Acoustic Modeling with Bootstrap and Restructuring for Low-resourced Languages

Xiaodong Cui, Jian Xue, Pierre L. Dognin, Upendra V. Chaudhari and Bowen Zhou

IBM T. J. Watson Research Center, Yorktown Heights, NY 10598, USA

{cuix, jxue, pdognin, uvc, zhou}@us.ibm.com

Abstract

This paper investigates an acoustic modeling approach for low-resourced languages based on bootstrap and model restructuring. The approach first creates an acoustic model with redundancy by averaging over bootstrapped models from resampled subsets of sparse training data, which is followed by model restructuring to scale down the model to a desired cardinality. A variety of techniques for Gaussian clustering and model refinement are discussed for the model restructuring. LVCSR experiments are carried out on Pashto language with up to 105 hours of training data. The proposed approach is shown to yield more robust acoustic models given sparse training data and obtain superior performance over the traditional training procedure.

Index Terms: speech recognition, bootstrap, model restructuring, low-resourced language

1. Introduction

Statistical acoustic modeling for automatic speech recognition (ASR) is often hindered by limited availability of training data. The data sparsity may cause poor model estimation which leads to unsatisfactory performance. This is particularly true for low-resourced languages such as Farsi, Dari or Pashto where the availability of high quality transcribed speech is limited due to the difficulty of extensive data collection and expensive labor for audio transcription. Therefore, dealing with the sparsity of the training data is not only necessary but also crucial for good ASR performance for those low-resourced languages.

In this paper, an approach based on bootstrap and restructuring is proposed and investigated for acoustic modeling of Pashto, one of the two major languages spoken in Afghanistan, with sparse training data. Bootstrap [1] as one of the resampling statistical methods has been widely practiced for estimating the properties of an estimator by sampling from the empirical distribution of the observed data, especially when the observed data is insufficient. It also finds its success in the fields of signal processing [2] and machine learning [3]. In particular, the bagging technique studied in [3] which belongs to a family of ensemble-based algorithms in machine learning can be considered an extension of bootstrap to improve the accuracy and stability of classifiers and to reduce the variance of estimates by majority voting or averaging on a set of weak learners.

The approach discussed in this paper attempts to yield more reliable model estimation out of the sparse training data by bootstrapping and model averaging. In this approach, the training data are resampled without replacement into subsets with each set covering a fraction of the total amount of training data. HMMs are trained on each individual subset and the final HMM is obtained by averaging across the HMMs trained from all of the subsets. The model averaging will generate a more robust estimate of the acoustic model which will be shown to give superior recognition performance. However, the HMM trained this way will have a much larger model cardinality than the traditional HMM and contain redundancy in parameterization. Therefore, the model is restructured afterwards to reduce the size of the parameter set. The restructuring process consists of Gaussian clustering followed by model refinement. A variety of criteria and methods for performing Gaussian clustering and model refinement will be studied and discussed.

The remainder of the paper is organized as follows. Section 2 gives an overview of the whole training scheme. Section 3 and Section 4 present the mathematical treatments of the bootstrap and model restructuring, respectively. Experimental results are shown in Section 5 followed by a summary in Section 6.

2. Algorithmic Overview

Fig.1 shows a diagram of the proposed bootstrap and restructuring scheme. The acoustic modeling investigated in this paper is targeted to ASR in device-based speech-to-speech translation applications. It is subject to constraints on both speed and memory due to which a static graph [4] of modest size is employed for fast decoding. Therefore, multiple LDAs and decision trees (DTs) are not feasible in the decoding strategy. Accordingly, there will be only one LDA and decision tree under consideration in this work.

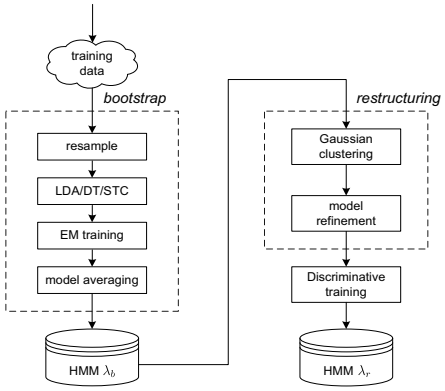

Figure 1: Diagram of acoustic modeling with bootstrap and restructuring.

As illustrated in Fig.1, given the training data, resampling without replacement is applied to generate an ensemble of subsets. The ensemble of bootstrapped subsets is first pooled together for the estimation of the LDA matrix, global semi-tied covariance (STC) and decision tree. With the LDA, STC and DT in place, separate HMMs with GMM distributions are estimated by the EM algorithm on each bootstrapped subset. Model

averaging is then performed across all the HMMs estimated from the subsets to create the final HMM. This concludes the bootstrap step. The model averaging on bootstrapped HMMs leads to an HMM with large cardinality which can be downsized via model restructuring. The restructuring process aims to reduce the model cardinality to a desirable size and also remove the underlying parameterization redundancy introduced by overlap between bootstrapped subsets. It is composed of two stages: Gaussian clustering is first performed to merge similar Gaussians based on some criteria, after which the clustered distributions are further refined according to certain criteria. After the model has been restructured to a desired cardinality, discriminative training is applied using the original training data. In the next two sections detailed mathematical treatments of bootstrap and model restructuring will be presented.

3. Model Averaging with Bootstrap

Suppose S is the training data set with R utterances, $|S| = R$. Assume these R samples are from an unknown underlying distribution $\mathcal{F}$. Any estimate of parameters or statistics is a function of this underlying distribution $\mathcal{F}$ with S being a set of observations. In bootstrap methods, $\mathcal{F}$ is approximated by the empirical distribution which assumes the observed samples are uniformly distributed with probability $\frac{1}{R}$ for each sample.

Out of the original data set S, generate N subsets of data, $\{S_1, S_2, \cdots, S_N\}$, by resampling from S without replacement. Each subset covers a fraction, r, of the original data, namely, $|S_i| = r \cdot |S|, 0 < r \leq 1, i = 1, \cdots, N$. An HMM, denoted as $\lambda_{\text{bs},i}$ for its parameters, is estimated from each individual subset S_i. The overall HMM is computed as

$$\lambda = \frac{1}{N} \sum_{i=1}^{N} \lambda_{\text{bs},i} \approx \mathcal{E}_{\mathcal{F}} \left[\lambda_{\text{bs}} \right] \tag{1}$$

which can be considered an approximation to the expectation of the estimated parameters with respect to the sample distribution according to the law of large numbers.

Based on the training framework described in Fig.1, the HMMs $\lambda_{\text{bs},i}$ share the same LDA, global STC and decision tree which are built on the whole ensemble of resampled subsets. Therefore, the average of models $\lambda_{\text{bs},i}$ in Eq.1 amounts to performing average on observation distributions $f_{\text{bs},i,s}(x)$ in each state s across all the bootstrapped HMMs $\lambda_{\text{bs},i}$. Its observation distribution $f_s(x)$ in each state can be computed as

$$f_s(x) = \frac{1}{N} \sum_{i=1}^{N} f_{\text{bs},i,s}(x) \tag{2}$$

In particular, when $f_{\text{bs},i,s}(x)$ is a GMM

$$f_{\text{bs},i,s}(x) = \sum_{k_{is}=1}^{K_{is}} c_{isk} \mathcal{N}(x; \mu_{isk}, \Sigma_{isk}) \tag{3}$$

Eq.2 can be written as

$$\begin{aligned} f_s(x) &= \frac{1}{N} \sum_{i=1}^{N} \sum_{k_{is}=1}^{K_{is}} c_{isk} \mathcal{N}(x; \mu_{isk}, \Sigma_{isk}) \\ &\triangleq \sum_{i=1}^{N} \sum_{k_{is}=1}^{K_{is}} w_{isk} \mathcal{N}(x; \mu_{isk}, \Sigma_{isk}) \end{aligned} \tag{4}$$

with $w_{isk} = c_{isk}/N$. In other words, the average of the HMMs estimated from N bootstrapped subsets results in an HMM with a larger GMM in each state. From Eq.4, there are $M_s = \sum_{i=1}^{N} K_{is}$ Gaussian components in the observation distribution in state s. Usually, $M_s \gg K_{is}, i = 1, \cdots, N$.

4. Model Restructuring

Intuitively, the averaged model with large cardinality as shown in Eq.4 can give a better modeling resolution and should be more robust given the sparse training data. However, there is also underlying redundancy in the parameterization since the bootstrapped subsets from the original data set have overlapped samples. Therefore, it is necessary to restructure the model down to a reasonable size while still maintaining decent performance. The restructuring includes Gaussian clustering and model refinement.

4.1. Gaussian Clustering

The Gaussian clustering is performed in a bottom-up greedy fashion in which every iteration the two Gaussians that are most similar under certain criterion are merged into one as given in Eqs.5 and 6.

$$\mu = \bar{w}_1 \mu_1 + \bar{w}_2 \mu_2 \tag{5}$$

$$\Sigma = \bar{w}_1 \Sigma_1 + \bar{w}_2 \Sigma_2 + \bar{w}_1 \bar{w}_2 (\mu_1 - \mu_2)(\mu_1 - \mu_2)^{\mathsf{T}} \tag{6}$$

where $\bar{w}_1 = \frac{w_1}{w_1 + w_2}$ and $\bar{w}_2 = \frac{w_2}{w_1 + w_2}$. This merging procedure keeps going until the total number of Gaussians meets the target number. In what follows various criteria to measure the Gaussian similarity are investigated.

4.1.1. KL Divergence

The KL divergence is commonly applied to measure the similarity of two distributions. Given two distributions, $f_1(x)$ and $f_2(x)$, the (symmetric) KL Divergence is defined as

$$D_{\text{kl}}(f_1 || f_2) = \int f_1(x) \log \frac{f_1(x)}{f_2(x)} dx \tag{7}$$

Accordingly, when f_1 and f_2 are Gaussian distributions,

$$\begin{aligned} D_{\text{kl}}(f_1 || f_2) &= \frac{1}{2} [\log \frac{|\Sigma_2|}{|\Sigma_1|} + \text{Tr}(\Sigma_2^{-1} \Sigma_1 - I_d) + \\ &\quad (\mu_1 - \mu_2)^{\mathsf{T}} \Sigma_2^{-1} (\mu_1 - \mu_2)] \end{aligned} \tag{8}$$

In implementation, a symmetric version of Eq.8 is used for Gaussian clustering in this paper.

4.1.2. Entropy

The entropy criterion measures the change of entropy after two distributions are merged, where the distributions are assumed Gaussians. It is defined as

$$D_{\text{ent}}(f_1 || f_2) = (w_1 + w_2) \log |\Sigma| - w_1 \log |\Sigma_1| - w_2 \log |\Sigma_2| \tag{9}$$

where Σ is the covariance after the merge, computed as in Eq.6. It is easy to prove that this entropy criterion is also equal to the change of likelihood caused by merging the two distributions.

4.1.3. Bayes Error

The Bayes error measures the overlap between two distributions. Given two distributions $f_1(x)$ and $f_2(x)$, the Bayes error is defined as

$$D_{\text{bayes}}(f_1 || f_2) = \int \min(f_1(x), f_2(x)) dx \tag{10}$$

There is no closed-form solution of the Bayes error even when $f_1(x)$ and $f_2(x)$ are multi-dimensional Gaussians. Under this condition, when both $f_1(x)$ and $f_2(x)$ have diagonal covariances, an approximation is given by the product of integral from each dimension d:

$$D_{\text{bayes}}(f_1||f_2) \approx \prod_d \int \min(f_1(x_d), f_2(x_d))dx_d \quad (11)$$

A more delicate but computationally expensive way of computing the Bayes error resorts to variational optimization based on Chernoff bounds [5], but is not used in this paper.

4.1.4. Weighted Local Maximum Likelihood

The weighted Local Maximum Likelihood (wLML) is a cost function that has been successfully used for agglomerative clustering, as described in details in [7]. This cost function is used within a greedy algorithm to select pairs of components to merge. Each pair offers minimum loss of likelihood due to the merge, under the constraint of diagonal covariance.

4.2. Model Refinement

The downsized model by Gaussian clustering can serve as a starting point in the parameter space for further refinement. The goal of the model refinement is to minimize the distance between the downsized model after Gaussian clustering, denoted $g(x)$, and the original averaged model, denoted $f(x)$, which can be treated as the "ground truth" for reference.

4.2.1. Variational EM

In [6], a variational EM algorithm was proposed and studied for minimizing the KL divergence between a reference GMM distribution $f(x)$ and a GMM distribution $g(x)$ whose parameters are to be optimized:

$$f(x) = \sum_a \pi_a f_a(x) = \sum_a \pi_a \mathcal{N}(x; \mu_a, \Sigma_a) \quad (12)$$

$$g(x) = \sum_b \omega_b g_b(x) = \sum_b \omega_b \mathcal{N}(x; \mu_b, \Sigma_b) \quad (13)$$

Since there is no closed-form solution to the KL divergence between two GMM distributions, variational technique is employed for the optimization. Let

$$L(f||g) \triangleq \int f(x)\log g(x)dx$$

It is trivial to show that minimizing the KL distance is equivalent to maximization of $L(f||g)$. Introducing the variational parameters $\phi_{b|a}$ which satisfy $\phi_{b|a} \geq 0$ and $\sum_b \phi_{b|a} = 1$, by Jensen's inequality,

$$\begin{aligned} L(f||g) &= \sum_a \pi_a \int f_a(x) \log \sum_b \phi_{b|a} \frac{\omega_b g_b(x)}{\phi_{b|a}} dx \\ &\geq \sum_a \pi_a \int f_a(x) \sum_b \log \frac{\omega_b g_b(x)}{\phi_{b|a}} dx \triangleq \mathcal{L}_\phi(f||g) \end{aligned}$$

where $\mathcal{L}_\phi(f||g)$ serves as a lower bound of $L(f||g)$ for all $\phi_{b|a}$ in which the tightest lower bound is achieved by maximizing $\mathcal{L}_\phi(f||g)$ over $\phi_{b|a}$. This corresponds to the E-step of the variational EM algorithm:

$$\hat{\phi}_{b|a} = \frac{\omega_b e^{-D_{\text{kl}}(f_a||g_b)}}{\sum_{b'} \omega_{b'} e^{-D_{\text{kl}}(f_a||g_{b'})}} \quad (14)$$

For the given $\phi_{b|a}$, $\mathcal{L}_\phi(f||g)$ is convex with respect to the parameters of g_b. By maximizing, the M-step leads to

$$\omega_b = \sum_a \pi_a \phi_{b|a}, \quad \mu_b = \frac{\sum_a \pi_a \phi_{b|a} \mu_a}{\sum_a \pi_a \phi_{b|a}} \quad (15)$$

$$\Sigma_b = \frac{\sum_a \pi_a \phi_{b|a} \left[\Sigma_a + (\mu_a - \mu_b)(\mu_a - \mu_b)^{\mathsf{T}}\right]}{\sum_a \pi_a \phi_{b|a}} \quad (16)$$

4.2.2. Monte Carlo based KL Minimization on GMM

Monte Carlo (MC) based KL minimization on GMM minimizes the KL divergence between $f(x)$ and $g(x)$,

$$g(x) = \underset{g(x)}{\text{argmin}}\, D_{\text{kl}}(f||g) = \underset{g(x)}{\text{argmin}} \int f(x) \log \frac{f(x)}{g(x)} dx \quad (17)$$

which amounts to the following maximization problem

$$\begin{aligned} g(x) &= \underset{g(x)}{\text{argmax}} \int f(x) \log g(x) dx = \underset{g(x)}{\text{argmax}}\, \mathbf{E}_f \left[\log g(x)\right] \\ &\approx \underset{g(x)}{\text{argmax}} \frac{1}{N} \sum_{i=1}^{N} \log g(x_i) \end{aligned} \quad (18)$$

where the last step is approximated by Monte Carlo methods in which N samples $\{x_i\}_{i=1}^N$ are produced by sampling $f(x)$. Closer inspection shows that Eq.18 is simply to fit a maximum likelihood model to the samples generated from distribution $f(x)$ starting from $g(x)$ where the traditional EM algorithm for GMM estimation is readily applied.

4.2.3. Monte Carlo based KL Minimization on HMM

As an extension of KL minimization on GMM, KL minimization on HMM follows an analogous mathematical argument to Section 4.2.2 provided a careful definition of the HMM is given in terms of a distribution (integrates to one) over all sequence lengths [8]. Following the definitions in [8], let $x_{1:n} \triangleq (x_1, \cdots, x_n)$ be a sequence of observations, $f(x_{1:n})$ and $g(x_{1:n})$ are the reference HMM and HMM to be refined respectively.

$$\begin{aligned} g(x) &= \underset{g(x_{1:n})}{\text{argmin}}\, D_{\text{kl}}(f||g) = \underset{g(x_{1:n})}{\text{argmax}}\, \mathbf{E}_f \left[\log g(x_{1:n})\right] \\ &\approx \underset{g(x_{1:n})}{\text{argmax}} \frac{1}{N} \sum_{i=1}^{N} \log g_i(x_{1:n}) \end{aligned} \quad (19)$$

where N utterances are sampled according to HMM $f(x_{1:n})$. If the ensemble of all bootstrapped utterances is treated as the N sequence samples according to the "ground truth" generative HMM $f(x_{1:n})$, then the KL minimization on HMM in Eq.19 is equivalent to ML estimation starting from HMM $g_i(x_{1:n})$ using the ensemble of all bootstrapped utterances. Distinguished from the KL minimization on GMM in which the optimization is carried out within states on independent observations, KL minimization on HMM refines the model on observation sequences.

5. Experimental Results

The Pashto data used by this paper was collected and transcribed under the DARPA Transtac project. There are in total less than 150 hours of data, delivered in batches. The proposed bootstrap and restructuring approach is investigated with 35 hours, 60 hours and 105 hours of training data respectively.

The feature space is constructed by splicing 9 frames of 24 dimensional PLP features and projecting down to a 40 dimensional space via LDA followed by a global STC. Context-dependent quinphone states are tied by a decision tree. Discriminative training is applied to both the feature space (fMPE[9]

/ fMMI[10]) and model space (MPE[11] / BMMI[10]). The training process follows Fig.1. In bootstrap, each resampled subset covers 70% of the original training set. Criteria discussed in Sections 4.1 and 4.2 for Gaussian clustering and model refinement are compared. A trigram language model with 1.2M n-grams is used for test, with a dictionary of 30K words.

	35h		60h		105h	
	size	WER	size	WER	size	WER
ml	1k/60k	47.7	2k/70k	46.2	3k/80k	41.1
ml(bs_ma)	3k/1M	45.3	4k/1.2M	44.4	5k/1.4M	39.5
ml(kl)	3k/60k	45.8	4k/70k	44.8	5k/80k	40.3
ml(bayes)	3k/60k	45.7	4k/70k	44.8	5k/80k	40.3
ml(wlml)	3k/60k	45.8	4k/70k	44.9	5k/80k	40.2
ml(ent)	3k/60k	45.8	4k/70k	44.8	5k/80k	40.1
ml(vem)	3k/60k	45.7	4k/70k	44.8	5k/80k	40.2
ml(mcgmm)	3k/60k	45.7	4k/70k	44.8	5k/80k	40.1
ml(mchmm)	3k/60k	45.8	4k/70k	44.6	5k/80k	39.6

Table 1: WERs of ML models on 35 hours, 60 hours and 105 hours of training data. The model refinement is performed with Gaussian clustering by the Entropy criterion.

	35h		60h		105h	
	size	WER	size	WER	size	WER
ml	1k/60k	47.7	2k/70k	46.2	3k/80k	41.1
fmpe+mpe	1k/60k	44.7	2k/70k	42.4	3k/80k	35.9
fmmi+bmmi	1k/60k	42.3	2k/70k	41.2	3k/80k	35.5
fmpe+mpe(bs)	3k/60k	43.6	4k/70k	41.8	5k/80k	35.4
fmmi+bmmi(bs)	3k/60k	41.6	4k/70k	40.3	5k/80k	34.9

Table 2: WERs of discriminatively trained models on 35 hours, 60 hours and 105 hours of training data where Gaussian clustering is performed under the Entropy criterion and model refinement by Monte Carlo based KL minimization on HMM.

model	1M	500k	300k	200k	100k	60k
WER	45.3	45.3	45.5	45.5	45.6	45.8

Table 3: WERs of ML models with various cardinalities clustered by the Entropy criterion on 35 hours of training data.

Table 1 shows the performance and the model sizes resulting from the proposed approach on ML models with 35 hours, 60 hours and 105 hours of training data. Experiments are carried out on a 10-hour held-out data set. The "size" columns give the number of states and Gaussians in the models. The first row gives the baseline performance where the ML model is trained by a traditional training recipe. The model averaging of bootstrapped model (**bs_ma**) leads to an ML model with large cardinality in both states and Gaussians. It yields significant improvements over the baseline. Gaussian clustering with KL divergence (**kl**), Bayes error (**bayes**), weighted local maximum likelihood (**wlml**) and Entropy (**ent**) can effectively scale down the model size while maintaining performance close to the averaged model. From what is observed in the table, there are no significant differences between these clustering criteria although **ent** looks slightly better. The model restructuring with variational EM (**vem**), Monte Carlo based KL minimization on GMM (**mcgmm**) and HMM (**mchmm**) is able to further improve the performance on top of just Gaussian clustering. In particular, **mchmm** obtains the best performance among the three restructuring techniques. Compared to the baseline, the proposed approach obtains 1.9%, 1.6% and 1.5% absolute improvements respectively for the three training sets. Table 2 shows the performance after both feature and model space discriminative training where Gaussian clustering is performed under the Entropy criterion and model restructuring under the Monte Carlo based KL minimization on HMM. Both fMPE+MPE and fMMI+BMMI with bootstrap and restructuring (**bs**) outperform their counterparts from the original discriminative training. The boosted MMI training (fMMI+BMMI) yields better performance than MPE training (fMPE+MPE) in all three cases. This demonstrates similar results to those reported in [10] where BMMI is reported to work better when the training data size is not very large. Overall, compared with the best performance delivered by FMMI+BMMI, the proposed approach obtains 0.6%-0.9% absolute improvement. This gives the best Pashto ASR performance in the Transtac project evaluation.

The model restructuring in Tables 1 and 2 tries to downsize the Gaussians to a comparable level to the original baseline models. However, it can do more by providing the flexibility of restructuring the large cardinality of averaged model to any desired size. Table 3 gives the performance of models of various cardinalities from 35 hours of training data. Given such limited training data, the proposed approach can create a model with 500K Gaussians which is 2.4% absolute better than the 60K Gaussian model trained by the traditional approach. In general, with traditional training, a model with 500K Gaussians estimated by 35 hours of training data can not obtain much improved performance due to unreliable estimation. The same can be said about the decision tree. With the ensemble of bootstrapped subsets, the decision tree can grow deep, which is difficult in traditional training for robust estimation out of insufficient data.

6. Summary

An acoustic modeling approach based on bootstrap and model restructuring is investigated for low-resourced languages in this paper. The approach deals with data sparsity by averaging models trained from bootstrapped subsets without replacement. Model restructuring is then applied to scale down the averaged model to a desired cardinality. Experiments on up to 105 hours of training data show that the approach can improve the robustness of modeling and outperform models trained traditionally.

7. References

[1] B. Efron, "Bootstrap methods: Another look at the jackknife", The Annals of Statistics, vol. 7, no. 1, pp. 1-26, 1979.

[2] A. M. Zoubir and D. R. Iskander, "Bootstrap techniques for signal processing", Cambridge University Process, 2004.

[3] L. Breiman, "Baggging predictors", Machine Learning, vol. 24, no. 2, pp. 123-140, 1996.

[4] G. Saon, D. Povey, and G. Zweig, Anatomy of an extremely fast LVCSR decoder, Proc. of Interspeech, pp. 549 552, 2005.

[5] P. A. Olsen and J. R. Hershey, "EVV confusability project", IBM internal technical report.

[6] P. L. Dognin, J. R. Hershey, V. Goel and P. A. Olsen, "Refactoring acoustic models using variational Expectation-Maximization", ICASSP 2009.

[7] P. L. Dognin, J. R. Hershey, V. Goel and P. A. Olsen, "Restructuring acoustic models for client and server-based automatic speech recognition", SQ2010, www.spokenquery.org, Mar, 2010.

[8] J. R. Hershey and P. A. Olsen, "Variational Bhattacharyya divergence for hidden Markov models", ICASSP 2008.

[9] D. Povey, et al. "fMPE: discriminatively trained features for speech recognition", ICASSP, 2005

[10] D. Povey, et al, "Boosted MMI for model and feature-space discriminative training", ICASSP, 2008.

[11] D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training", ICASSP, 2002.