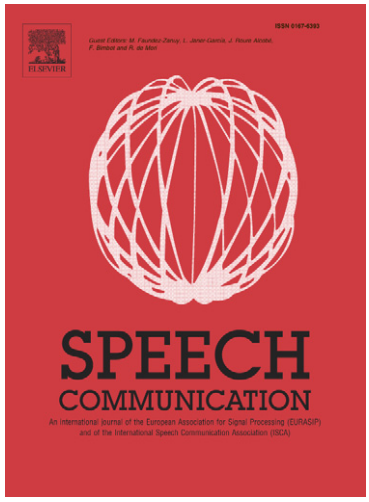
Linear Hidden Transformations for Adaptation of Hybrid ANN/HMM Models

Roberto Gemello, Franco Mana, Stefano Scanzio, Pietro Laface, Renato de Mori



Linear Hidden Transformations for Adaptation of Hybrid ANN/HMM Models

Roberto Gemello, Franco Mana, Stefano Scanzio, Pietro Laface, Renato De Mori

PII: S0167-6393(06)00178-6
DOI: 10.1016/j.specom.2006.11.005
Reference: SPECOM 1593

To appear in: *Speech Communication*

Received Date: 31 March 2006
Revised Date: 16 October 2006
Accepted Date: 29 November 2006

Please cite this article as: Gemello, R., Mana, F., Scanzio, S., Laface, P., De Mori, R., Linear Hidden Transformations for Adaptation of Hybrid ANN/HMM Models, *Speech Communication* (2006), doi: 10.1016/j.specom.2006.11.005

Linear Hidden Transformations for Adaptation of Hybrid ANN/HMM Models[1]

Roberto Gemello[1], Franco Mana[1], Stefano Scanzio[2], Pietro Laface[a2] and Renato De Mori[3]

[a] Corresponding Author
Pietro Laface
[2] *Politecnico di Torino*
Corso Duca degli Abruzzi, 24
10129 Torino Italy
Email: {Pietro.Laface, Stefano.Scanzio}@polito.it
Phone: +39 011 564-7004, Fax: +39 011 564-7099

Coauthors
[1] *LOQUENDO*
Via Val della Torre, 4 A
10149 Torino Italy
Email: {Roberto.Gemello, Franco Mana}@loquendo.com
Phone: +39 011 291-3458, Fax: +39 011 291…..

[3] *LIA - University of Avignon*
339, Chemin des Meinajaries
Agroparc BP 1228
84911 AVIGNON Cedex 9 France
Email: Renato.Demori@lia.univ-avignon.fr,
Phone: +33 49 0 84 3515, Fax: +33 49 0 84 3501

Abstract

This paper focuses on the adaptation of Automatic Speech Recognition systems using Hybrid models combining Artificial Neural Networks (ANN) with Hidden Markov Models (HMM).

Most adaptation techniques for ANNs reported in the literature consist in adding a linear transformation network connected to the input of the ANN. This paper describes the application of linear transformations not only to the input features, but also to the outputs of the internal layers. The motivation is that the outputs of an internal layer represent discriminative features of the input pattern suitable for the classification performed at the output of the ANN.

[1] This work was supported by the EU FP-6 IST Projects DIVINES and HIWIRE

In order to reduce the effect due to the lack of adaptation samples for some phonetic units we propose a new solution, called Conservative Training.
Supervised adaptation experiments with different corpora and for different types of adaptation are described. The results show that the proposed approach always outperforms the use of transformations in the feature space and yields even better results when combined with linear input transformations.

Keywords: Automatic Speech Recognition; Speaker Adaptation; Neural Network Adaptation; Catastrophic Forgetting

1 Introduction

The literature on adaptation of speaker, environment, and application is rich of techniques for refining Automatic Speech Recognition (ASR) systems by adapting the acoustic features and the parameters of stochastic models [1-5]. More recently, particular attention has been paid to discriminative training techniques and their application to the acoustic feature transformation [6,7].
Since discriminative methods are also used to train the acoustic-phonetic Artificial Neural Networks (ANN) models, it is worth exploring methods for adapting their features and model parameters. Several solutions to this problem have been proposed. Some of these techniques for adapting neural networks are compared in [8,9]. A classical approach consists in adding a linear transformation network (LIN) that acts as a pre-processor to the main network. Alternatively, it could be possible to simply adapt all the weights of the original network. A tied-posterior approach is proposed in [10] to combine Hidden Markov Models (HMM) with ANN adaptation strategies. The weights of a hybrid ANN/HMM system are adapted by optimizing the training set cross entropy. A sub-set of the hidden units is selected for this purpose. The adaptation data are propagated through the original ANN. The nodes that exhibit the highest variances are selected, since hidden nodes with a high variance transfer a larger amount of information to the output layer. Then, only the weights of the links coming out of the selected nodes are adapted.
Recent adaptation techniques have been proposed with the useful properties of not requiring to store the previously used training data, and to be effective even with a small amount of adaptation data. Methods based on speaker space adaptation [2] and eigenvoices [3] are of this type and can be applied both to Gaussian Mixture HMMs as well as to the ANN inputs as proposed in [11]. The parameters of the transformations are considered the components of a vector in a parameter adaptation space. The principal components of this space define a speaker space. Rapid adaptation consists in finding the values of the coordinates of a specific speaker point in the speaker space.
Another approach is the regularized adaptation proposed in [12], where the original weights of the networks, trained with unadapted data, are the a priori knowledge used to control the degree of adaptation, to avoid overfitting on adaptation data.
This paper explores a new possibility consisting in adapting ANN models with transformations of an entire set of internal model features. Values for these features are collected at the output of a hidden layer for which the number of outputs is usually of the

order of a few hundreds. These features are supposed to represent an internal structure of the input pattern. As for input feature transformation, a linear network can be used for hidden layer feature transformation. In both cases, the estimation of the parameters of the adaptation networks can be done with error Back-Propagation by keeping unchanged the values of the parameters of the ANN.

A problem, however, occurs in distributed connectionist learning when a network, trained with a large set of patterns, has to be adapted to classify input patterns that differ in some aspects from the ones used originally to train the network. A problem called "catastrophic forgetting" [13] arises when a network is adapted with new data that do not adequately represent the knowledge included in the original training data. This causes a particularly severe performance degradation. This happens when adaptation data do not contain examples for a subset of the output classes.
A review of several approaches that has been proposed to solve this problem is presented in [13]. One of them uses a set of pseudo-patterns, i.e. random patterns, associated to the output values produced by the connectionist network before adaptation. These pseudo-patterns are added to the set of the new patterns to be learned [14]. The attempt is to keep stable the classification boundaries related to classes that have few or no samples in the new set of patterns. This effectively decreases the catastrophic forgetting of the knowledge provided by originally learned patterns. Tests of this solution have been reported with small networks and low dimensional artificial input patterns. Unfortunately, they do not scale well because it is difficult to generate effective pseudo-patterns when the dimensionality of the input features is high.
For this reason, it has been proposed [15] to include examples of the missing classes, taken from the training set, in the adaptation set. However, the addition of a small subset of training examples related to the missing classes could redefine the class boundaries according to the distribution of these small subsets. This distribution would be different from the one of the complete training set. Moreover, this approach has a main practical problem: it is mandatory to store training set samples for the adaptation step. The number of samples should be large enough to provide a good preservation of the class boundaries. Finally, since the task independent network could be adapted to several applications, different sets of training patterns would be necessary to compensate classes missing in different adaptation sets.

This paper proposes a solution to this problem by introducing Conservative Training, a variation to the standard method of assigning the target values, which compensates for the lack of adaptation samples in some classes. The key idea of Conservative Training is that the probability of classes with no adaptation samples available should be replaced by the best available estimations of their real values. The only way to obtain these estimations is with the model provided by the original network.

Experimental results on the adaptation test for the Wall Street Journal task [16], using the proposed approaches, compare favorably with published results on the same task [10,16].

The paper is organized as follows: Section 2 gives a short overview of the acoustic-phonetic models of the ANN used by the ASR system, and presents the Linear Hidden Networks, which transform the features at the output of hidden layers. Section 3 is devoted to the

illustration of the problem of catastrophic forgetting in connectionist learning, and proposes our Conservative Training approach as a possible solution. Section 4 illustrates the benefits of Conservative Training using an artificial classification task of 16 classes. Section 5 reports the experiments performed on several databases with the aim of clarifying the behavior of the new adaptation techniques with respect to the classical LIN approach. Finally, the conclusions and future developments are presented in the last Section.

2 Feature transformations

The LOQUENDO-ASR decoder uses a hybrid combination of Hidden Markov Models (HMM) and a 4-layer Multi Layer Perceptron (MLP), where each phonetic unit is described in terms of a single or double state left-to-right automaton with self-loops. The HMM transition probabilities are uniform and fixed, and the emission probabilities are computed by a MLP [17]. The MLP has an input layer of 273 units (39 parameters of a 7 frame context), a first hidden layer of 315 units, a second hidden layer of 300 units and an output layer including a variable number of units, which is language dependent (600 to 1000). The advantage of using two hidden layers, rather than a larger single hidden layer, is that the total number of connections is reduced. Moreover, this architecture allows to consider the activation values of each hidden layer as a progressively refined projection of the input pattern in a space of features more suitable for classification.
The acoustic models are based on a set of vocabulary and gender independent units, including stationary context-independent phones and diphone-transition coarticulation models [17]. The models have been trained using large 8 kHz telephone speech databases, e.g. the SpeechDat corpora. The training set includes mainly phonetically balanced sentences, and some samples of application lists (e.g. digits, currency, yes-no). The weights of the ANN/HMM system are adapted by optimizing the training set cross entropy.
These models are the acoustic models of the 15 languages available with the LOQUENDO-ASR recognizer. They are used as seed models for the adaptation experiments of Section 5, unless differently specified.

2.1 Input feature transformations

The simplest and more popular approach to speaker adaptation with ANNs is Linear Input Transformation (LIN) [8,9]. A linear transformation rotates the input space to reduce the discrepancy between target and training conditions. A LIN, as shown in Figure 1, performs this transformation. The LIN weights are initialized with an identity matrix, and they are trained by minimizing the error at the output of the ANN system keeping fixed the weights of the original ANN.
Using few training data, the performance of the combined architecture LIN/ANN is usually better than the one obtained by adapting the weights of the whole network, because the adaptation involves a lower number of parameters.

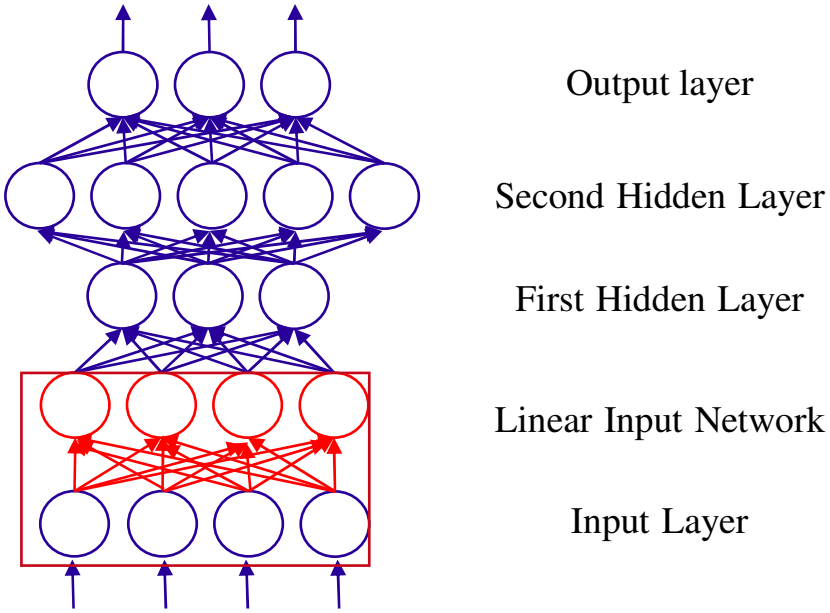


Fig. 1. Artificial Neural Network including a linear input layer

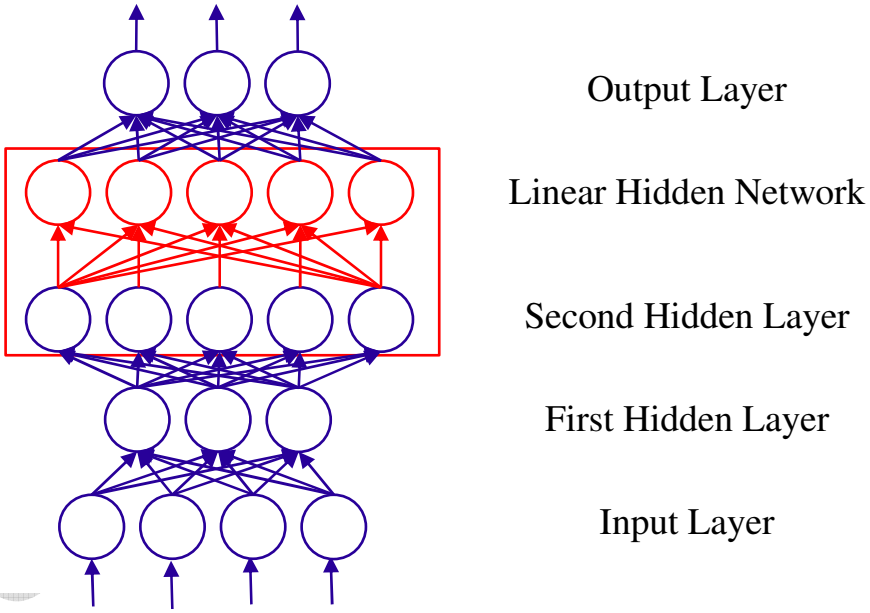


Fig. 2. Artificial Neural Network including a linear hidden layer

2.2 Hidden feature transformations

In a layered neural network, the activation values of each hidden layer are a progressively refined projection of the input pattern in a space of features more suitable for classification. The weights between the last hidden layer and the output layer perform a linear discrimination of the output classes. For this purpose, the weights of the lower layers of the network are trained to produce activations of the last hidden units that are linearly separable.
These activations can be considered as new features obtained by a non-linear discriminant analysis. Because of their properties, they are used - properly decorrelated - as observation features for Continuous Densities Gaussian Mixture models in the TANDEM approach [18].
Since the activation values of a hidden layer represent an internal structure of the input pattern in a space more suitable for classification, it is worth considering the adaptation of these features. A Linear Hidden Network (LHN) performs such an adaptation. Exactly as in the LIN case, the values of an identity matrix initialize the weights of the LHN. The weights are estimated using a standard Back-Propagation algorithm keeping frozen the weights of the original network. It is worth noting that, since the LHN performs a linear transformation, when the adaptation process is completed, the LHN can be removed. This can be done combining the LHN weights with the ones leading to the nodes of the next layer using the following simple matrix operations:

$$\begin{aligned} W_a &= W_{LHN} \times W_{SI} \\ B_a &= B_{SI} + B_{LHN} \times W_{SI} \end{aligned} \qquad (1)$$

where W_a and B_a are the weights and the biases of the adapted layer, W_{SI} and B_{SI} are the weights and biases of the layer above the LHN in the original Speaker Independent network, and W_{LHN} and B_{LHN} are the adapted weights and the biases of the linear hidden network.
In our experiments the LHN has been applied to the last hidden layer, but since the outputs of an internal layer can be considered as features more discriminative than the original ones, the LHN can be applied to whatever internal layer.

3 Catastrophic Forgetting

It is well known that in connectionist learning, acquiring new information in the adaptation process may cause a partial or total oblivion of the previously learned information [13,14]. This effect must be taken into account when adapting an ANN with a limited amount of data, i.e. when the probability of the absence of samples for some acoustic-phonetic units is high. The problem is more severe in the ANN modeling framework than in the classical Gaussian Mixture HMMs. The reason is that an ANN uses discriminative training to estimate the posterior probability of each acoustic-phonetic unit. The minimization of the output error is performed by means of the Back-Propagation algorithm that penalizes the units with no observations in the adaptation set by setting to zero the target value of the their output units for *every* adaptation frame. This target assignment policy reduces the ANN capability of correctly classifying the corresponding acoustic-phonetic units. On the contrary, the Gaussian Mixture models with little or no observations remain un-adapted, or share some adaptation transformations of their parameters with other similar acoustic models, maintaining the knowledge acquired before adaptation.

To mitigate the just introduced oblivion problem, it has been proposed [15] to include in the adaptation set examples of the missing classes taken from the training set. The disadvantage of this approach is that a substantial amount of the training set must be stored in order to have enough examples of the missing classes for each adaptation task. In [14], it has been proposed to approximate the real patterns with pseudo-patterns rather than using the training set. A pseudo-pattern is a pair of a random input activation and its corresponding output. These pseudo-patterns are included in the set of the new patterns to be learned to prevent catastrophic forgetting of the original patterns.
The proposed solutions have problems when applied to the adaptation of large ANNs. In fact, there are no criteria for selecting adaptation samples from the training data which are often not available when adaptation is performed. Moreover, the selected data should share some characteristics that make the adaptation environment different from the training one, but the elements of such a difference are often unknown.
Furthermore, it is unclear how effective pseudo-patterns can be generated when the dimensionality of the input features is high.

A solution, called Conservative Training (CT), is now proposed to mitigate the forgetting problem.
Since the Back-Propagation technique used for MLP training is discriminative, the units for which no observations are available in the adaptation set will have zero as a target value for all the adaptation samples. Thus, during adaptation, the weights of the MLP will be biased to favor the output activations of the units with samples in the adaptation set and to weaken the other units, which will always have a posterior probability getting closer to zero.
Conservative Training does not set to zero the value of the targets of the missing units; it uses instead the outputs computed by the original network as target values.
Regularization as proposed in [12] is another solution to the forgetting problem. Regularization has theoretical justifications and affects all the ANN outputs by constraining the network weight variations. Unfortunately, regularization does not directly address the problem of classes that do not appear in the adaptation set. We tested the regularization approach in a preliminary set of experiments, obtaining minor improvements. Furthermore, we found difficult to tune a single regularization parameter that could perform the adaptation avoiding catastrophic forgetting.
Conservative Training, on the contrary, takes explicitly all the output units into account, by providing target values that are estimated by the original ANN model using samples of units available in the adaptation set.

Let F_p be the set of phonetic units included in the adaptation set (p indicates presence), and let F_m be the set of the missing units. In Conservative Training the target values are assigned as follows:

$$
\begin{aligned}
&T(f_i \in F_m \mid O_t) = OUTPUT\_ORIGINAL\_NN(f_i \mid O_t) \\
&T(f_i \in F_p \mid O_t \quad \& \quad correct(f_i \mid O_t)) = \\
&\qquad (1.0 - \sum_{j \in F_m} OUTPUT\_ORIGINAL\_NN(f_j \mid O_t) \\
&T(f_i \in F_p \mid O_t \quad \& \quad !correct(f_i \mid O_t)) = 0.0
\end{aligned}
\tag{2}
$$

where $T(f_i \in F_p \mid O_t)$ is the target value associated to the input pattern O_t for a unit f_i that is present in the adaptation set. $T(f_i \in F_m \mid O_t)$ is a target value associated to the input pattern O_t for a unit not present in the adaptation set,. $OUTPUT\_ORIGINAL\_NN(f_i \mid O_t)$ is the output of the original network (before adaptation) for the phonetic unit *i* given the input pattern O_t, and $correct(f_i \mid O_t)$ is a predicate which is true if the phonetic unit f_i is the correct class for the input pattern O_t.

Thus, a phonetic unit that is missing in the adaptation set will keep the value that it would have had with the original un-adapted network, rather than obtaining a zero target value for each input pattern.

This policy, like many other target assignment policies, is not optimal. Nevertheless, it has the advantage of being applicable in practice to large and very large vocabulary ASR systems using information from the adaptation environment, and avoiding the destruction of the class boundaries of missing classes.

It is worth noting that in badly mismatched training and adaptation conditions, for example in some environmental adaptation tasks, acoustically mismatched adaptation samples may produce unpredictable activations in the target network. This is a real problem for all adaptation approaches: if the adaptation data are scarce and they have largely different characteristics – SNR, channel, speaker age, etc. – other normalization techniques have to be used for transforming the input patterns to a domain similar to the original acoustic space.

Although different strategies of target assignment can be devised, the experiments reported in the next sections have been performed using only this approach. Possible variations, within the same framework, include the fuzzy definition of missing classes and the interpolation of the original network output with the standard 0/1 targets.

4 Experimental results on artificial data

An artificial two-dimensional classification task has been used to investigate the effectiveness of the Conservative Training technique. The examples have been designed to illustrate the basic dynamics of the class boundaries. They reproduce the problems due to missing classes in the adaptation set, emphasizing them.

An MLP has been used to classify points belonging to 16 classes having the rectangular shapes shown by the green borders in Figure 3. The MLP has 2 input units, two 20 node hidden layers, and 16 output nodes. It has been trained using 2500 uniformly distributed patterns for each class.

Figure 3 shows the classification behavior of the MLP after training based on Back-Propagation. In particular, a dot has been plotted only if the score of the corresponding class

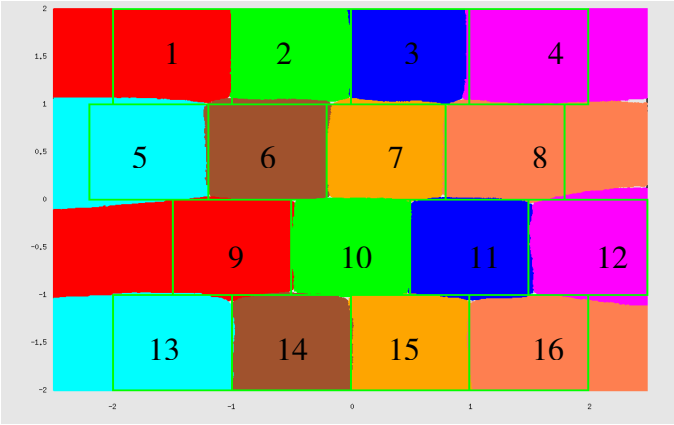


Fig. 3. Training 16 classes on a 4-layer network with 760 weights

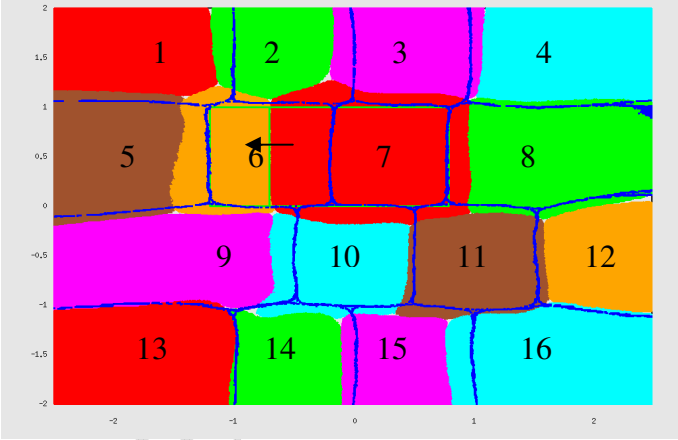


Fig. 4. Adaptation of all the network weights. The adaptation set includes examples of class 6 and class 7 only.

Adaptation method	*Forgetting mitigation technique*	*Average classification rate (%)*	*Class 6 classification rate (%)*	*Class 7 classification rate (%)*
1.None	None	95.9	98.5	93.3
2.Whole network	None	83.1	100.0	98
3.Whole network	CT	89.8	97.8	94.8
4.LIN	None	42.6	100	95.7
5.LIN	CT	69.0	99.0	91.8
6.LHN	None	65.4	99.6	97.2
7.LHN	CT	86.7	98.0	93.3

Table 1
Correct classification rates on the artificial data task.

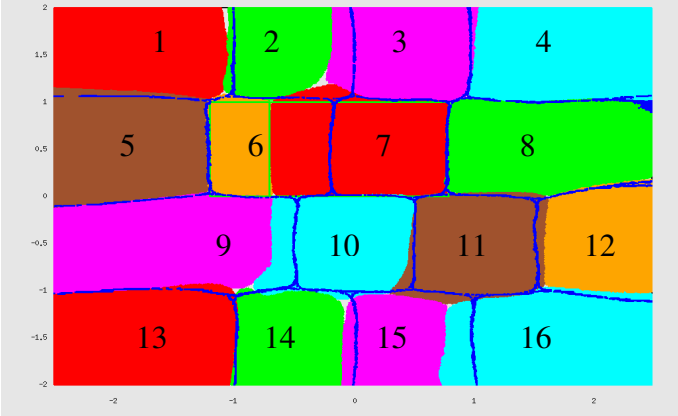


Fig. 5. Conservative Training adaptation of all the network weights.

was greater than 0.5. MLP outputs have also been plotted for test points belonging to regions that have not been trained, and outside the green rectangles: they are at the left and right sides of Figure 3. The average classification rate for all classes, and the classification rate for classes 6 and 7, is reported in the first row of Table 1.
Afterward, an adaptation set was defined to simulate an adaptation condition where only two of the 16 classes appear. The 5000 points in this set define a border between classes 6 and 7 shifted toward the left, as shown in Figure 4. In the first adaptation experiment, all the 760 MLP weights and 56 biases of the network were adapted. The catastrophic forgetting behavior of the adapted network is evident in Figure 4, where a blue grid has been superimposed to indicate the original class boundaries learned by full training.
Classes 6 and 7 do actually show a relevant increase of their correct classification rate, but they have a tendency to invade the neighbor classes. Moreover, a marked shift toward the left affects the classification regions of all classes, even the ones that are distant from the adapted classes. This undesired shift of the boundary surfaces induced by the adaptation process damages the overall average classification rate, as shown in the second row of Table 1.

To mitigate the catastrophic forgetting problem, the adaptation of the network has been performed using Conservative Training. Figure 5 shows how the trend of classes 6 and 7 to invade neighbor classes is largely reduced, Class 6 and 7 fit well their true classification regions, and although the left shift syndrome is still present, the adapted network performs better as shown by the average classification rate in the third row of Table 1.

Our artificial test-bed is not well suited to LIN adaptation because the classes cover rectangular regions: thus, a linear transformation matrix that is able to perform a single global rotation of the input features is ineffective. Moreover, the degree of freedom of this LIN is really poor: the LIN includes 4 weights and 2 biases only. These considerations are confirmed by the results reported in line 4 of Table 1. Classes 6 and 7 are well classified, but the average classification is very bad because the adaptation of the LIN weights to fit the

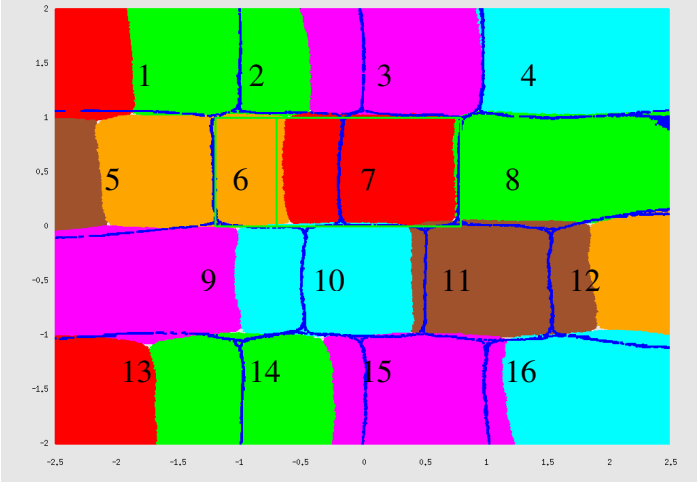


Fig. 6. Conservative Training LIN adaptation.

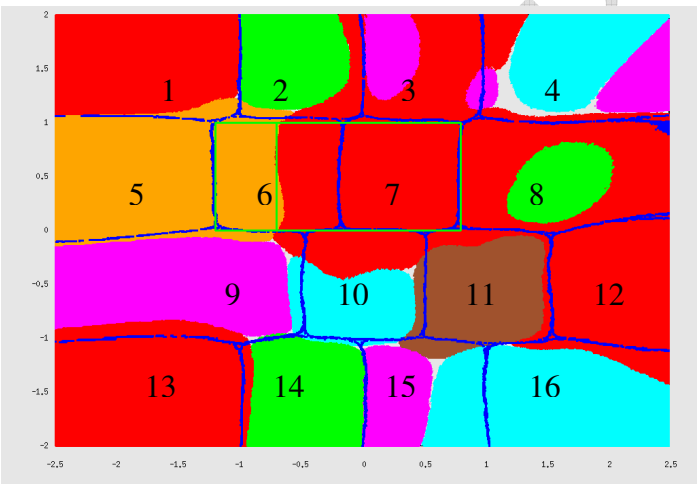


Fig. 7. LHN Adaptation.

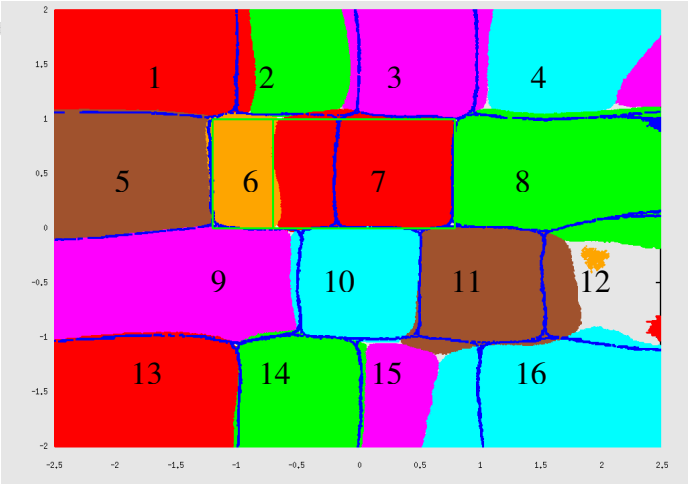


Fig. 8 Conservative Training LHN adaptation.

boundary between class 6 and 7, has the catastrophic forgetting effect of enlarging the regions of all classes.
The mitigation of these effects introduced by Conservative Training is shown in Figure 6, and in line 5 of Table 1. The shift toward left syndrome is still visible, but the horizontal boundary surfaces are correct.
If we add, instead, a LHN between last hidden layer and the output layer, and we adapt its 420 weights plus biases only, we obtain better results than LIN adaptation (see line 6 of Table 1). However, as Figure 7 shows, the class separation surfaces are ugly. Class 6, and especially class 7 are spread out, class 3 is split, and thus the average classification rate is unacceptable.
Conservative Training does again a very good job, as shown in Figure 8 and in last line of Table 1, even if class 12 does not present high scores.

5 Experimental results on speech recognition tasks

Adaptation to a specific application may involve the speakers, the channel, the environmental noise and the vocabulary, especially if the application uses specific lists of terms. The proposed techniques have been tested on a variety of cases requiring different types of adaptation. The adaptation tasks that have been considered are listed in sub-section 5.1 below. The LOQUENDO default speaker and task independent Italian models, described in Section 2, were the seed models for the adaptation.

The results of our experiments show that the problem of forgetting is dramatic especially when the adaptation set is not characterized by a good coverage of the phonemes of the language. The use of Conservative Training mitigates the forgetting problem, allowing adaptation with a limited performance decrease of the model on other tasks (some performance reductions are inevitable because the ANN is adapted to a specific condition and thus it is less general).

5.1 Tests on various adaptation tasks

Application adaptation: Directory Assistance

We tested the performance of models adapted to a Directory Assistance application. The corpus includes spontaneous utterances of the 9325 Italian city names. The adaptation set has 53713 utterances; the test set includes 3917 utterances.

Vocabulary adaptation: Command words

The lists A1-2-3 of the SpeechDat-2 Italian corpus, containing 30 command words, have been used. The adaptation and the test sets include 6189 and 3094 utterances respectively.

Channel-Environment adaptation: Aurora-3

The benchmark is the standard Aurora3 Italian corpus. The Well-Matched train set has been used for adaptation (2951 utterances), while the results on the Well-Matched test set (the noisy channel, ch1) are reported (654 utterances).

Adaptation task / Adaptation method	Application *Directory Assistance*	Vocabulary *Command Words*	Channel-Environment *Aurora3 Ch 1*
No adaptation	14.6	3.8	24.0
Whole network	10.5	3.2	10.7
LIN	11.2	3.4	11.0
LIN + CT	12.4	3.4	15.3
LHN	9.6	2.1	9.8
LHN + CT	10.1	2.3	10.4

Table 2
Adaptation results (WER %) on different tasks using various adaptation methods. The seed adaptation models are the standard LOQUENDO telephone models. The model are adapted on a given task and tested on sentences of the *same domain.*

Adaptation task / Adaptation method	*Directory Assistance Adapted Models*	*Command Words Adapted Models*	*Aurora3 Ch1 Adapted Models*
Whole network	36.3	*63.9*	*126.6*
LIN	36.3	42.7	*108.6*
LIN + CT	36.5	35.2	42.1
LHN	40.6	*63.7*	*152.1*
LHN + CT	40.7	45.3	44.2
No adaptation	29.3		

Table 3
Evaluation of the forgetting problem: recognition results (WER%) on Italian continuous speech with models adapted on *different* tasks.

The results on these tests, reported in Table 2, show that a linear transform on hidden units (LHN) always outperforms a linear transform on the input space (LIN). This indicates that the hidden units represent a projection of the input pattern in a space where it is easier to learn or adapt the classification expected at the output of the MLP. The *whole network* row in the table corresponds to the adaptation of all the ANN weights by incremental training of the original network. This adaptation is feasible only if many adaptation data are available, and it is less effective than LHN.
Conservative training imposes some constraints to the adaptation process because it tries to prevent forgetting missing classes. Thus, the performance of the LIN/LHN models, adapted on a given task and tested on sentences of the same domain, are slightly better than the performance of the corresponding models adapted by means of CT in addition to LIN/LHN. This happens because, using a large adaptation set, there are enough samples for most of the

outputs. Thus, there is little or no risk of catastrophic forgetting, and there is no need for the estimates obtained when input samples of different units are applied at the network input.

Conservative training, however, preserves the generality of the adapted model. This claim has been assessed by using a model adapted on a task and evaluating its recognition performance on a *different* generic common task. The generic task is large vocabulary (9.4k words) continuous speech recognition of 4296 phonetically balanced sentences uttered by 966 speakers. The average sentence length is 6 words. The results are obtained without language modeling. The adaptation tasks and the reference word error rate (29.3 %) achieved using un-adapted acoustic models on the generic task are given in the first and last row of Table 3 respectively.
Since the adapted models have been specialized to a specific condition, they are not expected to perform as well as the original model – task independent - on a continuous speech task. It was interesting, however, to have a measure of the generalization loss after adapting the standard model with different approaches.
Table 3 highlights the effects of catastrophic forgetting, which takes place when the vocabulary of the adaptation set is small and has a poor phonetic coverage. This is particularly evident for the models adapted on the *Command words* and *Aurora 3* tasks whose results on the generic continuous speech recognition task are emphasized in italics.
If the adaptation set has enough data for each class appearing in the test set of the new task, then LIN or LHN approach perform well without CT. The results reported in the first column of Table 3 confirm that the use of CT is not relevant when the model has been obtained using the phonetically rich adaptation set provided by the utterances collected for a Directory Assistance task. On the other hand, if the *test* set includes phonetic classes that are missing in the adaptation set, the use of CT in addition to LIN/LHN produces better results avoiding the catastrophic forgetting. This is shown in the second and third columns of Table 3, where the test set is still large vocabulary continuous speech, but the models have been adapted using command words and digits respectively.
Conservative Training, thus, mitigates the forgetting problem, preserving an acceptable performance of the adapted model on the task for which the original network was trained (open vocabulary, task independent speech recognition). The phonetic classes that were rarely represented, or were missing, in the adaptation set can be still reasonably well recognized by the adapted model.

5.2 Speaker Adaptation

Further experiments have been performed on the WSJ0 speaker adaptation test in several conditions. Three baseline models have been used:

- the default LOQUENDO 8 kHz telephone speech model (trained with LDC MACROPHONE [19] – referred as MCRP in the Tables);
- a model trained with the WSJ0 train set (SI-84), 16 kHz.
- a model trained with the WSJ0 train set (SI-84), down-sampled to 8 kHz.

Furthermore, we tested two architectures for each type of models: the standard one (STD), described in sub-section 2.1 and an improved (IMP) architecture, characterized by a wider input window modeling a time context of 250 ms [20], and by the presence a third 300 units hidden layer.

Train Set	*Net type*	*Adaptation method*	*Bigram LM*	*Trigram LM*
MCRP	STD	NO adaptation	16.4	13.6
		Standard LIN	14.6	11.6
		LIN+CT	13.9	11.3
		LHN+CT	12.1	9.9
		LIN+LHN+CT	11.2	9.0
WSJ0	STD	NO adaptation	13.4	10.8
		Standard LIN	14.2	11.6
		LIN+CT	11.8	9.7
		LHN+CT	10.4	8.3
		LIN+LHN+CT	9.7	7.9
WSJ0	IMP	NO adaptation	10.8	8.8
		Standard LIN	9.8	7.6
		LIN + CT	9.8	7.7
		LHN + CT	8.5	6.6
		LIN+LHN+CT	8.3	6.3

Table 4
Speaker Adaptation word error rate on the WSJ0 8 kHz tests using different adaptation approaches.

Train Set	*Net type*	*Adaptation method*	*Bigram LM*	*Trigram LM*
WSJ0	STD	NO adaptation	10.5	8.4
		Standard LIN	9.9	7.9
		LIN+CT	9.4	7.1
		LHN+CT	8.4	6.6
		LIN+LHN+CT	8.6	6.3
	IMP	NO adaptation	8.5	6.5
		Standard LIN	7.2	5.6
		LIN+CT	7.1	5.7
		LHN+CT	7.0	5.6
		LIN+LHN+CT	6.5	5.0

Table 5
Speaker Adaptation word error rate on the WSJ0 16 kHz tests using different adaptation approaches.

The adaptation set is the standard adaptation set of WSJ0 (`si_et_ad`, 8 speakers, 40 utterances per speaker), down-sampled to 8 kHz when necessary.
The test set is the standard SI 5K read NVP Senneheiser microphone (`si_et_05`, 8 speakers x ~40 utterances). The bigram or trigram standard Language Models provided by Lincoln Labs have been used in these experiments.

The results, reported in Tables 4 and 5, show that also in these tests LHN always achieves better performance that LIN. The combination of LIN and LHN (trained simultaneously) is usually better than the use of LHN alone.

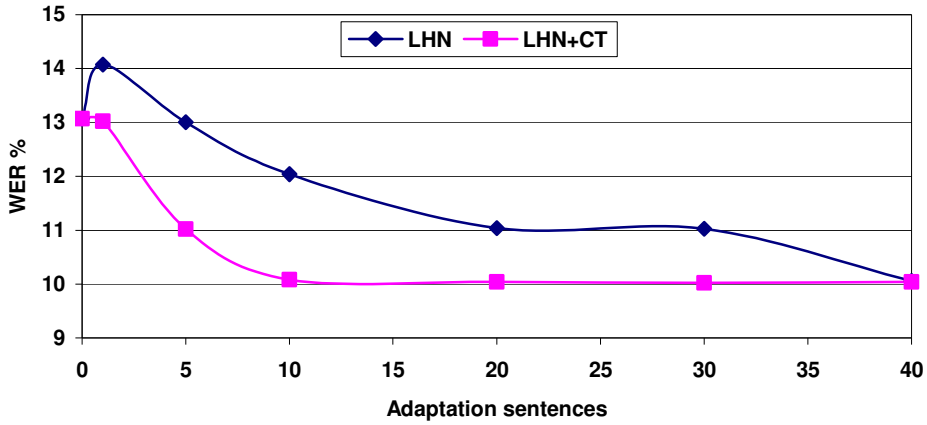


Figure 9 Word error rate on the WSJ0 8 kHz tests as a function of the amount of adaptation data (1, 5, 10, 20, 30, and 40 sentences).

Conservative training effects are of minor importance in these tests because the adaptation set has a good phonetic coverage and the problem of unseen phonetic classes is not dramatic. Nevertheless, its use improves the performance (compare Standard LIN and LIN+CT), because it avoids the adaptation of prior probabilities of the phonetic classes on the (poor) prior statistics of the adaptation set.
Finally, Figure 9 shows how CT influences the performance of models adapted with a different amount of adaptation data. The results, obtained with a bigram language model, refer to LHN adaptation of the standard models (STD) using up to 40 sentences of each speaker. The sentences are down-sampled to 8 kHz. The first point in the graph shows the 13.4% WER achieved using the standard model without adaptation.
Conservative training is particularly useful when the adaptation set is very small (a few sentences), because in that case the problem of the missing phonetic classes is more relevant. In particular, using a single sentence the error rate for LHN adaptation alone actually increases.

The experiments show an overall benefit of CT, considering that the minimal performance degradation reported for the improved network models adapted with LIN has been obtained using the trigram LM, which could mask the acoustic quality of the models.

6 Conclusions

A method has been proposed for adapting all the outputs of the hidden layer of ANN acoustic models and for reducing the effects of catastrophic forgetting when the adaptation set does not contain examples for some classes. Experiments have been performed for the adaptation of an existing ANN to a new application, a new vocabulary, a new noisy environment and

new speakers. They show the benefits of CT, and that LHN outperforms LIN. Furthermore, experiments on speaker adaptation show that further improvements are obtained by the simultaneous use of LHN and LIN showing that linear transformations at different levels produce different positive effects that can be effectively combined.

An overall WER of 5% after adaptation on WSJ0 using the standard trigram LM and without across word specific acoustic models compares favorably with published results.

References

[1] Gauvain, J. L., Lee, C. H., 1994. Maximum a posteriori estimation for multivariate gaussian mixture observations of Markov chain. IEEE Transactions on Speech and Audio Processing, Vol. 2, n. 2, pp. 291-298.

[2] Gales, M.J.F., 1998. Maximum likelihood linear transformations for HMM-based speech recognition. Computer Speech and Language, Vol. 12, pp. 75-98.

[3] Kuhn, R., Junqua, J.-C, Nguyen, P., Niedzielski N., 2000. Rapid speaker adaptation in eigenvoice space", IEEE Transactions on Speech and Audio Processing, Vol. 8, no. 4, pp. 695–707.

[4] Sagayama, S., Shinoda, K., Nakai, M., Shimodaira, H., 2001. Analytic methods for acoustic model adaptation: A review. Proc. Adaptation Methods for Speech Recognition, ISCA ITR-Workshop, pp. 67–76.

[5] Lee, C.-H., Huo, Q., 2000. On adaptive decision rules and decision parameter adaptation for automatic speech recognition. Proc. IEEE, vol. 88, no. 8, pp. 1241–1269.

[6] Hsiao, R., Mak, B., 2004. Discriminative feature transformation by guided discriminative training. Proc. ICASSP-04, Montreal, pp. 897-900.

[7] Liu, X., Gales, M.J.F., 2004. "Model complexity control and compression using discriminative growth functions. Proc. ICASSP-04, pp. 797-800.

[8] Abrash, V., Franco, H., Sankar, A., Cohen, M., 1995. Connectionist speaker normalization and adaptation. Proc. EUROSPEECH 1995, pp. 2183–2186.

[9] Neto, J., Almeida, L., Hochberg, M., Martins, C., Nunes, L., Renals, S., Robinson, T., 1995. Speaker-adaptation for hybrid HMM-ANN continuous speech recognition system. Proc. EUROSPEECH 1995, pp. 2171–2174, 1995.

[10] Stadermann, J., Rigoll, G., 2005. Two-stage speaker adaptation of hybrid tied-posterior acoustic models. Proc. ICASSP-05, Philadelphia, pp. I-997-1000.

[11] Dupont, S., Cheboub. L., 2000. Fast speaker adaptation of artificial neural networks for automatic speech recognition. Proc. ICASSP-00, pp. 1795-1798.

[12] Li, X., Bilmes, J., 2006. Regularized adaptation of discriminative classifiers. Proc. ICASSP-06, pp. 237-240.

[13] French, M., 1994. Catastrophic forgetting in connectionist networks: causes, consequences and solutions. Trends in Cognitive Sciences, 3(4), pp. 128-135.

[14] Robins, A., 1995. Catastrophic forgetting, rehearsal, and pseudo-rehearsal. Connection Science, 7, 123 – 146.

[15] BenZeghiba, M.F., Bourlard, H. 2003. Hybrid HMM/ANN and GMM combination for user-customized password speaker verification**.** Proc. ICASSP-03, pp. 225-228.

[16] Pallett, D. S., Fiscus, J. G., Fisher, W. M., Garofolo, J. S., Lund, B. A., Przybocki, M. A., 1994. 1993 Benchmark tests for the ARPA spoken language program. Proc. of the Human Language Technology Workshop, pp. 49–74.

[17] Albesano, D., Gemello, R., Mana, F., 1997. Hybrid HMM-NN modeling of stationary-transitional units for continuous speech recognition. Proc. Neural Information Processing, pp. 1112–1115.

[18] Hermansky, H., Ellis, D., Sharma, S., 2000. Tandem connectionist feature extraction for conventional HMM systems. Proc. ICASSP-00, pp.1635-1638.

[19] Available at http://www.ldc.upenn.edu

[20] Dupont, S., Ris, C., Couvreur L., Boite, J. M., 2005. A study of implicit and explicit modeling of coarticulation and pronunciation variation. Proc. Interspeech-05, pp. 1353-1356.