# Online Speaker Adaptation with Pre-computed FMLLR Transformations

*Volker Fischer, Siegfried Kunzmann*

European Media Laboratory GmbH,
Schloß-Wolfsbrunnenweg 33, D-69118 Heidelberg, Germany
volker.fischer@eml.villa-bosch.de, kunzmann@eml.villa-bosch.de

## Abstract

This paper presents a memory efficient single pass speech recognizer that makes use of pre-computed FMLLR transformations for online speaker adaptation. For that purpose we apply unsupervised segment clustering to the training corpus, create a transformation matrix for each cluster, and train a text-independent Gaussian mixture classifier for cluster selection during runtime. We use the RWTH Aachen University open source speech recognition toolkit for evaluation and compare the results to a standard speaker adaptive two pass decoding strategy. Results indicate that the method improves single pass recognition in VTLN feature space almost without overhead due to cluster selection, and show a relative improvement of up to 15 percent over speaker adaptative decoding, if only little data is available for unsupervised online adaptation.

**Index Terms**:speaker adaptation, cluster adaptive training, one pass speech recognition.

## 1. Introduction

State of the art large vocabulary continuous speech recognizers, like e.g. [1, 2], usually make use of a two pass decoding strategy: Speaker independent acoustic models are used in a first decoding pass in order to provide a preliminary transcript, which is then used for online speaker adaptation, and which is followed by a second decoding pass that utilize speaker adaptive models for the creation of a final, more accurate transcript. Popular normalization and adaptation techniques employed between the two decoder passes are vocal tract length normalization (VTLN), cf. [3], and (feature-space) maximum likelihood linear regression (fMLLR, MLLR) [4, 5]. Prior to first pass decoding the sketched setup may make use of a segmentation and clustering component. While segmentation of the audio into speech and silence segments is performed in order to avoid the unnecessary decoding of non-speech segments, clustering helps to identify speaker changes (for example, in broadcast news shows), and/or is used to find signal portions that can undergo the same transformation during adaptation.

Speaker normalization, online adaptation and second pass decoding can provide gains of 30 percent and more in word error rate [2], but do so at the cost of increased computing time. For example, the real time (1xRT) system described in [6] spends only 25 percent of its time in the acoustic frontend and first pass decoding, whereas the remaining 75 percent are shared between adaptation (roughly 20 percent, including alignment) and second pass decoding (55 percent).

The work presented here aims to avoid these drawbacks by borrowing ideas from clustering of training speakers, which has been successfully applied in the past in dictation systems [7, 8]. For that purpose, Section 2 briefly reviews speaker clustering and its use during recognition. Section 3 presents a modified approach that considers the needs of todays applications, like e.g. memory efficiency for high scalability, or the need for *text independent* cluster selection. Section 4 gives results obtained with a state-of-the-art open-source speech recognizer [1, 9], before Section 5 concludes with an outlook on future work.

## 2. Pre-clustering of training data revisited

### 2.1. Method

Introduced for the use in desktop dictation systems, preclustering of training speakers was mainly designed to provide a better out-of-box word error rate or to reduce the need for (offline) speaker adaptation [7]. More recently, it has been found beneficial for applications like, for example, Voice Search, which must face great acoustic variablity and quite weak language model constraints at the same time [10]. Cluster adaptive training comprises the following steps:

1. Partitioning: The training corpus is divided into a given number of acoustically similar clusters of speakers. For that purpose, either a priori knowledge (like gender and/or dialect information [8]) or an *alignment based* characterization of the speakers' acoustic space can be used. For example, in [7] a speaker is represented by the concatenation of single-gaussian emission densities of a phone based Hidden-Markov-Model system. Partitions are created in a bottom-up

clustering procedure that evaluates the Euclidean distance between these supervectors.

2. Model training: Following the standard recipes for the training of Hidden-Markov-Model based speech recognizers, an acoustic model is trained for each cluster. Dependent on the amount of data, a cluster acoustic model may be created from the partition data only, or it may result from adaptation of a general acoustic model trained with all data.

3. Cluster selection and recognition: Based on a small amount of audio collected from a test speaker, the best suited cluster is identified using the same distance measure as in training. In a desktop dictation system, cluster identification is typically done "under the cover", for example, by prompting the user for a phonetically well designed speech sample during microphone setup. The selected acoustic model is then used for the recognition of the test speaker's utterances; it may also be subject to further (offline) adaptation once a sufficient amount of user data has been collected.

### 2.2. Discussion

Being developed as a method for offline speaker adaptation that takes place during the setup of a dictation system, preclustering of training speakers shows the following disadvantages with respect to today's application scenarios:

- Manifold of acoustic models: Storing and/or (dynamic) loading of several acoustic models becomes a cost factor for applications that run tens or hundreds of speech recognizers in parallel. In particular, it may be prohibitive, if the models provide a high acoustic resolution by using a quite large number of Gaussians. Instead, for a resource efficient runtime behaviour it is preferable to store a memory-mapped version of a general acoustic model and provide a feature space transformation per cluster that is applied based on the result of the cluster selection process.

- Runtime cluster selection: Prompting the user for a (phonetically balanced) portion of speech is not an option in almost any scenario. Instead, a text independent cluster selection is preferred in order to avoid any kind of first pass decoding and alignment and to keep the recognizer architecture simple.

On the other hand, there are some potential benefits of the method:

- Robustness of adaptation: Being computed on a fairly large amount of training data, parameter estimation may be more robust (but also less specific) than in an online scenario.

- Supervised vs. unsupervised adaptation: Clustered training is based on the correct script rather than on the erroneous transcript of first pass decoding.

## 3. Online adaptation with pre-stored FMLLR transformations

Following above considerations we train a speaker adaptive clustered acoustic model and a text-independent cluster selection mechanism as follows:

Partitioning of the training corpus is unsupervised and based on segments, i.e. different utterances from the same speaker are allowed do belong to different clusters. Segments are represented by the mean and full covariance of MFCC feature vectors and clustering is based on generalized likelihood ratio. The number of partitions is controlled via a Bayesian Information Criterion (BIC, cf. [11]).

Given the reference transcripts for all training segments belonging to a cluster, a VTLN-adapted acoustic model is used for the estimation of a single FMLLR transform for each cluster. Subsequently, the FMLLR-VTLN canonical features are used in a standard EM procedure for training a "cluster adaptive" acoustic model. Finally, the transformation matrices are stored for further use during single pass decoding.

In order to enable single pass decoding the cluster selection mechanism must be independent of alignment. For that purpose, we train a Gaussian mixture classifier using a different feature vector (12 MFCCs, 12 delta coefficient, and one acceleration coefficient). In this model, each cluster is represented by a single Gaussian with diagonal covariance matrix. A similar classifier using the very same feature vector is also trained for fast, text independent warping factor estimation, cf. [3].

During runtime each utterance or segment is first evaluated by both Gaussian mixture classifiers, resulting in a warping factor and a cluster identifier. Associated with the identifier is a FMLLR transformation matrix that was estimated and stored during training; this transformation is now applied to the warped MFCC feature vectors and the transformed features are finally used in a single recognition pass.

## 4. Experiments

All experiments reported in this section were performed with the RWTH Aachen University open source speech recognition toolkit [9] that offers all components needed to setup the clustered recognition system presented here. Notably these components are:

- speaker adaptive training in VTLN feature space,
- a Gaussian mixture classifier for fast and text-independent cluster selection during runtime,
- unsupervised segment clustering (used here for the creation of speaker clusters), and
- an efficient tree search decoder.

We made use of the freely available CMU Census Database for which the RWTH toolkit provides recipes for the development of a 100 word speech recognizer. The database consists of roughly 30 minutes of training material (948 utterances from 74 speakers, 21 of them females), and 6 minutes of test data (130 utterances from 10 speakers, 3 of them females). We downsampled the database to telephony speech bandwidth and used the training transcripts for the creation of a trigram language model with modified Kneser-Ney smoothing [12]. Reconfiguration of the acoustic frontend, clustering of training speakers, and the introduction of cluster selection during recognition were easily done by adapting the configuration files provided with the toolkit for the components mentioned above.

We trained single state triphone HMMs with Gaussian mixture emission probabilities and a globally pooled, diagonal covariance matrix. Models of different acoustic resolution with 3.000 – 14.000 Gaussians were created by performing up to eight splits of the mixtures set. The baseline acoustic models for single and two pass pass decoding are trained in VTLN and VTLN-FMLLR feature space, respectively. Speaker adaptive training makes use of the speaker information provided in the training database. For the cluster adaptive training proposed in Section 3 the 948 utterances of the training corpus were automatically grouped into five clusters, resulting in five FMLLR adaptation matrices being stored for cluster adaptive recognition.

The so created acoustic models were used in four different recognition experiments:

1. **VTL**: single pass decoding in VTLN feature space,
2. **VTL-SAT-L**: two pass decoding with speaker labels obtained from unsupervised test corpus clustering,
3. **VTL-SAT**: two pass decoding without speaker labels, and
4. **VTL-CLU**: single pass cluster adaptive decoding, i.e. selection and use of pre-computed FMLLR training transformations, also without speaker labels.

Whereas VTL and VTL-SAT-L are the standard setups for single and two pass decoding, VTL-CLU evaluates the recognizer proposed in Section 3. Both VTL-CLU and VTL-SAT make no use of speaker information, but determine a FMLLR transformation for each individual utterance of the test corpus, either by cluster selection (VTL-CLU) or by online adaptation (VTL-SAT). With an average test utterance length of only 2.75 seconds, both experiments aim at the simulation of scenarios like, for example, Voicemail transcription or Voice Search, where short utterances and a small amount of adaptation data are predominant.

Recognition results and normalized real time factors for VTL, VTL-SAT-L, and VTL-CLU are shown in Table 1. It can be seen that cluster dependent decoding outperforms single pass recognition both with respect to word error rate and real time (despite of the additional effort for cluster selection), but in terms of word error rate can not compete with standard two pass decoding.

| | 3k | 5.6k | 8.8k | 12k | 14k |
|---|---|---|---|---|---|
| VTL | 9.83 | 9.96 | 8.67 | 9.06 | 9.70 |
| VTL-SAT-L | 9.06 | 7.63 | 7.63 | 7.63 | 8.15 |
| VTL-CLU | 8.93 | 8.93 | 8.41 | 8.41 | 9.18 |

| | 3k | 5.6k | 8.8k | 12k | 14k |
|---|---|---|---|---|---|
| VTL | 0.65 | 0.54 | 0.47 | 0.42 | 0.38 |
| VTL-SAT-L | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| VTL-CLU | 0.50 | 0.40 | 0.33 | 0.32 | 0.31 |

Table 1: *Word error rates (top) and normalized real time factors (bottom) for models of different acoustic resolution.*

However, the picture changes, if the two pass strategy does not make use of test corpus clustering, but estimates an individual FMLLR transform for each short utterance (VTL-SAT), cf. Table 2. In this case, VTL-CLU outperforms VTL-SAT, suggesting that the "average" FMLLR transformation computed on the *training data* is better suited, if only little adaptation data can be gathered online.

| | 3k | 5.6k | 8.8k | 12k | 14k |
|---|---|---|---|---|---|
| VTL-SAT | 9.70 | 10.22 | 8.93 | 9.93 | 9.44 |
| VTL-CLU | 8.93 | 8.93 | 8.41 | 8.41 | 9.18 |

Table 2: *Word error rates for utterance based online adaptation (VTL-SAT) and adaptation with pre-computed transforms.*

Finally, Table 3 shows *oracle* word error rates for single pass decoding with pre-computed FMLLR transformations. The results demonstrate that VTL-CLU can compete with the standard approach to two pass decoding (VTL-SAT-L), if the transformation that yields lowest word error rate is known in advance. Therefore, the results suggest to put further efforts in the cluster selection scheme.

| | 3k | 5.6k | 8.8k | 12k | 14k |
|---|---|---|---|---|---|
| VTL-SAT-L | 9.06 | 7.63 | 7.63 | 7.63 | 8.15 |
| VTL-CLU-o | 7.24 | 7.50 | 7.50 | 7.12 | 7.24 |

Table 3: *Word error rates for online adaptation (VTL-SAT-L) vs. oracle word error rates for adaptation with pre-computed transforms (VTL-CLU-o).*

## 5. Conclusion and Outlook

In this paper, we presented a cluster adaptative single pass speech recognizer with a Gaussian mixture classifier for text-independent cluster selection. Additional memory requirements for the multiple acoustic models are minimal, since each cluster model is represented only by a transformation matrix of size less than 20 kilobytes. Results obtained with an open source state of the art speech recognition toolkit and a public available data set demonstrate the feasibility of the approach, but also show runtime cluster selection as a major source for further improvement.

More recently, we have started to carry over the approach presented here to discriminatively trained models using MPE [13] on top of the clustered acoustic model. In doing so, unsupervised clustering of several tens of thousands of speakers via generalized likelihood ratio and BIC turned out to be a major computational effort that needs to be reduced.

## 6. Acknowledgements

The authors thank David Rybach, Markus Nußbaum-Thom, and David Nolden, all with RWTH University of Aachen, for their kind help with the RWTH speech recognition toolkit.

## 7. References

[1] J. Lööf, C. Gollan, S. Hahn, G. Heigold, B. Hoffmeister, C. Plahl, D. Rybach, R. Schlüter, and H. Ney, "The RWTH 2007 TC-STAR Evaluation System for European English and Spanish", in *Proc. of Interspeech 2007*, Antwerp, Belgium, 2007, pp. 2145–2148.

[2] H. Soltau, G. Saon, and B. Kingsbury, "The IBM Attila Speech Recognition Toolkit", in *IEEE Workshop on Spoken Language Technology*, Berkeley, CA., 2010, pp. 85–90.

[3] L. Welling, S. Kanthak, and H. Ney, "Improved Methods for Vocal Tract Normalization", in *Proc. of the Int. Conf. on Acoustics, Speech, and Signal Processing*, Phoenix, AZ., 1999, pp. 761–764.

[4] M.J.F. Gales, "Maximum Likelihood Linear Transformations for HMM-based Speech Recognition," *Computer Speech and Language*, vol. 12, no. 2, pp. 75–98, 1998.

[5] C. Leggetter and P. Woodland, "Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density Hidden Markov Models," *Computer Speech and Language*, vol. 9, no. 2, pp. 171–185, 1995.

[6] S.F. Chen, B. Kingsbury, L. Mangua, D. Povey, G. Saon, H. Soltau, G. Zweig, J. Lai, and R. Mercer, "Advances in Speech Transcription at IBM under the DARPA EARS Program," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1596–1608, 2006.

[7] Y. Gao, M. Padmanabhan, and M. Picheny, "Speaker Adaptation based on Pre-Clustering Training Speakers", in *Proc. of the 5th European Conference on Speech Communication and Technology*, Rhodes, Greece, 1997, pp. 2091–2094.

[8] V. Fischer, Y. Gao, and E. Janke, "Speaker Independent Upfront Dialect Adaptation in a Large Vocabulary Continuous Speech Recognizer", in *Proc. of the 5th Int. Conf. on Spoken Language Processing*, Sydney, 1998.

[9] D. Rybach, C. Gollan, G. Heigold, B. Hoffmeister, J. Lööf, R. Schlüter, and H. Ney, "The RWTH Aachen University Open Source Speech Recognition System", in *Proc. of Interspeech 2009*, Brighton, UK, 2009, pp. 2111–2114.

[10] F. Beaufays, V. Vanhoucke, and B. Strope, "Unsupervised Discovery and Training of Maximally Dissimilar Cluster Models", in *Proc. of Interspeech 2010*, Makuhari, Chiba, Japan, 2010, pp. 66–69.

[11] S. Chen and P. Gopalakrishnan, "Clustering via the Bayesian Information Criterion with Applications to Speech Recognition", in *Proc. of the IEEE Int. Conference on Acoustics, Speech, and Signal Processing*, Seattle, 1998, pp. 645–648.

[12] S. Chen and J. Goodman, "An Empirical Study of Smoothing Techniques for Language Modelling," Tech. Rep. Technical Report TR-10-98, Harvard University, 1998.

[13] D. Povey and P.Woodland, "Minimum Phone Error and I-smoothing for Improved Discriminative Training", in *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Orlando, Fl., 2002, pp. 105–108.