Creating Lithuanian and Latvian Speech Corpora from Inaccurately Annotated Web Data

Askars Salimbajevs
Tilde, Vienibas gatve 75A, Riga, Latvia
University of Latvia, Raina bulv 19, Riga, Latvia
askars.salimbajevs@tilde.lv

Abstract

This paper describes the method that was used to produce additional acoustic model training data for the less-resourced languages of Lithuanian and Latvian. The method uses existing baseline speech recognition systems for Latvian and Lithuanian to align audio data from the Web with imprecise non-normalised transcripts.

From 690 hours of Web data (300h for Latvian, 390h for Lithuanian), we have created additional 378 hours of training data (186h for Latvian and 192 for Lithuanian). Combining this additional data with baseline training data allowed to significantly improve word error rate for Lithuanian from 40% to 23%. Word error rate for the Latvian system was improved from 19% to 17%.

Keywords: speech recognition, web data, alignment, Lithuanian, Latvian, speech corpora, low-resourced languages

1. Introduction

Training of an acoustic model for an automatic speech recognition (ASR) system requires large amounts of transcribed audio, especially for the state-of-the art deep neural network models. For Low Resource Languages (LRLs), where there may only be a few hours of transcribed audio, this is a very serious problem. For Latvian and Lithuanian languages the situation is better, for example, there are a 100h Lithuanian speech corpus created for the LIEPA project[1], an 84h corpus from BMMG(Alumäe and Tilk, 2016) and a 100h Latvian speech corpus(Pinnis et al., 2014). However, even this is not much, as neural network acoustic models can make use of many thousands of hours of data and achieve significant improvement in recognition accuracy.

In recent years, improvements in data storage and networking technology have made it feasible to provide Internet users with access to large amounts of multimedia content. This content can be automatically collected and processed for the purpose of training statistical models. However, in many cases, this content is not structured or organised in an accurate and machine-readable form.

For example, both the Lithuanian Parliament (Seimas) and Latvian Parliament (Saeima) websites contain a large archive of video recordings of parliamentary sessions and edited transcripts. One may want to collect this data and use it for the development of a general purpose speech recognition system. This data can also be used for adapting existing ASR for transcription of Seimas sessions. Edited transcripts can be used as the training data for the language model, and video recordings can be used for adapting the acoustic model.

Unfortunately, edited transcripts do not have any timing information. Also, these transcripts are non-normalised. This means that numbers, dates, percent signs, etc. are written with digits and symbols, not as words. Converting these tokens back to words is complicated and error-prone for inflected languages like Latvian and Lithuanian.

While speaking humans can make grammar mistakes, repeat words, make corrections and restart whole sentences. Edited transcript is a written document, thus it contains only final, grammatically correct and re-formulated sentences, which makes it not 100% accurate.

In this paper, we describe the method we used to obtain additional training data by aligning audio and edited transcripts from Lithuanian and Latvian parliament websites. This helped us to significantly improve the Lithuanian general purpose ASR and also resulted in noticeable improvement for the Latvian ASR.

The alignment between long audios and their corresponding transcripts has been previously studied in the context of various applications. (Panayotov et al., 2015) use existing ASR and audio-alignment techniques for creation of a large training corpus from public domain audio-books, and (Anguera et al., 2014) use ASR for different languages and a clever phoneme-based alignment approach for training speech recognition with very limited language resources. (Prahallad and Black, 2011) describe the creation of aligned corpora for building text-to-speech systems, and (Hazen, 2006) focuses on the automatic alignment and correction of inaccurate text transcripts through an iterative process.

Also, Lithuanian is one of the development languages within the IARPA BABEL research program (Harper, 2013), and therefore, Lithuanian is one of the test languages in many papers that have studied low-resource training methods for speech recognition, e.g. (Mendels et al., 2015; Davel et al., 2015; Gales et al., 2015) and many others. While BABEL focuses more on conversational telephone speech (Lileikyte et al., 2018), broadcast speech recognition for Lithuanian was addressed within the Quaero research program. In (Lileikyte et al., 2016), a semi-supervised method was used to train an acoustic model with only three hours of transcribed data and 360

[1] LIEPA (Services Controlled by Lithuanian Voice), Vilnius University, LEU, Siauliai University, Institute of Lithuanian language. https://www.xn–ratija-ckb.lt/liepa

hours of untranscribed data. The untranscribed data was iteratively automatically transcribed and added to the training data. The system achieved a remarkable word error rate (WER) of 18.3% on Quaero broadcast speech evaluation data. Unfortunately, we can not compare this result with our method, as Quaero systems and evaluation data are not publicly available.

The method that we are using in this paper is similar to (Panayotov et al., 2015) and (Hazen, 2006). The main differences are the use of the SpkDiarization toolkit(Rouvier et al., 2013) for segmenting large audio recordings into smaller manageable segments and for providing speaker diarisation, an optional intermediate step where we use the retrained model for better alignment, a different aligned segment extraction process and the fact that it is being done for the less-researched and less-resourced Latvian and Lithuanian languages.

2. Alignment of Parliament session transcripts

2.1. Data collection and processing

A web-crawler script is used for collecting video recordings of parliamentary sessions and the corresponding human-edited reference transcripts from both Lithuanian Seimas and Latvian Saeima websites.

Then, audio is extracted from each video file and processed by the LIUM SpkDiarization toolkit(Rouvier et al., 2013), which segments audio into smaller parts and groups them into clusters (that should correspond to different speakers). Later we will add these segments directly to the training data, so each segment should be reasonably short and contain speech only from one speaker. Also, clustering by speaker is important for correct Speaker Adaptive Training(Anastasakos et al., 1997; Miao et al., 2014).

Reference transcripts are normalised, all punctuation and non-alphabetic characters are removed, all words are lowercased. We also rewrite numbers and dates into words, a module from a text-to-speech engine is used for this purpose for Latvian, but for Lithuanian we just use the nominal forms as we do not have the necessary tools.

Audio segments are then processed by ASR. Clustering information is used during recognition for speaker adaptation.

After all these processing steps, from each video file the following files are obtained:

- A corresponding inaccurate reference transcript in normalised form.
- A set of short audio files that roughly correspond to separate utterances. This set is sorted in chronological order and clustered into different speakers.
- A raw ASR transcript for each short audio file.
- Word alignment information (time, when each word is pronounced and length of pronunciation) from ASR for each audio file.

2.2. First Alignment Step

Next, we take each reference transcript file and perform a global alignment between the inaccurate reference text and per-utterance ASR transcripts. Similarly to (Panayotov et al., 2015), we use the Smith-Waterman alignment algorithm (Smith and Waterman, 1981). An example of such alignment is shown in Figure 1. The grey boxes represent boundaries between different utterances obtained by the LIUM SpkDiarization toolkit.

After the alignment, in each utterance, we select continuous sequences of matched words that are longer than some threshold (e.g. 3 in Figure 1). Using word alignment from ASR, these word sequences are extracted from utterance audio files and are added to the new training data set together with their transcripts from ASR. Speaker diarisation is also preserved, so that utterances from the same speaker are grouped together.

A length threshold is needed to filter out possible alignment errors, for example, short word sequences like ”un

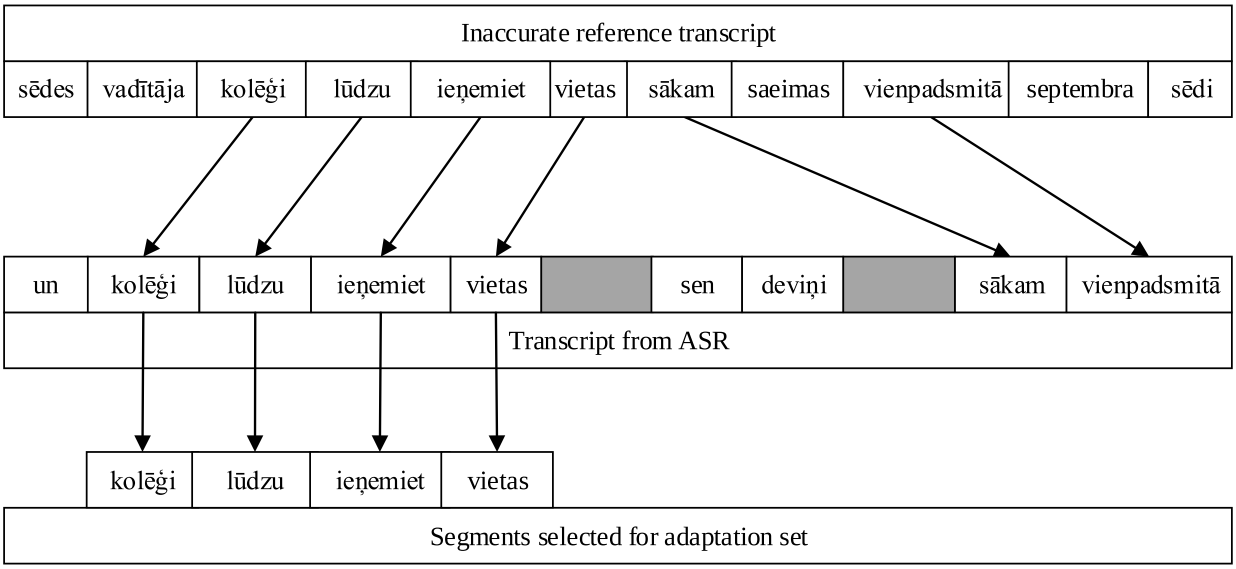

Figure 1: Alignment between the reference transcript and ASR output.

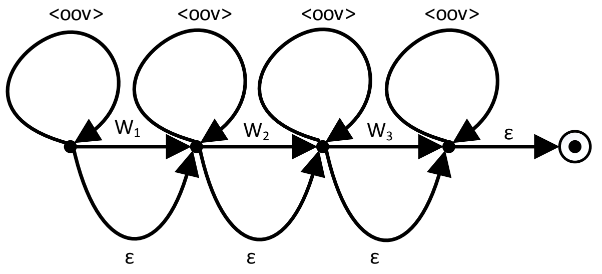

Figure 2: Example FSA for the pseudo-forced alignment of the word sequence w_1, w_2, w_3 with insertions, deletions and substitutions allowed.

tas ir" ("and that is"), "un ir" ("and is"), etc. are rather frequent and can be either misrecognised by ASR or misaligned. Also, we assume that ASR word alignment for longer sequences is more accurate, so extracting longer sequences is less likely to cut off word beginnings and endings. In the first alignment step we use a threshold of 5 or more consecutive words in this work.

2.3. Second Alignment Step

Word sequences extracted in the first alignment step can already be used for training acoustic models. However, the improvement from adding these sequences to the training data will be limited because existing ASR already recognised them correctly; they already match the acoustic model quite well. The parts that were not recognised accurately (and not aligned) can be much more useful, as they are examples of when the existing acoustic model is not good enough.

Successfully extracted segments could be used as "anchor points" so that the audio between anchor points will be mapped to the text between anchor points. This mapping then can be used to help ASR to recognise this part correctly and produce a better alignment.

However, before that, there is an optional intermediate second alignment step that is needed to improve the alignment and get more anchor points. We append the data extracted in the first step to the training data, retrain the acoustic model (we will typically retrain only the DNN part of the acoustic model). The retrained model used for repeated decoding and alignment.

The extraction threshold is changed from 5 consecutive words to 22 consecutive phones. The number 22 represents the average length of 3 average words in both Latvian and Lithuanian and is calculated on vocabularies of both systems. This less strict threshold creates more anchor points.

2.4. Pseudo-Force Alignment

After obtaining mappings between misrecognised audio segments and reference text, the natural choice would be to perform a classic force alignment. However, in our case, it's not suitable, as the reference text is not 100% accurate, so instead we perform a pseudo-force alignment step like in (Hazen, 2006).

We use successfully extracted segments as "anchor points" so that the audio between anchor points will be mapped to the text between anchor points. In our case of edited transcripts, reference text can contain long regions of insertions and audio can contain long non-speech regions, so we have limits for text and audio length. Mappings that are too long are filtered out.

We assume that errors in the transcript are possible, and we allow insertions of new words, deletions and substitutions for existing words. This process is realised through the composition of a pseudo-forced alignment finite state acceptor (FSA) with a lexical transducer from the baseline ASR. An example alignment FSA that allows insertions, substitutions and deletions is shown in Figure 2.

Insertions are modelled through the use of an out-of-vocabulary (OOV) word filler model. This is are single phone model with 5 HMM states. During baseline acoustic model training this model was used to consume OOV and foreign words, fragmented words and spoken noise.

After decoding the pseudo-force aligned audio segments, we select segments with a length of 22 or more phonemes and append them to the training data.

3. Results

3.1. Baseline Speech Recognition Systems

Baseline Latvian and Lithuanian ASR systems are implemented in the open-source Kaldi toolkit (Povey et al., 2011). The recipes for both systems are very similar:

- Grapheme-based pronunciation model.
- HMM-DNN p-norm (Zhang et al., 2014) acoustic model with iVectors for speaker adaptation (Miao et al., 2014).
- Vocabulary of about 800,000 word forms.
- Language models are trained on texts collected from web news portals. After filtering, each training corpus contains about 40M sentences.
- 2-gram language model for decoding and 3-gram for rescoring.

The Lithuanian ASR is trained on speech recordings collected by the LIEPA project and 40h of Seimas recordings from year 2016. About a half of the LIEPA corpus is silence (because it contains isolated voice commands), so the resulting size of training data set is 92h. While the Latvian ASR is trained on a specifically designed 100h Latvian Speech Recognition Corpus (LSRC)(Pinnis et al., 2014).

3.2. Improving Latvian Speech Recognition

For this research, we downloaded and processed about 300 hours of video recordings of Saeima sessions in the period of 2011-2014. After the first alignment step we obtained 120 hours of data.

Next, we evaluated the impact of adding these 120h to the acoustic model training data. For this evaluation, we used a 1 hour test set that is recorded using smartphones and contains recordings of 10 non-professional speakers reading news on different topics from different news web pages and some excerpts from fiction literature. The corpus was manually annotated.

	Size, h	WER, %
Google Cloud Speech	n/d	33-44
Baseline (LSRC)	100	19.6
+ Saeima (1st step)	220	17.0
+ Saeima (2nd step)	249	17.0
+ Pseudo-force aligned	286	**16.9**

Table 1: Evaluation of Latvian speech recognition on the general domain test set.

An improvement in recognition quality was observed (see Table 1) after adding these 120 hours of Saeima recordings to the acoustic model training data. Word error rate was reduced from 19.6% to 17.0% (13% relative).

We have also processed the same test set with the Google Cloud Speech service for comparison. Unfortunately, Cloud Speech service filters out parts with low confidence, so direct calculation of WER, shows very poor result (44%) with a lot of deletions. If deleted segments are not counted WER is 33%. We believe that real WER for Google's system should be somewhere in the middle of this interval.

In the second alignment step, the ASR improved by the new training data from the first step was used to repeat the decoding and alignment procedure. This allowed to improve the alignment, first to 136 hours and then, by changing the threshold to 22 characters, to 149 hours. However, no improvement in WER was detected in either case.

Because a large part of the data has been successfully decoded in the first two steps, only 57 hours of data were selected for pseudo-force alignment. 37 hours were successfully aligned and added to the training set, the WER of ASR trained on all of the data combined is 16.9%.

3.3. Improving Lithuanian Speech Recognition

For the evaluation of Lithuanian speech recognition, we used a 1-hour "general domain" test set that consists of manually transcribed randomly picked audio segments from various radio and TV shows (mainly "Atviras pokalbis" and "Labas rytas, Lietuva"), and Seimas sessions from 2017. The corpus does not have speaker information, so speaker adaptation is done only a the utterance level.

We have also processed the same test set with the Google Cloud Speech service, however, as in previous case, the service omits large segments for which it has low confidence, so the WER is high (40%) because of deletions (if these segments are not counted, WER is about 27%). We have also evaluated ASR from (Alumäe and Tilk, 2016) on this set (compounding was not performed).

For first experiment, we have downloaded about 270 hours of video recordings of Seimas sessions from years 2015 and processed with our baseline system. After alignment and extraction, we obtained an additional 52 hours of data. The retrained ASR system achieved a WER of 25.5%. Repeating retraining and changing the threshold to 22 characters allowed obtaining 13 more hours (65 hours of aligned data in total), but no improvement in WER was detected.

Pseudo-force alignment produced 71 hours of additional aligned data (from 138 hours). Appending this data to the training data resulted in an improvement of WER from 25.5% to 24.4% (4% relative).

As alignment resulted in only about 130h of data (comparing to 180h for Latvian), we decided to download more Seimas audio (120h of recordings in the period from November 2016 till April 2017) and process it in the exactly same way. This allowed to obtain 56h of additional aligned data. Appending this data to the training data resulted in an improvement of WER from 24.4% to 23.3% (5% relative). The results are summarised in Table 2.

4. Discussion

In the first and second alignment steps, we extract segments of data that have a 100% match with the reference transcript. This means that our existing ASR already recognises such segments correctly. However, after adding these segments to the training, we see an improvement in WER.

We believe that this is caused by (1) better senone coverage in the larger training set and (2) a large "language model bias" that allowed some segments to be recognised correctly even when the acoustic score was too low.

When doing pseudo-force alignment, this "language model bias" is even larger, so adding pseudo-force aligned data should improve word error rate even more. However, experiments showed a small improvement for both languages. This means that not all of the pseudo-force aligned data are complementary to the data extracted in the first two steps.

The data aligned in the first step can be used to adapt the acoustic model and repeat the decoding. Our experiments show that this improves the second alignment (i.e. more data are successfully aligned), however adding new data from the second alignment to the training, does not result in WER improvements for both languages.

Both ASR systems significantly outperform Google ASR for given languages on our test set, but this is probably because Google systems are strongly adapted to specific domain and specific usage scenario. Also, it's problematic to correctly evaluate Google systems as a lot of transcript post-processing is performed (filtering of low-confidence segments, inverse text normalization etc).

Also, by adding automatically aligned Seimas data to the training we managed to outperform the Lithuanian speech recognition by (Alumäe and Tilk, 2016). It is unknown how our system will perform on the broadcast speech recognition, however it seems that result will be comparable to Quaero(Lileikyte et al., 2016) and (Alumäe and Tilk, 2016).

	Size, h	WER, %
Google Cloud Speech	n/d	27-40
(Alumäe and Tilk, 2016)	84	25.2
Baseline	92	27.4
+ Seimas 2015 (1nd step)	144	25.5
+ Seimas 2015 (2nd step)	157	25.5
+ Pseudo-force aligned	228	24.4
+ Seima 2017 + PFA	284	**23.3**

Table 2: Evaluation of Lithuanian speech recognition on the general domain test set.

5. Conclusions

In this work, we have automatically aligned audio recordings with imprecise transcriptions and created additional training data for the less-resourced languages of Latvian and Lithuanian (about 190h for each). The data was automatically collected from the websites of the Latvian and Lithuanian parliaments.

Training ASR with new training data allowed improving WER for both Latvian and Lithuanian languages. This suggests that the collected training data are complementary to the baseline training corpora. The improvement is similar for both languages and is about 13% relative in this work.

We believe that experiment results show that the method is language independent and can be used to collect more training data for other less-resourced languages.

In future work, we plan to use this method to process more recordings for both languages and obtain a larger speech corpora.

6. Acknowledgements

The research has been supported by the European Regional Development Fund within the project ”Neural Network Modelling for Inflected Natural Languages” No. 1.1.1.1/16/A/215.

7. Bibliographical References

Alumäe, T. and Tilk, O. (2016). Automatic speech recognition system for lithuanian broadcast audio. In *Human Language Technologies - the Baltic Perspective : Proceedings of the Seventh International Conference, Baltic HLT 2016*, volume 289, pages 39–45. IOS Press.

Anastasakos, T., McDonough, J., and Makhoul, J. (1997). Speaker adaptive training: a maximum likelihood approach to speaker normalization. *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2:0–3.

Anguera, X., Luque, J., and Gracia, C. (2014). Audio-to-text alignment for speech recognition with very limited resources. In *Proceedings of the 15th Annual Conference of the International Speech Communication Association (INTERSPEECH'2014)*, pages 1405–1409.

Davel, M., Barnard, E., Van Heerden, C., Hartmann, W., Karakos, D., Schwartz, R., and Tsakalidis, S. (2015). Exploring minimal pronunciation modeling for low resource languages. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2015-January, pages 538–542.

Gales, M. J., Knill, K. M., and Ragni, A. (2015). Unicode-based graphemic systems for limited resource languages. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2015-August, pages 5186–5190.

Harper, M. (2013). The BABEL program and low resource speech technology. In *ASRU*.

Hazen, T. J. (2006). Automatic alignment and error correction of human generated transcripts for long speech recordings. In *Proceedings of the 9th International Conference on Spoken Language Processing (INTERSPEECH'2006)*, volume 2006, pages 1606–1609.

Lileikyte, R., Gorin, A., Lamel, L., Gauvain, J. L., and Fraga-Silva, T. (2016). Lithuanian Broadcast Speech Transcription Using Semi-supervised Acoustic Model Training. In *Procedia Computer Science*, volume 81, pages 107–113.

Lileikyte, R., Lamel, L., Gauvain, J. L., and Gorin, A. (2018). Conversational telephone speech recognition for Lithuanian. *Computer Speech and Language*, 49:71–82.

Mendels, G., Cooper, E., Soto, V., Hirschberg, J., Gales, M., Knill, K., Ragni, A., and Wang, H. (2015). Improving speech recognition and keyword search for low resource languages using web data. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2015-January, pages 829–833.

Miao, Y., Zhang, H., and Metze, F. (2014). Towards speaker adaptive training of deep neural network acoustic models.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE.

Pinnis, M., Auzina, I., and Goba, K. (2014). Designing the Latvian Speech Recognition Corpus. In *Proceedings of the 9th edition of the Language Resources and Evaluation Conference (LREC'14)*, pages 1547–1553.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., and Vesely, K. (2011). The Kaldi Speech Recognition Toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society.

Prahallad, K. and Black, A. W. (2011). Segmentation of monologues in audio books for building synthetic voices. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(5):1444–1449.

Rouvier, M., Dupuy, G., Gay, P., Khoury, E., Merlin, T., and Meignier, S. (2013). An open-source state-of-the-art toolbox for broadcast news diarization. In *Proceedings of the 14th Annual Conference of the International Speech Communication Association (INTERSPEECH'2013)*.

Smith, T. F. and Waterman, M. S. (1981). Identification of common molecular subsequences. *Journal of molecular biology*, 147(1):195–197.

Zhang, X., Trmal, J., Povey, D., and Khudanpur, S. (2014). Improving deep neural network acoustic models using generalized maxout networks. *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 215–219.