Adaptation of Artificial Neural Networks Avoiding Catastrophic Forgetting

Dario Albesano[1], Roberto Gemello[1], Pietro Laface[2], Franco Mana[1], Stefano Scanzio[2]

***Abstract*— In connectionist learning, one relevant problem is "catastrophic forgetting" that may occur when a network, trained with a large set of patterns, has to learn new input patterns, or has to be adapted to a different environment. The risk of catastrophic forgetting is particularly high when a network is adapted with new data that do not adequately represent the knowledge included in the original training data.**

Two original solutions are proposed to reduce the risk that the network focuses on new data only, loosing its generalization capability. The first one, Conservative Training, is a variant to the target assignment policy, while the second approach, Support Vector Rehearsal, selects from the training set the patterns that lay near the borders of the classes not included in the adaptation set. These patterns are used as sentinels that try to keep unchanged the original boundaries of these classes.

Moreover, we investigated the extension of the classical approach consisting in applying 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 a projection of the input pattern into a space where it should be easier to learn the classification or transformation expected at the output of the network.

We illustrate the problems using an artificial test-bed, and apply our techniques to a set of adaptation tasks in the domain of Automatic Speech Recognition (ASR) based on Artificial Neural Networks. Supervised ASR adaptation experiments with several corpora and for different adaptation types are described. We report on the adaptation potential of different techniques, and on the generalization capability of the adapted networks. The results show that the combination of the proposed approaches mitigates the catastrophic forgetting effects, and always outperforms the use of the classical transformations in the feature space.

I. Introduction

Connectionist learning, in particular error back-propagation in Multi Layer Perceptron networks, is based on the repeated presentation of all the patterns that must be learned until the network weights converge to stable values. Once the network has been trained, it is able to generalize, i.e. to correctly classify input patterns that present similar characteristics. A problem, however, occurs when a network, trained with a large set of patterns, has to learn new input patterns. This problem, called "catastrophic forgetting," [1] is particularly high when a connectionist network is adapted with new data that do not adequately represent the knowledge included in the original training data. This effect is evident when adaptation data do not contain examples for a subset of the output classes. If the outputs of the missing classes are forced to a value close to zero for all the adaptation samples, there is a risk that the network becomes less sensitive to input data belonging to these classes.

A review of several approaches that has been proposed to solve this problem is presented in [1]. Among them, the use of a series 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 [4] to try keeping stable the classification boundaries related to classes that have few samples or are even missing in the new set of patterns. This effectively decreases catastrophic forgetting of the originally learned patterns. Since it seems difficult to generate these pseudo-patterns when the dimensionality of the input features is high, it has been proposed [5] to include in the adaptation set examples of the missing classes taken from the training set.

This paper proposes, instead, two solutions to mitigate these problems introducing Conservative Training, a variant to the standard method of assigning the target values, which compensates for the lack of adaptation patterns in some classes, and Support Vector Rehearsal, which selects from the training set the patterns that lay near the borders of the classes not included in the adaptation set. These patterns are used as sentinels that try to keep unchanged the original boundaries of these classes.

We illustrate the problems, and our solutions, using an artificial two-dimensional classification task where 16 classes must be discriminated. Our main interest, however, is devoted to solutions that scale up to large training sets and networks. Thus, we applied our approach on several adaptation tasks in the domain of Automatic Speech Recognition (ASR) based on Artificial Neural Networks (ANNs), where adaptation is important because the recognition accuracy of speaker-independent recognition systems is sensible to speaker variability. In these systems, significant performance degradations are experienced with outlier or non-native speakers. Environment, channel, and microphone variability is another important source of errors.

The literature on speakers, environments, and applications adaptation is rich of techniques for refining ASR systems by

D. Albesano, R. Gemello, and F. Mana are with Loquendo SpA, Italy, (phone: +39 011 2913421 fax: +39 011 2913199; e-mail: {Dario.Albesano,Roberto.Gemello,Franco.Mana}@loquendo.com)

P. Laface, and S. Scanzio are with the Departimento di Automatica e Informatica, Politecnico di Torino, Italy. (e-mail: {Pietro.Laface, Stafano.Scanzio}@polito.it).

This work was supported by the EC FP-6 IST Project DIVINES – Diagnostic and Intrinsic Variabilities in Natural Speech, and by the EC FP-6 IST Project HIWIRE – Human Input that Works in Real Environments

adapting the acoustic features and the parameters of stochastic models [6]-[11]. In comparison, far less proposals have been made for the adaptation of the features and model parameters of systems that use the hybrid Hidden Markov Model-ANN approach. Some of these techniques for adapting neural networks are compared in [12][13]. A classical approach consists in adding a linear transformation network that acts as a pre-processor to the main network, or adapting all the weights of the original network.

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.

We show that the Conservative Training approach is particularly important in this framework to mitigate the forgetting effects due to the lack of adaptation samples in some classes. Experimental results on the adaptation test for the Wall Street Journal corpus [14] using the proposed approach compare favorably with published results on the same task [14][15].

The paper is organized as follows: Section II gives a short overview of the acoustic-phonetic models of the ANN used by the Loquendo ASR system, and presents the Linear Hidden Networks, which transform the features at the output of hidden layers. Section III is devoted to the illustration of the problem of catastrophic forgetting in connectionist learning, and proposes our Conservative Training and Support Vector Rehearsal approaches as possible solutions. The results of these approaches are given in Section IV, where an artificial classification task of 16 classes is addressed. Section V reports on the experiments performed on several speech recognition tasks with the aim of clarifying the behavior of the proposed adaptation techniques that mitigates the effects of forgetting. Finally the conclusions are given in the last Section.

II. NEURAL NETWORK ADAPTATION

As mentioned in the introduction, our interest is devoted to solutions that scale up to large training sets and networks. Thus, we present in this Section the architecture of a large vocabulary, task independent, Automatic Speech Recognition system based on ANNs. This system has been used to produce the baseline results, and to benchmark our techniques.

A. The ANN Architecture

The Loquendo-ASR system uses acoustic models based on a hybrid combination of Hidden Markov Models (HMM) and 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 an MLP [16]. This MLP has an input layer of 273 units, a first hidden layer of 315 units, a second hidden layer of 315 units and an output layer including a variable number of units that is language dependent (600 to 1000). Using two hidden layers, rather than a larger single hidden layer, has the advantage of reducing the total number of connections. Moreover, it allows considering 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 [16]. These acoustic models have been successfully used for the 15 languages released with the Loquendo ASR recognizer, and are the seed models for the adaptation experiments of Section V, if not differently specified.

B. Input Feature Transformations

The simplest and more popular approach to speaker adaptation with ANNs is Linear Input Transformation [12][13][17]. The input space is rotated – and shifted – by a linear transformation to make the target conditions more consistent with the training conditions. The transformation is performed by a linear layer interface (referred to, in this paper, as linear input network or LIN) introduced between the input observation vectors and the input layer of the trained ANN. The LIN weights are initialized with an identity matrix, and they are trained by minimizing the error at the output of the ANN keeping fixed the weights of the original ANN.

Using few training data, the performance of the combined architecture LIN/ANN is usually better than adapting the whole network, because it involves the estimation of fewer parameters.

C. Hidden feature transformations

Assuming that the activation values of a hidden layer represent an internal structure of the input pattern in a space more suitable for classification, a linear transformation can be applied to the activations of the internal layers. Such a transformation is performed by a Linear Hidden Network (LHN). As for the LIN, the values of an identity matrix are used to initialize the weights of the LHN. The weights are trained 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, once the adaptation process is completed, the LHN can be removed combining LHN weights with the ones 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 $\boldsymbol{W_a}$ and $\boldsymbol{B_a}$ are the weights and the biases of the adapted layer, $\boldsymbol{W_{SI}}$ and $\boldsymbol{B_{SI}}$ are the weights and biases of the layer following the LHN in the original Speaker Independent network, and $\boldsymbol{W}_{LHN}$ and $\boldsymbol{B}_{LHN}$ are the adapted weights and the biases of the linear hidden network.

III. Catastrophic Forgetting

It is well known that in connectionist learning, acquiring new information in the adaptation process can damage previously learned information [1]-[4]. This effect must be taken into account when adapting an ANN with limited amount of data, which do not include enough samples for all the classes. The problem is more severe in the ANN modeling framework than in a classical Gaussian Mixture classifier. The reason is that an ANN uses discriminative training to estimate the posterior probability of each class. The minimization of the output error is performed by means of the Back-Propagation algorithm that penalizes the units with no observations by assigning to them a zero target value for every adaptation frame. That induces in the ANN a forgetting of the capability to classify the corresponding classes. Thus, while the Gaussian Mixture models with little or no observations remain un-adapted or share some adaptation transformations of their parameters with other acoustic similar models, the units with little or no observations in the ANN model loose their characterization rather than staying not adapted. Thus, adaptation may destroy the correct behavior of the network for the unseen units.

To mitigate the problem of loosing characterization of the units with little of no observations, it has been proposed [5] 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 so that examples of the missing classes can be retrieved for each adaptation task. In [4], it has been proposed to approximate the real patterns with pseudo-patterns rather than using the training set. Pseudo-patterns consist of pairs of random input activations and the 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. It seems difficult, however, to generate these pseudo-patterns when the dimensionality of the input space is high.

Here we propose two approaches to mitigate the catastrophic forgetting problem:

– We refer to the first one as Conservative Training (CT). It is a variant to the target assignment policy that sets the output computed by the original network as target value for the classes not represented in the adaptation set, rather than assigning to them the target value zero, as usual.

– The second one is referred to as Support Vector Rehearsal, a name borrowed by the analogy with Support Vector Machines. It is an original solution with respect to the pseudo-patterns [4], or the missing class patterns [5] approaches. Rather than selecting samples supposed to cover the feature space of the missing classes, it selects from the training set the Support Vectors: patterns that lay near the borders of the classes not included in the adaptation set. The Support Vectors are used as sentinels that try to keep unchanged the original boundaries of these classes.

A. Conservative Training

Since ANN training is discriminative, the units for which no observations are available will have zero as a target for all the adaptation samples. Thus, during adaptation, the weights of the acoustic ANN 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 tend to always have a posterior probability close to zero.

Conservative Training avoids associating the value zero to the target of the missing units, using instead as target the outputs computed by the original network.

Let C_p be the set of classes included in the adaptation set (p indicates presence), and and let C_m be the complement set of the missing classes. In Conservative Training the target values are assigned as follows:

$$\begin{aligned}
&T(c_i \in C_m \mid X_t) = OUTPUT\_ORIGINAL\_NN(c_i \mid X_t) \\
&T(c_i \in C_p \mid X_t \quad \& \quad correct(c_i \mid X_t)) = \\
&\qquad (1.0 - \textstyle\sum_{j \in C_m} OUTPUT\_ORIGINAL\_NN(c_j \mid X_t) \\
&T(c_i \in C_p \mid X_t \quad \& \quad !correct(c_i \mid X_t)) = 0.0
\end{aligned} \tag{2}$$

where $T(c_i \in C_p|X_t)$ is the target value associated to the input pattern X_t for a class c_i that appears in the adaptation set, $T(c_i \in C_m|X_t)$ is a target value associated to the input pattern X_t for a class c_i that is missing in the adaptation set, $OUTPUT\_ORIGINAL\_NN(c_i|X_t)$ is the output of the original network (before adaptation) for class c_i given the input pattern X_t , and $correct(c_i|X_t)$ is a predicate which is true if c_i is the correct class for the input pattern X_t.

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

B. Support Vector Rehearsal

We define “support vectors” (SV) a subset of the training patterns that are topologically located near the borders of the classes. The idea of Support Vector Rehearsal is to add a set of SVs to the adaptation set. The target assigned to a SV is the output computed by the original network. Thus, the SVs act as sentinels and try to keep the classification boundaries of the adapted network close to the ones of the original network for classes not represented in the adaptation set.

1) Support Vector selection: since SVs should be patterns located near the class borders, where the decision of the ANN is less confident, a good criterion for selecting a pattern as a SV is the entropy of the corresponding ANN outputs. A better measure, which allows obtaining a [0,1] range is the Normalized entropy obtained by normalizing the entropy by the logarithm of the number N of output classes:

$$H' = \frac{-\sum_{i=1}^{N} o_i \log(o_i)}{\log(N)} \qquad (3)$$

where o_i is the output of class i.

Using this measure, the SV set for a trained ANN can be defined as $S = \{p_i \mid H'(net\_output(p_i)) > K\}$, $K \in [0,1]$, where p_i is an input pattern and the threshold K tunes the sieve.

2) Association of Support Vectors to Classes: each element of S belongs to a class c_i, and can be used as a sentinel of the border between two or more class separation surfaces. Thus we associate a SV, belonging to a class c_i, to one or more "borders", defined by a pair of classes c_i,c_j.

$S(c_i,c_j) \subset S$, is the subset of the SVs that will be used to try keeping unaffected the border between c_i and c_j.

Let $p_i \in S$ be a SV belonging to class c_i. To identify its neighbor classes, the following assignment rule is used: excluding the contribution of o_i from (3), the SV p_i is assigned to the pair (c_i,c_j), where c_j is the class for which the contribution to H' of o_j is maximum.

To find all the neighbouring classes, this rule is iterated, excluding from the contribution to H' the already assigned classes, until $H' < K$. Thus, a SV p_i can be associated to several class pairs.

3) Exclusion of the SVs belonging to the adaptation set: all the pairs of classes where at least one class is present in the adaptation set are eliminated from S; therefore a new S' is generated according to the following formula:

$$S' = \bigcup_{i,j} \{S(c_i,c_j) \mid c_i \in C_m \vee c_j \in C_m\} \qquad (4)$$

The Support Vector set S' complements the adaptation set A, obtaining an enlarged adaptation set $A' = A \cup S'$.

Using A' the ANN can be adapted while preserving the boundaries surfaces of the missing classes.

4) Reduction of the cardinality of the SV set: since an important practical issue for large training databases is the number of SVs that must be stored, the cardinality of each Support Vector set $S'(c_i,c_j)$ can be sensibly reduced by clustering the included SVs, and keeping only the cluster centroids.

IV. EXPERIMENTAL RESULTS ON ARTIFICIAL DATA

An artificial two-dimensional classification task has been used to investigate the effectiveness of the Conservative Training and Support Vector Rehearsal techniques. An MLP has been used to classify points belonging to 16 classes having the rectangular shapes shown by the green borders in Figure 1. 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 1 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 was greater than 0.5. At left and right sides of Figure 1, MLP outputs have also been plotted for test points belonging to regions that have not been trained and outside the green rectangles. The average classification rate for all classes, and particularly 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 2. In the first adaptation experiment, all the 760 MLP weights and 56 biases were adapted. The catastrophic forgetting behavior of the adapted network is evident in Figure 2, 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 can be noticed for the classification regions of all classes, even the ones far from the adapted classes. This undesired drag of the boundary surfaces induced by the adaptation process damages the overall average classification rate as shown in the second row of Table 1.

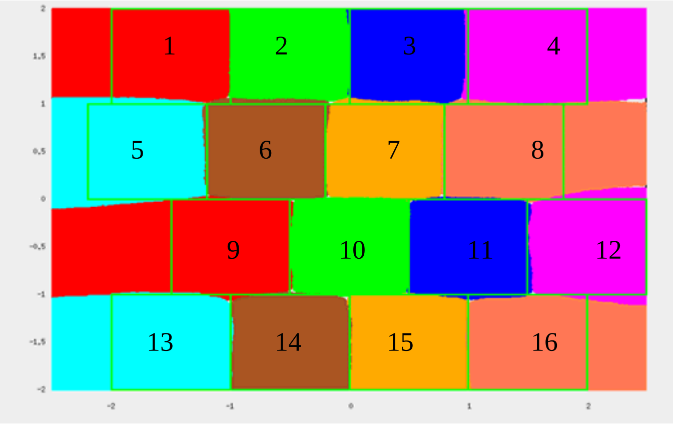

Fig. 1 Training 16 classes on a 4 layer network with 760 weights

TABLE I

CORRECT CLASSIFICATION RATES ON THE ARTIFICIAL DATA TASK

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. whole network	**SV**	**96.8**	**99.1**	**94.8**
5. LIN	**none**	**42.6**	**100**	**95.7**
6. LIN	**CT**	**69.0**	**99.0**	**91.8**
7. LIN	**SV**	**68,6**	**100**	**92.4**
8. LHN	**none**	**65.4**	**99.6**	**97.2**
9. LHN	**CT**	**86.7**	**98.0**	**93.3**
10. LHN	**SV**	**96.8**	**99.0**	**95.8**
11. whole network	**clustered SV**	**94.1**	**100**	**97.9**

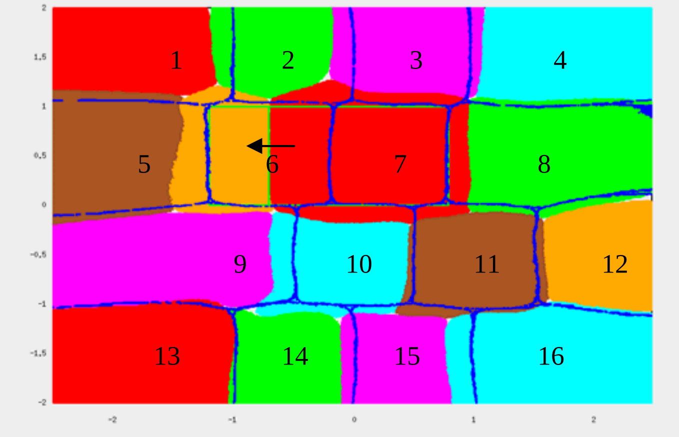

Fig. 2 Adaptation of all the network weights.
The adaptation set includes examples of class 6 and class 7 only.

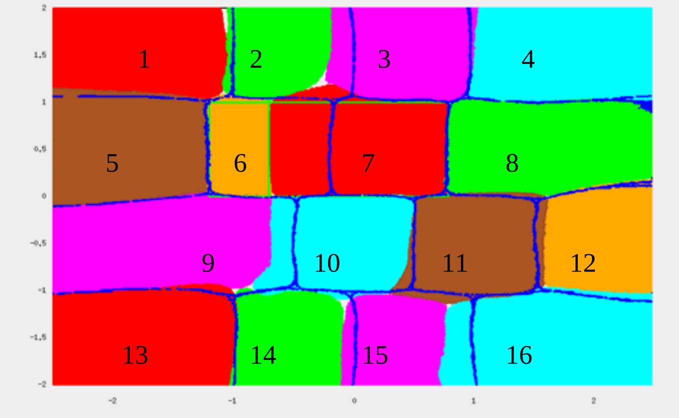

Fig. 3 Conservative Training Adaptation of the network

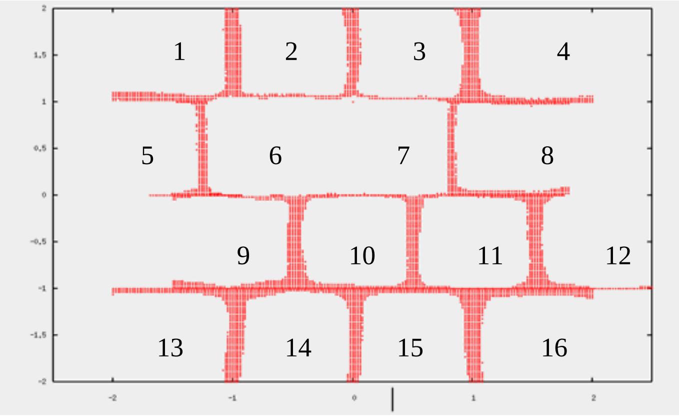

Fig. 4 Support Vectors for the adaptation set

To mitigate the catastrophic forgetting problem, the adaptation of the network has been performed using Conservative Training. Figure 3 shows how the trend of classes 6 and 7 to invade neighbor classes is largely reduced: these classes well fit their true classification regions, and although the left shift syndrome is still present, the adapted network performs better as shown in the third row of Table 1.

We then added to the adaptation samples a set of Support Vectors. In this example, 7635 samples, corresponding to the 19.1% of the patterns used to train all the weights of the MLP, have been selected according to the entropy criterion proposed in Section III, with a threshold K of 0.1. An image of the selected Support Vectors is given in Figure 4.

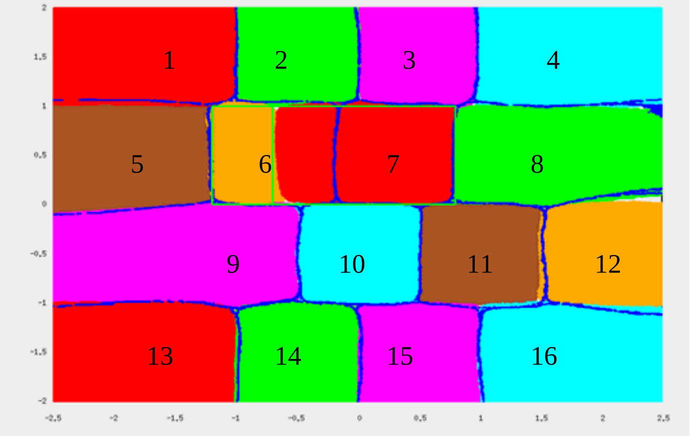

Fig. 5 Network adaptation using Support Vectors

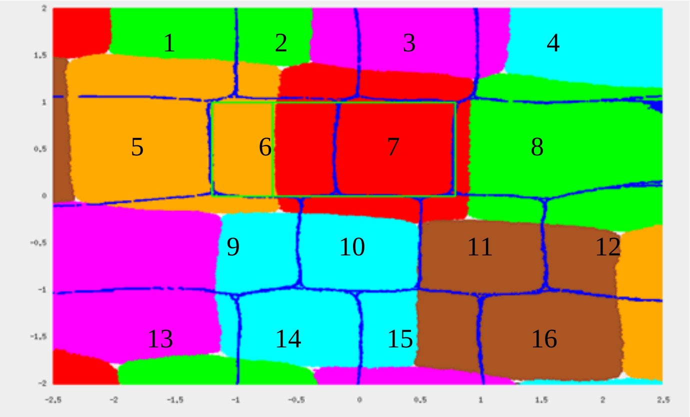

Fig. 6 Adaptation by a LIN

As shown in Figure 5 and line 4 of Table 1, the use of the Support Vectors brings again the performance to a level comparable to the one achieved using the whole train set of 40.000 patterns.

It is interesting considering the results obtained by adding a linear layer (LIN) in front of the trained network, and adapting only the weights of the LIN network. This configuration is often used in Speech Recognition experiments because the LIN layer performs a transformation of the input features to better fit the adaptation environment. Moreover, the amount of adaptation data is usually not sufficient to adapt all the network weights (possibly greater than 300.000).

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 does not help too much. Moreover the degree of freedom of this LIN is really poor: the LIN includes 4 weights and 2 biases only. These considerations are confirmed in Figure 6 and line 5 of the

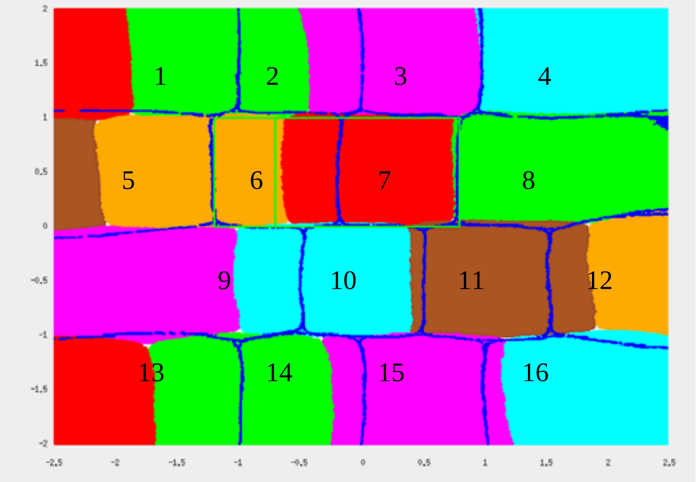

Fig. 7 Conservative Training adaptation by a LIN

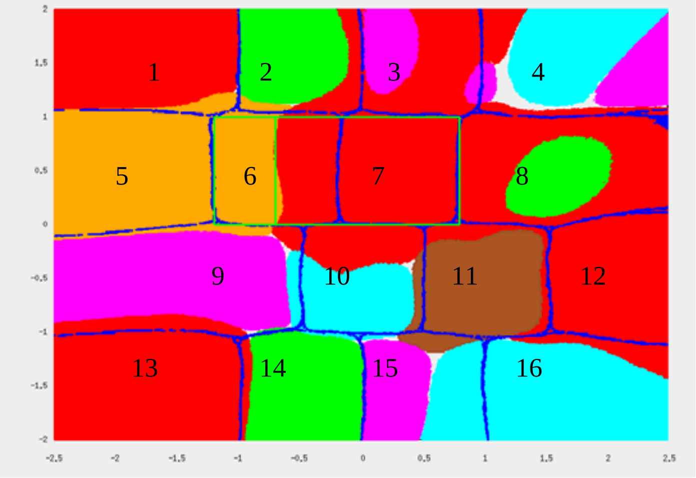

Fig. 8 Adaptation by a LHN.

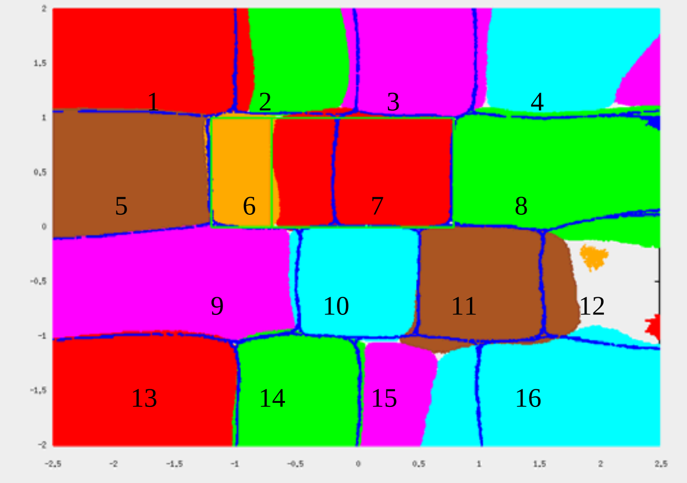

Fig. 9 Conservative Training adaptation by a LHN

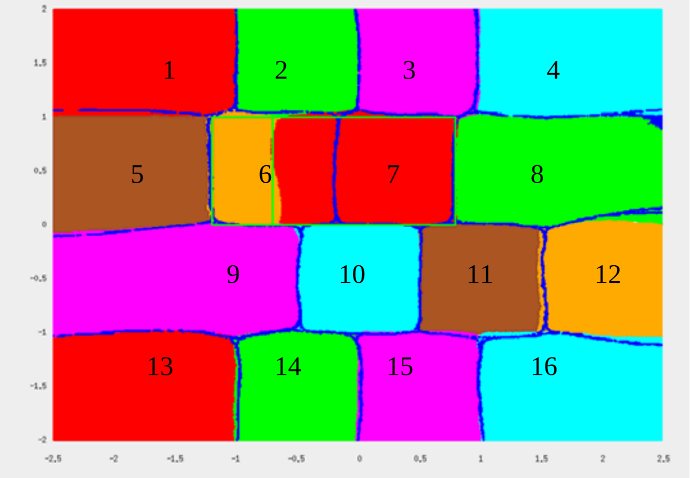

Fig. 10 Adaptation by a LHN using Support Vectors

summary 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 boundary between class 6 and 7 has the catastrophic forgetting effect of enlarging the regions of all the classes. The mitigation of these effects introduced by Conservative Training is shown in Figure 7 and at line 6 of Table 1. The drag toward left syndrome is still visible, but the horizontal boundary surfaces are now correct. Since we have too restricted freedom degrees, the inclusion of Support Vectors in the adaptation set does not give any contribution to the performance (see line 7 of Table 1).

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 8 of Table 1). However, as Figure 8 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 9, even if class 12 does not present high scores. Finally, adding the set of the Support Vectors to adaptation set allows reaching a performance that is comparable or even better than the one obtained with the original training. The estimated classification regions are shown in Figure 10, and the classification rate details are reported at line 10 of Table 1.

Finally, to reduce the number of Support Vectors that must be kept for incremental learning/adaptation, they have been clustered. By adding to the adaptation set 32 Support Vectors per class only, corresponding to 512 Support Vectors (1.28% of original training set), and adapting all the network weights, the classification rate reported at the last line of Table 1 favorably compares with the one reported at line 4, achieved using all the Support Vectors. This shows that it is possible to reduce SV cardinality while maintaining the advantages of the method.

V. EXPERIMENTAL RESULTS ON SPEECH RECOGNITION TASKS

We report in this Section the results of several experiments referring to the adaptation of ANN devoted to Speech Recognition. The results 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 task with a limited performance decrease of the original model on other tasks (some performance reductions are inevitable because the ANN is adapted to a specific condition and thus it is less general). The experimentation with the Support Vectors Rehearsal technique is in progress, its results will be reported in a future communication.

A. Test on Different Speech Adaptation Tasks

Adaptation to a specific speech application may involve the speakers, the channel, the environmental noise and the vocabulary, especially if the application uses specific list of terms. The proposed techniques have been tested on a variety of test-beds requiring different types of adaptation. The adaptation tasks that have been considered are listed below.

1) Application adaptation: Directory Assistance

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

2) 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.

3) Channel-Environment adaptation: Aurora-3

The benchmark is the standard Aurora3 Italian corpus, made by connected digits recorded in car environment. The Well-Matched train set has been used for adaptation (2951 utterances). The reported results have been obtained on the Well-Matched test set (the noisy channel, ch1 - 654 utterances).

The results are given in terms of Word Error Rate percentage (WER%) defined as:

$$WER\% = \frac{S + D + I}{N_r} \times \quad 100$$

where N_r is the number of words in the reference transcription, S is the substitution count, D the deletion count and I the insertion count. Note that this measure has no upper bound.

The results on these tests, reported in Table II, 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 adaptation of the whole net is feasible only if many adaptation data are available, and is less effective than LHN. As expected, CT slightly reduces the performance, but the CT adapted models have greater generalization capabilities, as shown in Table III. Table III shows the effects of forgetting by testing the adapted models on a generic task (continuous speech with a large vocabulary). Last line of Table III shows the reference recognition rate achieved using unadapted acoustic models on the same task.

Table II

ADAPTATION RESULTS (WER%) ON DIFFERENT TASKS AND METHODS. THE *LOQUENDO* TELEPHONE MODELS ARE THE SEED MODELS

Adaptation task ***Adaptation method***	***Application*** **Directory Assistance**	***Vocabulary*** **Command Words**	***Channel-Environ.*** **Aurora3 Ch1**
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 III

EVALUATION OF THE FORGETTING PROBLEM: RECOGNITION RESULTS (WER%) ON ITALIAN CONTINUOUS SPEECH WITH DIFFERENT ADAPTED MODELS

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**		

Because the adapted models have been specialized to a specific condition, some performance reductions are acceptable. The problem is catastrophic forgetting, which takes place when the vocabulary of the adaptation set is small and has a poor phonetic coverage, as shown by the *Command words* and *Aurora 3* results emphasized in *italic* in Table III. Conservative Training mitigates the problem, preserving an acceptable performance of the adapted model on the task for which it was originally trained (open vocabulary speech recognition).

B. Speaker Adaptation (WSJ0)

Further experiments have been performed on the Wall Street Journal WSJ0 speaker adaptation corpus in several conditions. Three baseline models have been used:

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

Moreover, two architectures are tested for each type of model: a standard one (STD), described in sub-section II.A and an improved one (IMP), characterized by a wider input window modeling a time context of 250 ms [18], and by the inclusion of a third 300 units hidden layer.

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 Sennheiser microphone (si_et_05, 8 speakers x ~40 utterances) with bigram or trigram standard LM provided by Lincoln Labs.

TABLE IV
SPEAKER ADAPTATION RESULTS (WER%) – WSJ0 8 KHZ

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

TABLE V
SPEAKER ADAPTATION RESULTS (WER%) – WSJ0 16 KHZ

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

The results, reported in Tables IV and V, show that also in these experiments LHN is always better that LIN. The combination of LIN and LHN (simultaneously trained) is usually better that the use of LHN alone. Conservative Training effects are of minor importance in WSJ0 because the adaptation set has a good phonetic coverage, and the problem of missing phonetic classes is not dramatic. Nevertheless, CT use improves, even in this case, the performance (compare LIN and LIN+CT rows in the Tables).

VI. CONCLUSION

The Conservative Training method has been proposed for reducing the effects of catastrophic forgetting when the adaptation set does not contains patterns for some classes. The advantages of adapting the outputs of the last hidden layer of ANN acoustic models in association with Conservative Training have been studied on an artificial test-bed, and presented as results of several experiments for the adaptation of a speaker independent phonetic ANN to a new application, a new vocabulary, a new noisy environment and new speakers. Furthermore, experiments on speaker adaptation show that the simultaneous use of linear transformations at different layers increase the adapted system performance.

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

Work is in progress using the Support Vectors Rehearsal technique to find a good tradeoff between the cardinality of the support vectors that must be selected and the accuracy of the classification of the adapted networks.

REFERENCES

[1] M McCloskey,. and N.Cohen, *Catastrophic interference in connectionist networks: The sequential learning problem*. In G. H. Bower (ed.) The Psychology of Learning and Motivation: Vol. 24, 109-164, NY: Academic Press, 1989.

[2] R. Ratcliff *Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions*. Psychological Review, 97, 285-308, 1990

[3] M. French, “Catastrophic Forgetting in Connectionist Networks: Causes, Consequences and Solutions”, in *Catastrofic Forgetting in Connectionist Networks, Trends in Cognitive Sciences*, 3(4), pp. 128-135, 1999.

[4] A. Robins, “Catastrophic forgetting, rehearsal, and pseudo-rehearsal.” *Connection Science*, 7, 123 – 146, 1995.

[5] M.F. BenZeghiba, and H. Bourlard, **“**Hybrid HMM/ANN and GMM Combination for User-Customized Password Speaker Verification**,”** ICASSP-03, 2003.

[6] J. L. Gauvain, and C. H. Lee, “Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains”, *IEEE Trans. on Speech and Audio Processing*, Vol. 2, n. 2, pp. 291-298, 1994.

[7] M.J.F. Gales, “Maximum Likelihood Linear Transformations for HMM-based Speech Recognition”, *Computer Speech and Language*, Vol. 12, pp. 75-98, 1998.

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

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

[10] R Hsiao and B. Mak, Discriminative feature transformation by guided discriminative training, Proc. ICASSP-04, Montreal, pp. 797-900, 2004.

[11] X. Liu and M.J.F. Gales, Model complexity control and compression using discriminative growth functions, Proc. ICASSP-04, Montreal, pp. 897-800, 2004.

[12] V. Abrash, H. Franco, A. Sankar, and M. Cohen, “Connectionist Speaker Normalization and Adaptation,” Proc. EUROSPEECH 1995, pp. 2183–2186, 1995.

[13] J. Neto, L. Almeida, M. Hochberg, C. Martins, L. Nunes, and S. Renals, T. Robinson, “Speaker-adaptation for Hybrid HMM-ANN Continuous Speech Recognition System,” Proc. EUROSPEECH 1995, pp. 2171–2174, 1995..

[14] D. S. Pallett, J. G. Fiscus, W. M. Fisher, J. S. Garofolo, B. A. Lund, and M. A. Przybocki, “1993 Benchmark Tests for the ARPA Spoken Language Program,” In Proc. of the Human Language Technology Workshop, pp. 49–74, Plainsboro, 1994.

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

[16] D. Albesano, R. Gemello, and F. Mana, “Hybrid HMM-NN Modelling of Stationary-Transitional Units for Continuous Speech Recognition”, Int. Conf. On Neural Information Processing, pp. 1112–1115, 1997.

[17] R. Gemello, F. Mana, and D. Albesano, “Linear Input Network based Speaker Adaptation in Dialogos System”, Proc. IJCNN 1998, pp. 2190-2193, 1998.

[18] S. Dupont, C. Ris, L. Couvreur, and J. M. Boite. “A study of implicit and explicit modelling of coarticulation and pronunciation variation”, Proc. Interspeech-05, Lisbon, 2005.