Initial Experiments With Estonian Speech Recognition

Anton Ragni
Department of Physics
University of Tartu
50090 Tartu, Estonia
ragni@ut.ee

Abstract

This paper presents a short description of work recently done at University of Tartu to construct a word–based speech recognition system. The experimental settings used in our experiments are very basic: a simple trigram language model with cross–word triphone acoustic models are used by one–pass best hypothesis recognizer to perform the decoding of test data. The best accuracy reported in this paper has a word error rate (WER) of 40.4% which is a common figure for languages like Estonian. The system described in this paper will be used as a baseline for our subsequent experiments on Estonian speech recognition.

1 Introduction

Estonian belongs to a family of inflectional and agglutinative languages – one of many other Slavic [Byrne et al., 2001, Maučec et al., 2003], Arabic [Kirchhoff et al., 2003, Choueiter et al., 2006] and Asian [Kwon et al., 1999, Sinha et al., 2006] languages which received a particular attention in recent years. A single base word form by means of inflections and compounding may have a huge number of derivative words. This greatly complicates the problem of building a speech recognition system with comparable WER performance to English systems. A common approach is to employ some type of subword systems, the goodness of which can be then compared to each other and/or a word–based system. This paper is devoted to the building of such word–based system and reports on results we obtained.

The first comprehensive description of work done on Estonian speech recognition appeared only recently [Alumäe, 2006]. A huge number of experiments is conducted on two databases: Estonian part of Babel multi–language database [Eek and Meister, 1998] and Estonian SpeechDat–like database [Meister et al., 2003]. The language modeling is performed both on a word and subword level. Our set of experiments is much more modest as compared to that work. However, we do not replicate the work already done but provide a completely independent set of results on Estonian part of Babel multi–language database.

The rest of the paper is organized as follows: in Section 2 we describe different language models we built and evaluate them on a testing part of corpus

	Training Part	Testing Part
Total words	76,823,686	1,182,376
Unique words	1,689,206	141,724
No. sentences	5,727,566	82,814

Table 1: Statistics of training and testing parts of MCE

used for language modeling. Section 3 introduces speech database we used for acoustic modeling. Audio data preprocessing is described comprehensively. Two ways for unit selection choice is presented. The section ends with a detailed description of acoustic model training procedure. Section 4 describes experimental settings and reports on results we obtained. Section 5 makes conclusions drawn from this study.

2 Language Modeling

2.1 Mixed Corpus of Estonian

Experimental work on language modeling is conducted on the Mixed Corpus of Estonian (MCE) – a set of written texts collected and maintained by University of Tartu [Computer Linguistics Group]. The training part of the corpus consists of articles from the daily newspapers "Eesti Express" (6.5M), "Postimees" (32M) and magazine "Horizont" (0.25M). Another significant part constitute translations of European Union and Estonian laws (9.5M), shorthand records from the state's assembly (12M) and a corpus of written language from the years 1890–1990 (4M). The total size of training corpus is approximately 64M words excluding such special tags like sentence beginning (`<s>`) and ending (`</s>`) symbols. The testing part of MCE is composed from the articles of "Postimees" from the year 1995 (0.5M) and 2001 (0.5M).

A single preprocessing strategy is applied throughout the corpora. Sentence boundaries are determined heuristically. All numbers are mapped to a common tag `<NUMBER>` since there is no known to us application capable of expanding them into verbal representations. For inflectional languages like Estonian this is not a trivial task since all numbers like any other part of speech should agree in number, case and gender with corresponding nouns.

Basic corpus statistics is given in Table 1. Average sentence length is 13.4 words for the training part of MCE and 14.2 for the testing part. The number of unique words in the training part is higher than the size of a typical vocabulary (65,000) by more than one order of magnitude.

2.2 Trigram Language Models

A number of competitive trigram language models is created and evaluated using the HTK toolkit [Young et al., 2006]. But first, the vocabulary is fixed to 65,000 most frequent words with addition of all words found in the transcriptions of training audio data. All the diversity of language models is obtained by application of different cut–off values to the number of bigrams and trigrams left in the model. The cutoff value specifies the least number of times any n–gram should have been seen in the training corpus to be included in the model.

Name	Cut–off	Bigrams	Trigrams	Size	PP
–	0	11,676,757	34,166,450	–	–
tg1-1	1	3,855,881	5,760,565	111.7	992.1
tg2-2	2	2,318,933	2,759,140	58.9	1067.6
tg3-3	3	1,697,321	1,814,180	41.0	1132.3
tg6-6	6	974,714	872,274	22.2	1278.8
tg10-10	10	635,555	507,714	14.3	1415.0
tg20-20	20	340,807	239,389	7.9	1640.4
tg30-30	30	230,928	152,495	5.6	1793.6
tg100-100	100	66,440	38,885	2.4	2317.3

Table 2: Parameters of trigram language models

Standard Good–Turing discounting is applied to refine parameters of language models. The discounting factor k is kept greater from the cutoff value by seven for both bigrams and trigrams.

Evaluation of language models is performed on the testing part of MCE. Table 2 provides information about number of n–grams, size in megabytes (ARPA–compatible textual representation) and perplexity for each created model. The first unnamed row in the table provides a reference for the total number of different n–grams found in the training corpus. The out–of–vocabulary (OOV) rate is 11.2% which means that on average each testing sentence contains at least one unknown word. High perplexities and OOV ratio originate from the well–known fact that for inflectional and agglutinative languages a sub–word language modeling as a rule is more appropriate in terms of memory size and perplexity figures [Whittaker and Woodland, 1998, Maučec et al., 2003, Hirsimäki et al., 2005].

3 Acoustic Modeling

3.1 Babel Multi–Language Speech Database

Experimental work on acoustic modeling is conducted on Estonian part of Babel speech database [Eek and Meister, 1998]. The database consists of three subsets:

- very few talker set – 2 talkers (1 male and 1 female)
- few talker set – 8 talkers (4 male and 4 female)
- many talker set – 60 talkers (30 male and 30 female)

The recordings are made from a set of 40 text passages, 2 sets of numbers and 4 sets of sentences with multiple occurrence of acoustically confusable words (e.g., *Lina* and *liina*, *türi* and *tüüri*) in a clean recording environment. The recorded speech is sampled at 20,000 Hz and digitized using 16-bit integers.

The training part in this study is composed from the very few and many talker sets. The few talker set is used for testing. Basic statistics for training and testing parts is summarized in Table 3.

no.	Training Part	Testing Part
passage sets	163	80
sentence sets	67	8
number sets	64	8
hours	7.4	1.2

Table 3: Statistics of training and testing parts of Babel Speech Corpus

3.2 Data Preprocessing

All data in this study is preprocessed using Mel–Frequency Cepstral Coefficients (MFCC) feature extraction scheme. Each audio data file is split into a number of speech segments with 10 ms duration called *frames*. A composite speech structure called *window* is constructed around any given frame. Each window consists of samples from previous, current and next frames. 7.5 ms of samples from previous frame is appended with 10 ms of samples from current frame and terminated by 7.5 ms samples from the next frame.

A preemphasis (first–order finite impulse response (FIR) filter) is applied to each window to increase amplitudes of high frequencies since laters are usually suppressed by the following processing stages. The preemphasis is defined by:

$$y_n = x_n - 0.97 \cdot x_{n-1}$$

where each window sample y_n is substituted by a weighted combination of two adjacent window samples x_n and x_{n-1}. Each window is then multiplied by a Hamming window to avoid introduction of non–existent frequencies due to the adverse effect of cutting speech segments out of continuous waveforms. The Hamming window is defined by

$$H(n; N) = 0.54 - 0.46 \cdot \cos\left(\frac{2\pi n}{N-1}\right)$$

where $N = 400$ is the length of window in samples. Finally, each window is appended with 112 zeros to apply radix–2 Fast Fourier Transform.

The frequency spectrum is integrated with 26 Mel–spaced band–pass filters to reduce the number of data points in each window from 256 to 26. The output of each band–pass filter equals the amount of energy contained in the frequency band where this filter is defined. Natural logarithm is taken from the output of filter banks to make the energy statistics approximately Gaussian. Finally, the Discrete Cosine Transform is applied to map the window into 13–dimensional cepstral space. The first cepstral coefficient is substituted with the log–energy value of current frame. The window is filtered (or liftered) for the final time using the following filter formula:

$$y_n = \left(1 + \frac{L}{2}\sin\left(\frac{\pi n}{L}\right)\right) \cdot x_n$$

where $L = 22$ controls the amount of filtering, x_n is initial and y_n is filtered coefficient.

The first (delta) and second order (delta–delta) derivatives are calculated for each cepstral window using the following regression formula:

$$\Delta c_n = \frac{\sum_{\theta=1}^{\Theta} \theta \cdot (c_{n+\theta} - c_{n-\theta})}{2 \cdot \sum_{\theta=1}^{\Theta} \theta^2}$$

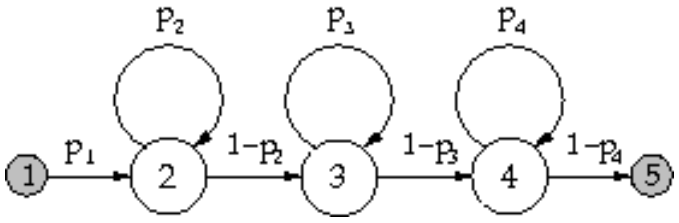

Figure 1: 3-state left-to-right HMM

where $\Theta = 2$ is a size of regression window. The final window called *observation vector* is constructed from 13 cepstral, 13 delta (Δ) cepstral and 13 delta–delta ($\Delta\Delta$) cepstral coefficients. These 39–dimensional vectors are used for acoustic model training in Section 3.4.

3.3 Unit Selection

The first step in acoustic modeling is to decide upon basic modeling units. There are many options to choose from: words, syllables, phonemes. The large vocabulary speech recognition is best done with phoneme units. There are two possible phoneme sets: orthographic and phonetic set. Experiments conducted on two different Estonian speech corpora revealed no preference in WER figures between these two representations [Alumäe, 2006]. The orthographic representation used in this study is based on the letters of Estonian alphabet with some minor modifications to the loaned letters such as *c, q, x,* etc. These letters are substituted with a sequence of common letters following the generic rules of Estonian pronunciation.

There are 32 letters in Estonian alphabet and 27 of them are considered to be common letters. The remaining 5 letters are substituted with one or more letters from the first set. In addition to these 27 models two models are created for representing *short pause* (usually between two words) and *silence* (usually between two phrases or sentences) events. Thus the monophone set consists of 29 models:

```
a, b, d, e, f, g, h, i, j,
k, l, m, n, o, p, r, s, sh,
z, zh, t, u, v, io, ae, oe, ue,
sp, sil
```

where `sh` corresponds to *š* letter, `zh` to *ž*, `io` to *õ*, `ae` to *ä*, `oe` to *ö* and `ue` to *ü*.

3.4 Acoustic Models

A single 3–state left–to–right hidden Markov model (HMM) is constructed for each monophone except for short–pause (`sp`) model. Fig 1. shows the topology of all 3–state HMM models. The mean and covariance are initialized from the global mean and covariance computed from the all available training data. Parameters of monophone models are reestimated using embedded Baum–Welch training procedure three times. Once the models are trained the 1–state `sp` model is created and its single state is tied with the center state of silence model

Model	Passages	Numbers	Sentences	Total
tg100-100	56.2%	8.1%	41.2%	45.5%
tg30-30	55.7%	7.5%	41.0%	44.2%
tg20-20	55.5%	7.4%	40.6%	44.9%
tg10-10	55.3%	7.3%	40.5%	43.9%
tg6-6	55.2%	7.2%	40.1%	43.9%
tg3-3	54.8%	7.1%	40.0%	43.6%
tg2-2	54.6%	6.9%	39.9%	43.4%
tg1-1	54.5%	6.5%	39.6%	43.3%

Table 4: Word error rates for different parts of testing set

(`sil`). In the `sp` model there is a direct transition between initial and final state so no observations need to be produced to traverse the model. Two additional transitions are created for the `sil` model to connect the second and fourth states in both directions. The parameters of monophone set are reestimated two more times, however, pronunciation of each word is appended with the `sp` model. For example, the vocabulary word *isa* will be transformed into the 4–phoneme sequence `i, s, a, sp`.

Once the monophone models are trained, the next stage of training procedure is to create a set of cross–word triphone models. Transcriptions of training audio files are used to induce the initial set of triphone models the parameters of which are re–estimated two times. The vocabulary of language model is used to produce the additional set of models which may be required during the recognition of test utterances. Parameters of all triphone models are tied using a phonetic decision–tree state tying procedure [Young et al., 1994] implemented in HTK. The tied set of triphones is trained twice to produce the final set of single mixture models.

The number of mixtures is gradually increased using the following strategy: the probability density function (pdf) of each state is copied into a new mixture component. Weights of both mixtures are divided by two and a mean vector of pdf is shifted away by -0.2 and $+0.2$ standard deviations for the first and second mixture component. Each time a new mixture is created the parameters of triphone models are reestimated three times. We have created 8 sets of triphone models each with a distinct number of mixture components in it ($1, 2, \ldots, 8$ mixtures).

4 Experiments

We have conducted a number of experiments for exploring different language models built in Section 2. The testing set of Babel multi–language database (see Section 3.1) is split in parts to evaluate the performance of speech recognizer on different types of speech data: text passages, numbers, random sentences. A large vocabulary speech recognizer implemented in HTK toolkit (`HDecode`) is used to transcribe test sentences. Table 4 gives WER figures for different parts of the testing set. As it can be noted the lowest error rates are obtained on the set of numbers (WER $< 9\%$). Sets of passages and sentences have the WER figure of the same order of magnitude ($40\% <$ WER $< 60\%$). The difference in size between the first and last language model is more than 100 MB, however,

Model	Deletions	Substitutions	Insertions	WER
tg100-100	312	2428	578	45.5%
tg30-30	417	2366	439	44.2%
tg20-20	320	2390	562	44.9%
tg10-10	418	2350	431	43.9%
tg6-6	418	2348	429	43.9%
tg3-3	419	2330	426	43.6%
tg2-2	418	2321	422	43.4%
tg1-1	422	2309	421	43.3%

Table 5: Number of deletion, substitution and insertion errors

the improvement in WER is only 2.2% absolute or 4.8% relative.

The most number of errors in recognizing test data comes from substitution of correct word with any other word in the vocabulary. This amounts to approximately 75% of all errors made by recognizer (see Table 5). The number of insertions and deletions can be controlled by tuning a word insertion penalty parameter available in the recognizer. The word insertion penalty is a fixed value added to each token when it transits from the end of one word to the beginning of another [Young et al., 2006]. By penalizing inter–word tokens we can force introduction of new words only when their probability becomes sufficiently high. However to reduce the number of substitution errors we need to construct better acoustic and language models. This means more training audio data and more appropriate and careful language modeling.

We have performed a number of experiments with different values of word insertion penalty parameter. Table 6 summarizes the results of this evaluation. The first row in the table (with 5 subrows) shows the performance of recognizer when the word insertion penalty varies between 0.0 and -100.0. The optimal value of parameter lies somewhere between 0.0 and -50.0 which gives the improvement in WER around 5%. The remaining rows show the performance of recognizer for the rest of models when the word insertion penalty is 0.0 and -50.0. The results show that the WER can be lowered by 3% when the word insertion is penalized ($penalty \neq 0$).

The best recognition accuracy is obtained when the recognizer uses trigram language model with cutoff value 20 for both bigrams and trigrams, and the word insertion is penalized by -50.0 log probability. This gives word error ratio of 40.4%.

5 Conclusions

In this paper we described briefly the initial set of experiments on Estonian speech recognition using multi–language Babel speech database. Word–based modeling of inflectional and agglutinative language reveals very high perplexity and WER figures. So the future set of experiments will be focused on a subword language modeling.

The best accuracy reported in this paper (WER = 40.4%) can be compared to a recently reported value of 36.2% [Alumäe, 2006] if we account for the reduced amount of training data used in our set of experiments. Some minor improvement can be obtained by constructing language models more carefully –

Model	Penalty	Deletions	Substitutions	Insertions	WER
tg100-100	0.0	312	2428	578	45.5%
tg100-100	-25.0	503	2208	259	40.8%
tg100-100	-50.0	660	2193	129	40.9%
tg100-100	-75.0	831	2430	98	46.1%
tg100-100	-100.0	1031	2815	82	53.9%
tg30-30	0.0	417	2366	439	44.2%
tg30-30	-50.0	562	2210	253	41.5%
tg20-20	0.0	320	2390	562	44.9%
tg20-20	-50.0	660	2164	122	40.4%
tg10-10	0.0	418	2350	431	43.9%
tg10-10	-50.0	563	2192	248	41.2%
tg6-6	0.0	418	2348	429	43.9%
tg6-6	-50.0	563	2183	245	41.1%
tg3-3	0.0	419	2330	426	43.6%
tg3-3	-50.0	567	2172	241	40.9%
tg2-2	0.0	418	2321	422	43.4%
tg2-2	-50.0	567	2169	241	40.9%
tg1-1	0.0	422	2309	421	43.3%
tg1-1	-50.0	568	2157	240	40.7%

Table 6: The effect of penalizing word insertion probability on the number of errors and WER

training corpus used in this paper contains a lot of non–linguistic entries which should be removed but frequently appear at the output of recognizer. This has a clear influence on WER figures.

The recognition of test data is performed using a single–pass best hypothesis strategy which generally loses considerably to multi–pass N–best list strategies with a lattice stage rescored using more comprehensive language models. However, this is an example of things needed to be done in the future using baseline systems built and described in this paper.

References

T. Alumäe. *Methods for Estonian Large Vocabulary Speech Recognition.* PhD thesis, Tallinn University of Technology, 2006.

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 *Proceedings of Eurospeech*, Scandinavia, 2001.

G. Choueiter, D. Povey, S. F. Chen, and G. Zweig. Morpheme-based Language Modeling for Arabic LVCSR. In *Proceedings of the ICASSP*, pages 1053–1056, Toulouse, 2006.

Computer Linguistics Group. Mixed Corpus of Estonian. Available online from http://www.cl.ut.ee. University of Tartu.

A. Eek and E. Meister. Estonian speech in the babel multi–language database: Phonetic–phonological problems revealed in the text corpus. In *LP*, volume II, pages 529–546, 1998.

T. Hirsimäki, M. Creutz, V. Siivola, and M. Kurimo. Morphologically motivated language models in speech recognition. In *International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning*, pages 121–126, Espoo, Finland, 2005.

K. Kirchhoff, J. Bilmes, S. Das, N. Duta, M. Egan, G. Ji, F. He, J. Henderson, D. Liu, M. Noamany, P. Schone, R. Schwartz, and D. Vergyri. Novel Approaches to Arabic Speech Recognition: Report from the 2002 JHU Summer Workshop. In *Proceedings of the ICASSP*, Hong Kong, 2003.

O. Kwon, K. Hwang, and J. Park. Korean large vocabulary continuous speech recognition using pseudomorpheme units. In *Eurospeech'99*, pages 483–486, Budapest, 1999.

M. Maučec, T. Rotovnik, and M. Zemljak. Modelling highly inflected slovenian language. *International Journal of Speech Technology*, 6:245–257, 2003.

E. Meister, J. Lasn, and L. Meister. Development of the Estonian SpeechDat–like database. In *Proceedings of Eurospeech*, pages 1601–1604, Geneva, Switzerland, 2003.

R. Sinha, M. J. F. Gales, D. Y. Kim, X. A. Liu, K. C. Sim, and P. C. Woodland. The CU-HTK Mandarin Broadcast News Transcription System. In *Proceedings of the ICASSP*, Toulouse, 2006.

E. Whittaker and P. Woodland. Comparison of language modelling techniques for russian and english. In *Proceedings of ICSLP*, Sydney, Australia, 1998.

S. Young, J. Odell, and P. Woodland. Tree–based state tying for high accuracy acoustic modelling. In *Proceedings of DARPA Workshop on Human Language Technology*, pages 307–312, 1994.

S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *The HTK Book (for HTK Version 3.4)*. Cambridge University Press, 2006.