Sequence-discriminative training of deep neural networks

Karel Veselý1, Arnab Ghoshal2, Lukáš Burget1, Daniel Povey3

1Brno University of Technology, Czech Republic
2Centre for Speech Technology Research, University of Edinburgh, UK
3Center for Language and Speech Processing, Johns Hopkins University, USA
iveselyk@fit.vutbr.cz, a.ghoshal@ed.ac.uk, burget@fit.vutbr.cz, dpovey1@jhu.edu

Abstract

Sequence-discriminative training of deep neural networks (DNNs) is investigated on a 300 hour American English conversational telephone speech task. Different sequence-discriminative criteria — maximum mutual information (MMI), minimum phone error (MPE), state-level minimum Bayes risk (sMBR), and boosted MMI — are compared. Two different heuristics are investigated to improve the performance of the DNNs trained using sequence-based criteria — lattices are regenerated after the first iteration of training; and, for MMI and BMMI, the frames where the numerator and denominator hypotheses are disjoint are removed from the gradient computation. Starting from a competitive DNN baseline trained using cross-entropy, different sequence-discriminative criteria are shown to lower word error rates by 8-9% relative, on average. Little difference is noticed between the different sequence-based criteria that are investigated. The experiments are done using the open-source Kaldi toolkit, which makes it possible for the wider community to reproduce these results.

Index Terms: speech recognition, deep learning, sequence-criterion training, neural networks, reproducible research

1. Introduction

This paper presents a reproducible set of experiments on speech recognition with a deep neural network (DNN) - hidden Markov model (HMM) hybrid. In such hybrid setups the DNN is used to provide pseudo-likelihoods (“scaled likelihoods”) for the states of an HMM [1]. While computational constraints limited earlier uses of hybrid systems to estimating scaled likelihoods for monophones using a two layered network [2] and recurrent networks [3], recent years have seen a resurgence in their use [4, 5, 6, 7]. The principal modeling and algorithmic difference to previous systems is the use of RBM pretraining [8].

Neural networks (NNs) for speech recognition are typically trained to classify individual frames based on a cross-entropy criterion (section 2.1). Speech recognition, however, is inherently a sequence classification problem. As such, speech recognizers using Gaussian mixture model (GMM) as the emission density of an HMM achieve state-of-the-art performance when trained using sequence-discriminative criteria like maximum mutual information (MMI) [9, 10], boosted MMI (BMMI) [11], minimum phone error (MPE) [12] or minimum Bayes risk (MBR) [13, 14, 15]. It is possible to efficiently estimate the parameters based on any of these criteria using statistics collected from lattices [10, 12].

The theory for sequence-discriminative training of neural networks was also developed in early literature [16, 17]. In fact, the “clamped” and “free” posterior described in [16] are same as the numerator and denominator occupancies [10] used in discriminative training of GMM-HMM systems. The logical extension of this fact is that sequence-discriminative training of NNs can take advantage of the lattice-based computations that are routinely used for GMM-HMM systems. This was pointed out in [18], where it was shown that the sequence-discriminative training can improve upon networks trained using cross-entropy. Subsequent results reported in [19, 20, 6] have also shown consistent gains from sequence-discriminative training of NNs. However, there is some disagreement about which of the criteria is suitable: [18, 20] suggest using a state-level minimum Bayes risk (sMBR) criterion, while [19] finds MMI to work better than MPE, and [6] only provide results using MML.

Needless to say, such empirical observations depend on the choice of the dataset and specific details of the implementation. In this paper, we present a comparison of the different training criteria for DNNs on the 300-hour Switchboard conversational telephone speech task, which has also been used in [5, 20]. We do this using the Kaldi speech recognition toolkit [21], which is a free, open-source toolkit for speech recognition research. The tools and scripts used to produce the results reported in this paper are publicly available as part of the Kaldi toolkit1, and anyone with access to the data should be able to reproduce our results.

2. Acoustic modeling with DNNs

In a DNN-HMM hybrid system, the DNN is trained to provide posterior probability estimates for the HMM states. Specifically, for an observation $o_{ut}$ corresponding to time $t$ in utterance $u$, the output $y_{ut}(s)$ of the DNN for the HMM state $s$ is obtained using the softmax activation function:

$$y_{ut}(s) \triangleq P(s | o_{ut}) = \frac{\exp(a_{ut}(s))}{\sum_{s'} \exp(a_{ut}(s'))},$$

(1)

1Available from http://kaldi.sf.net/
where $a_{ut}(s)$ is the activation at the output layer corresponding to state $s$. The recognizer uses a pseudo log-likelihood of state $s$ given observation $o_{ut}$,

$$\log p(o_{ut}|s) = \log y_{ut}(s) - \log P(s),$$  

(2)

where $P(s)$ is the prior probability of state $s$ calculated from the training data [1].

The networks are trained to optimize a given training objective function using the standard error backpropagation procedure [22]. Typically, cross-entropy is used as the objective and the optimization is done through stochastic gradient descent (SGD). For any given objective, the important quantity to calculate is its gradient with respect to the activations at the output layer. The gradients for all the parameters of the network can be derived from this one quantity based on the back-propagation procedure.

### 2.1. Cross-Entropy

For multi-class classification, it is common to use the negative log posterior as the objective:

$$F_{CE} = -\sum_u \sum_{t=1}^{T_u} \log y_{ut}(s_{ut}),$$  

(3)

where $s_{ut}$ is the reference state label at time $t$ for utterance $u$. This is also the expected cross-entropy between the distribution represented by the reference labels and the predicted distribution $y(s)$. The necessary gradient is:

$$\frac{\partial F_{CE}}{\partial a_{ut}(s)} = -\frac{\partial \log y_{ut}(s_{ut})}{\partial a_{ut}(s)} = y_{ut}(s) - \delta_{s,s_{ut}},$$  

(4)

where $\delta_{s,s_{ut}}$ is the Kronecker delta function. Minimizing the cross-entropy is the same as maximizing the mutual information between $y(s)$ and $\delta_{s,s_{ut}}$ computed at the frame-level.

### 2.2. MMI

The MMI criterion used in ASR [9] is the mutual information between the distributions of the observation and word sequences. With $O_u = \{o_{ut}, \ldots, o_{t_ut_u}\}$ as the sequence of all observations, and $W_u$ as the word-sequence in the reference for utterance $u$, the MMI criterion is:

$$F_{MMI} = \sum_u \log \frac{p(O_u|S_u)^n P(W_u)}{\sum_W p(O_u|S_u)^n P(W)},$$  

(5)

where $S_u = \{s_{ut}, \ldots, s_{t_ut_u}\}$ is the sequence of states corresponding to $W_u$; and $\kappa$ is the acoustic scaling factor. The sum in the denominator is taken over all word sequences in the decoded speech lattice for utterance $u$. Differentiating (5) w.r.t. the log-likelihood $\log p(o_{ut}|r)$ for state $r$, we get:

$$\frac{\partial F_{MMI}}{\partial \log p(o_{ut}|r)} = \kappa \delta_{r,s_{ut}} - \kappa \frac{\sum_{W,s_{ut}} p(O_u|S_u)^n P(W)}{\sum_W p(O_u|S_u)^n P(W)},$$

$$= \kappa (\delta_{r,s_{ut}} - \gamma_{ut}^{DEN}(r)), $$  

(6)

where $\gamma_{ut}^{DEN}(r)$ is the posterior probability of being in state $r$ at time $t$, computed over the denominator lattices for utterance $u$. The required gradient w.r.t. the activations is obtained as:

$$\frac{\partial F_{MMI}}{\partial a_{ut}(s)} = \sum_r \frac{\partial F_{MMI}}{\partial \log p(o_{ut}|r)} \frac{\partial \log p(o_{ut}|r)}{\partial a_{ut}(s)},$$

$$= \kappa (\delta_{s,s_{ut}} - \gamma_{ut}^{DEN}(s)).$$  

(7)

Note that in this work we have assumed that the reference state labels are obtained through a forced alignment of the acoustics with the word transcript. More generally, one may use forward-backward over the word reference to obtain the numerator occupancies $\gamma_{ut}^{NUM}(s)$ instead of using $\delta_{s,s_{ut}}$ in equations (4) and (7).

### 2.3. MPE/sMBR

While minimizing $F_{CE}$ minimizes expected frame-error, maximizing $F_{MMI}$ minimizes expected sentence error. The MBR family of objectives are explicitly designed to minimize the expected error corresponding to different granularity of labels [14]:

$$F_{MBR} = \sum_u \frac{\sum_W p(O_u|S_u)^n P(W) A(W,W_u)}{\sum_W p(O_u|S_u)^n P(W)},$$  

(8)

where $A(W,W_u)$ is the raw accuracy, that is, the number of correct phone labels (for MPE) or state labels (for sMBR) corresponding to the word sequence $W$ with respect to that corresponding to the reference $W_u$. Differentiating (8) w.r.t. $\log p(o_{ut}|r)$, we get:

$$\frac{\partial F_{MBR}}{\partial \log p(o_{ut}|r)} = \kappa \gamma_{ut}^{DEN}(r) \{\tilde{A}_u(s_t) = r - \tilde{A}_u\},$$

$$= \kappa \gamma_{ut}^{MBR}(r),$$

where $\tilde{A}_u(s_t) = r$ is the average accuracy of all paths in the lattice for utterance $u$ that pass through state $r$ at time $t$; $\tilde{A}_u$ is the average accuracy of all paths in the lattice; and $\gamma_{ut}^{MBR}(r)$ is the MBR “posterior” as defined in [12]. Like before, we get:

$$\frac{\partial F_{MBR}}{\partial a_{ut}(s)} = \kappa \gamma_{ut}^{MBR}(s).$$  

(9)

### 2.4. Boosted MMI

In boosted MMI [11], the MMI objective 5 is modified to boost the likelihood of paths that contain more errors:

$$F_{BMMI} = \sum_u \log \frac{p(O_u|S_u)^n P(W_u)}{\sum_W p(O_u|S_u)^n P(W) e^{-b A(W,W_u)}},$$  

(10)

where $b$ is the boosting factor. The BMMI criterion may also be interpreted as a margin term in the MMI objective [23]. The gradient computation is identical to that of MMI (eq. (7)), with the effect of the boosting showing up in $\gamma_{ut}^{DEN}(s)$.

3. Experimental setup

In this paper, we report experiments on the 300 hour Switchboard conversational telephone speech task. Specifically, we use Switchboard-1 Release 2 (LDC97S62) as the training set, together with the Mississippi State transcripts\textsuperscript{2} and the 30K-word lexicon released with those transcripts. The lexicon contains pronunciations for all words and word fragments in the English Part 1 transcripts (LDC2004T19). The LMs are trained

\textsuperscript{2}Available from: http://www.isip.piconepress.com/
using interpolated Kneser-Ney smoothing and the interpolated LM has 950K trigrams and 1064K bigrams.

The acoustic models (both GMM-HMM and DNNs) are trained on features that are obtained by splicing together 7 frames (3 on each side of the current frame) of 13-dimensional MFCCs (C0-C12) and projecting down to 40 dimensions using linear discriminant analysis (LDA). The MFCCs are normalized to have zero mean per speaker. We also use a single semi-tied covariance (STC) transform [24] on the features obtained using LDA. The combined features are referred to as LDA+STC. Moreover, speaker adaptive training (SAT) is done using a single feature-space maximum likelihood linear regression (FMLLR) transform estimated per speaker. We select the first 100K utterances from the training data to create a second smaller training set with 110 hours of speech, in order to achieve faster turnaround times for the different tuning experiments.

### 3.1. Baseline GMM-HMM systems

The baseline GMM-HMM systems are trained on the LDA+STC+FMLLR features described above. The models trained on the full 300 hour training set contain 8859 tied triphone states and 200K Gaussians. In Table 1, we compare the results of the maximum likelihood (ML) trained models with those trained using BMNI with a boosting factor $b = 0.1$ (cf. equation (10)). It is worth pointing out that the Hub5’00 data contain 20 conversations from Switchboard (SWBD) and 20 conversations from CallHome English (CHE). The CallHome data tends to be harder to recognize, partly due to a greater prevalence of foreign-accented speech. Here, we present results on both of these subsets as well as the complete Hub5’00 evaluation set. Only the results in the SWB column should be compared with the Hub5’00 results presented in [5] and [20]. The models trained on the 110 hour training set contain 4234 tied triphone states and 90K Gaussians, the results for which are similarly presented in Table 1. For either of the training conditions, the leaves of the phonetic decision tree used for the GMM-HMM system correspond to the output units of the respective DNNs.

### 3.2. DNNs trained using cross-entropy

The DNNs are trained on the same LDA+STC+FMLLR features as the GMM-HMM baselines, except that the features are globally normalized to have zero mean and unit variance. The FMLLR transforms are the same as those estimated for the GMM-HMM baselines, except that the features are similarly presented in Table 1. For either of the training conditions, the leaves of the phonetic decision tree used for the GMM-HMM system correspond to the output units of the respective DNNs.

### 3.3. Sequence-discriminative training of DNNs

Just like with GMM-HMM systems, sequence-discriminative training of DNNs start from a set of alignments and lattices that are generated by decoding the training data with a unigram LM. For each training condition, the alignments and lattices are generated using the corresponding DNN trained using cross-entropy. The cross-entropy trained models are also used as the starting point for the sequence-discriminative training. As with CE training, the posterior probability computation using the DNN and the backpropagation are done on a GPU, while the lattice-based computations run on a CPU. It is possible to speed-up the training by using a distributed algorithm [20, 25]. However, this has not been done in this initial implementation.

Through some initial benchmarking experiments with MMI as the objective function, we found 1e-5 to be a suitable learning rate for training DNNs.

---

1. We do not use different names (e.g. deep belief networks) depending on how the networks are initialized.

---

### Table 1: Results (% WER) for the baseline GMM-HMM systems on the subsets of the Hub5’00 evaluation set.

<table>
<thead>
<tr>
<th>System</th>
<th>Hours</th>
<th>SWB</th>
<th>CHE</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML SAT GMM</td>
<td>300</td>
<td>21.2</td>
<td>36.4</td>
<td>28.8</td>
</tr>
<tr>
<td>BMNI SAT GMM</td>
<td>300</td>
<td>18.6</td>
<td>33.0</td>
<td>25.8</td>
</tr>
<tr>
<td>ML SAT GMM</td>
<td>110</td>
<td>23.8</td>
<td>38.6</td>
<td>31.2</td>
</tr>
<tr>
<td>BMNI SAT GMM</td>
<td>110</td>
<td>21.0</td>
<td>35.6</td>
<td>28.3</td>
</tr>
</tbody>
</table>

### Table 2: Results (% WER) for the DNN systems on the subsets of the Hub5’00 evaluation set. The DNNs are trained on LDA+STC+FMLLR features using the cross-entropy criterion.

<table>
<thead>
<tr>
<th>System</th>
<th>Hours</th>
<th>SWB</th>
<th>CHE</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNN 7 layers</td>
<td>Rand</td>
<td>110</td>
<td>17.1</td>
<td>29.6</td>
</tr>
<tr>
<td>DNN 5 layers</td>
<td>Rand</td>
<td>110</td>
<td>17.1</td>
<td>29.6</td>
</tr>
</tbody>
</table>
Table 3: Results (% WER) of the DNNs trained on the full 300 hour training set using different criteria.

<table>
<thead>
<tr>
<th>System</th>
<th>Hub5 ’00</th>
<th></th>
<th>Hub5 ’01</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SWB</td>
<td>CHE</td>
<td>Total</td>
<td>SWB</td>
</tr>
<tr>
<td>GMM BMI</td>
<td>18.6</td>
<td>33.0</td>
<td>25.8</td>
<td>18.9</td>
</tr>
<tr>
<td>DNN CE</td>
<td>14.2</td>
<td>25.7</td>
<td>20.0</td>
<td>14.5</td>
</tr>
<tr>
<td>DNN MMI</td>
<td>12.9</td>
<td>24.6</td>
<td>18.8</td>
<td>13.3</td>
</tr>
<tr>
<td>DNN sMBR</td>
<td>12.6</td>
<td>24.1</td>
<td>18.4</td>
<td>13.0</td>
</tr>
<tr>
<td>DNN MPE</td>
<td>12.9</td>
<td>24.1</td>
<td>18.5</td>
<td>13.2</td>
</tr>
<tr>
<td>DNN BMMI</td>
<td>12.9</td>
<td>24.5</td>
<td>18.7</td>
<td>13.2</td>
</tr>
</tbody>
</table>

Figure 1: Hub5 ’00 results: DNNs trained with MMI on 110h set, with and without frame rejection (FR).

Figure 2: Histogram of lengths of rejected frame intervals.

Figure 3: Hub5 ’00: DNNs trained on 110h set, various criteria.

Figure 4: Hub5 ’00 results: Lattice regeneration after 1st epoch (indicated by “lat” suffix), trained on the 110h set.

4. Conclusions

We have presented experiments with DNN-HMM hybrid systems trained using frame-based cross-entropy and different sequence-discriminative criteria on the 300 hour Switchboard conversational telephone speech task. We achieved state-of-the-art results on this task. While the gain from sequence-discriminative training is found to be 8-9% on average, it varies depending on how matched the test set is to the train set. The system building scripts and the neural network training code are released as part of the free and open-source Kaldi toolkit, making it possible for the wider speech recognition research community to use these state-of-the-art techniques in their work.
5. References


