AN EXPERIMENTAL STUDY OF ACOUSTIC ADAPTATION ALGORITHMS*

Ananth Sankar *Leonardo Neumeyer* *Mitchel Weintraub*

Speech Technology And Research Laboratory
SRI International
Menlo Park, CA

ABSTRACT

Recently there has been much interest in the area of adaptation for improved speech recognition in the presence of mismatches between the training and testing conditions. In this paper we focus on transformation-based maximum-likelihood (ML) adaptation. Some of the important adaptation parameters include whether the adaptation is sbibperformed in the *feature-space* or *model-space*, and whether the adaptation is *supervised* or *unsupervised*. An additional parameter is the adaptation data. For example adaptation may be performed using an independent dataset or the test data itself. The latter is referred to as *transcription-mode adaptation*. In this paper, we experimentally study the effect of these various parameters, and report on our findings.

1. INTRODUCTION

Recently, there has been much interest in the area of transformation-based ML adaptation to reduce the recognition degradation caused by acoustic mismatches between the training and testing conditions [1, 2, 3]. It is assumed that the speech hidden Markov models (HMMs) estimated in the training condition and an adaptation data set collected in the testing condition are available. The problem is to transform either the features or the models to reduce the mismatch between the two, and consequently reduce the degradation in performance caused by the mismatch. Depending on whether the features or models are being transformed, the methods are classified as *feature-space* or *model-space* algorithms [1, 4]. The form of the transformation is hypothesized, and its parameters are estimated by maximizing the likelihood of the adaptation data using the expectation-maximization (EM) algorithm [5].

In this paper, we study the effect of different ML-based adaptation parameters. These parameters include whether adaptation is performed in the feature-space or model-space, whether the method is supervised or unsupervised, and whether the adaptation is done on an independent adaptation set or on the test set (transcription-mode adaptation). In this work, we find that the model-space adaptation paradigm is more versatile than the feature-space paradigm in that several adaptation schemes can be easily explored. The model-space approach does not necessarily follow an underlying assumption of the actual distortion in the feature-space. However, as opposed to the feature-space, it makes it easy to estimate complex transformations of the HMM models. Our experiments show that the model-space approach results in a significant improvement over the feature-space approach.

On our test database, where the operating word error-rate was about 20%, we find that supervised methods, where the transcriptions for the adaptation data are available, perform only slightly better than unsupervised methods, which use the transcriptions from the recognition process. At error-rates lower than 20%, since the recognition transcriptions will have even fewer errors, we expect this observation will continue to hold. This indicates that at error-rates of about 20%, unsupervised adaptation can be used over supervised adaptation, thus removing the necessity of speaker enrollment.

Finally, our experiments show that transcription-mode adaptation performs almost as well as supervised adaptation. This result is mainly of academic interest, since if the test data is available for adaptation, it makes intuitive sense to use it. We were motivated, however, to measure the improvement resulting from transcription-mode adaptation.

2. ALGORITHM DESCRIPTION

2.1. Feature-Space and Model-Space Adaptation

We assume that we are given the adaptation data Y, and the trained HMMs Λ_X, where X is the training data. In *feature-space* adaptation, we assume that the adaptation data Y can be mapped to an estimate of the original features X by a transformation $F_\nu(Y)$ so that the original models Λ_X can be used for recognition. In *model-space* adaptation, we assume that the original models Λ_X can be mapped to the transformed models Λ_Y by a function $G_\eta(\Lambda_X)$. The models Λ_Y can then be used to recognize the test speech Y. The parameters of these transformations are estimated by maximizing the likelihood of the adaptation data. Thus, in the feature-space we need to find ν' such that

$$\nu' = \underset{\nu}{\operatorname{argmax}}\, p(Y|\nu, \Lambda_X), \tag{1}$$

and in the model-space we need to find η' such that

$$\eta' = \underset{\eta}{\operatorname{argmax}}\, p(Y|\eta, \Lambda_X). \tag{2}$$

*THIS WORK WAS SPONSORED BY ARPA THROUGH NAVAL COMMAND AND CONTROL OCEAN SURVEILLANCE CENTER UNDER CONTRACT N66001-94-C-6048.

The ML problem is solved by the EM algorithm [5] which iteratively finds the new estimates of the transformation parameters, given the current estimates. The EM algorithm is particularly useful when closed-form reestimation formulae for the transformation parameters can be obtained in each iteration.

In our work on transformation-based ML estimation, separate transformations are applied to separate gaussian clusters [2]. More complex functions can be implemented using a greater number of transformations, each tied to a separate gaussian cluster. However, as the number of transforms is increased, a larger amount of adaptation data is needed to estimate their parameters.

In computing the likelihoods above, we make use of the HMM state-conditioned observation probability density function (pdf) $p_y(y_t|s_t, \Lambda_X)$, where y_t is the tth feature vector, and s_t is the state at time t. In the case of a feature-space transformation, assuming that each vector y_t is transformed according to an invertible function $f_\nu(y_t)$, this pdf can be written as

$$p_y(y_t|s_t, \nu, \Lambda_X) = \frac{p_x(f_\nu(y_t)|s_t, \Lambda_X)}{|J_\nu(y_t)|}, \tag{3}$$

where $J_\nu(y_t)$ is the Jacobian of the inverse transformation $f_\nu^{-1}(x_t)$. Note that the inverse transformation must exist.

We have experimented with simple affine transformations of each feature component in the feature-space:

$$x_i = a y_i + b. \tag{4}$$

In this case, closed-form reestimation expressions for the transformation parameters can be obtained [1, 2]. Note that the above transformation can be implemented equivalently in the model-space by appropriately transforming the means and variances of the HMMs.

For more complex transformations such as

$$x = Ay + b, \tag{5}$$

where A is a full matrix, the Jacobian computation in Equation 3 makes closed-form reestimation expressions intractable. However, if we assume that only the HMM means are transformed by the affine transformation of Equation 5, and the variances remain unchanged, then closed-form reestimation expressions can be derived for the parameters A and b [3]. This method can be rationalized as a model-space transformation where there is no underlying feature-space assumption. We have found that this approach performs better than the component-wise feature-space approach since it makes use of the dependencies between the different feature components.

We also note that the model-space approach is more versatile than the feature space approach. For example, we may separately transform the variances and means of the models [1, 4]. In addition, we can also explore complex non-linear transformations such as the neural-network transformation approach described in [6].

2.2. Supervised and Unsupervised Adaptation

Equations 1 and 2 give the estimation method for the feature-space and model-space, respectively. In what follows, we restrict ourselves to the model-space case, since the equations are similar for the feature-space. Equation 2 can be rewritten as

$$\eta' = \underset{\eta}{\text{argmax}} \sum_W p(Y|\eta, \Lambda_X, W) P(W), \tag{6}$$

where the summation is over all possible word-strings W, and $P(W)$ is the apriori probability of the word-string W. $P(W)$ is usually computed using a statistical language model.

When the correct word-string is known, the method is a *supervised adaptation* scheme. In this case, we assign a unity probability to the correct word-string W^*, and Equation 6 becomes

$$\eta' = \underset{\eta}{\text{argmax}}\, p(Y|\eta, \Lambda_X, W^*). \tag{7}$$

In *unsupervised adaptation*, the correct word-string is not known. In this case, all possible word-strings must be considered as in Equation 6. However, we may use an algorithm based on the recognized word-string W_Y, that is,

$$\eta' = \underset{\eta}{\text{argmax}}\, p(Y|\eta, \Lambda_X, W_Y). \tag{8}$$

This is the usual approach to unsupervised adaptation. Alternately, we may use Equation 6, but consider only the top N word-strings in an N-best framework [7] to reduce the number of computations.

Note that in the unsupervised methods described above, it is necessary to first recognize the adaptation data to get the best word-string or the N-best word-strings. For a large recognition grammar, this is a significant overhead. In this paper, we propose a new scheme that removes this overhead. This is done by writing Equation 2 as

$$\eta' = \underset{\eta}{\text{argmax}} \sum_{ph} p(Y|\eta, \Lambda_X, ph) P(ph), \tag{9}$$

where ph is an acoustic subword sequence. We have considered sequences of context-independent (CI) phones for this purpose, and evaluated $P(ph)$ by using a model comprising a loop of CI phones, where every phone can follow every other phone with the same probability. While this is a simple model, we note that it is possible to incorporate additional knowledge by using a phone bigram or trigram model in order to compute $P(ph)$.

2.3. Transcription-Mode Adaptation vs. Adapting on an Independent Set

Unsupervised adaptation may be performed on the test data itself. This is called transcription-mode adaptation. This can be contrasted to adapting on an independent dataset. In applications such as non-real-time recognition of taped speech, it makes sense to use transcription-mode adaptation, since the adaptation and test data are identical. Transcription-mode adaptation can also be used to adapt to a particular test sentence. We refer to this as "self-adaptation". Intuitively, we expect transcription-mode adaptation to perform better than unsupervised adaptation on an independent data set. In this paper, we experimentally study the effect of using transcription-mode adaptation.

Number of Transforms	2	5	10	20
Full Matrix	17.9	18.4	19.1	21.5
Block-diagonal	18.6	18.1	17.6	18.2

Table 1. Word Error Rates (percent) for Supervised Adaptation using Full Matrix and Block-Diagonal Approaches for different number of transformations

Number of Transforms	2	5	10	20
Supervised	18.6	18.1	17.6	18.2
Unsupervised CI phone loop	18.6	18.8	18.8	20.9

Table 2. Word Error Rates (percent) for Supervised and Unsupervised (CI phone loop) Approaches for different number of transformations

3. EXPERIMENTS

We have experimented with a subset of the 1993 ARPA Wall Street Journal (WSJ) corpus, using a 20,000-word vocabulary bigram language model. In the experiments reported here, the speaker-independent (SI) models were trained using the WSJ native American speech training database, and adaptation was used to improve the performance for *native American test speakers*. The test set contained 10 male speakers each uttering about 25 sentences for a total of 230 utterances. For supervised adaptation we used the 40 common adaptation utterances from each speaker. For unsupervised adaptation, we used either the 40 common sentences or the test sentences from each speaker (transcription-mode adaptation).

The SI word error-rate on this database was 20.9%. We summarize the results of our adaptation experiments below.

3.1. Feature vs. Model Space

For feature-space adaptation, we used the component-wise affine transformation given by Equation 4. We observed no improvement over the SI performance (20.9% word error-rate). This demonstrates the lack of power in the component-wise transformation to improve performance for native speakers. We have previously reported significant improvement using this approach for non-native speakers [2] and noisy speech recognition [8].

In the model-space approach, we used the full-matrix affine transformation of Equation 5. Two different approaches were used to estimate the matrix. In the first approach, the matrix transformed the entire gaussian mean vector corresponding to the cepstrum, delta cepstrum, and delta-delta cepstrum. In the second approach, a separate transform was used for the cepstrum, delta-cepstrum, and delta-delta-cepstrum [4]. Thus the first approach uses a full matrix whereas the second approach uses a block-diagonal matrix. This leads to fewer parameters, and hence to more robust estimation with a limited amount of adaptation data. We have previously reported on the performance of these methods in [4]. In this paper we tabulate the results according to the number of transformations used for each approach in Table 1. Recall from Section 2.1 that a separate transformation is used for each gaussian cluster.

From the table, it can be seen that the model-space approach significantly reduced the error-rate to 17.6% as compared to the SI error-rate of 20.9%, whereas, as stated above, the feature-space approach gave no improvement. We also see that except for the case of two transformations, the block-diagonal approach is consistently superior to the full-matrix approach. This can be explained by considering the effect of the diagonal and off-diagonal blocks. In the block-diagonal approach, the diagonal blocks separately transform the cepstrum, delta cepstrum, and delta-delta cepstrum, and the off-diagonal blocks have no effect. In the full-matrix case, all blocks in a row affect the transformation of the corresponding feature. Intuitively we expect the off-diagonal blocks to have lesser importance than the diagonal blocks. This was also observed by examining the estimated transformation matrices from the full-matrix case. For any given number of transformations, the estimation of the block-diagonal transformation is more robust since it has fewer parameters than the full-matrix transformation. This explains the better performance of the block-diagonal approach. However, for the case of only two transformations, it is possible that the estimation of additional off-diagonal blocks offsets the fact that only two transformations are used, and hence results in superior performance for the full-matrix case as compared to the block-diagonal approach.

The table also shows that as we increase the number of transformations, the performance of the full-matrix approach deteriorates whereas the error-rate for the block-diagonal approach decreases to a minimum of 17.6% at 10 transformations. This can again be explained by the fact that far more parameters need to be estimated for the full-matrix case, making it less robust, especially for a larger number of transformations. In all the following experiments, we used the block-diagonal approach.

Before concluding this section, we observe that the full-matrix and block-diagonal approaches above are only two examples of the model-space transformation approach. In addition to the two methods described above, we have also used this paradigm to separately transform the variance and the means of the gaussians in the HMMs [4], and to derive a neural-network-based non-linear transformation approach [6].

3.2. Supervised vs. Unsupervised Methods

The model-space block-diagonal approach used in Section 3.1 was tested in both supervised and unsupervised modes. For unsupervised adaptation, we used the CI phone loop described in Equation 9. The error-rates for supervised and unsupervised adaptation are given in Table 2.

The table shows that as the number of transformations is increased, the performance of the unsupervised approach deteriorates. This can be explained by observing that the unsupervised algorithm makes use of the CI phone loop for guiding the adaptation as opposed to the correct word string in the supervised case, resulting in worse alignments with the HMM models and hence poorer estimates of the transformation parameters, especially for larger number of

Number of Transforms	2	5	10	20
Transcription-mode (Recognized string)	18.0	18.2	18.4	19.3
Transcription-mode (CI phone loop)	18.4	18.6	19.6	20.9
Supervised	18.6	18.1	17.6	18.2
Unsupervised CI phone loop	18.6	18.8	18.8	20.9

Table 3. Word Error Rates (percent) for Transcription-Mode adaptation, and comparison to supervised and unsupervised adaptation

transformations. The best performance of the unsupervised algorithm (18.6%) is comparable to that of the supervised method (17.6%). We note that using the recognized word-string instead of the CI phone loop is likely to improve performance. Furthermore, unsupervised adaptation can be run in an iterative fashion, where at each iteration the newly adapted models are used to recognize the adaptation data to generate word-strings for the next adaptation iteration. In another experiment on non-native speaker adaptation using a 5000 word vocabulary, the SI error-rate was 20.7%, and the supervised and unsupervised performance was 15.3% and 16.0% respectively. In this experiment, the unsupervised method used the recognized word-string instead of the CI phone loop to guide the adaptation. This shows that at operating error-rates of about 20%, it is possible to use unsupervised algorithms in order to remove the necessity of speaker enrollment which is required in supervised algorithms.

3.3. Transcription-mode vs. Adapting on Independent Set

We carried out transcription mode adaptation on the 10 native speaker data set described above using both the recognized word-strings (Equation 8) and the CI phone loop (Equation 9). These results are shown in Table 3. The table also replicates from Table 2 the results of using supervised and unsupervised adaptation on an independent dataset. We point out that while the independent dataset had 40 sentences, the transcription mode adaptation uses on average 23 sentences per speaker. This works to the detriment of the latter algorithm.

As with the unsupervised case, both the transcription mode methods deteriorate as the number of transformations is increased. This is probably due to the poorer alignments as compared to the supervised case as explained in Section 3.2. However, for two transformations, we see that the transcription mode adaptation approach is slightly better than the supervised technique. This can be explained by observing that the transcription mode algorithm adapts on the test data, whereas the supervised method adapts on an independent dataset. The best performance of the transcription-mode method (18%) is only slightly worse than that of the supervised method (17.6%). Transcription-mode adaptation has applications in such areas as transcribing tapes of recorded speech, and the results above motivate further research in this area. It is perhaps possible to exploit the test data for adaptation to decrease the error-rate even further. It is also interesting in the context of "self-adaptation" on a single test sentence.

4. CONCLUSIONS

In this paper, we have presented transformation-based ML adaptation approaches, and described the effect of the various parameters of these algorithms. In our experiments on native American test speakers, model-space adaptation gave a significant improvement over SI performance, while feature-space adaptation resulted in no improvement. Furthermore, model-space techniques can be used to explore a variety of possible adaptation algorithms, and are hence more flexible than feature-space techniques. We note that in previously reported results, the feature-space approach has given us a significant improvement for non-native test speakers [2] and for noisy speech [8]. We found that unsupervised methods performed almost as well as supervised approaches at operating error-rates of about 20%. This shows that we can use unsupervised techniques at these error-rates to obviate the need for speaker enrollment which is required in supervised adaptation. Finally, it was found that transcription-mode methods performed almost as well as supervised methods, even though the amount of adaptation data used for transcription-mode adaptation in our experiments was less than that for supervised adaptation.

REFERENCES

[1] A. Sankar and C.-H. Lee, "A Maximum-Likelihood Approach to Stochastic Matching for Robust Speech Recognition," *IEEE TSAP*, 1995, to appear.

[2] V. Digalakis, D. Rtischev, and L. Neumeyer, "Speaker adaptation using constrained reestimation of gaussian mixtures," *IEEE TSAP*, vol. 3, no. 5, pp. 357–366, 1995.

[3] C. J. Legetter and P. C. Woodland, "Flexible Speaker Adaptation using Maximum Likelihood Linear Regression," in *Proceedings of the Spoken Language Systems Technology Workshop*, pp. 110–115, 1995.

[4] L. Neumeyer, A. Sankar, and V. Digalakis, "A Comparative Study of Speaker Adaptation Techniques," in *Proceedings of EUROSPEECH*, 1995.

[5] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. Roy. Stat. Soc.*, vol. 39, no. 1, pp. 1–38, 1977.

[6] V. Abrash, A. Sankar, H. Franco, and M. Cohen, "Acoustic Adaptation using Non-Linear Transformations of HMM Parameters," in *Proceedings ICASSP*, 1996.

[7] R. Schwartz and Y.-L. Chow, "A Comparison of Several Approximate Algorithms for Finding Multiple (N-Best) Sentence Hypotheses," in *Proceedings ICASSP*, pp. 701–704, 1991.

[8] L. Neumeyer and M. Weintraub, "Robust Speech Recognition in Noise using Mapping and Adaptation Techniques," in *Proceedings ICASSP*, pp. 141–144, 1995.