DISCRIMINATIVE TRAINING FOR SPEECH RECOGNITION IS COMPENSATING FOR STATISTICAL DEPENDENCE IN THE HMM FRAMEWORK

Dan Gillick, Steven Wegmann

International Computer Science Institute
Berkeley, CA, USA

Larry Gillick

EnglishCentral, Inc.
Arlington, MA, USA

ABSTRACT

The parameters of the standard Hidden Markov Model framework for speech recognition are typically trained via Maximum Likelihood. However, better recognition performance is achievable with discriminative training criteria like Maximum Mutual Information or Minimum Phone Error. While it is generally accepted that these discriminative criteria are better suited to minimizing Word Error Rate, there is very little qualitative intuition for how the improvements are achieved. Through a series of "resampling" experiments, we show that discriminative training (MPE in particular) appears to be compensating for a specific incorrect assumption of the HMM—that speech frames are conditionally independent.

Index Terms— speech, discriminative training, MMI, MPE, sampling, statistical independence

1. INTRODUCTION

For many years, the model of choice for speech recognition systems has been a Hidden Markov Model (HMM) where each hidden state generates a mixture of Gaussians to represent output frames. Traditionally, the parameters of these models have been trained via the Expectation Maximization (EM) algorithm so as to maximize the likelihood of manually transcribed speech data. Such Maximum Likelihood (ML) estimation has the following desirable property: if the data satisfies the assumptions of the model, then as the amount of training data goes to infinity, the global parameter estimate will be optimal in that it is asymptotically unbiased with minimum variance [1].

In practice, however, training sets are limited in size, EM only guarantees convergence to a local, rather than global optimum, and actual speech data clearly violates the model assumptions. Most conspicuously, the output distribution of each HMM state is not truly Gaussian, and the independence assumption, that a speech frame is independent of all other frames conditional on its generating state, is false.

As a result, a variety of other estimation procedures can yield parameters that give better performance. In particular, discriminative training schemes like Maximum Mutual Information (MMI) [2, 3] and more recently, Minimum Phone Error (MPE) [4] have shown significant improvement over ML.

In general, ML estimation seeks to maximize the probability of the acoustic observations given the correct transcriptions $P(O|S)$, while discriminative training reverses the direction of the conditioning, to maximize something like $P(S|O)$[1]. Why does this work? Is there a meaningful qualitative description of how the discriminatively trained parameters differ from the maximum likelihood parameters? To the best of our knowledge, there has been no empirical investigation of this matter for speech recognition.

We present a series of experiments to demonstrate that the standard discriminative training procedures do not improve the models of the states' output distributions; somewhat surprisingly, they appear to compensate for the incorrect assumptions of independence, even beyond the state level. We hope that beginning to understand how discriminative training improves performance points the way toward more targeted research programs.

The primary statistical tool for these experiments is a version of *resampling*, as introduced in [5]—creating pseudo speech data by stringing together samples of real speech segments. By manipulating the test data to include or remove specific statistical properties, we can tease apart differences between ML-trained models and models trained discriminatively.

Section 2 briefly reviews the objective functions in question, Section 3 outlines the data and models used in the series of experiments described in Section 4; we conclude with a short discussion in Section 5.

2. OBJECTIVE FUNCTIONS

The standard objective function used in Maximum Likelihood training can be written as:

$$\mathcal{F}_{ML}(\lambda) = \sum_r \log P_\lambda(o_r|s_r) \quad (1)$$

[1] This is precisely the MMI criterion; MPE is a kind of smoothed version (see equations (2) and (3)).

Here, s_r is the correct transcription of utterance o_r. While Maximum Mutual Information (MMI) was the discriminative criterion of choice for some time, Minimum Phone Error tends to give slightly better performance. In the interest of parsimony, we restrict our analysis to MPE, though we give the objective function for MMI as well to help with intuition:

$$\mathcal{F}_{MMI}(\lambda) = \sum_r \log \frac{P_\lambda(o_r|s_r)\,P(s_r)}{\sum_s P_\lambda(o_r|s)\,P(s)} \quad (2)$$

$$\mathcal{F}_{MPE}(\lambda) = \sum_r \frac{\sum_s P_\lambda(o_r|s)\,P(s)\,A(s,s_r)}{\sum_u P_\lambda(o_r|u)\,P(u)} \quad (3)$$

$P(s)$ is the language model probability for sentence s[2]. At the sentence level, the MMI criterion is simply the posterior probability of the correct transcription: the probability of the correct transcription in the numerator, and the sum over all possible transcriptions in the denominator. In practice, the denominator is estimated from a lattice of competitive alternative transcriptions. The MPE objective is quite similar, but measures success with $A(s, s_r)$, the raw phone transcription accuracy of a sentence s relative to the reference s_r. Thus MPE favors word transcriptions that have the best phone accuracy relative to competing transcriptions. For more extensive discussion, see [6].

3. DATA AND MODELS

We show experimental results on both Wall Street Journal (WSJ), carefully read news reports in controlled quiet conditions, and Switchboard (SWB), spontaneous telephone conversations in uncontrolled environments. The training data include 66 hours of the WSJ SI-200 dataset and 300 hours of Switchboard I. Each dataset is split into two speaker-disjoint sets, one for training recognition models, and one for training models used for forced alignment and resampling (we'll call these the recognition models and the alignment models, respectively). Statistics of the training and test sets are given in Table 1; for more details, see [5].

Dataset	Speakers	Utterances	Words	Hours
WSJ align	100	13,857	249,557	32
WSJ rec	100	14,852	250,904	32
SWB align	256	105,629	1,366,704	135
SWB rec	255	100,750	1,343,286	132
WSJ test	18	576	9,381	1.2
SWB test	23	954	10,727	1.1

Table 1. Training set (top) and test set (bottom) statistics.

We use version 3.4 of the HTK toolkit to train and test our models [7]. In particular, we use the standard HTK front-end to produce a 39 dimensional feature vector every 10 ms: 13 Mel-cepstral coefficients, including energy, plus their first and second differences over a 25 ms window. The cepstral coefficients are mean-normalized at the utterance level. We use version 0.6 of the CMU pronunciation dictionary (stress removed) for both WSJ and SWB models.

The acoustic models are cross-word triphones modeled by a three-state HMM with a discrete linear transition structure (no skipping) and one diagonal Gaussian per state. Significantly better performance can be achieved with mixture models, but the simplicity of a single component makes analysis easier and helps highlight performance differences in our experiments.

Maximum likelihood training roughly follows the HTK tutorial: monophone models are estimated from a "flat start", duplicated to form triphone models, clustered (2500 states for WSJ and 5000 states for SWB), and re-estimated. The Baum-Welch algorithm is used for estimation.

MPE training roughly follows [6, 8], starting from the ML-trained models: First, a weak bigram language model is estimated from the training transcripts. Second, word-level numerator lattices are generated using the training transcripts and the training dictionary, and word-level denominator lattices are generated by running recognition on the training data using the weak bigram LM and the seed acoustic models. Finally, forced alignment, again using the seed acoustic models, is performed on these word lattices to obtain phone-marked numerator and denominator lattices. At this point, we run lattice-based MPE using the extended Baum-Welch algorithm for estimation, with standard settings ($E = 2$; $\tau = 50$)[3].

Recognition experiments are performed with a 5k bigram language model for WSJ and a 20k trigram language model for SWB, using HTK's HDecode for decoding with a wide search beam (300), a word-insertion penalty of -4, and the language model scale factor set to 15.

4. EXPERIMENTS

4.1. Classifying frames

Before delving into resampling, we begin with a more straightforward experiment. One way MPE could improve over ML is by providing better classification decisions among states. Since a single Gaussian is not the correct generative model for a state, and discriminative training in general is agnostic with regards to the underlying distributions, the discriminative framework could be learning to distinguish between states more effectively than ML.

We set up an experiment to test this possibility: First, we run forced alignment on the test data using the alignment

[2]We've omitted the scaling factor κ for simplicity, though this is in fact important for training.

[3]Our code for training and testing models (using HTK tools), including discriminative training is available here: https://code.google.com/p/pyhtk.

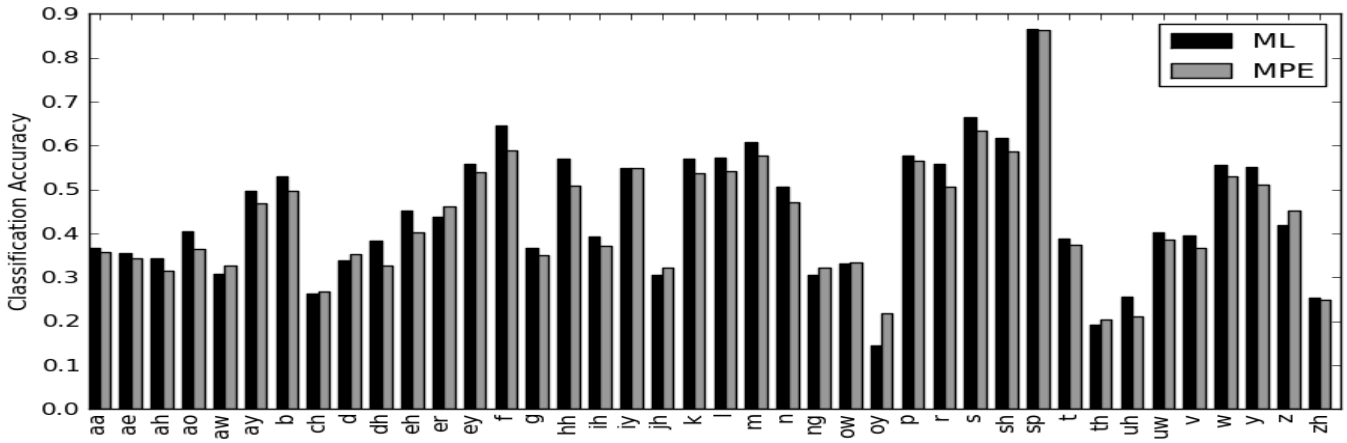

Fig. 1. Classification accuracies for ML and MPE models, broken down by phoneme (WSJ data).

model; this gives a label for each frame[4]. Next, we use the ML recognition models as well as the MPE recognition models to classify each frame (the recognition models are used to rule out any cheating with respect to the forced alignment): The posterior probability of a frame is computed using each of the state models (recall there are 2500 WSJ states and 5000 SWB states) and the predicted state is the one with the highest posterior.

We assume a uniform prior over states (though including a prior does not qualitatively change the results). In total, 427,318 frames are classified in the WSJ test, and 370,781 in the SWB test. Table 2 shows overall classification results, demonstrating that in all cases, the ML models outperform the MPE models in both state classification and phoneme classification (where the prediction is correct so long as it matches the correct phoneme)[5] Figure 1 breaks down classification accuracy into phonemes; ML is consistently better than MPE, though the diphthong "oy" is a notable exception.

	State %acc		**Phone %acc**	
Test	ML	MPE	ML	MPE
WSJ	26.7	24.2	58.1	56.6
SWB	5.3	5.2	34.2	33.6

Table 2. Overall state and phoneme classification accuracies.

These results show that MPE training does not improve on ML training at the level of frame classification. This must mean that MPE's benefit only appears across sequences of frames, a somewhat surprising result given that it can only adjust parameters within an HMM framework that assumes conditional independence among frames (and thus assigns a separate probability to each frame).

[4]The classification results are the same regardless of whether this alignment model is trained with ML or MPE.

[5]ML also outperforms MPE if correctness is defined by: Rank(correct)$\leq k$, for $k \in \{2, 5, 10, 20\}$.

4.2. Resampling

The objective of resampling, introduced in [5], is to create pseudo test data that shares the output distributions of real test data, but satisfies various independence assumptions of the HMM that are violated by real data. First, the alignment model is used to create a forced alignment of the utterances used to train that model, so that each speech frame is annotated with its most likely generating state. Next, we walk through this alignment, filling an urn for each state with its representative frames; at the end of this process, each urn is populated with frames representing its empirical distribution. To generate resampled data, we use the alignment model to create a forced alignment of the original test data, and then sample a frame (at random, with replacement) from the appropriate urn for each frame position; these resampled frames are concatenated. With this frame-level resampling, the pseudo test data is exactly the same length as the original, and has the same underlying alignment, but the frames are now conditionally independent.

By placing entire state regions—sequences of frames—in the urns, and then resampling (again, concatenating samples), we end up with pseudo test data with dependence among frames within state regions, but independence across state boundaries (note that resampling units larger than single frames produces pseudo test data that may be a different length from the original). We can further extend this idea to phonemes and to words; in both cases, the urn labels include the triphone context ("eh-N-ih" or "n-THE-f", for example) to respect the cross-word triphone structure of the models.

Table 3 shows the results of the resampling experiments. Pseudo test data is created by resampling frames, states, phonemes, and words, and then the two recognition models, ML and MPE, are used for decoding. Each resampling experiment is repeated five times and the mean WERs are shown in the table; the standard errors range from 0.01 (frame resampling) to 0.22 (word resampling), but all the WER differences

Test	WSJ WER ML	WSJ WER MPE	WSJ WER Change	SWB WER ML	SWB WER MPE	SWB WER Change	Independence Assumptions Satisfied $o_i \perp\!\!\!\perp o_j \mid s$
Resampled frames	0.5	0.9	+90%	4.5	5.9	+34%	yes
Resampled states	2.8	2.2	-20%	28.2	22.0	-22%	with i and j in different states
Resampled phones	6.2	4.8	-23%	42.1	32.3	-23%	with i and j in different phonemes
Resampled words	13.6	10.5	-23%	56.4	44.4	-21%	with i and j in different words
Original	15.2	11.2	-26%	61.5	50.2	-18%	no

Table 3. Recognition results with ML and MPE models when the test data is assembled by resampling.

between recognition models are highly significant.

Perhaps the most striking thing about the resampling results is how small the error rates are when all the independence assumptions are satisfied by the data; this is addressed at length in [5]. However, here we are most interested in the relative differences between ML and MPE. There are a few important points:

First, we observe that with test data assembled by resampling frames, the ML-trained models outperform the MPE-trained models, a result consistent with the frame classification experiment (above). In removing all dependence among frames, we also seem to have removed any advantage that the MPE models had over the ML models.

Second, we observe that when we instead resample whole states, MPE outperforms ML by about 20%. What have we changed about the test data in switching from frame resampling to state resampling? We have introduced within-state dependence among frames. Thus, this result strongly suggests that MPE is winning by compensating for dependence at the sub-phone state level.

Third, we observe that as we continue to add dependence, resampling phonemes, words, and finally running recognition on the original data, the relative advantage of MPE over ML does not change very much. This suggests that the within-state dependence is the primary issue addressed by MPE; we cannot however, rule out the possibility that MPE compensates for dependence across phonemes or words.

5. DISCUSSION

Our experiments show that (1) MPE does not improve on ML in modeling sub-phone state output distributions, and (2) MPE appears to compensate for statistical dependence, in particular within states. While discriminative training may also improve on ML in other ways, this is a fairly indirect method for temporal dynamics. If we can understand a little more about how MPE adjusts for dependence—perhaps by normalizing phoneme or state-level scores—we might be able to benefit by modeling dependence directly. Such insight would be especially valuable given the halting progress of segmental modeling techniques that aim to model such dependence.

6. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1015930.

7. REFERENCES

[1] A. Nadas, "A decision theorectic formulation of a training problem in speech recognition and a comparison of training by unconditional versus conditional maximum likelihood," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 31, no. 4, pp. 814–817, 1983.

[2] L. Bahl, P. Brown, P. De Souza, and R. Mercer, "Maximum Mutual Information Estimation of Hidden Markov Model Parameters for Speech Recognition," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'86.* IEEE, 1986, vol. 11, pp. 49–52.

[3] P.C. Woodland and D. Povey, "Large scale discriminative training of hidden markov models for speech recognition," *Computer Speech & Language*, vol. 16, no. 1, pp. 25–47, 2002.

[4] D. Povey and P.C. Woodland, "Minimum Phone Error and I-smoothing for Improved Discriminative Training," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*. IEEE, 2002, vol. 1, pp. I–105.

[5] D. Gillick, L. Gillick, and S. Wegmann, "Dont Multiply Lightly: Quantifying Problems with the Acoustic Model Assumptions in Speech Recognition," in *Proceedings of ASRU*, 2011.

[6] D. Povey, "Discriminative training for large vocabulary speech recognition," *Cambridge, UK: Cambridge University*, 2004.

[7] S.J. Young, G. Evermann, M.J.F. Gales, D. Kershaw, G. Moore, J.J. Odell, D.G. Ollason, D. Povey, V. Valtchev, and P.C. Woodland, "The HTK Book version 3.4," 2006.

[8] B. Gold, N. Morgan, and D. Ellis, *Speech and Audio Signal Processing – Processing and Perception of Speech and Music, Second Edition*, Wiley, 2011.