Prosodically Motivated Features for Confidence Measures

Silke Goronzy, Krzysztof Marasek, Andreas Haag, Ralf Kompe

Sony International (Europe) GmbH, Advanced Technology Center Stuttgart (ATCS)
Home Network Company Europe, Man Machine Interfaces
Hedelfinger Str. 61, D-70327 Stuttgart, Germany
Fon: +49-711-5858-456, Fax: +49-711-5858-199
{goronzy, marasek, haag, kompe}@sony.de

ABSTRACT

In this paper new, phone-duration-based features for confidence measures (CMs) using a classifier are proposed. In misrecognized utterances, the segmentation and thus the phoneme durations often deviate severely from what can be observed in the training data. Also the found segmentation for one recognized phoneme often covers several 'real' phonemes, that have different spectral properties. So such phoneme durations often indicate that a misrecognition took place and we derived some new features based on these durations. In addition to these new features we used some related to the acoustic score of the N-best hypotheses. Using the full set of 46 features we achieve a correct classification rate of 90% at a false rejection rate of 5.1% on an isolated word, command&control task using a rather simple neural network (NN) classifier. Simultaneously, we try to detect out of vocabulary (OOV) words with the same approach and succeed in 91% of the cases. We then combine this CM with unsupervised MAP and MLLR speaker adaptation. The adaptation is guided by the CM and the acoustic models are only modified if the utterance was recognized with high confidence.

1. INTRODUCTION

Recognition rates of state-of-the-art recognition systems are often far below 100%, especially for large vocabularies, continuous speech or in adverse conditions. Often it is beneficial to know if a misrecognition took place, in particular in command&control applications. Here the cost of executing a misrecognized and thus wrong command often far exceeds the cost of re-prompting the user if it is known that the utterance was misrecognized. The detection of misrecognized utterances can be achieved by CMs, that try to calculate a probability of correct recognition. Another problem that occurs, is that of OOV words. The recognizer always outputs the best matching word, but if the uttered word was not included in the vocabulary, this will result in a misrecognition and increases the overall error rate of the recognizer. Thus our approach tries to detect OOV words simultaneously to calculating the confidence.

There are many approaches to deal with the problem of assigning confidence to the recognizer output. Many use the statistical hypothesis testing, cf. [1, 2, 3], which often involves the training of so called anti-models, either on phone- or word basis. These anti-models then represent everything but the word/phone under consideration. Each decoded hypothesis is tested against the corresponding anti-model(s) and if the resulting value(s) fall below a certain threshold, the utterance is considered as unreliable and rejected. Other approaches collect a set of features during the search and then combine these to formulate the final CM. Several studies tested the combination of a set of features and compared this to the performance of each feature alone and found that combining them outperforms either feature if taken alone, cf. [4, 5].

We also employ such a two-step procedure. In the first step the utterance is recognized and several features are extracted during the search and from the recognizer output, which is a N-best list. In the second step the features are standardized and fed into the NN, which computes the probability that the utterance was correctly recognized. Simultaneously with classifying the recognition result we try to judge whether a misrecognized utterance was an OOV word or not. Then we combine this approach with unsupervised MLLR and MAP speaker adaptation, such that adaptation is conducted in a semi-supervised manner, i.e., only those words that were recognized with high confidence are used for the on-line adaptation of the acoustic models. All other words are discarded from the adaptation to avoid the adaptation of the wrong models.

The following section describes the features used in our approach. Section 3 describes the NN we use and summarizes the results we obtain. We then outline our semi-supervised adaptation approach and show the results.

2. CM FEATURES

Phone-Duration-Based Features

It can be observed that when a word is misrecognized, there is a severe mismatch between the segmentation (and thus the phoneme durations) found by the recognizer and the durations that can be found in the training data. This motivated the use of some new features that are related to phoneme durations. During the training the distributions of durations of all phones are determined, based on a forced alignment of the training data. The durations are additionally smoothed and are later compared to the phoneme durations that were determined by the alignment of the recog-

nizer during testing. Since it is well known that the speaking rate strongly influences the phoneme durations, we estimated the speaking rate and normalized the found durations accordingly. The speaking rate was estimated as follows, cf. [6]:

$$\alpha = \frac{1}{N} \sum_{i=1}^{N} \frac{d_i}{\bar{x}_p} , \quad (1)$$

where *N* denotes the number of phones in the observed utterance and/or in the past few utterances, d_i is the duration of the *i-th* phone segment (recognized as phone *p*) in the utterance and $\bar{x}_p$ is the mean length of the corresponding phone *p* learned during training.

Some of the features were multiply used, e.g. not normalized, normalized by number of frames, by acoustic score of the best hypothesis or by the speaking rate.

The feature set comprises the following features:

1. *n_toolong01, n_toolong05*: Number of phones in the best hypothesis that are longer than the 0.01 and 0.05 percentile, respectively, compared to the training data
2. *n_tooshort01, n_tooshort05*: See above for too short durations
3. *sequence*: Number of sequences of phone pairs within one utterance where the first phoneme was too long and the second one too short (or vice versa) (using the 0.01 percentiles).
4. *avg_tempo*: The average speaking rate
5. *stdev_tempo*: The standard deviation of the speaking rate (w.r.t. to average speaking rate of last n utterances)
6. *diff_tempo*: The absolute difference between the average and the actual speaking rate.
7. *tempo*: Current speaking rate

To show the relation between the features chosen and the correct/misrecognized classification we show some box-and-whiskers plots. Box-and-whiskers plots are a way to look at the overall shape of the data. The central box shows the data between the 'hinges' (roughly quartiles), with the median presented by a line. 'Whiskers' go out to the extremes of the data, and very extreme points are shown by themselves, cf. [7]. We show the plots of the features *n_toolong05, n_tooshort05 and avg_tempo* in Figures 1, 2 and 3, respectively. Although several outliers are present we can see a good distinction capability between the correct/misrecognized classes of the corresponding feature.

Additional Features

In addition to the duration-based features described above we used some more features that have proven to be useful for CMs in the past. These consist of the following:

1. *n_frames*: Total number of frames of the utterance
2. *n_phones*: Total number of phones of the utterance

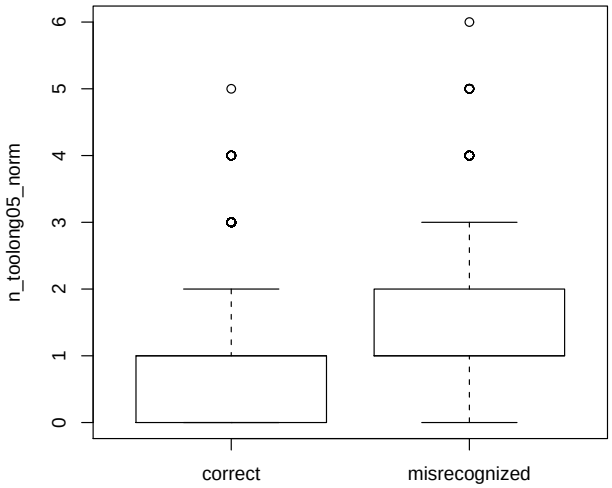

Figure 1: box-and-whiskers plot for feature n_too_long05 normalized by the number of frames

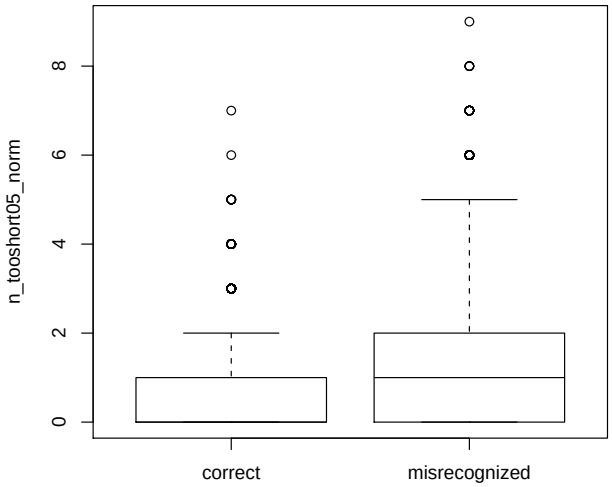

Figure 2: box-and-whiskers plot for feature n_too_short05 normalized by the number of frames

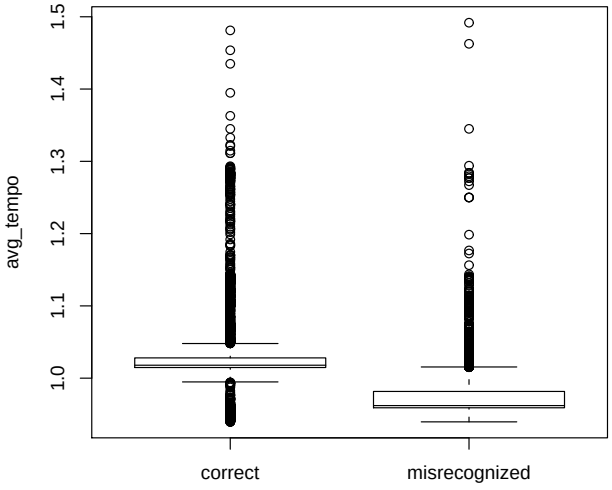

Figure 3: box-and-whiskers plot for feature avg_tempo

3. *n_frames_nosil*: Total number of frames (without silence)

4. *first_score*: Acoustic score of the best hypothesis

5. *first_second*: Difference in acoustic scores between the first and second-best hypothesis

6. *avg*: Average acoustic score for the N-best hypotheses

7. *first_avg*: Difference between *first_score* and the average score

8. *first_last*: Difference between the first and last-best hypothesis

9. *first_beambest*: For each frame and all active states the distances between the Gaussians and the feature vectors are computed. The best possible distance, i.e., the minimal one in the beam, is compared to the distance found for the state belonging to the best state sequence.

10. *first_beambest_zeros*: The number of frames for which the score difference (see *first_beambest*) is zero

11. *first_beambest_largest*: The largest continuous difference between the first-best and the best state sequence in beam

12. *best*: The best possible score in the beam

13. *first_best*: See *first_beambest*, taking into account the transition probabilities.

14. *worst_phonescore*: The worst phone score in the best hypothesis

15. *avg_phonescore*: The average phone score in the best hypothesis

16. *stdev_phonescore*: The change of score within one phone

17. *worst_phone_best*: The difference between the best possible and worst phoneme score in the best hypothesis

18. *worst_frame_score*: Worst frame score for all frames in the best hypothesis

19. *best_in_beam*: The sum of the differences between the best frame scores for the hypotheses in the beam and of the best hypothesis

20. *snr*: signal-to-noise ratio

We again exemplarily show a box-and-whiskers plot of one of the features, the normalized *worst_phonescore* in Figure 4. Again a good separation between the correct/misrecognized classes can be observed.

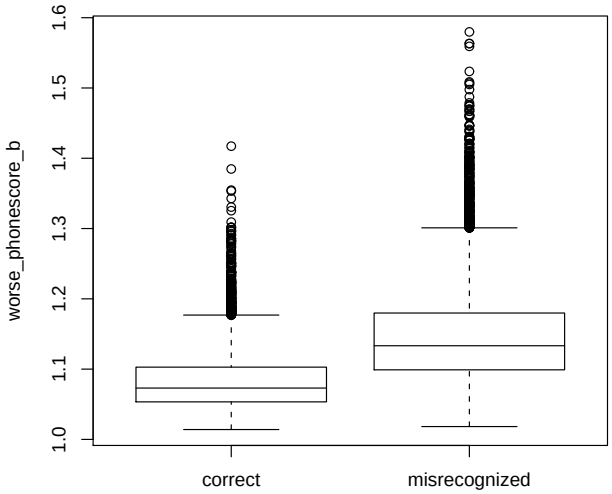

Figure 4: box-and-whiskers plot for feature worst_phonescore

	# patterns	corr	wrong	OOV
train	37718	18859	3022	15837
eval	250	1795	330	2125
test	16186	7071	1022	8093

Table 1: Number of patterns used for training, evaluation and testing of the NN classifier

3. CLASSIFICATION

For testing the 46 derived features we constructed a feature vector for each utterance of the command&control test task and used a NN [1] to classify the recognition result as either correct or wrong. For the NN we used a feed forward net, that consisted of one hidden layer only. We used 46 input nodes, 8 hidden nodes and 2 or 3 output nodes, respectively. The training data for the NN comprised clean speech only and also included a large amount of OOV words. The detailed statistics concerning the patterns used for training, evaluation and testing can be found in Table 1. One pattern corresponds to the features extracted from one utterance. The data for training the NN was obtained using our standard recognizer, that will be described in more detail in section 4. We used different data for training the monophone models for the recognizer than we used for training the NN. Since we use some features related to the acoustic score, this was necessary to avoid a possible influence of the training data on the acoustic score if the same data was used for training the monophone models and determining the reference durations. The NN training data was labelled as being correctly recognized or not. So the target output for the 2 output nodes of the NN were either '1 0' or '0 1', '1 0' means the utterance was correctly recognized and '0 1', it was misrecognized.

In a second set of experiments we use a NN with 3 output nodes. The first two output nodes have the same meaning as before, the third output node is to indicate, whether a misrecognized word was an OOV word or not ('1' means

[1] we used SNNS to build our nets, [8]

	CER	C_a	F_a	C_r	F_r
baseline (SI)	-	43.7	56.3	-	-
2 out	9.67	38.45	4.62	51.55	5.05
3 out	10.29	37.9	4.62	51.62	5.67

Table 2: Classification results of a NN with 46 input nodes, one hidden layer with 8 nodes and 2 and 3 output nodes, respectively

OOV). So possible outputs (in the training data) are '0 1 1' or '0 1 0' in case of a misrecognition. For correctly recognized words only '1 0 0' is possible. During testing the NN outputs values between 0 and 1. The final decision threshold for the values between 0 and 1 is then simply 0.5. This means that if the first output is greater than 0.5 and the second is smaller than 0.5, the utterance is considered as being correctly recognized. If both values are greater or smaller then this threshold, the utterance cannot be classified at all. This happened in 0.3% of the cases only. Correspondingly, if the third output is greater than 0.5, the word is classified as being OOV.

4. EXPERIMENTS AND RESULTS

Experimental Setup

Training and testing were conducted using a German command&control isolated word data base recorded in our sound treated studio. It consists mainly of isolated words and short phrases. The vocabulary size was 375. The speech was sampled at 16 kHz and coded into 25 ms frames with a frame shift of 10 ms. Each speech frame was represented by a 38-component vector consisting of 12 MFCC coefficients (no energy) and their first and second time derivatives. The first and second time derivatives of the energy are also included. We trained 3-state, monophone HMM models with 1 Gaussian per state using 34281 utterances from 80 speakers. The choice of such a simple model was due to the memory and speed requirements of a command&control application. The corpora we used for testing were a German address corpus with approximately 23 utterances per speaker and a command&control corpus with approximately 234 utterances per speaker, so around 260 utterances per speaker in total. We then added the same number of OOV utterances for each speaker, resulting in a total number of 540 utterances per speaker. The test set consisted of 35 speakers.

NN results

When using the full set of features, we achieved the results that are listed in Table 2. The performance of our CM was measured in terms of classification error rate (CER), which is the number of misclassified patterns divided by the number of total patterns. The first row shows the results for the baseline speaker independent (SI) system. Also we listed the correct and false alarm rates (C_a and F_a, respectively) and correct and false rejection rates (Cr and Fr, respectively). The C_a rate directly corresponds to the recognition rate of the SI system. The relatively low initial recognition rates of 43.7% can be explained by the fact that we included 50% of OOV utterances into the test set.

	corr	wrong
3 out	88.6	11.4

Table 3: OOV classification results (in %) for the NN with 3 outputs, determined on the C_r-cases only

False alarms are those utterances that have been classified as being correctly recognized although they were misrecognized, so this corresponds to the WER of the SI system. Both NNs correctly reject (C_r) more than 50% of the utterances (remember that the more than 50% OOV were included in the test set). Although this is at the cost of rejecting 5.1% of the correctly recognized ones (F_r). 4.6% of the utterances were accepted although they were misrecognized (F_a). Applying this CM to the SI system would reduce the WER by more than 90% (from 56.3 to 4.6%). All utterances that were classified as unreliably recognized would be rejected and the user could be re-prompted for these utterances.

The results for the NN with 3 output are slightly worse. However, the 3-output net provides us with further information. This information is shown in Table 3. For all cases, in which an utterance was classified as misrecognized, we tried to judge, whether they were OOV utterances or not. In 88.6% of these (C_r-)cases of the baseline system the NN classified OOV words correctly.

In dialogue systems it could be beneficial for the course of the dialogue, to not just classify a word or sentence as being misrecognized, but to also know if it was an OOV word or not. This knowledge can greatly influence the following dialogue steps.

If we look at the large number of OOV patterns in the test set, we see that 88.8% of the 'misrecognized' patterns were OOV words. So if we simply classify all misrecognized words as OOV, we would be wrong in 11.2% of the cases on this test set, which is almost the same result delivered by the NN. So in this case our NN does not seem to be better than guessing. But since we do not detect 100% of the misrecognized words correctly, this cannot be directly compared. Furthermore when testing the CM in an online demonstration system, we saw that the OOV rejection works reasonably.

To assess the contribution of the prosodic features, we also trained NNs with 13, 31 and 35 input features, respectively. The 35 features comprised all acoustic-score based features and the features related to the speaking rate. The 31 features included the acoustic score-related features only. The 13 features were those related to the phoneme durations and speaking rates. The results are given in Table 4. As can be seen there the CER for the acoustic purely score-related features are the lowest followed by all features, the acoustic and tempo features and the purely duration features. Combining the duration-based features with the acoustic score-based ones unexpectedly does not yield any improvements. It seems that all information captured in the phonemes durations is also kept in the acoustic score related features. However, taking the duration-based features alone still shows

	CER	C_a	F_a	C_r	F_r
baseline (SI)	-	43.7	56.3	-	-
ac feat only	9.03	39.48	5.15	50.8	3.88
ac + spk rate	8.95	37.39	3.73	51.19	5.21
dur feat only	16.4	36.09	8.85	47.47	7.57
all	10.79	37.5	4.6	51.72	6.19

Table 4: Classification results (%) for different subsets of features

acceptable performance. One major disadvantage of the acoustic score-based features is that they strongly depend on the topology of the recognizer, the front-end, etc. So whenever there is a change in one of the above mentioned parameters, a new training of the NN would become necessary. On the contrary the duration-based features are independent of these parameters. So the use of these features would make the CM independent of the recognizer, which is a big advantage if it is to be used for different applications or on different platforms.

Semi-supervised Adaptation

For most command&control applications the use of speaker dependent (SD) systems is not feasible. However if the devices are used for a longer time by the same person, some kind of speaker adaptation should be employed to improve the performance of the SI system. Supervised adaptation schemes, that need a relatively large amount of adaptation data are also not desired. We want an approach that allows the user to start using the system right away, but which has the capability to adapt to the user's voice while he is actually using the system and is not aware that some adaptation is going on. The weakness of such unsupervised adaptation schemes is, that they often fail if the baseline performance of the recognizer is too low and too many misrecognitions occur. Since the adaptation has to rely on the recognizer output to be the spoken word, misrecognitions cause the wrong models to be adapted. If this happens repeatedly, the performance may even decrease.

At this point CMs can be applied to judge how reliable the recognizer result was and accept only those utterances for adaptation that were recognized with high confidence. We incorporated our CM using the full feature set into our recognizer and it guided the adaptation, such that we kind of semi-supervised the adaptation. All utterances that were marked as unreliable by the CM are excluded from the online adaptation. For the adaptation it is not necessary to know whether an unreliable word was an OOV word or not, but for other components of the system it might be important. So we used the NN with 3 outputs and the complete feature set for the following experiments. For adaptation we used a combination of MLLR and MAP adaptation, in which for the first 15 utterances MLLR with one global regression class is conducted. The resulting adapted models are then used as prior information for MAP from the 15^{th} utterance on. This was chosen to achieve a fast (but coarse) adaptation to the channel and speaker using MLLR and then doing a more specific adaptation (using MAP) as more adaptation data becomes available. Our adaptation approach is described in more detail in [9, 10].

#utterances	400	600	1000	2000
unsupervised	42.9	38.7	39.6	42.5
semi-sup. our CM	39.6	37.7	37.7	37.3
semi-sup. perf. CM	46.2	37.7	40.6	44.8

Table 5: Improvements in % WER w.r.t. the SI system using different adaptation approaches after a different number of utterances

For our adaptation experiments we used a different test set. 6 speakers that were recorded in a clean studio environment and the same front-end than described above were used. This time the test set comprised commands only and no OOV words, the initial WER of the SI system was 22.8%. For each of these speakers we have approximately 2000 utterances that were split into 10 sets of 200 utterances each. Set 10 was used for testing always.

Table 5 shows the results in WER when testing was conducted after a different number of utterances. It can be seen, that our unsupervised adaptation approach outperforms the SI models and further improves if more adaptation data becomes available. In the unsupervised approach all utterances no matter if they were misrecognized or not were used for adaptation. The WER can be reduced by 42.5%. When the semi-supervised adaptation is used, no further improvements can be observed. Here only utterances that were classified as being reliable were used for adaptation. We additionally added the results for a simulated 'perfect CM', in which we took only those utterances that we knew were recognized correctly using the SI system. The testing set of course remained unchanged and comprised also misrecognized utterances. Even using this perfect CM for supervision does only yield slight improvements. We thus conclude, that since the number of erroneous utterances that were used for unsupervised adaptation was quite small, this does not have an adverse effect on performance. Or at least the positive effect of using only correctly recognized utterances is nullified by the reduced number of utterances (resulting from the rejection of unreliable utterances). On the other hand, these results demonstrates the robustness of our unsupervised adaptation approach.

5. CONCLUSION

We presented new features for a CM approach, that uses a NN as a classifier. These features are based on phoneme duration statistics, that were obtained from the training data. Together with features that are related to the acoustic score present in the N-best output of the recognizer we achieved a classification rate of 90% at a false rejection rate of 5.1%. Simultaneously we succeeded in identifying OOV words in 91% of the C_r-cases. The application of this CM reduces the WER of the baseline recognizer by more than 90%. For all these utterances the user could e.g. be re-prompted.

When training NNs that use only subsets of the complete feature set, we found that the duration features perform slightly worse (with a CER of 16.4%) than the acoustic score-based features (with a CER of 9.7%). However this

performance is still acceptable and using only the duration-based features has the great advantage that the NN is completely independent from the recognizer and the front-end. This is not the case if acoustic score-related features are used. Combining both feature sets did not improve the classification results.

The features we used for formulating the CM are mostly related to single words. However, this approach can be easily extended to LVCSR systems.

We combined this CM using the full feature set with unsupervised speaker adaptation, such that adaptation is conducted in a semi-supervised manner and only those words that were recognized with high confidence are used for adaptation. WERs can be improved by 42.5% (w.r.t. the SI system) using the unsupervised approach, while using the semi-supervised approach did not improve the results any further. However also using a perfect CM for guiding the adaptation did yield only slight improvements, which demonstrates the robustness of our unsupervised adaptation approach.

REFERENCES

[1] T. Kawahara, C.-H. Lee, and B.-H. Juang. Flexible Speech Understanding based on Combined Key-Phrase Detection and Verification. *Transactions on Speech and Audio Processing*, pages 558–568, November 1998.

[2] P. Modi and M. Rahim. Discriminative Utterance Verification Using Multiple Confidence Measures. In Eurospeech97, Rhodes, Greece [11], pages 103–106.

[3] R. A. Sukkar and C.-H. Lee. Vocabulary Independent Discriminative Utterance Verification for Non-keyword Rejection in Subword based SR. *Transactions on Speech and Audio Processing*, pages 420–429, 1996.

[4] A. Wendemuth, R. C. Rose, and J. G. A. Dolfing. Advances In Confidence Measures For Large Vocabulary. In *1999 International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 705–708. IEEE, 1999.

[5] T. Kemp and T. Schaaf. Estimating Confidence Using Word Lattices. In Eurospeech97, Rhodes, Greece [11], pages 827–830.

[6] Ralf Kompe. *Prosody in Speech Understanding Systems*. Springer Verlag, 1997.

[7] Venables and Ripley. *Modern Applied Statistics with S-Plus.* Springer Verlag, 1997.

[8] Institute of Parallel and Distributed High Performance Systems, University of Tübingen, Department of Computer Architecture. *SNNS, Stuttgart Neural Network Simulator, User Manual v4.2*, 1998. http://www-ra.informatik.uni-tuebingen.de/SNNS.

[9] S. Goronzy and R. Kompe. A MAP-like weighting scheme for MLLR speaker adaptation. In *6th European Conference on Speech Communication and Technology*, volume 1, pages 5–8. European Speech Communication Association (ESCA), 1999.

[10] S. Goronzy and R. Kompe. A Combined MAP + MLLR approach for speaker adaptation. In *Proceedings of the Sony Research Forum 99*, volume 1, pages 9–14, 1999.

[11] European Speech Communication Association (ESCA). *5th European Conference on Speech Communication and Technology*, 1997.