see oli pogromm nende vastu	.. MAAINIMESED ..
LAULULOOJA oli huvitatud AINU AUTORSUSEST	LAULU LOOJA .. AINUAUTORSUSEST
kuigi broileriks nimetatakse noort kana saab maitsva prae ka BROILER KÜÜLIKUST	.. BROILERKÜÜLIKUST

Table 3: Some sample compound word reconstruction errors from the SpeechDat test set.

ation of Information Technology and Telecommunications.

References

Tanel Alumäe. 2006. *Methods for Estonian large vocabulary speech recognition.* Ph.D. thesis, Tallinn University of Technology.

Arvo Eek and Einar Meister. 1999. Estonian speech in the BABEL multi-language database: Phonetic-phonological problems revealed in the text corpus. In *Proceedings of LP'98. Vol II.*, pages 529–546.

Heiki-Jaan Kaalep and Kadri Muischnek. 2005. The corpora of Estonian at the University of Tartu: the current situation. In *The Second Baltic Conference on Human Language Technologies : Proceedings*, pages 267–272, Tallinn, Estonia.

Heiki-Jaan Kaalep and Tarmo Vaino. 2001. Complete morphological analysis in the linguist's toolbox. In *Congressus Nonus Internationalis Fenno-Ugristarum Pars V*, pages 9–16, Tartu, Estonia.

Einar Meister, Jürgen Lasn, and Lya Meister. 2002. Estonian SpeechDat: a project in progress. In *Fonetiikan Päivät 2002 — The Phonetics Symposium 2002*, pages 21–26.

P. Placeway, S. Chen, M. Eskenazi, U. Jain, V. Parikh, B. Raj, M. Ravishankar, R. Rosenfeld, K. Seymore, M. Siegler, R. Stern, and E. Thayer. 1997. Hub-4 Sphinx-3 system. In *Proceedings of the DARPA Speech Recognition Workshop*, pages 95–100.

Vesa Siivola, Teemu Hirsimäki, Mathias Creutz, and Mikko Kurimo. 2003. Unlimited vocabulary speech recognition based on morphs discovered in an unsupervised manner. In *Proceedings of Eurospeech*, Geneva, Switzerland.

M. Spies. 1995. A language model for compound words. In *Proceedings of Eurospeech*, pages 1767–1779.

A. Stolcke and E. Shriberg. 1996. Automatic linguistic segmentation of conversational speech. In *Proceedings of ICSLP*, volume 2, pages 1005–1008, Philadelphia, PA, USA.

A. Stolcke, E. Shriberg, R. Bates, M. Ostendorf, D. Hakkani, M. Plauche, G. Tur, and Y. Lu. 1998. Automatic detection of sentence boundaries and disfluencies based on recognized words. In *Proceedings of ICSLP*, volume 5, pages 2247–2250, Sydney, Australia.

Andreas Stolcke. 2002. SRILM – an extensible language modeling toolkit. In *Proceedings of ICSLP*, volume 2, pages 901–904, Denver, USA.