Discriminative Objective Functions in Neural Network Based Acoustic Modeling

Awni Hannun

August 27, 2013

1 Intro

In automatic speech recognition (ASR) we are given an input utterance of length T as a series of observations $O = \{o_1, ..., o_T\}$ which are real valued n-dimensional vectors, and we attempt to find the most likely sequence of words given those observations, $P(W|O)$. However, this distribution is too dificult to model directly, thus using bayes rule we write it as,

$$P(W|O) = \frac{P(O|W)P(W)}{P(O)} \tag{1}$$

In 1 we must model *likelihood* of the data given a word string, $P(W|O)$ and the *prior* probability of that word string, $P(W)$. The distribution over observations, $P(O)$, can be safely ignored since we are looking to maximize 1 and thus the normalization factor is not needed. Using the likelihood and the prior we can then calculate the *posterior* distribution, $P(W|O)$, (or at least a distribution proportional to it) and choose the word string W which has the highest probability.

The likelihood of the observation sequence given the word string is known as the *acoustic model* and the prior distribution over word strings is the *language model*. Typically the acoustic model is represented with a mixture of gaussians (GMM) whose parameters are estimated using maximum likelihood estimation (MLE). When attempting to model the distribution $P(O|W)$, we usually don't model the full observation directly, but instead model $P(o_i|s)$ where s is the state of the HMM at time i.

Using a Neural Network instead of modeling the likelihood of an observation given a state, we instead model the posterior distribution over states given an observation, $P(s|o)$. Using this posterior distribution we can then calculate psuedo-likelihoods that can be used as input for the HMM during the decoding process. By Bayes Rule (in log probability space), we model the likelihoods as

$$\log P(o|s) = \log P(s|o) - \log P(s) \tag{2}$$

Above we are simply modeling the conditional probability distribution over states given an observation, which we will use a Neural Network for, and the prior probability over states, which we can model simply by gathering sufficient statistics from our forced alignment. Note also in 2 that we have dropped the prior distribution over observations from the calculation of the likelihoods. This can be done without worry since we are more interested in the relative difference between likelihoods than the actual values themselves.

State of the are results in LVCSR have been achieved using feedforward neural networks with many hidden layers, commonly known as a deep neural networks (DNNs) to model the posterior distribution over states given observations. Several different loss functions exist for which to train the DNN. The most commonly employed is the Cross Entropy (CE) loss as discussed in section 2.1. This loss function only minimizes the framewise error which is not perfectly correlated with the goal of minimal word error rate over the entire utterance. To ameliorate this problem, several sequence based discriminative loss functions exist. In section 3.1, we present the Maximum Mutual Information (MMI) objective criterion and a derivation of the gradient needed to train a Neural Network using this loss function. In section 3.3 we present and derive the gradient of the Minimum Phone Error (MPE) rate objective function, and in section 3.4 we present and derive the gradient of the Minimum Bayes Risk (MBR) objective function.

Here we use a standard feedforward neural network with nonlinearities at each hidden unit (typically rectified linear units or tanh sigmoidal units). The output of the neural network to calculate all the objective functions we discuss is given by a softmax in order to properly model the framewise distribution over states given observations. The softmax is given by,

$$p(s_i|o) = \gamma_i = \frac{e^{a(s_i)}}{\sum_{s'} e^{a(s')}} \tag{3}$$

Above the sum in the denominator is taken over all states. Also $a(s)$ is the activation of the output unit of the DNN corresponding to state s. The gradient of the softmax with respect to the activation at state s_i is given by

$$\frac{\partial \gamma_i}{\partial a(s_i)} = \gamma_i(1 - \gamma_i) \tag{4}$$

And with respect to the activation as state s_j, $j \neq i$ is

$$\frac{\partial \gamma_i}{\partial a(s_j)} = -\gamma_j \gamma_i \tag{5}$$

2 Framewise Objectives

In this section we discuss several objective functions which are local to individual frames of the observation rather than a a sequence of frames e.g. the full utterance.

2.1 Cross Entropy

In the Cross Entropy loss function we attempt to minimize the cross entropy between the conditional distribution over states given observations as predicted by the neural network and the ground truth distribution as given by the forced alignments (i.e. a distribution with all the mass at the correct label). The CE criterion over all utterances $u \in U$ is given by

$$\mathcal{F}_{CE} = -\sum_{u} \hat{P}(S|O) \log P(S|O) \tag{6}$$

Thus we attempt to minimize the cross entropy between the ground truth distribution $\hat{P}$ and the DNN output distribution P. In order to minimize the above function we use the iterative first order method, stochastic gradient descent (SGD). In order to perform the update step in the SGD algorithm and to propagate the error backwards through the DNN to update the hidden layer weights, we need the gradient of 6. With respect to a single utterance, we derive the gradient at the activation $a(s_i)$,

$$\begin{aligned}
\frac{\partial \mathcal{F}_{CE}}{\partial a(s_i)} &= -\frac{\partial}{\partial a(s_i)} \hat{p}(s|o) \log p(s|o) \\
&= -\frac{\partial}{\partial a(s_i)} \hat{p}(s|o) \log \frac{e^{a(s)}}{\sum_{s'} e^{a(s')}} \\
&= -\frac{\partial}{\partial a(s_i)} \left(\hat{p}(s|o) a(s) - \log \sum_{s'} e^{a(s')} \right) \\
&= \delta_{s_i,s} - \frac{e^{a(s_i)}}{\sum_{s}' e^{a(s')}} \\
&= \delta_{s_i,s} - \gamma_i
\end{aligned} \tag{7}$$

Above $\delta_{i,j}$ is the standard Kronecker delta function taking value 1 if $i = j$ and 0 otherwise. Also since the gradient is a linear operation, in order to find the gradient at the activation $a(s_i)$ over the full training set (or minibatch as may be the case in SGD) simply sum over the gradients w.r.t each utterance.

2.2 Squared Error

2.3 Hinge Loss

3 Sequence-Discriminative Objectives

3.1 Maximum Mutual Information

In the Maximum Mutual Information (MMI) objective we attempt to maximize the mutual information between the observation sequences and the corresponding word sequences. Since the language model is fixed during training, the mutual information reduces to maximizing the posterior distribution over the full transcription given the observation sequence.

The MMI criterion can be motivated by the minimization of conditional entropy between the probability distribution over sentences and the distribution over acoustic observations with a fixed language model. In minimizing this conditional entropy, $H(W|O)$, we seek to minimize the average uncertainty in (or number of bits needed for encoding) the word transcription given an acoustic observation. We can write the conditional entropy as,

$$H(W|O) = H(W) - I(W;O) \tag{8}$$

Given 8 we see that there are two ways of decreasing the conditional entropy. Either we decrease the marginal entropy, $H(W)$, of the distribution over sentences or increase the mutual information between the two distribtuions, $I(W;O)$. Since we assume our language model is fixed at training time, the marginal entropy $H(W)$ is fixed as well given that the language model specifies the distribution $P(W)$. Thus we seek to maximize the mutual information,

$$I(W;O) = \sum_{W,O} P(W,O) \log \frac{P(W,O)}{P(W)P(O)} \tag{9}$$

Note in 9 we've discretized both the space of word strings and the space of acoustic observations for simplicity. This could be remedied easily by replacing the summation with an integral for the corrseponding continuous versions. Since the joint distribution $P(W,O)$ is unknown we use the training set as representative. Let the set of utterances be given by $\mathcal{U}$, where each utterance is a acoustic observation and ground truth transcript pair, thus we write 9 as,

$$I(W;O) \propto \sum_{u \in \mathcal{U}} \log \frac{P(W_u, O_u)}{P(W_u)P(O_u)} \tag{10}$$

However, again using the fact that the language model is fixed, the objective can be written as

$$\sum_{u \in \mathcal{U}} \log \frac{P(W_u, O_u)}{P(O_u)} = \sum_{u \in \mathcal{U}} \log P(W_u|O_u) \tag{11}$$

Thus we see that maximizing the mutual information under the above assumptions is equivalent to maximizing the conditional likelihood of the training set. Using Bayes Theorem, the final MMI objective function for discriminative acoustic models is given by

$$\mathcal{F}_{MMI} = \sum_{u} \log \frac{P(O|S_u)^k P(W_u)}{\sum_{W'} P(O|S')^k P(W')} \tag{12}$$

In 12 the likelihood $P(O|S_u)$ is over the full observation given the underlying state string $S_u = \{s_1, ..., s_T\}$ where the utterance u is of length T. The prior over the word string $P(W_u)$ is given by the language model using the reference transcript. In the denominator we sum over all possible word strings W' multiplied by the likelihood of the observation given the corresponding state string S'. Note that we've also scaled the likelihoods by k.

In practice the denominator of 12 is intractable to calculate due to the huge number of possible word strings we must search over. Thus to calculate the denominator we use a lattice which efficiently stores a set of highly probably word strings. This set of word strings is however only a small fraction of all possible word strings.

Again in order to update the weights of the DNN using SGD we must calculate the gradient of the activation w.r.t to the $\mathcal{F}_{MMI}$ objective function. We first calculate the gradient for a single utterance w.r.t. the log-likelihood of an observation at time t given state r, $\log p(o_t|r)$,

$$\begin{aligned}\frac{\partial \mathcal{F}_{MMI}}{\partial \log p(o_t|r)} &= \frac{\partial}{\partial \log p(o_t|r)} \log \frac{P(O|S_u)^k P(W_u)}{\sum_{W'} P(O|S')^k P(W')} \\ &= \frac{\partial}{\partial \log p(o_t|r)} \left[k \log P(O|S_u) + \log P(W_u) - \log \left(\sum_{W'} P(O|S')^k P(W') \right) \right]\end{aligned}$$

We analyze each term separately. The first term gives,

$$\begin{aligned}&\frac{\partial}{\partial \log p(o_t|r)} k \log P(O|S_u) \\ =&\frac{\partial}{\partial \log p(o_t|r)} k \log \prod_{i=1}^{T} p(o_i|s_i) \\ =&\frac{\partial}{\partial \log p(o_t|r)} k \sum_{i=1}^{T} \log p(o_i|s_i) \\ =&k\delta_{s_t,r}\end{aligned}$$

The second term simply drops since the prior distribution over words does not depend on the likelihood of the observation,

$$\frac{\partial}{\partial \log p(o_t|r)} \log P(W_u) = 0$$

And the third term becomes,

$$
\begin{aligned}
&\frac{\partial}{\partial \log p(o_t|r)} \log \left(\sum_{W'} P(O|S')^k P(W') \right) \\
=&\frac{1}{\sum_{W'} P(O|S')^k P(W')} \frac{\partial}{\partial \log p(o_t|r)} \sum_{W'} P(O|S')^k P(W') \\
=&\frac{1}{\sum_{W'} P(O|S')^k P(W')} \frac{\partial}{\partial \log p(o_t|r)} \sum_{W':s_t=r} e^{\log P(O|S')^k} P(W') \\
=&\frac{1}{\sum_{W'} P(O|S')^k P(W')} \frac{\partial}{\partial \log p(o_t|r)} \sum_{W':s_t=r} e^{k \sum \log p(o_t|s'_t)} P(W') \\
=&\frac{1}{\sum_{W'} P(O|S')^k P(W')} \sum_{W':s_t=r} e^{k \sum \log p(o_t|s'_t)} P(W') \frac{\partial}{\partial \log p(o_t|r)} k \sum logp(o_t|s'_t) \\
=&\frac{1}{\sum_{W'} P(O|S')^k P(W')} \sum_{W':s_t=r} e^{k \sum \log p(o_t|s'_t)} P(W') k \\
=&\frac{k \sum_{W':s_t=r} P(O|S')^k P(W')}{\sum_{W'} P(O|S')^k P(W')} = k\gamma_{tr}^{DEN}
\end{aligned}
$$

Note the above term γ_{tr}^{DEN} is simply the posterior probability of being in state r at time t. Putting the terms together our final gradient w.r.t. the log-likelihood of an observation at time t given state r we have,

$$\frac{\partial \mathcal{F}_{MMI}}{\partial \log p(o_t|r)} = k \left(\delta_{s_t,r} - \gamma_{tr}^{DEN} \right) \tag{13}$$

Now we use the chain rule to calculate the gradient w.r.t. the activation of state s at time t, $a_t(s)$,

$$\frac{\partial \mathcal{F}_{MMI}}{\partial a_t(s)} = \sum_r \frac{\partial \mathcal{F}_{MMI}}{\partial \log p(o_t|r)} \frac{\partial \log p(o_t|r)}{\partial a_t(s)} \tag{14}$$

The summation over all states r follows from the fact that the $\mathcal{F}_{MMI}$ objective is a function of the likelihoods at all states which in turn all depend on the activation at time t for a specific state s. Also, using the posteriors from the DNN softmax output layer, we calculate the psuedo log-likelihoods as $\log p(o_t|r) = \log p(r|o_t) - \log p(r)$. When $r \neq s$, we have

$$
\begin{aligned}
\frac{\partial \log p(o_t|r)}{\partial a_t(s)} &= \frac{\partial}{\partial a_t(s)} \left(\log p(r|o_t) - \log p(r) \right) \\
&= \frac{\partial}{\partial a_t(s)} \log \gamma_{tr} = \frac{1}{\gamma_{tr}} \frac{\partial}{\partial a_t(s)} \gamma_{tr} \\
&= \frac{1}{\gamma_{tr}} (-\gamma_{ts}\gamma_{tr}) = -\gamma_{ts}
\end{aligned}
$$

And when $r = s$ we have,

$$\begin{aligned}
\frac{\partial \log p(o_t|r)}{\partial a_t(s)} &= \frac{\partial}{\partial a_t(s)} \left(\log p(s|o_t) - \log p(s)\right) \\
&= \frac{\partial}{\partial a_t(s)} \log \gamma_{ts} = \frac{1}{\gamma_{ts}} \frac{\partial}{\partial a_t(s)} \gamma_{ts} \\
&= \frac{1}{\gamma_{ts}} \gamma_{ts}(1 - \gamma_{ts}) = (1 - \gamma_{ts})
\end{aligned}$$

Returning to 14, the gradient becomes,

$$\begin{aligned}
\frac{\partial \mathcal{F}_{MMI}}{\partial a_t(s)} &= \frac{\partial \mathcal{F}_{MMI}}{\partial \log p(o_t|s)}(1 - \gamma_{ts}) + \sum_{r \neq s} \frac{\partial \mathcal{F}_{MMI}}{\partial \log p(o_t|r)}(-\gamma_{ts}) \\
&= \frac{\partial \mathcal{F}_{MMI}}{\partial \log p(o_t|s)} - \gamma_{ts} \sum_{r} \frac{\partial \mathcal{F}_{MMI}}{\partial \log p(o_t|r)} \\
&= k\left(\delta_{s_t,s} - \gamma_{ts}^{DEN}\right) - \gamma_{ts} \sum_{r} k\left(\delta_{s_t,r} - \gamma_{tr}^{DEN}\right) \\
&= k\left(\delta_{s_t,s} - \gamma_{ts}^{DEN}\right) - \gamma_{ts} k \left(\sum_{r} \delta_{s_t,r} - \sum_{r} \gamma_{tr}^{DEN}\right) \\
&= k\left(\delta_{s_t,s} - \gamma_{ts}^{DEN}\right) - \gamma_{ts} k\,(1 - 1) \\
&= k\left(\delta_{s_t,s} - \gamma_{ts}^{DEN}\right)
\end{aligned} \tag{15}$$

In the second to last step we use the fact that the sum over all possible states of the posterior distribution for being in state r at time t must be 1. Also, $\delta_{s_t,r}$ evaluates to true for only one state thus also sums to 1 over all states.

3.2 Boosted MMI

With boosted MMI (BMMI) we attempt to more heavily penalize the cost from word strings that contain more errors in the denominator of 12. If we let $A(W_u, W)$ denote the raw accuracy over phone state labels or sonene state labels between the utterance reference string W_u and another word string W, usually given as a path through the lattice, the objective function for BMMI is,

$$\mathcal{F}_{BMMI} = \sum_{u} \log \frac{P(O|S_u)^k P(W_u)}{\sum_{W'} P(O|S')^k P(W') e^{-bA(W_u,W)}} \tag{16}$$

The derivation of the gradient for 16 with respect to the activation for state s at time t is exactly the same as that of 12 with the inclusion of the boosting term in the calculation of the posterior probability γ_{st}^{DEN} of being in state s at time t given the observation sequence O.

3.3 Minimum Phone Error

3.4 Minimum Bayes Risk

4 Lattice Based MMI

$$\alpha_l = \sum_k \alpha_k P_{acoust}(w_{k,l}) P_{lang}(w_{k,l}) \tag{17}$$

$$\beta_l = \sum_k \beta_k P_{acoust}(w_{k,l}) P_{lang}(w_{k,l}) \tag{18}$$

$$\gamma_{k,l} = \frac{\alpha_k P_{acoust}(w_{k,l}) P_{lang}(w_{k,l}) \beta_l}{\mathcal{L}} \tag{19}$$

References

[1] P. F. Brown. *The Acoustic-Modeling Problem in Automatic Speech Recognition.* PhD thesis, Carnegie Mellon University, May 1987.

[2] D. Povey. *Discriminative Training for Large Vocabulary Speech Recognition.* PhD thesis, University of Cambridge, March 2003.

[3] K. Vesely, A. Ghoshal, L. Burget, and D. Povey. Sequence-discriminative training of deep neural networks. *Interspeech*, 2013.

[4] P. C. Woodland and D. Povey. Large scale discriminative training for speech recognition. *Proceedings of International Workshop on Automatic Speech Recognition*, 2000.