A Fast Online Algorithm for Large Margin Training of Continuous Density Hidden Markov Models

Chih-Chieh Cheng[1], *Fei Sha*[2], *Lawrence K. Saul*[1]

[1]Department of Computer Science and Engineering, University of California, San Diego
[2]Department of Computer Science, University of Southern California

chc028@cs.ucsd.edu, feisha@usc.edu, saul@cs.ucsd.edu

Abstract

We propose an online learning algorithm for large margin training of continuous density hidden Markov models. The online algorithm updates the model parameters incrementally after the decoding of each training utterance. For large margin training, the algorithm attempts to separate the log-likelihoods of correct and incorrect transcriptions by an amount proportional to their Hamming distance. We evaluate this approach to hidden Markov modeling on the TIMIT speech database. We find that the algorithm yields significantly lower phone error rates than other approaches—both online and batch—that do not attempt to enforce a large margin. We also find that the algorithm converges much more quickly than analogous batch optimizations for large margin training.

Index Terms: hidden Markov models, online learning, large margin classification, discriminative training, automatic speech recognition

1. Introduction

For nearly two decades, most state-of-the-art systems for automatic speech recognition (ASR) have relied at their core on the statistical framework provided by continuous density hidden Markov models (CD-HMMs) [1]. In many ways, the basic form of these models has not changed over time. However, researchers continue to experiment with new and improved methods for parameter estimation.

Recently, several researchers have proposed methods for large margin training of CD-HMMs [2, 3, 4, 5, 6, 7]. In large margin training, acoustic models are estimated to assign significantly higher scores to correct transcriptions than competing ones; in particular, the margin between these scores may be required to grow in proportion to the total number of recognition errors [3, 6, 7]. Empirically, large margin training has improved the performance of many systems beyond other leading discriminative approaches.

Large margin training in CD-HMMs has the same basic computational requirements as other discriminative approaches. The updates depend on computing statistics of hidden states as well as gradients with respect to various model parameters. For each update, these quantities must be computed and accumulated over all the utterances in the training corpus. To cope with large corpora, researchers often parallelize this batch computation across many different nodes, then combine the individual results as needed to average over all the training utterances.

In this paper, we investigate a different, simpler approach for accelerating large margin training of CD-HMMs. We replace the batch computation described above by an *online, sequential* computation. Specifically, we optimize the CD-HMM parameters in an incremental fashion, updating them after the decoding of each training utterance. We find that this approach converges much more quickly than our own batch optimizations of large margin CD-HMMs. We also find that it yields significantly more accurate acoustic models than other approaches—both online and batch—that do not attempt to enforce a large margin [8].

The paper is organized as follows. In section 2, we describe our online algorithm for large margin training and contrast it with competing approaches. In section 3, we present results on the TIMIT speech database; to highlight the gaps in performance between differently trained acoustic models, we report frame and phone error rates in the absence of a language model. Finally, in section 4, we present our conclusions and ideas for future work.

2. Model

2.1. Background and previous work

In ASR, we seek to model the joint distributions $\mathcal{P}(\mathbf{s}, \mathbf{x})$ over sequences of hidden (phonetic) states $\mathbf{s} = \{s_1, s_2, \cdots, s_T\}$ and acoustic observations or feature vectors $\mathbf{x} = \{x_1, x_2, \cdots, x_T\}$. In continuous density hidden Markov models (CD-HMMs), the distribution is parameterized with an initial state distribution $\mathcal{P}(s_1)$, state transition probabilities $\mathcal{P}(s_{t+1}|s_t)$, and emission densities $\mathcal{P}(x_t|s_t)$. Concretely,

$$\mathcal{P}(\mathbf{s}, \mathbf{x}) = \mathcal{P}(s_1) \prod_{t=1}^{T-1} \mathcal{P}(s_{t+1}|s_t) \prod_{t=1}^{T} \mathcal{P}(x_t|s_t). \quad (1)$$

The emission densities for ASR are usually computed from Gaussian mixture models (GMMs). For each state s and mixture component c, we denote the mean vector by μ_{sc} and the covariance matrix by Σ_{sc}. Using this notation, the emission densities are given by:

$$\mathcal{P}(x_t|s_t) = \sum_c \mathcal{P}(c|s) \mathcal{N}(\mu_{sc}, \Sigma_{sc}), \quad (2)$$

where $\mathcal{N}(\mu, \Sigma)$ denotes a multivariate Gaussian and $\mathcal{P}(c|s)$ denote mixture weights. The parameters in CD-HMMs are most easily estimated using an Expectation-Maximization (EM) algorithm. In this context, the EM algorithm attempts to maximizes the log-likelihood of observation sequences.

Because maximum likelihood estimation (MLE) does not directly optimize the error rates for ASR, many researchers have developed so-called discriminative methods for parameter estimation [9, 10, 11, 12]. Unlike MLE, these methods

directly seek to minimize the number of classification errors. Though discriminative training generally yields better performance for ASR, it also requires much more computation. The extra computation arises for two reasons. First, for each training utterance, discriminative methods must compute likelihoods not only for the target transcription, but also for all incorrect transcriptions that may have possibly higher likelihoods. Second, many update rules for discriminative training require fine-tuning of one or more learning rates. They do not converge as quickly as the EM algorithm for MLE in practice.

2.2. Online Training

In this work we explore a mistake-driven, online algorithm for discriminative training of CD-HMMs. Our approach builds on earlier work on perceptron training of discrete HMMs [13] and CD-HMMs [8]. We briefly review the latter before considering its extension to large margin training in the next section. Our earlier work [8] began by defining a discriminant function over observation and state transition sequences:

$$\mathcal{D}(\mathbf{x}, \mathbf{s}) = \log \mathcal{P}(s_1) + \sum_{t=1}^{T-1} \log \mathcal{P}(s_{t+1}|s_t) + \sum_t \log \mathcal{P}(x_t|s_t). \tag{3}$$

The discriminant function in eq. (3) is simply the logarithm of the joint distribution in eq. (1). Let $\mathbf{y}$ denote the correct transcription of the observation sequence $\mathbf{x}$. For correct recognition, we require that

$$\forall \mathbf{s} \neq \mathbf{y}, \quad \mathcal{D}(\mathbf{x}, \mathbf{y}) > \mathcal{D}(\mathbf{x}, \mathbf{s}); \tag{4}$$

note that eq. (4) defines a set of inequalities for all incorrect transcriptions $\mathbf{s} \neq \mathbf{y}$. We use $\mathbf{s}^*$ to denote the optimal decoding

$$\mathbf{s}^* = \text{argmax}_{\mathbf{s}} \mathcal{D}(\mathbf{x}, \mathbf{s}), \tag{5}$$

which can be efficiently computed by the Viterbi algorithm. Let Θ denote the parameters of the CD-HMM, and let $(\mathbf{x}_n, \mathbf{y}_n)$ denote the acoustic observations and (ground truth) hidden state transcriptions of the nth training utterance. (We assume these transcriptions to be known.) In our earlier work, we updated the CD-HMM parameters using the online learning rule:

$$\Theta \leftarrow \Theta + \eta \frac{\partial}{\partial \Theta} \left[\mathcal{D}(\mathbf{x}_n, \mathbf{y}_n) - \mathcal{D}(\mathbf{x}_n, \mathbf{s}_n^*) \right], \tag{6}$$

where $\eta > 0$ was a carefully chosen learning rate. The update in eq. (6) attempts to close the gap between $\mathcal{D}(\mathbf{x}_n, \mathbf{y}_n)$ and $\mathcal{D}(\mathbf{x}_n, \mathbf{s}_n^*)$ whenever an error occurs in recognition. In practice, training utterances are often processed in random order. Furthermore, multiple passes are made through the training corpus with each utterance being presented once in each "pass".

In general, this mistake-driven approach will not converge to a fixed set of parameters. However, convergence to a fixed set can be obtained by averaging parameters across different updates of eq. (6); the averaging also gives a better result after a finite number of iterations through the training set [14, 15]. The averaged parameter estimates after r updates are given by:

$$\hat{\Theta}^{(r)} = \frac{1}{r} \sum_{j=1}^{r} \Theta^{(j)}. \tag{7}$$

Note that this averaging does not affect training process: it only affects the parameters used for evaluating the model on held-out data. In our earlier work, we applied the online update in eq. (6) and the parameter averaging in eq. (7), looping through the training utterances until the CD-HMM (with averaged parameters) reached its minimum error rate on a held-out set.

2.3. Large margin training

Large margin training of CD-HMMs seeks not only to minimize the empirical error rate, but also to separate the scores of correct and incorrect transcriptions by the largest possible amount, thus achieving better generalization on unseen data. This idea has been independently investigated by many researchers in ASR [2, 3, 4, 5]. Our main contribution in this work is to investigate a simple, online method for large margin training of CD-HMMs.

Let $(\mathbf{x}, \mathbf{y})$ denote an observation sequence and its ground truth transcription. The essence of large margin training lies in the following observation: whereas for correct recognition we merely require the inequalities in eq. (4), for correct recognition *by a large margin*, we additionally require that

$$\forall \mathbf{s} \neq \mathbf{y}, \quad \mathcal{D}(\mathbf{x}, \mathbf{y}) > \mathcal{D}(\mathbf{x}, \mathbf{s}) + \rho \mathcal{H}(\mathbf{s}, \mathbf{y}), \tag{8}$$

where $\mathcal{H}(\mathbf{s}, \mathbf{y})$ is the Hamming distance between two hidden state sequences of the same length, and $\rho > 0$ is a constant margin scaling factor. In other words, for large margin training, the score of the correct transcription should exceed the score of any incorrect transcription by an amount that grows in proportion to the number of recognition errors.

We can use dynamic programming to compute the hidden state sequence that most egregiously violates the margin constraint in eq. (8). We use $\tilde{\mathbf{s}}^*$ to denote this hidden state sequence. From eq. (8), we have:

$$\tilde{\mathbf{s}}^* = \text{argmax}_{\mathbf{s}} \left[\mathcal{D}(\mathbf{x}, \mathbf{s}) + \rho \mathcal{H}(\mathbf{s}, \mathbf{y}) \right]. \tag{9}$$

The right hand side of eq. (9) can be maximized by a simple variant of the standard Viterbi algorithm [16].

For online training of large margin CD-HMMs, we consider the following update rule:

$$\Theta \leftarrow \Theta + \eta \frac{\partial}{\partial \Theta} \left[\mathcal{D}(\mathbf{x}_n, \mathbf{y}_n) - \mathcal{D}(\mathbf{x}_n, \tilde{\mathbf{s}}_n^*) \right]. \tag{10}$$

Eq. (10) differs from eq. (6) in one critical aspect: namely, we replace the usual Viterbi sequence $\mathbf{s}_n^* = \text{argmax}_{\mathbf{s}} \mathcal{D}(\mathbf{x}_n, \mathbf{s})$ by the sequence $\tilde{\mathbf{s}}_n^* = \text{argmax}_{\mathbf{s}} \left[\mathcal{D}(\mathbf{x}_n, \mathbf{s}) + \rho \mathcal{H}(\mathbf{y}_n, \mathbf{s}) \right]$ from margin-based decoding. Though the margin scaling factor ρ does not appear explicitly in eq. (10), it directly affects the computation of $\tilde{\mathbf{s}}_n^*$. In fact, our experiments in section 3 will show that the subtle change in eq. (10) leads to profoundly different updates. To obtain smoother parameter estimates over time, the results from eq. (10) can also be averaged as in eq. (7). We performed this averaging in all of our experiments.

2.4. Parameterization

The online updates in eqs. (6) and (10) are written in terms of the parameters Θ of the CD-HMM. In this paper, we adopt a particular parameterization of CD-HMMs that has proven useful in earlier work [8, 17]. Also, in all our experiments, we only adapt the parameters of the GMMs, not the transition probabilities of the CD-HMMs. The latter generally play a less significant role in ASR; moreover, we have found that they are easily over-trained.

We briefly review the parameterization for GMMs described in earlier work [8, 17]. A single Gaussian distribution $\mathcal{P}(\mathbf{x}|\mu, \mathbf{\Sigma})$ is conventionally parameterized in terms of its mean μ and covariance matrix $\mathbf{\Sigma}$. Let $\gamma = -\log[1/(2\pi)^d|\Sigma]$ denote the log of the scalar prefactor that normalizes the distribution. In terms of these parameters, we consider the matrix:

$$\Phi = \begin{bmatrix} \Sigma^{-1} & -\Sigma^{-1}\mu \\ -\mu^\top \Sigma^{-1} & \mu^\top \Sigma^{-1} \mu + \gamma \end{bmatrix}. \tag{11}$$

Note that in terms of this matrix, we can write the Gaussian distribution as:

$$\mathcal{P}(\mathbf{x}|\mu, \mathbf{\Sigma}) = e^{-\frac{1}{2}\mathbf{z}^\top \Phi \mathbf{z}} \quad \text{where} \quad \mathbf{z} = \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}. \tag{12}$$

By a further reparameterization, we can also make explicit that the matrix Φ should be positive semidefinite. To this end, we consider the matrix factorization:

$$\Phi = \Lambda\Lambda^\top, \tag{13}$$

where the matrix Λ is the same size as the matrix Φ. In practice, we do not constrain the matrix Λ to preserve the normalization of the Gaussian distribution. However, normalized Gaussians are not needed to interpret CD-HMMs as discriminative models.

For online learning in CD-HMMs, we parameterize the Gaussian in each state s and mixture component c (after absorbing the mixture weight $\log \mathcal{P}(c|s)$ into γ) by a square matrix Λ_{sc}. We then update the parameters $\Theta = \{\Lambda_{sc}\}$ over all states and mixture components using the rules in eqs. (6) and (10). As in earlier work [8], we obtain smoothed parameter estimates (for testing on held-out data) by averaging the matrices $\Phi_{sc} = \Lambda_{sc}\Lambda_{sc}^\top$ over time using eq. (7).

3. Experiments

We performed experiments on the TIMIT speech corpus [18], whose signals have been manually segmented and aligned with phonetic transcriptions. We adopted the same front end as recent benchmarks for phone recognition on this data set [17]. We computed 39-dimensional acoustic feature vectors of mel-frequency cepstral coefficients on sliding windows of speech. We also followed the standard partition of the TIMIT corpus, yielding roughly 1.1 million, 120K, and 57K frames respectively for training, test, and holdout data.

We built recognizers using monophone CD-HMMs in which each of 48 states represented a context-independent phoneme. We experimented with models of different sizes by varying the number of Gaussian mixture components in each state. We evaluated the performance of each CD-HMM by comparing the hidden state sequences inferred by Viterbi decoding to the "ground-truth" phonetic transcriptions provided by the TIMIT corpus. We report two types of errors: the frame error rate (FER), computed simply as the percentage of misclassified frames, and the phone error rate (PER), computed from the edit distances between ground truth and Viterbi decodings. In calculating the errors, we follow the standard of mapping 48 phonetic classes down to broader 39 categories [19]. The performance of our baseline models with maximum likelihood estimation is similar to those previously reported [3, 8].

All CD-HMMs were initialized by maximum likelihood estimation. Starting from these baseline CD-HMMs, we then compared the performance of the different online updates in eq. (6) and (10). For the margin-based update, the results of training depend on the margin scaling factor ρ. We experimented with a wide range of values for this scaling factor.

Table 1 shows the results from the best models trained in this way. (For the margin-based results, we chose the scaling factor ρ that yielded the lowest phone error rates on the held-out development set.) The results show that online learning with margin-based decoding significantly reduces the frame and phone error rates across all model sizes. The results for online learning with margin-based decoding are also comparable or better than previously published benchmarks for batch implementations of discriminative training on this task [17]. In

# mixture component	Frame Error Rate (%)		
	Maximum likelihood	Online w/o margin	Online w/ margin
1	39.3	30.0	28.3
2	37.1	27.6	26.5
4	31.4	26.0	25.0
8	28.1	26.5	25.0

# mixture component	Phone Error Rate (%)		
	Maximum likelihood	Online w/o margin	Online w/ margin
1	42.0	35.2	33.5
2	38.6	33.2	31.8
4	34.8	31.2	30.3
8	32.5	31.9	30.2

Table 1: Frame error rates (*top*) and phone error rates (*bottom*) on the TIMIT test set for CD-HMMs of varying size, as obtained by maximum likelihood (ML) estimation, online training, and online training with margin-based decoding.

general, the frame error rates improve more than the phone error rates; this discrepancy reflects the fact that the margin-based updates more closely track the Hamming distance (not the edit distance) between target and Viterbi phone sequences.

While Table 1 quantifies the effects of margin-based decoding on error rates, Fig. 1 graphically illustrates the profound influence it exerts during training. To create this figure, we computed the Hamming distance between the Viterbi decoding $\mathbf{s}^*$ in eq. (5) and the margin-based decoding $\tilde{\mathbf{s}}^*$ in eq. (9) for each utterance during one online pass through the training corpus. The figure shows a histogram of these Hamming distances after they have been normalized by the number of frames in the utterance. The histogram's peak away from zero shows that margin-based decoding yields very different competing transcriptions for discriminative training than standard Viterbi decoding.

The frame and phone error rates from large margin training depend on the value of the margin scaling factor ρ. Fig. 2 shows this dependence for CD-HMMs with 4-component GMMs in each state. More generally, for phone error rates on the development set, the optimal values of ρ were respectively 0.8, 1.0, 0.7, and 1.0 for CD-HMMs with 1, 2, 4, and 8-component GMMs. Training with $\rho = 0$ (i.e., without margin-based decoding) produces the results shown in the middle columns of Table 1.

Finally, Fig. 3 illustrates the fast convergence of online training. The figure shows the frame error rates on the devel-

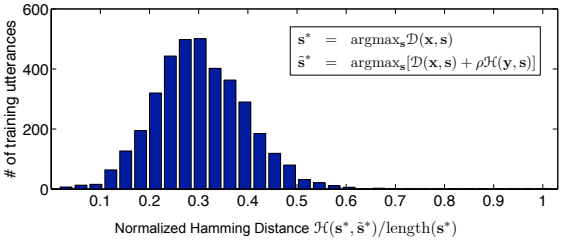

Figure 1: Histogram of normalized Hamming distances between sequences from Viterbi and margin-based decoding. The distances were computed during the fifth iteration through the training corpus for the large margin CD-HMM with two Gaussian mixture components.

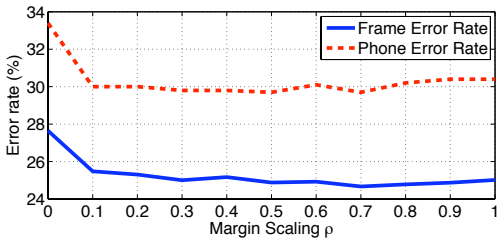

Figure 2: Frame and phone error rates on the development set as a function of the margin scaling factor ρ. Results are shown for CD-HMMs with 4-component GMMs.

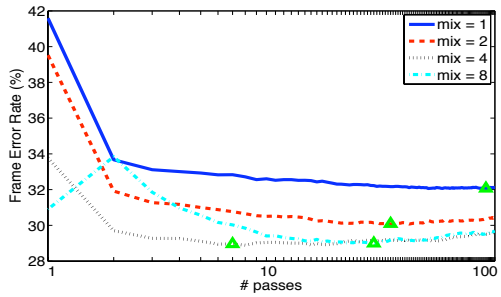

Figure 3: Frame error rates on the development set during training. The triangles mark the best models obtained for different numbers of Gaussian mixture components.

opment data set during training. For all model sizes, most of the improvement from discriminative training occurs during the first 10-20 passes through the training corpus.

4. Discussion

Online learning is an active area of research in machine learning [20, 21]. Our main contribution in this work lies in adapting various recent approaches [13, 8] to large margin training of CD-HMMs. On TIMIT phoneme recognition, we have shown that our approach is effective and efficient, not only attaining better error rates than standard batch algorithms [17], but also speeding up training time significantly. Anecdotally, we have attained similar performance as our own batch implementation of large margin training in roughly one third of the training time.

Scaling our approach to large vocabulary ASR presents several challenges. Online algorithms tend to update parameters very aggressively, thus exploring the parameter space more quickly than batch algorithms but also exhibiting larger variance on consecutive updates. Future work will explore how to balance these tendencies. One possible strategy is to chunk large amounts of data into small subsets, then to update the model parameters using statistics on subsets as opposed to individual utterances. This "minibatch" scheme lends itself naturally to parallelization since the computations on subsets of utterances can be distributed across multiple machines. Within this approach, however, further research is needed to determine the optimal subset size. We are actively working in these directions.

5. Acknowledgement

This work was supported by NSF Award 0812576. Fei Sha is partially supported by Charles Lee Powell Foundation.

6. References

[1] X. Huang, A. Acero, and H.-W. Hon, *Spoken language processing*. Prentice-Hall, 2001.

[2] H. Jiang, X. Li, and C. Liu, "Large margin hidden markov models for speech recognition," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1584–1595, 2006.

[3] F. Sha and L. K. Saul, "Comparison of large margin training to other discriminative methods for phonetic recognition by hidden Markov models," in *Proceedings of ICASSP 2007*, Honolulu, HI, 2007, pp. 313–316.

[4] J. Li, M. Yuan, and C. Lee, "Approximate test risk bound minimization through soft margin estimation," *IEEE Trans. on Speech, Audio and Language Processing*, vol. 15, no. 8, pp. 2392–2404, 2007.

[5] D. Yu, L. Deng, X. He, and A. Acero, "Large-margin minimum classification error training for large-scale speech recognition tasks," in *Proc. of ICASSP 2007.*, vol. 4, 2007, pp. IV–1137–IV–1140.

[6] D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariahy, "Boosted MMI for model and feature-space discriminative training," in *ICASSP 2008*, 2008.

[7] G. Saon and D. Povey, "Penalty function maximization for large margin HMM training," in *Interspeech 2008*, 2008.

[8] C. C. Cheng, F. Sha, and L. K. Saul, "Matrix updates for perceptron training of continuous density hidden markov models," in *Proc. of ICML*, 2009.

[9] L. R. Bahl, P. F. Brown, P. V. de Souza, and R. L. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," in *Proc. of ICASSP*, Tokyo, 1986, pp. 49–52.

[10] B.-H. Juang and S. Katagiri, "Discriminative learning for minimum error classification," *IEEE Trans. Sig. Proc.*, vol. 40, no. 12, pp. 3043–3054, 1992.

[11] P. C. Woodland and D. Povey, "Large scale discriminative training for speech recognition," in *Proc. ISCA ITRW ASR2000*, 2000.

[12] D. Povey and P. Woodland, "Minimum phone error and i-smoothing for improved discriminative training," in *Proc. ICASSP 2002*, Orlando, FL, 2002, pp. 105–108.

[13] M. Collins, "Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms," in *EMNLP 2002*, 2002.

[14] Y. Freund and R. E. Schapire, "Large margin classification using the perceptron algorithm," in *Machine Learning*, 1999, pp. 277–296.

[15] C. Gentile, "A new approximate maximal margin classification algorithm," *J. Mach. Learn. Res.*, vol. 2, pp. 213–242, 2002.

[16] F. Sha and L. K. Saul, "Large margin hidden markov models for automatic speech recognition," in *Advances in Neural Information Processing Systems 19*, B. Schölkopf, J. Platt, and T. Hoffman, Eds. Cambridge, MA: MIT Press, 2007, pp. 1249–1256.

[17] ——, "Large margin training of continuous density hidden markov models," in *Automatic Speech and Speaker Recognition: Large Margin and Kernel Methods*, J. Keshet and S. Bengio, Eds. Wiley-Blackwell, 2009.

[18] L. F. Lamel, R. H. Kassel, and S. Seneff, "Speech database development: design and analysis of the acoustic-phonetic corpus," in *Proceedings of the DARPA Speech Recognition Workshop*, L. S. Baumann, Ed., 1986, pp. 100–109.

[19] K. F. Lee and H. W. Hon, "Speaker-independent phone recognition using hidden markov models," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 37, pp. 1641–1648, 1988.

[20] L. Bottou and Y. LeCun, "Large scale online learning," in *Advances in Neural Information Processing Systems 16*. Cambridge, MA: MIT Press, 2004.

[21] L. Bottou and O. Bousquet, "The tradeoffs of large scale learning," in *Advances in Neural Information Processing Systems*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. MIT Press, 2008, vol. 20, pp. 161–168.