Speeding-up Neural Network Training Using Sentence and Frame Selection

Speeding-up Neural Network Training Using Sentence and Frame Selection

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

[1] Politecnico di Torino, Italy, [2] Loquendo, Torino, Italy

{Stefano.Scanzio, Pietro.Laface}@polito.it

{Roberto.Gemello, Franco.Mana}@loquendo.com

Abstract

Training Artificial Neural Networks (ANNs) with large amounts of speech data is a time intensive task due to the intrinsically sequential nature of the back-propagation algorithm.

This paper presents an approach for training ANNs using sentence and frame selection. The goal is to speed-up the training process, and to balance the phonetic coverage of the selected frames, trying to mitigate the classification problems related to the prior probabilities of the individual phonetic classes.

These techniques, together with a three-step training approach and software optimizations, reduced by an order of magnitude the training time of our models.

Index Terms: speech recognition, neural networks training, sentence selection, frame selection

1. Introduction

The increasing dimensions of the available training corpora offer great opportunities to improve the accuracy of the models used in Automatic Speech Recognition.

It is easy to distribute on several computers the task of computing the sufficient statistics needed to train Gaussian Mixture Hidden Markov Models. Distributing the training task of Artificial Neural Networks (ANNs) is more difficult. The computational load of the standard ANNs back-propagation training algorithm can be distributed performing batch update of the model weights, but it is known that sequential reestimation of the model weights [1] gives more accurate models. Usually, a good tradeoff between speed and accuracy is obtained reestimating the weights after processing 4-16 frames. Moreover, the time needed for training ANNs with large corpora considerably increases because larger networks can be reliably estimated.

Another classification problem with ANNs is related to the a priori probabilities of the phonetic classes. The posterior probability estimation provided by the ANN models [2] is approximate. The reasons for this are the lack of training data, the scarcity of model parameters, and the convergence of the weight estimates to a local rather than to a global optimum. If the frequency of the phonetic classes which occur in the training set is largely unbalanced, the network will be biased toward predicting the most represented ones.

To reduce training time, without performance loss, data selection can be performed. The selection may also have different objectives: to have a homogeneous coverage of the phonetic classes, to focus on decision boundaries, to exclude outliers [3], or simply to incrementally train larger and larger networks [4].

We present a strategy that performs both sentence and frame selection for reducing training time, and for equalizing the phonetic coverage of the frames provided to the back-propagation algorithm. The selection of a subset of sentences from the training corpus is used to create a robust and phonetically balanced bootstrap model. Furthermore, we use probabilistic selection of the frames in the whole training set to mitigate the classification problems related to the prior probabilities of the individual phonetic classes.

The paper is organized as follows: Section 2 introduces the architecture of the Loquendo ANN models and their training procedure. Section 3 presents our approach for selecting appropriate sentences for training the bootstrap model. Section 4 is devoted to the probabilistic frame selection. Section 5 illustrates how the presented techniques, together with other software optimizations, speed-up the training process. The conclusions are given in Section 6.

2. Loquendo ANN Architecture

The Loquendo-ASR decoder uses a hybrid HMM-ANN model, where each phonetic unit is described in terms of a single or double state left-to-right automaton with self-loops. The models are based on a set of vocabulary and gender independent units including stationary context-independent phones and diphone-transition coarticulation models. The HMM transition probabilities are uniform and fixed [5].The ANN is a Multi Layer Perceptron with two hidden layers. Having two layers, rather than a larger single hidden layer, has the advantage of reducing the total number of connections without any performance degradation. The network has 273 inputs (39 parameters of a 7 frame context), 315 units for the first hidden layer, and 300 for the second hidden layer. Softmax is applied to the output layer, which includes a language dependent number of units (in the range ~700 to ~1000).

The ANN models of the Loquendo ASR recognizer are trained using 15 epochs of the back-propagation algorithm. During these epochs the initial segmentation of each sentence (obtained with a rather simple Gaussian Mixture Hidden Markov Model) is progressively changed according to the segmentation derived from forced alignment with the current model. A focus of attention mechanism is used during training to skip the back-propagation step for frames that are classified with a low error by the currently estimated model.

It is worth noting that:

- the models provided with the Loquendo ASR are application independent.
- the training procedure, and the quite large number of 15 back-propagation epochs, were set after years of tests on several languages (currently 17) and on a development set consisting of sentences of a standard set of application grammars such as digit, date, currency, and many other recognition objects including continuous speech.
- for some of these grammars the standard prior normalization to convert posterior probabilities to class-conditional probabilities is not always the best strategy.

Using larger databases and larger ANN models, the previous training approach would be too time expensive.

To reduce learning time without harming recognition performance, in the next Section we present our approach for sentence selection associated with a three-step training technique. Further time-saving is obtained by the probabilistic frame selection described in Section 4.

3. Sentence selection

Let $S^T = \{s_1, s_2, \ldots, s_N\}$ be a training set of N sentences. Our training approach is based on the following three steps:

1. Train a bootstrap model M_B with a reduced set of training sentences $S^B \subset S^T$. The initial segmentation of the sentences is produced by forced alignment using Gaussian Mixture HMMs. The initial segmentation is refined during the training process.
2. Use the obtained bootstrap model M_B to segment every sentence of the complete training set S^T.
3. Train a new model M_{new} using the complete training set, keeping the segmentation fixed and using M_B as the initial network. Keeping the segmentation fixed reduces the computational costs, as explained in Section 5.

The selection of the bootstrap sentences in the first step is particularly important because a good bootstrap model M^B allows to obtain better segmentation, and more accurate final models.

In this work we compared the performance of two selection criteria. The first one looks for the minimum number of sentences that allow obtaining at least k frames for each target phonetic class of the ANN. This problem can be directly formulated as a linear programming problem with boolean variables. Its objective function is to minimize the number of selected sentences, with constraints $n(S,c) > k \quad \forall\ c$, where $n(S,c)$ is the number of frames belonging to the sentences in set S having an associated phonetic class c. Since this problem is solved by a standard simplex procedure [6], we will refer to this approach as "Simplex". The second criterion selects the set of sentences that ensure not only the required minimum coverage of k frames per class, but also a homogeneous coverage. In particular, the objective function of this approach is to maximize the entropy of a set of sentences and also to satisfy the minimum coverage per class:

$$\max \sum_{c=1}^{C} p(S,c)\cdot \log\left(p(S,c)\right) \qquad n(S,c) > k \ \forall c \tag{1}$$

where $p(S,c)$ is the frequency of class c in the set of sentences S, and C is the number of phonetic classes.

The goal is to have a phonetically balanced subset of bootstrap sentences with a sufficient number of samples per class. This allows us to train an initial model not biased toward the phonetic classes that appear more frequently in the complete database. The maximization of this criterion is obtained by using a sub-optimal greedy algorithm:

1. Initialize the set of sentence selected for the bootstrap set S^B to *NULL*.
2. Compute the total number of frames with an associated phonetic class c in the training set, $n(S^T,c)$.
3. Sort the class labels, in ascending order of $n(S^T,c)$.
4. While c in the sorted class labels has $n(S^B,c) <= k$ do
   a. For each sentence $s \notin S^B$ compute the normalized entropy

$$E_s = \frac{\sum_{c=1}^{C} p\left(S^B \cup s, c\right)\cdot \log p\left(S^B \cup s, c\right)}{\log\left(\#c \quad in \quad S^B \cup s\right)} \tag{2}$$

   where $p(S^B \cup s, c)$ is the frequency of class c in the union set $S^B \cup s$, and the denominator normalizes the entropy by the number of phonetic classes that appear in the sentences included in the set $S^B \cup s$.
   b. Add to set S^B the sentence $s^* = argmax_s\ E_s$

The experiments of Section 5 compare the performance of these selection criteria both in terms of the number of selected sentences and in terms of the quality of the models obtained after the three-step training procedure.

4. Frame selection

Improved models can be obtained reducing the effects of the a priori biases in training data. In [2] the a posterior probabilities $P(t_i \mid x)$ produced by the ANN are normalized by the a priori class probability $P(t_i)$ to convert them to the emission probabilities $P(x \mid t_i)$ used by the HMMs. This normalization can be simply obtained by appropriately correcting the biases of the ANN output layer [7].

Rather than acting a posteriori on the output layer biases, we alleviate the problem due to the uneven distribution of the phonetic classes in the training set directly by selecting and equalizing the training samples before and during the creation of the model. We select the sentences according to the entropy approach before the creation of the model, and we select the frames according to probabilistic sampling during training.

The frames are chosen using an approach similar to the frequency balancing method of [8], where the training patterns are probabilistically selected based on a precomputed, class dependent, probability.

Since the frequency of the silence frames is usually higher than the frequency of the other phonetic classes, two different thresholds are used to adjust their selection probability.

The selection probability for silence frames is defined as:

$$prob(sil) = \theta_{sil} \cdot \frac{\sum_{c \neq sil} n(S^T, c)}{n(S^T, sil)} \tag{3}$$

where θ_{sil} is the desired percentage of silence frames with respect to the voice frames.

To compute the selection probability for voice frames, first their average number per class is computed as:

$$\bar{n} = \frac{1}{C-1} \cdot \sum_{c \neq sil} n(S^T, c) \tag{4}$$

and the selection probability for voice frames is defined as:

$$prob(c) = \theta_{voice} \cdot \frac{\bar{n}}{n(S^T, c)} \quad \forall\, c \neq sil \tag{5}$$

where θ_{voice} controls the amount of selected frames. In particular, every phonetic class (excluding silence) will be trained using, at each epoch, a maximum number of frames not greater than θ_{voice} times $\bar{n}$.

For classes scarcely represented in the training set, $prob(c)$ in (5) may exceed one. In this case, all frames of class c are used for training.

Since a frame is selected probabilistically, different frames will contribute to the training of their related class in different epochs.

It is worth noting that this procedure is completely different from the one proposed in [3], because we do not perform frame selection based on their 'usefulness', measured by the frame

Table 1. *Number of sentences, number of frames, and mean entropy for the sentences selected with different criteria on a small corpus.*

Approach	# sentences	# frames	Entropy
Simplex	7943	2814000	4.992
Entropy	8765	3026585	5.027
Random	10800	2893518	4.900
Full S^T	17839	4795454	4.898

entropy, and we do not use a smaller network for this task.

5. Experimental results

In this Section, first we briefly recall the techniques and the software optimizations that were used for a fast implementation of the back-propagation algorithm. We then compare the results of the sentence and frame selection approaches in terms of accuracy and training speed-up on different databases. The first set of experiments was performed on relatively small databases. In the final experiments, illustrated in subsection 5.4, a large database was used to train a model integrating all the presented techniques.

5.1. Back-propagation optimization

The forward and the back-propagation steps of the ANN training algorithm, which were usually executed by means of matrix by vector operations, have been implemented as matrix by matrix operations. The same operation is efficiently executed for a buffer of frames rather than for a single frame[9][10][11]. Using the Intel high performance libraries [12][13], these changes accelerated our training procedure by 3.3 times on a Xeon 3.06 processor.

5.2. Sentence selection

A set of experiments has been performed to evaluate the sentence selection criteria.

Table 1 shows the number of sentences, the number of 10ms frames, and the mean entropy obtained applying three different criteria on the Italian SpeechDat 2 training corpus, including 1581 speakers and 17839 training sentences. Using the constraint $n(S^B,c) > 2000$ for each class c, the Simplex approach selected 7944 sentences. It produced, as expected, the smallest bootstrap set. The set produced by the Entropy method, although slightly larger, presents a more balanced phonetic coverage, as suggested by its higher mean entropy. In Table 1 we also present the outcome of a random selection of the sentences which have approximately the same total number of frames. More sentences are selected using the random approach, and they are not as well balanced compared to the other techniques.

The second set of experiments was performed to verify how many epochs were necessary in the third step to reach recognition performances similar to the ones obtained using the models trained with the complete training corpus.

Figure 1 shows the word accuracy obtained using models trained with the three-step approach using the three sentence selection criteria. The tests were performed on 9400 word vocabulary a continuous speech Italian corpus including 4295 sentences of 966 speakers. No language model was used in these tests. The results, using prior's normalization [2], are given as a function of the number of epochs performed by the third step of our training approach. The sentence selection based on entropy has the best performance, while the Simplex and random selection approaches not only produce poorer

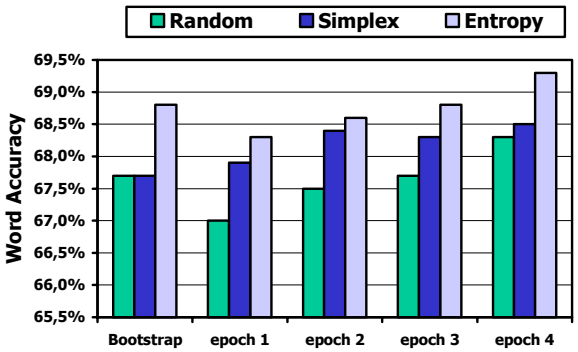

Figure 1. *Word accuracy using models trained by means of the three-step approach with different selection criteria.*

bootstrap models, but are also unable to reach the performance of the baseline model (69.3% WA), a result obtained by the entropy approach after just 4 training epochs. Since we retrain the bootstrap model using a high initial learning rate to escape local minima, it is not surprising that the performance of the models obtained in the first epochs decreases. Further iterations after the fourth epoch do not improve performance.

It has been experimentally verified that a minimum number of frames per class greater than 2000 allows the estimation of an acceptable bootstrap model. As shown in Table 1, ~3M frames are back-propagated for 15 epochs to train the bootstrap model. The first step, thus, requires a fixed computational cost. The ratio of the total computational costs for the three-steps and the original approach is given by:

$$\frac{15+k*4}{k*15}=\frac{1}{k}+\frac{4}{15} \qquad (6)$$

where k is the ratio between the number of frames in the complete and in the bootstrap training databases, and 15 and 4 are the number of back-propagation epochs for the original and the three-step approach respectively. The focus of attention mechanism has minor impact on this ratio because the proportion of the frames that are skipped is roughly equivalent in both approaches.

The time saving obtained in these experiments is limited to about 10% due to the small dimensions of the complete training set (13 hours of speech) with respect to the bootstrap set (k=1.58). The potential speed-up for this approach is, however, more than 70% (4 versus 15 epochs) for much larger corpora ($k \rightarrow \infty$). For the training database presented in Section 5.4, only 14% of its sentences and 16% of its frames are selected by the entropy criterion for training the bootstrap model, and the computational cost is reduced to ~43%. Moreover, the actual speed-up increases because larger networks can be trained using larger databases.

5.3. Frame selection

Three corpora of different languages have been used to assess the performance of the frame selection approach, and to verify that the threshold θ_{sil} and θ_{voice} settings were language and database independent:

- the previously described Italian SpeechDat 2 corpus.
- the German SpeechDat 2 corpus, which has characteristics similar to the Italian one, with 20030 training and 3395 test sentences respectively.
- the 8KHz down-sampled US-English Wall Street Journal WSJ0 including 8086 training and the 330 test sentences.

Table 2. *Word accuracy improvement w.r.t. the baseline results obtained using different normalizations of the priors.*

Corpus / Approach	Italian SpeechDat 2	German SpeechDat 2	WSJ 0
Baseline	**69.4%**	**65.9%**	**87.5%**
Prior normalization	**+0.6%**	**+1.4%**	**+1.1%**
Frame selection	**+0.9%**	**+2.3%**	**+2.0%**
Best thresholds	**+1.2%**	**+2.3%**	**+2.0%**

The results on SpeechDat tests are obtained without language model, while the standard bigram language model has been employed for the WSJ0 tests.

Table 2 shows the word accuracy obtained with the baseline model, trained with the standard 15 iterations, and without any prior normalization. The other rows in Table 2 give the word accuracy improvement provided by the prior normalization approaches. The second row refers to standard prior normalization, performed by correcting the biases of the ANN output layer of the baseline models [7]. The frame selection approach, keeping the threshold values $\theta_{sil} = 0.075$ and $\theta_{voice} = 10$ fixed for every language, gives the results reported in the third row. Last row shows the best case results that could be obtained using database dependent thresholds.

The frame selection approach outperforms the standard prior normalization technique, and the frame selection thresholds are fairly database independent, as can be observed by comparing the last two rows of Table 2.

5.4. Integration of sentence and frame selection

To validate the proposed techniques, and to quantify the training speed-up which occurs on a real task, a US-English model has been trained using our largest database including 108661 sentences for a total of 65 hours of telephone speech.

The results of the tests, performed on the same test component of the WSJ0, are summarized in Figure 2.

The baseline result has been obtained with a model trained with 15 epochs of the complete training set.

The bootstrap model has been trained selecting a subset of 15372 sentences (16% of the frames of the complete corpus) according to the entropy criterion and satisfying the usual constraint that every phonetic class includes at least 2000 frames. Using the bootstrap model, all the sentences in the database have been segmented, and a new model has been trained, in the third step, using the frame selection technique. Keeping the segmentation fixed is important for reducing the computational cost of the third training step. To estimate a new segmentation, a forward-run step of the network would be required even for frames that are not selected by the probabilistic sampling approach.

Figure 2 shows the word accuracy, and the 95% confidence intervals, obtained using the models trained with a different number of epochs. The data labels above the bars represent the percentage of time required to train the corresponding model with respect to the baseline model (100%).

A word accuracy improvement of 0.3% with respect to the baseline result can be observed after a single training epoch on the entire database. This result requires only 22% of the time needed for training the baseline model. The best performance is reached after 3 training epochs, with a 1.3% absolute improvement, and a 3 time faster training. Further iterations increase, of course, the computational efforts, but do not improve the performance, which remains inside the confidence interval of the best result.

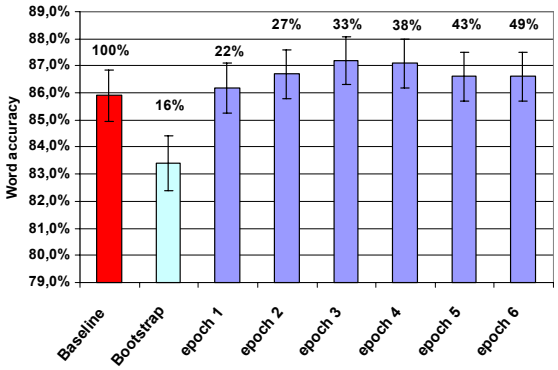

Figure 2. *Word accuracy and relative training time of models trained with the traditional and with the three-step approach.*

6. Conclusions

A technique has been presented that performs prior normalization during training rather than acting a posteriori on the trained model. This technique, together with a three-step approach, based on sentence selection, and aiming at homogeneous minimal coverage of the phonetic classes, allows the reduction of the training time to one third. Compared to the original procedure, the new approach, and the software optimizations, provides a 10 times faster training time.

7. References

[1] D. R. Wilson, and T. R. Martinez, "The General Inefficiency of Batch Training for Gradient Descent Learning," Neural Networks, Vol. 16, No. 10, pp. 1429 -1451, 2003.

[2] H. Bourlard, and N. Morgan, "Continuous Speech Recognition by Connectionist Statistical Methods," IEEE Trans. On Neural Networks, Vol 4, No. 6, Nov. 1993, pp. 893-909, 1993.

[3] C. Pelaez-Moreno, Q. Zhu, B. Chen, and N. Morgan, "Automatic data selection for MLP-based feature extraction for ASR," Proc. Interspeech 2005, pp. 229-232, 2005.

[4] Q. Zhu, A. Stolcke, B. Y. Chen, and N. Morgan, "Using MLP Features in SRI.s Conversational Speech Recognition System," in Proc. Interspeech 2005, pp. 2141-2144, 2005.

[5] 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 1997, pp. 1112–1115, 1997.

[6] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, "Introduction to Algorithms," MIT Press and McGraw-Hill, 2001.

[7] D. Albesano, R. Gemello, and F. Mana, "Hybrid HMM-NN for Speech recognition and Prior Class Probabilities," in Proceedings of the 9th International Conference on Neural Information Processing, vol. 5, pp.2391-2394, 2002.

[8] L. Yaeger, R. Lyon, and B. Webb, "Effective Training of Neural Network Character Classifier for Word Recognition," In Advances in Neural Information Processing Systems 9, MIT press, 1997.

[9] D. Anguita, G. Parodi, and R. Zunino, "An Efficient Implementation of BP on RISC-based Workstations," Neurocomputing, no. 6, pp. 57-65, 1994.

[10] J. Bilmes, K. Asanovic, C. Chin and J. Demmel, "Using Phipac to Speed Error Back-Propagation Learning," ICASSP 1997, vol 5, pp. 4153-4157, 1997.

[11] H. Schwenk, "Efficient Training of Large Neural Networks for Language Modeling," IJCNN 2004, pp. 3059-3062, 2004.

[12] Intel's MKL Math Kernel Library" http://www.intel.com/cd/software/products/asmona/eng/perflib/mkl

[13] Intel's IPP Integrated Performance Primitives, http://www.intel.com/cd/software/products/asmona/eng/perflib/ipp