Air Traffic Control Speech Recognition System Cross-Task & Speaker Adaptation

R. de Cordoba, J. Ferreiros, R. San-Segundo, J. Macias-Guarasa,
J. M. Montero, F. Fernandez, L.F. D'Haro, J.M. Pardo
Universidad Politécnica de Madrid

ABSTRACT

We present an overview of the most common techniques used in automatic speech recognition to adapt a general system to a different environment (known as cross-task adaptation) such as in an air traffic control system (ATC). The conditions present in ATC are very specific: very spontaneous, the presence of noise, and high speed speech. So, with a typical speech recognizer the recognition results are unsatisfactory. We have to decide on the best option for the modeling: to develop acoustic models specific to those conditions from scratch using the data available for the new environment, or to carry out cross-task adaptation starting from reliable HMM models (usually requiring less data in the target domain).

We begin with a description of the main techniques considered for cross-task adaptation, namely Maximum A Posteriori (MAP), Maximum Likelihood Linear Regression (MLLR), and the two together. We have applied each in two speech recognizers for air traffic control tasks, one for spontaneous speech and the other for a command interface. We show the performance of these techniques and compare them with the development of a new system from scratch. We also show the results obtained for speaker adaptation using a variable amount of adaptation data. The main conclusion is that MLLR can outperform MAP when a large number of transforms is used, and MLLR followed by MAP is the best option. All of these techniques are better than developing a new system from scratch, showing the effectiveness of mean and variance adaptation.

INTRODUCTION

The potential benefits of speech recognition are obvious for several types of applications, especially where the subject would improve his performance by not using his hands when performing a task, as is the case for an air traffic controller or a pilot. The difficulty is that speech recognition can be considered a very difficult problem in real-life environments because of several factors: the great variability between speakers and even for the same speaker resulting from stress amongst other things, significant variations between channels and/or environments, the presence of noise, etc. All these factors contribute to a reduction in the speech recognizer success rate that can lead to an unsatisfactory experience for the user. If there are too many recognition mistakes, the user is forced to correct the system which takes too long, it is a nuisance, and the user will finally reject the system. A high error rate is not acceptable for critical tasks, such as in ATC environments, which is probably the main reason for the low use of speech interfaces in ATC. *What can be done to deal with this problem?* Well, a lot of work has been carried out recently in several areas:

- *Increasing the size of the training database:*
  It is well known that the bigger the database used to train the system, the more robust the results from the recognizer. So, larger recordings are being used (even thousands of hours) that have greatly reduced the error rate of the recognizer. The problem with this approach, though, is that it is clearly at saturation point now; when there are more than enough recordings, very little additional improvement can be obtained by increasing its size.

- *Improving ASR system robustness:*
  Several techniques have been developed to

Author's Current Address:
R. de Cordoba, J. Ferreiros, R. San-Segundo, J. Macias-Guarasa, J. M. Montero, F. Fernandez, L.F. D'Haro and J.M. Pardo, Speech Technology Group. Dept. of Electronic Engineering, Universidad Politécnica de Madrid, E.T.S.I. Telecomunicacion. Ciudad Universitaria s/n, 28040-Madrid, Spain.

Manuscript received December 9, 2005; revised June 8, 2006.

improve the robustness of the recognizer against different and varying types of noise that are present in the speech signal.

- *Adapting the ASR system to speaker & task:* The focus of this paper. The recognizer adapts its models to different environments, conditions, speakers, and styles of speaking.

As we will see, very significant improvements can be obtained with this approach.

Although most commercial systems claim to be generic or speaker independent, that is, they are able to recognize any person, anywhere, and under any circumstances, the truth is that the error rate will be much lower if the system is adapted to the speaker, the environment, etc.

When we encounter a new environment for a speech recognition system, we have to take into account that the usual recognizers often perform well when tested on data similar to that used in training, but produce much higher error rates when tested on data from a new task. So, we have to consider two options. In the first place, we can begin from scratch using a lot of task-specific data. Nevertheless, collecting large amounts of data involves a great effort, it is very costly, and it is often impractical. The second option is to carry out cross-task adaptation as we did in a previous work [I]. We need a generic and robust recognition system that works well over a range of tasks. Then, with a small set of adaptation data, we adapt it to the new environment.

We have considered the two main adaptation techniques that can be applied to cross-task adaptation: Maximum A Posteriori (MAP) estimation [2] and Maximum Likelihood Linear Regression (MLLR) [3, 4]. We will show the behavior of each technique in both systems with varying sizes and characteristics. We will also present the effect of speaker adaptation in the command interface for Spanish, using the same techniques and varying the size of the adaptation set to find the point where MAP outperforms supervised MLLR. Other relevant works in cross-task adaptation are [5, 6], where MAP and MLLR are compared for different environments. In this paper, we provide additional refinements of these techniques, and we apply them in air traffic control tasks.

This work has been carried out under the INVOCA project, for the public company AENA, which manages Spanish airports and air navigation systems [7]. We have worked with two different systems, the first is a command interface used to control the radar display in an ATC position, and the second is a spontaneous speech system with conversations between controllers and pilots. Both were implemented in two languages: Spanish and English; nevertheless, we will show the results only for Spanish, as similar conclusions can be extracted for English and they add no relevant information.

Another field in speech processing where model adaptation can be useful too is speaker recognition [8], if only a reduced amount of data is available for each speaker, its model can be obtained adapting a generic one using the techniques presented herein. It can also be useful for channel adaptation, when the speaker changes the channel used for the recognition.

OVERVIEW OF ADAPTATION TECHNIQUES

We will focus on model-based adaptation, where the acoustic models are the parameters being adapted. Another possibility for adapting is, for example, to apply a spectral transform to the feature vector or to try to classify the speaker/environment in one of several categories or groups, but model-based adaptation is especially successful, as we will see.

Adaptation in speech recognition has been an issue for several years and several techniques have been proposed. These techniques can be classified according to several criteria, the main one being the amount of adaptation data available. If the amount of adaptation data is very small, e.g., only the speech recorded from a phone call is available, it is called rapid adaptation, which is obviously the most difficult task and specific techniques need to be applied. If the amount of adaptation data is medium or large, several techniques can be used successfully, namely Maximum a Posteriori (MAP) [2], Maximum Likelihood Linear Regression (MLLR) [3, 4], and variations of both. We will focus on the comparison between these techniques in the scope of cross-task and speaker adaptation. First, we will present a brief description of both. Further details can be found in the aforementioned references.

In speech recognition, we work with continuous Hidden Markov Models (HMM) [9], where every phone in the model set is modeled using a set of states – usually three – each made up of a set of Gaussian distributions (characterized by a mean vector and a covariance matrix for each distribution, and a vector with distribution weights). In a model-based adaptation, all of these parameters are adapted using the adaptation data.

MAP

Also known as Bayesian adaptation, MAP adaptation involves the use of prior knowledge about the model parameter distribution. The idea is to use a previously well-trained model as the prior knowledge.

The main advantage of MAP is that when enough adaptation data is available, the estimation converges to the maximum likelihood criterion, which is the optimum for estimating the parameters from scratch. So, it should be the best technique for large adaptation sets.

The disadvantage is that it does not modify the parameters that do not appear in the adaptation data. Then, it can be a bad choice when the adaptation set is small.

MLLR

In MLLR, a set of transformations for the model parameters is computed which reduces the mismatch between an initial model set and the adaptation data. The effect of these transformations is to shift the component means and to alter the variances in the initial system so that each state in the model is more likely to generate the adaptation data. The mathematics behind the transformation matrix is complex, so the reader should take a look at [3, 4] to find out the details. The main idea is that the transformation matrix is obtained by solving a maximization problem using the Expectation-Maximization (EM) technique, using the likelihood of the adaptation data as the maximization criterion.

One important issue is that it is not feasible to compute a transformation matrix for every unit in the model set. The solution is to group the most similar units given a similarity measure or distance between units and then to compute a common transform for all. Several distances can be considered: Euclidean, Symmetric likelihood, the average of two Kullback-Leibler distances between two Gaussians, etc.

So, a regression-class tree is created using the original model parameters and a top-down strategy. First, all units are grouped into a single cluster. Then, every cluster is divided iteratively into two clusters and the units are reassigned to their closest cluster (using one of the aforementioned distance measures). This procedure is repeated several times for each division and the number of final clusters has to be chosen according to the amount of data available.

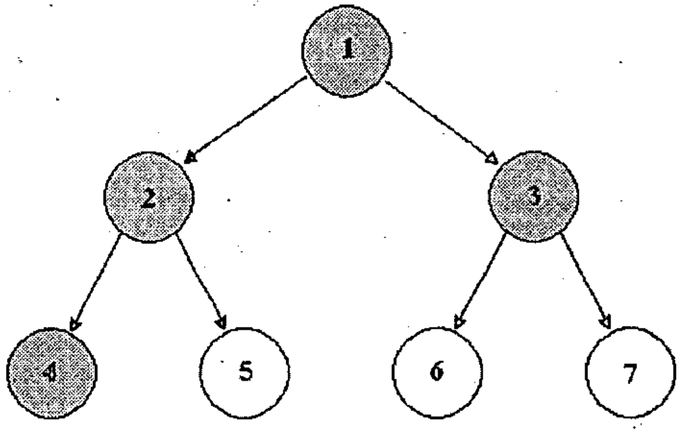

Fig. 1. Regression Class Tree

Then, during adaptation and according to the amount of adaptation data available, the set of transformations to be estimated can be chosen (a threshold is established). As we will need to compute one transformation matrix for each cluster, the matrix will not be reliable if there is insufficient data. That is why the threshold has to be estimated carefully. In Figure 1, we can see an example of a regression class tree where only nodes 1 to 4 have enough data, so only the transformation matrixes for nodes 2, 3, and 4 will be computed (1 is a non-terminal node and its two children are above the threshold).

In this paper, we show the results obtained for MLLR using different values for the number of clusters/transforms considered.

In summary, the main advantage of MLLR is that it shares the transforms between similar units, so that every parameter in the model set gets updated in the adaptation process even though it does not appear in the adaptation set. So, this technique should be better than MAP for smaller adaptation sets (medium size).

SYSTEM SETUP

Databases Used

We have used two different databases:

- *An isolated speech database*, used in a command interface to control the ATC position. It contains single words as well as some compound words. A few examples are as follows: "Altitude Filter," "Category Filter One," "Freeze Flight Plan," etc.

- *A spontaneous speech database*, which consists of conversations between controllers and pilots. It is a difficult task, noisy and spontaneous. Examples: "lufthansa four two seven nine start up approved clear to frankfurt standard departure somosierra one echo three six left squawk one zero two three report parking position," "airfrance two one zero one start up approved clear Paris de gaulle standard departure somosierra one echo squawk one zero seven three."

General Conditions of the Experiments

The system uses a front-end with PLP parameters in the cepstral domain (a common parameterization technique used for speech recognition) derived from a Mel-scale filter bank (MF-PLP), with 13 coefficients including their first- and second-order differentials, giving a total of 39 parameters for each 10 msec. frame.

As channel conditions are noisy, we decided to apply two normalization techniques that are especially designed to compensate for channel variations: Cepstral Mean Normalization (CMN) and Cepstral Variance Normalization (CVN). The effect of inserting a transmission channel into the input speech is to multiply the speech spectrum by the channel transfer function. In the log cepstral domain, this multiplication becomes a simple addition which can be removed by subtracting the cepstral mean from all input vectors. This is the objective of CMN: subtract the mean of all vectors. Its only drawback is that the mean has to be estimated over a limited amount of speech data, so the subtraction will not be perfect. Nevertheless, this simple technique is very effective in practice where it compensates for long-term spectral effects such as those caused by different microphones and audio channels. CVN adds another normalization: every parameter is multiplied by the quotient of the standard deviation of the parameter in the whole database and the deviation of the parameter in the specific file. This way, the variability of parameters throughout the database is compensated.

All systems use context-dependent continuous HMM models [10] (they take into account the allophones which are adjacent to the current one) built using decision-tree state clustering.

Isolated Word Recognition Experimental Setup

A specific database was recorded in the Invoca project [1]. The vocabulary of the task comprises 228 different commands (words or compound-words). We had 16 hours of speech, and we assigned 11 hours to training/cross-task adaptation and 5 hours to validation. We had a total of 30 different speakers, all identified, so we could do speaker adaptation experiments. For cross-task adaptation, we have used the Spanish SpeechDat database as a starting point (only the isolated speech part, 41.8 hours). In this database, 4,000 speakers uttered the following items: application words, isolated digits, cities, companies, names, and surnames.

Spontaneous Speech Recognition Experimental Setup

Another database was created for these experiments. It consists of live recordings on five air traffic control real positions (Arrivals, Departures, Grounds North, Grounds South, and Clearances). As Barajas is an International Airport, both Spanish and English utterances have been obtained interwoven. The recordings proceeded for about one week per position on a channel where only the controller speech was captured. During these recordings, a group of about 30 different controllers for each position contributed to the database with their voices.

To train the HMMs from scratch and to carry out cross-task adaptation, we used speech from the Clearances position. We had 9 hours of speech, and we dedicated 8 hours to training/cross-task adaptation and 1 hour to validation. The vocabulary size is 1104 words. For cross-task adaptation, we have used the SpeechDat database as a starting point again, but its continuous speech part, with 4,000 speakers and a total of 43.2 hours for training.

EXPERIMENTS AND RESULTS FOR ISOLATED WORD RECOGNITION

New System From Scratch

We used the training set described above to create HMM models from scratch. Using context-dependent models with 1509 states after the tree-based clustering, each state with 6 mixture components, the error rate was 0.90% for the vocabulary with 228 commands.

Cross-Task Adaptation

We used robust context-dependent HMM models trained with the SpeechDat database (as we have seen in the previous section, this database has 42 hours of speech versus 11 hours available in the Invoca database). Without adaptation, the error rate is 2.1%, worse than the system from scratch, showing the mismatch between both environments. Beginning from those models, we have considered two types of adaptation: MAP and supervised MLLR (in supervised mode, the transcription of the adaptation data is used and the estimated transforms are more reliable). For MLLR, we have considered regression-class trees of different sizes (between 64 and 1024 transforms) and several iterations were run. We can see the results in Table 1 for the optimum iteration (usually, the fourth one).

Table 1. MLLR Adaptation (Isolated) (% error rate)

	64	128	256	511	1000
Means Adaptation	1.48	0.99	0.89	0.91	0.84
Means and Variances Adaptation	1.39	0.91	0.82	0.84	0.83

Table 2. MAP Adaptation (Isolated)

	% Error Rate
Means Adaptation	1.00
Means and Variances Adaptation	0.81
MLLR + MAP	0.79

Table 3. MAP & MLLR Speaker Adaptation

	# Words Adaptation	MAP	MLLR
Means and Variances Adaptation	50	0.56	0.54
	228	0.27	0.27
	456	0.17	0.27
	684	0.17	0.15

The results for MAP can be seen in Table 2. We can see that they are better than the results from scratch (a 10% improvement), showing that the original database is useful and complements the adaptation database, as we wanted. We can also see that variances adaptation is clearly needed to improve the system trained from scratch.

We can see that MAP outperforms MLLR when the number of transforms is low (up to 128), but both can obtain similar results with 256-1024 transforms. The results also show that

Table 4. MLLR Cross-Task Adaptation (Spontaneous)

# Nodes	64	128	255	507	1004	1956	3694
Error Rate	14.78	14.00	13.08	12.60	12.11	12.08	12.05

there is enough data to estimate this number of transforms. When applying MAP to the best MLLR models the result increases slightly, thus providing the best performance. This confirms the results obtained in [1], where MLLR+MAP obtained similar results to just MAP.

Speaker Adaptation

For speaker adaptation we began from the best models so far, obtained using MAP with means and variances adaptation (0.81% error rate), and we varied the amount of data dedicated to the task. In this database, every speaker uttered the list of 228 application commands five times per command. We dedicated up to three of these repetitions for speaker adaptation and carried out the test with the other two repetitions (the error rate for this new test set is 0.73%). The results for MAP and MLLR speaker adaptation are shown in Table 3. With more transforms the results are similar, as very few transforms were actually used (occupation threshold not reached).

We can extract some interesting conclusions from these results:

- Both techniques provide very similar results. We are probably very close to the maximum performance of the system.
- With only 50 words of speaker adaptation MLLR slightly outperforms MAP (as could be expected as discussed in the comparison of both techniques), and the relative improvement is a remarkable 26% for MLLR.
- With 456 words, MAP outperforms MLLR, but surprisingly with 684 words MLLR is slightly better than MAP.

In any case, both techniques are close to a limit in performance. 0.15% equals 6 mistakes (from 4,086 files). This limit would be very difficult to surpass.

EXPERIMENTS AND RESULTS FOR SPONTANEOUS SPEECH RECOGNITION

New System From Scratch

We used the 8-hour training set to create the HMM models from scratch. All adaptation results refer to the 503 test sentences with a vocabulary of 1,104 words. Using context-dependent models with 1506 clustered states, each one with 8 mixtures, the error rate was 12.70%.

Cross-Task Adaptation

Again, we used context-dependent HMM models trained with the SpeechDat database (43.2 hours) with a total of 1,807 states and 7 mixture components per state. Using these models without adaptation, the result is a 19.51% error rate, so they are clearly worse than beginning from scratch. There is a clear mismatch between both tasks; the most remarkable aspect is the spontaneity of the Invoca database, whereas SpeechDat is read speech. After the experience with isolated speech, we decided to carry out means and variances adaptation. Using MAP, the error rate was 12.43%. We can see the results for MLLR (% error rate) in Table 4.

We can extract the following conclusions from these results:

- MAP outperforms MLLR with the typical number of transforms (up to 512), as could be expected resulting from the large size of the adaptation set, but we can see that using MLLR with a large number of nodes is better than MAP (a 4% relative improvement).
- There is enough data to train up to 2000-4000 transforms in MLLR.
- Both cross-task MAP and MLLR adaptation are again better than beginning from scratch. The reason for this improvement in cross-task adaptation is that the adaptation set is much smaller than the training set in SpeechDat, so that we can take advantage of some information from the original system.

We then applied MAP to the best MLLR models, and the results improved to 11.66%. So, unlike the results obtained in [1], where MLLR+MAP obtained similar results to just MAP, and our results for the isolated task (a low improvement, from 0.81 to 0.79), in this task the improvement is remarkable over MLLR alone (2.4% relative improvement).

Again, the best result using adaptation – 11.66% – is much better than system trained from scratch – 12.70% – with a remarkable 8.2% relative improvement. We should also mention that all these techniques can be applied in real-time. In fact, they mean no additional processing time to the usual recognition stage as they are applied in the training stage, where real-time is not a must.

CONCLUSION

We have shown a whole set of adaptation experiments using MAP, MLLR, and both in two different tasks.

For the isolated speech task, the cross-task experiments show that MAP and MLLR obtain similar results when using more than 500 transforms, the best solution being MLLR followed by MAP. All are better than creating new models from scratch. In the speaker adaptation experiments, we showed that: 50 words are enough for a remarkable improvement; with 50 words, MLLR slightly outperforms MAP; using more words, both techniques have similar results; the best result means a 79.5% relative improvement over no-speaker adaptation with a resulting negligible error rate.

For the spontaneous speech system, the cross-task experiments show that MLLR outperforms MAP when using 1024 or more transforms, and now the best is clearly MLLR followed by MAP, with a relative improvement of 6.2% over MAP alone and 2.4% over MLLR alone.

In summary, all the options considered for cross-task adaptation are better than beginning from scratch, showing the appropriateness of this approach. Another crucial conclusion is that a generic speech recognizer is a bad option for a spontaneous ATC task (19.51% error rate even with the language model adapted to the task), so it is absolutely necessary to use task specific data to obtain an acceptable error rate in speech recognition for ATC.

ACKNOWLEDGEMENTS

The authors thank Jose D. Romeral (in memoriam) for his important contributions in many experiments in this paper. We also thank those at the Human-Computer Technology Lab (Universidad Aut6noma de Madrid) who recorded the Isolated Speech Database; and AENA staff who participated in the recordings of the Spontaneous Database.

REFERENCES

[1] Cordoba, R, Woodland, P.C. and Gales, M.J.F.,
Improved Cross-Task Recognition Using MMIE Training,
IEEE ICASSP 2002, pp. 85-88.

[2] Gauvain, J.L. and Lee, C.H.,
Maximum A-Posteriori Estimation for Multivariate Gaussian
Mixture Observations of Markov Chains,
IEEE Trans. SAP, Vol. 2, pp. 291-298, 1994.

[3] Gales, M.J.F. and Woodland, P.C.,
Mean and Variance Adaptation Within the MLLR Framework,
Computer Speech & Language, Vol. 10, pp. 249-264, 1996.

[4] Leggetter, C.J. and Woodland, P.C.,
Flexible Speaker Adaptation Using Maximum Likelihood
Linear Regression,
Proc. ARPA SLT Workshop, pp. 104-109.
Morgan Kaufmann. 1995.

[5] Gales, M.J.F., Dong, Y., Povey, D. and Woodland, P.C.,
Porting: Switchboard to the Voicemail Task,
IEEE ICASSP 2003, pp. 1-536-539.

[6] Lefevre, F., Gauvain, J.L. and Lamel, L.,
Improving Genericity for Task-Independent Speech Recognition,
Proceedings of Eurospeech 2001, pp. 1241-1244.

[7] *INVOCA Project Synopses*. Eurocontrol. Analysis of R&D in
European Programmes.
http://www.eurocontrol.int/ardep-arda/public/jsp/Ardep.jsp?MENU
ITEM=1014&Proj=AEN043.

[8] Faundez-Zanuy, M. and Monte-Moreno, E.,
State-of-the-art in speaker recognition,
IEEE Aerospace and Electronic Systems Magazine,
Vol. 20, Issue 5, March 2005, pp. 7-12.

[9] Huang, X.D., Ariki, J. and Jack, M.A.,.
Hidden Markov Models for Speech Recognition,
Edinburgh University Press, 1990.

[10] Cordoba, R., Macias-Guarasa, J., Ferreiros, J., Montero, J.M. and
Pardo, J.M.,
State Clustering Improvements for Continuous HMMs in a
Spanish Large Vocabulary Recognition System,
ICSLP 2002, pp. 677-680.