Procesamiento del Lenguaje Natural, núm. 35 (2005), pp. 21-27 recibido 27-04-2005; aceptado 01-06-2005

New Advances in Cross-Task and Speaker Adaptation for Air Traffic Control Tasks

Ricardo Córdoba, Javier Macías-Guarasa, Valentín Sama, Roberto Barra, José Manuel Pardo
Grupo de Tecnología del Habla. Departamento de Ingeniería Electrónica. UPM.
E.T.S.I. Telecomunicación. Ciudad Universitaria s/n, 28040 Madrid, Spain.
{cordoba, macias, vsama, barra, pardo}@die.upm.es

Resumen: En este artículo exploramos diversas opciones para la adaptación a tarea en reconocimiento de habla y las comparamos con desarrollar el sistema nuevo desde cero. Comparamos adaptación a tarea mediante MAP y MLLR, y ambas en serie, en dos reconocedores de habla para tareas de control del tráfico aéreo, una para habla espontánea y la otra para una interfaz de comandos. Vamos a mostrar cómo MLLR puede incluso superar a MAP cuando se utilizan un número elevado de transformadas, cómo MLLR seguido de MAP es la mejor opción, y también vamos a proporcionar varias pistas de cuáles son las mejores opciones para la creación de los árboles de clases de regresión utilizados en MLLR. En todos los casos, demostramos la efectividad de la adaptación conjunta de medias y varianzas. Para la interfaz de comandos, también incluimos la comparación entre MAP y MLLR para adaptación a locutor utilizando una cantidad variable de datos de adaptación.

Palabras clave: adaptación a tarea, adaptación a locutor, reconocimiento de habla, MAP, MLLR.

Abstract: In this paper we explore several options for cross-task adaptation in speech recognition and compare them to develop the new system from scratch. We compare cross-task MAP and MLLR adaptation, and both of them together, in two speech recognizers for air traffic control tasks, one for spontaneous speech and the other one for a command interface. We show how MLLR can even outperform MAP when a big number of transforms is used, how MLLR followed by MAP is the best option, and we also provide some hints of which are the best options to create the MLLR regression class trees. In all cases, we show the effectiveness of means and variance adaptation. For the command interface, we also include the comparison between MAP and MLLR for speaker adaptation using a variable amount of adaptation data.

Keywords: cross-task adaptation, speaker adaptation, speech recognition, MAP, MLLR.

1 Introduction

To develop a speech recognition system in a new environment we have to take into account that the usual speech recognition systems often perform well when tested on data similar to that used in training, but give much higher error rates when tested on data from a new task.

So, we have to consider two options. In first place, we can begin from scratch, but collecting a large amount of task-specific data needs a great effort, it is very costly, and is often impractical. The second option is to do cross-task adaptation as we did in a previous work (Cordoba, 2002a) and (Cordoba, 2004). 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.

This work has been done under the project INVOCA, for the public company AENA, which manages Spanish airports and air navigations systems (INVOCA, 2002). We have worked with two different systems, the first one is a command interface, used to control the air traffic controller position, and the second one is a spontaneous speech system with conversations between controllers and pilots. Both were implemented in two languages, Spanish and English; therefore, we have worked with four different databases in total.

ISSN: 1135-5948

We have considered the two main adaptation techniques that can be applied to cross-task adaptation: maximum a posteriori (MAP) estimation (Gauvain, 94) and maximum likelihood linear regression (MLLR) (Gales, 96; Leggetter, 95). We will show the behavior of each technique in both systems with varying sizes and characteristics. In both techniques, we will see the effect of adapting the means alone or the means and variances together.

We will also see 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.

The paper is organized as follows. In section 2 we present the database used in the experiments and the general conditions of the experiments. In section 3, the results for the command interface and the spontaneous speech systems are described. The conclusions are given in Section 4.

2 System Setup

2.1 Databases used

We have used two different databases:

- An isolated speech database, used in a command interface to control the air traffic controller position. In fact, it contains some compound words.
- A spontaneous speech database, which consists of conversations between controllers and pilots. It is a very difficult task, noisy and very spontaneous.

2.2 General conditions of the experiments

The system uses a front-end with PLP coefficients derived from a Mel-scale filter bank (MF-PLP), with 13 coefficients including c_0 and their first and second-order differentials, giving a total of 39 parameters for each 10 msec. frame.

As the channel conditions are noisy, we decided to apply CMN plus CVN. CMN plus CVN meant a 15% improvement in average over CMN alone in preliminary experiments.

For all experiments, we have considered very detailed sets of allophones. In Spanish, we used a set of 45 units: we differentiate between stressed / unstressed / nasalized vowels, we include different variants for the vibrant 'r' in Spanish, different units for the diphthongs, the fricative version of 'b', 'd', 'g', and the affricates version of 'y' (like 'ayer' and 'cónyuge'). In (Córdoba, 2002b) we show the results with three different sets of units, where the best was for 45 units (12% over a 23 units set).

In English, we defined a very detailed set of 61 units: we have 19 vowels and diphthongs plus 16 of them stressed. The remaining units are consonants.

All systems use context-dependent continuous HMM models built using decision-tree state clustering. We have developed our own rules using phonetic relevant information in Spanish and English.

2.3 Isolated word recognition experimental setup

A database specific to the project Invoca was recorded. The vocabulary of the task consists of 228 different commands (words or compound-words). For Spanish, we had 16 hours of speech, and dedicated 11 hours to training/cross-task adaptation and 5 to validation. For English, we had 10 hours and dedicated 6 to training/cross-task adaptation and 4 to validation. In both languages, we had a total of 30 different speakers, all identified, so we could do speaker adaptation experiments.

For cross-task adaptation, we have used as starting point the SpeechDat database for Spanish (its isolated speech part), with 4,000 speakers who utter the following items: application words, isolated digits, cities, companies, names, and surnames. There are a total of 44,000 files for training (41.8 hours).

2.4 Spontaneous speech recognition experimental setup

Another database was created for these experiments. It consists of recordings on five air traffic control real positions (Arrivals, Departures, Madrid Barajas North taxing, Madrid Barajas South taxing, and Clearances). As Barajas is an International Airport, both Spanish and English utterances have been obtained interleaved. 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 with their voices to the database. Although they knew, for legal requirements, that they were being recorded, they were doing their real work and the speech produced was

fully spontaneous. In fact, recording equipment was in a different room from the actual controlling facility, and thus, no disturbance has been produced on their work.

The only drawback for our purposes is that they did not allow us to control the identity of each speaker, so we could not do speaker adaptation experiments in this task.

Expert labelers that marked each sentence with relevant information regarding both the correct grapheme and the artifacts that actually appeared in the speech realization processed the recordings.

To train the HMMs from scratch and to do cross-task adaptation, we used speech from the Clearances position. Table 1 shows the database details. The vocabulary size is 1104 words, and the test set perplexity of the bigram language model that we used is 15.2. We decided to use a bigram for two reasons: the phraseology used by the controllers is very regular, so a bigram could be enough, and the text that we had available was clearly too small to train a trigram LM.

Table 1. Database for continuous speech

Sentences/hours	Spanish	English
Training set	4,588 / 8.0	2,700 / 5.7
Test set	503 / 0.9	453 / 0.9

For cross-task adaptation, we have used as starting point the SpeechDat database (its continuous speech part), with 4,000 speakers who utter 9 phonetically rich sentences. Removing sentences with mistakes and 500 sentences for test, we used a total of 31,393 sentences for training (43.2 hours).

3 Experiments and Results

3.1 Isolated word recognition - Spanish

3.1.1 New system from scratch

We used the train set described in Section 2.3 to create HMM models from scratch. First, we estimated context independent (CI) models with 10 mixture components per state: we got 2.6% error rate with the vocabulary of 228 commands. Then, we estimated context dependent (CD) models with 1509 states after the tree-based clustering, each state with 6 mixture components. The error rate with that system was **0.90%**.

3.1.2 Cross-task adaptation

We use robust context-dependent HMM models trained with the SpeechDat database. The optimum error rate obtained in that environment was 3.8% with a 500 words dictionary, using a total of 1900 states in the HMMs and 8 mixture components per state.

Using those models without adaptation, the result is 2.1% error rate, so they are worse than the system from scratch, showing that there is a mismatch between both environments.

Beginning from those models, we have considered two types of adaptation: MAP (Gauvain, 94) and supervised MLLR (Gales, 96; Leggetter, 95), as we know the transcription of the adaptation data.

For MLLR, we have considered regression class trees of different sizes (from 64 until 1024 transforms), block-diagonal linear transformations and several iterations were run. We can see the results in Table 2. Results with 2048 transforms were worse and have not been included. The benefits of iterating in MLLR are clearly more remarkable when using fewer transforms, although some saturation in the results is observed as we approach the 0.80 'limit'.

Table 2. MLLR adaptation (Isolated-Sp.) (% error rate)

	# nodes	Iteration number				
		1	2	3	4	5
Means adaptation	64	1.59	1.55	1.51	**1.48**	1.50
	128	1.37	1.20	1.07	1.04	**0.99**
	256	1.24	1.04	0.93	0.91	**0.89**
	511	1.09	0.99	0.93	0.92	**0.91**
	1000	0.98	0.89	0.85	**0.84**	**0.84**
Means and variances adaptation	64	1.61	1.54	1.40	1.41	**1.39**
	128	1.18	1.06	1.01	0.95	**0.91**
	256	1.12	0.90	**0.80**	0.82	0.89
	511	0.86	0.84	0.85	**0.84**	0.89
	1000	**0.83**	**0.83**	0.84	0.86	0.88

The results for MAP can be seen in Table 3. We can see that the results are better than the ones obtained beginning from scratch (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.

Table 3. MAP adaptation (Isolated-Spanish)

	% error rate
Means adaptation	1.00
Means and variances adaptation	**0.81**
MLLR + MAP	**0.79**

We can see that MAP outperforms MLLR when the number of transforms is low (up to 128), but can obtain similar results with 256-1024 transforms. The results also show that there is enough data to estimate this number of transforms. When applying MAP to the best MLLR models the result increases slightly providing the best performance. This confirms the results obtained in (Cordoba, 2002a), where MLLR+MAP obtained similar results to just MAP (the best in this case).

3.1.3 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).

In this case, we are going to vary the amount of data dedicated to the adaptation. In this database, every speaker uttered five times the list of 228 commands defined for the application. We are going to dedicate up to three of those repetitions for speaker adaptation and do the test with the other two repetitions (there are a total of 4,086 files for the test set in these experiments). Considering this new test set the error rate is 0.73%. The results for MAP and MLLR speaker adaptation are shown in Table 4. For MLLR, several iterations were run again, and the results usually converged after 4 iterations, so the results for the forth iteration are shown. We present the results using 128 transforms. We also considered the use of bigger trees, but results were similar or equal, as very few transforms were actually used because the occupation threshold was not reached.

Table 4. MAP & MLLR speaker adaptation

	Adaptation set (words)	MAP	MLLR
Means adaptation	50	0.56	0.61
	228	0.29	0.47
	456	**0.17**	0.39
	684	**0.17**	0.29
Means and variances adaptation	50	0.56	0.54
	228	0.27	0.27
	456	**0.17**	0.27
	684	**0.17**	**0.15**

We can extract some interesting conclusions from this results:

- Variance adaptation has very little effect on MAP, but for MLLR the improvement is obvious (30% error rate reduction in average).
- Using variance adaptation, 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 outperforms MAP slightly (as could be expected), 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) that may be are impossible to recover.

3.1.4 Hints for regression class tree creation in MLLR

MLLR makes use of a regression class tree to group the Gaussians in the model set, so that the set of transformations to be estimated can be chosen according to the amount of adaptation data that is available. Before the experiments for MLLR presented in the previous sections were made, we considered several options for the creation of the regression class tree, which we will describe now. The conclusion of these experiments is that, at least in this task, tree creation is not crucial when a large number of transforms is used or several iterations of MLLR are run, probably because we are too close to the best performance that can be obtained with the system (close to 0.8% error rate). Nevertheless, differences can be observed if we consider an intermediate number of transforms and the first iteration. So, we want to describe the alternatives that we have considered and their overall results to serve as guidelines for tree creation.

A. Balanced / unbalanced tree

In a balanced tree, all nodes are split when the next level of the tree is created. In an unbalanced tree, the 'biggest' node is selected for splitting. Several criteria can be used to decide the biggest node, e.g., the largest total distance between its Gaussians (the largest intra-cluster dispersion). Our experiments show that the balanced tree provides better results

than the unbalanced one, especially when small trees are used (3-4% relative error reduction). At the same time, the number of nodes that have more than 2 Gaussians is bigger for the balanced tree.

B. Minimum number of Gaussians per node
We experimented with several values for the threshold to be applied to the number of Gaussians in each node during the tree creation process, and the best results were obtained using 2 Gaussians per node, although similar but slightly worse results were obtained using 6 Gaussians. A bigger threshold obtained worse results.

C. Distance between Gaussians
To compute the distance between Gaussians, which is used for the clustering algorithms, several alternatives can be used. We have considered the following ones:

1. Mahalanobis

$$d(p,q) = \left(\mu^{(p)} - \mu^{(q)}\right)^T \cdot \Sigma^{-1} \cdot \left(\mu^{(p)} - \mu^{(q)}\right)$$

2. Symmetric likelihood: measures the decrease in likelihood after joining Gaussians p and q into Gaussian g.

$$d(p,q) = \left[\log lik(p) + \log lik(q)\right] - \log lik(g)$$

3. J-Divergence: the average of two Kullback-Leibler distances between the two Gaussians.

$$J(p,q) = \frac{KL(p \| q) + KL(q \| p)}{2}$$

4. Euclidean distance.

In Table 5, we can see the average word error rate for each distance for the first iteration of MLLR using 128 and 256 transforms for cross-task adaptation and 128 transforms with the 4 adaptation sets considered for speaker adaptation (means and variances adaptation in all cases). The best overall distance is the Symmetric likelihood, clearly better than using the usual Euclidean in both cases. So, this is the one that we have used in all results presented in this paper.

Table 5. WER for several distances used in regression class tree creation

	Mahala-nobis	Sym. likelihood	J-Diverg.	Euclidean
Cross-task	1.19	**1.15**	1.18	1.18
Speaker	**0.372**	**0.373**	0.398	0.408

3.2 Isolated word recognition - English

Again, we used the train set described in Section 2.3 to create HMM models from scratch. First, we estimated context independent (CI) models with 10 mixture components per state: 8.2% error rate with the vocabulary of 270 commands. Then, we estimated context dependent (CD) models with 1400 states after the tree-based clustering, each state with 8 mixture components. The error rate was **2.7%**.

The error rate was clearly worse than in the Spanish system with a similar dictionary. The reason is probably that the speakers were in fact Spanish (non-native) and we observed that many pronunciations were quite different from the phoneme transcriptions we had used (native English). We included some alternative pronunciations in the dictionary trying to cover the different possibilities, but we could not get a performance similar to the Spanish system.

In this system, we did not do cross-task adaptation because we did not have a previous robust and general system trained for English. We did not do either speaker adaptation because error rates were low enough to fulfill the project specifications.

3.3 Spontaneous speech recognition

3.3.1 New system from scratch

We used the train set with 8 hours (see Table 1) to create HMM models from scratch. All adaptation results refer to the 503 test sentences with a vocabulary of 1,104 words. First, we created context independent (CI) models with 10 mixture components per state: 16.7% error rate. Then we created context dependent (CD) models with 1506 clustered states, each state with 8 mixture components. The error rate with that system was **12.70%**. We created two other systems with 1203 and 1803 states, but results were slightly lower for them.

3.3.2 Cross-task adaptation

Again, we used context-dependent HMM models trained with the SpeechDat database (43.2 hours). The optimum error rate obtained in that environment was 4.2% with a 3,065 words dictionary, using a total of 1,807 states in the HMMs and 7 mixture components per state.

Using those models without adaptation, the result is **19.51%** error rate, so they are even worse than CI models 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 the isolated database, we decided to do means and variances adaptation, as means only adaptation was worse in all cases. Using MAP, the error rate was **12.43%**. We can see the results for MLLR (% error rate) in Table 6.

Table 6. MLLR cross-task adaptation (Spont)

# nodes	1	2	3	4	5	6
64	16.05	15.28	14.94	14.87	**14.78**	**14.78**
128	15.40	14.69	14.42	14.11	14.17	**14.00**
255	14.42	13.71	13.40	13.20	13.09	**13.08**
507	13.61	12.94	12.81	12.66	**12.60**	12.64
1004	13.16	12.68	12.39	12.42	12.15	**12.11**
1956	13.03	12.32	12.35	**11.94**	12.08	12.09
3694	13.04	12.52	12.32	12.19	**12.05**	12.05

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 due to the big size of the adaptation set, but we can see that using bigger trees MLLR behaves much better than MAP (a 4% relative improvement using MLLR).
- There is enough data to train up to 2000-4000 transforms in MLLR.
- Both cross-task MAP and MLLR adaptation are better that beginning from scratch in this case. The reason for this improvement in cross-task adaptation is that the adaptation set is much smaller than the train set in SpeechDat, so that we can take advantage of some information from the original system. The improvement is remarkable, especially considering that there is a clear mismatch between tasks: Invoca is very spontaneous and SpeechDat is read speech.

We applied then MAP to the best MLLR models, and the results improved to **11.66%**. So, unlike the results obtained in (Cordoba, 2002a), where MLLR+MAP obtained similar results to just MAP, and our results for the isolated task (see the low improvement in section 3.1.2), in this task the improvement is remarkable over MLLR alone (2.4% relative improvement).

3.4 Spontaneous speech recognition - English

We used the train set with 5.7 hours (see Table 1) to create HMM models from scratch. All adaptation results refer to the Clearances task (453 test sentences) with a vocabulary of 793 words. As before, we created context independent (CI) models with 9 mixture components per state: we obtained 28.7% error rate.

Then we created context dependent (CD) models with 901 clustered states, each state with 8 mixture components. The error rate with that system was **22.2%**. We created another three systems using 599, 1205 and 1499 states, but the optimum was using only 901.

We can see that the results are clearly worse than in Spanish. We have found two reasons for that: first, the train set is almost half the size and is clearly too small, as the optimum was found for only 901 states; second, the controllers are non-native speakers and their pronunciation is quite Spanish, especially in airline, airport and city names, and even some greetings and goodbyes are in Spanish. In fact, first results were even worse, so we included alternative pronunciations with a remarkable improvement.

4 Conclusions

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 then 500 transforms, being the best solution using MLLR followed by MAP. All of them 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 outperforms MAP slightly; using more words, both techniques have similar results; the best result means a 79.5% relative improvement over no speaker adaptation with a negligible error rate.

For regression class tree creation in MLLR, it is better to use balanced trees and a symmetric likelihood distance.

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. All the options are better than beginning from scratch.

5 Acknowledgements

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

This work has been partially funded by the Spanish Ministry of Science and Technology under contracts DPI2001-3652-C02-02 (URBANO-IVANHOE), TIC2003-09192-C11-07 (MIDAS-INAUDITO), and DPI2004-07908-C02-02 (ROBINT).

6 References

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

Córdoba, R., Macías-Guarasa, J., Ferreiros, J., Montero, J.M., Pardo, J.M., "State Clustering Improvements for Continuous HMMs in a Spanish Large Vocabulary Recognition System", ICSLP 2002, pp. 677-680.

Córdoba, R., J. Ferreiros, J.M. Montero, F. Fernández, J. Macías-Guarasa, S. Díaz. "Cross-Task Adaptation and Speaker Adaptation in Air Traffic Control Tasks". III Jornadas en Tecnología del Habla, pp. 93-97. Noviembre 2004. ISBN: 84-9705-714-7.

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

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

INVOCA 2002. Project Synopses. Eurocontrol. Analysis of Research & Development in European Programmes. Available at http://www.eurocontrol.int/eatmp/ardep-arda/servlets/SVLT014?Proj=AEN043

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