HYBRID DNN-LATENT STRUCTURED SVM ACOUSTIC MODELS FOR CONTINUOUS SPEECH RECOGNITION

Suman Ravuri

International Computer Science Institute, Berkeley, CA
University of California - Berkeley, Berkeley, CA

ABSTRACT

In this work, we propose Deep Neural Network (DNN)-Latent Structured Support Vector Machine (LSSVM) Acoustic Models as replacement for more standard sequence-discriminative trained DNN-HMM hybrid acoustic models. Compared to existing methods, approaches based on margin maximization, as is considered in this work, enjoy better theoretical justification. In addition to a max-margin based criteria, we also extend the Structured SVM model to include latent variables in the model to account for uncertainty in state alignments. Introducing latent structure allows for better sample complexity, often requiring 33% to 66% fewer utterances to converge compared to alternate criteria. On an 8-hour independent test set of conversational speech, the proposed method decreases word error rate by 9% relative to a cross-entropy trained hybrid system, while the best existing system decreases the word error rate by 6.5% relative.

Index Terms— Structured SVM, Deep Learning, Sequence-Discriminative Training, Large Margin, Acoustic Modeling

1. INTRODUCTION

Statistical speech recognition reposes on the assumption that a word sequence W and its associated acoustics O is a stochastic process distributed according to $O, W \sim P_{true}(O, W)$. Modern Automatic Speech Recognition (ASR) systems attempt to model this process with $P_M(O, W)$, typically comprising four major components: neural networks for frame-level triphone classification, Hidden Markov Models (HMMs) for state-level sequence classification, a lexicon for phone-to-word transduction, and a language model that estimates the likelihood of word sequences. An application of elementary probability theory allows us to combine these separate models:

$$\begin{aligned}\hat{W} &= \operatorname*{argmax}_W P_M(W|O) = \operatorname*{argmax}_W P_M(O|W)P_M(W) \\ &= \operatorname*{argmax}_W \sum_S P_M(O, S|W)P_M(W) \\ &= \operatorname*{argmax}_W \sum_S P_M(O|S)P_M(S|W)P_M(W) \\ &\approx \operatorname*{argmax}_{W,S} P_M(O|S)P_M(S|W)P_M(W) \\ &= \operatorname*{argmax}_{W,S\in\mathcal{S}_W} P_M(O|S)P_M(W)\end{aligned}$$

where S denotes the state sequence and $P_M(S|W)$ the lexicon.[1] Since in general the typical dictionaries only define allowable phone sequences without more detailed relative probabilities, the decode equation in the final line searches over $\mathcal{S}_W$, the set of states consistent with word sequences.

If $P_M(O, W)$ could accurately model $P_{true}(O, W)$, then independent training of each of these separate components, and Minimum Bayes Risk (MBR) decoding [1] would likely yield an accurate recognition. HMM modeling assumptions, however, are rather poor: [2] showed that if the data were actually distributed according to the HMM modeling assumptions, word error rates would drop from 30-60% to 1-5% even with weak frame-level classification and suboptimal MAP decoding (which minimizes expected sentence instead of expected word error rate). Moreover, since MBR decoding presupposes accurate estimates of $P_{true}(W|O)$, it is perhaps not surprising that implementing Minimum Bayes Risk decoding with model posteriors more modestly improves recognition performance than expected.

Many of the standard fixes used to improve word recognition performance, such as raising language model scores by a scaling factor in the exponent, violate the traditional rules of probability while partially fixing poor modeling assumptions. As mentioned by [3], the result is that the full decoding model more resembles a log-linear model. Denoting h_t to be the last hidden layer of a Deep Neural Network (DNN) system (augmented by 1 to accommodate a bias term), α_s as the logistic regression layer weights for state s (augmented by $b_s - \log P(s)$, the bias with the log prior of state s subtracted), the model transition log-probabilities α_{s_{i-1},s_i}, and the language model scaling factor α_{lmsf}, we can more accurately represent the decoding problem $\operatorname{argmax}_{W,S\in\mathcal{S}_W} \log P_M(O|S) +$

[1] The lexicon defines phone, not state, sequences, but the two are trivially related.

$\log P_M(W)$ as:

$$
\begin{aligned}
&= \underset{W,S_i \in \mathcal{S}_W}{\operatorname{argmax}} \sum_i (\log P_M(O_i|S_i) + \log P_M(S_i|S_{i-1})) \\
&\qquad + \alpha_{lmsf} \log P_M(W) \\
&= \underset{W,S \in \mathcal{S}_W}{\operatorname{argmax}} \sum_i (\alpha_{s_i}^\intercal h_{t_i} + \alpha_{s_{i-1},s_i}) + \alpha_{lmsf} \log P_M(W)
\end{aligned}
$$

Even this log-linear model does not accurately model P_{true}, so let us discard the probabilistic interpretation – i.e., we keep the same model parameters as before but eschew the notion that decoding scores are representative of probabilities – and ask a slightly different question: for model parameters α in model family $\mathcal{A}$, what parameters will minimize our true risk

$$\min_{\alpha \in \mathcal{A}} \mathcal{R} \equiv \min_{\alpha \in \mathcal{A}} E_{P_{true}(W,O)}[\mathcal{L}(\hat{W}, W)]$$

where $\mathcal{L}$ is the word error rate. Here risk corresponds to the expected word error from a random test set.

While directly optimizing for risk is likely difficult for a finite training set (though [4] showed theoretically that one could use perceptron-like update to obtain an exact loss gradient on an "infinite" training set), there exist methods which minimize surrogate objectives that also give nice theoretical guarantees. Structured Support Vector Machines (SVMs) provide one such method for linear models, providing theoretical guarantees that true risk is not much higher than training set error (see [5] for a typical proof). Such a guarantee assumes that the input features to the Structured SVM are bounded, which is trivially true for hidden units with sigmoid non-linearities as used in this work.[2]

2. RELATED WORK

Max-margin methods are not the only way to approach approximate optimization of Bayes risk, and sequence-discriminative training criteria have long attempted to minimize the risk equation through different approximations. The minimum phone error (MPE) [6] and state-level minimum Bayes risk (sMBR) [7, 8] criteria directly try to optimize for the risk through the approximation:

$$\underset{\alpha \in \mathcal{A}}{\operatorname{argmin}} \mathcal{R} \approx \underset{\alpha \in \mathcal{A}}{\operatorname{argmax}} E_{P_{Emp}(O)} E_{P_{Model}(W|O)}[\mathcal{P}(\hat{S}, S)]$$

where S are phones for MPE and triphone states for sMBR, and the raw accuracy $\mathcal{P}$ is the number of correct units minus the number of insertions, calculated without substitutions or deletions for efficiency purposes. Maximum Mutual Information (MMI) [9] and boosted MMI [10] make somewhat different approximation:

$$E_{Emp(O,W)}[\log(1+ \sum_{\hat{W} \neq W} \exp(-(b\mathcal{P}(\hat{S}, S) + \log \frac{P_M(W|O)}{P_M(\hat{W}|O)}))]$$

which substitutes empirical risk for true risk, and a log-loss for true loss. Boosted MMI uses a soft margin, inspired by the work of [11], who applied large margin Gaussian Mixture Models (GMMs) to phoneme recognition. To the best of our knowledge, neither of these approximations have theoretical guarantees on test set error.

There have been some more recent attempts to include Structured SVM criteria – first introduced in [12] and later extended by [13] – into speech recognition: [14] augments the standard ASR model with per-phone acoustic model scaling factors learned through a cutting-plane algorithm, while more recent work on hybrid systems attempt to learn the output and transition model parameters using a frame-based loss [15], showing an improvement over cross-entropy trained neural networks on TIMIT phone recognition. There have also been attempts to incorporate Structured SVM criteria into segmental ASR models: see [16] for a comparison of different segmental models under different loss functions. Finally, [17] directly incorporated margin-terms into MMI and MPE criteria for a hidden CRF extension to GMMs, but were unable significant improve upon results of the MPE baseline on a large-vocabulary recognition task.

3. LATENT STRUCTURED SVM HYBRID ACOUSTIC MODELS

To connect speech recognition to Structured SVMs, note that the log-linear speech recognition model can be compactly expressed as:

$$\log p(W|O) = \alpha^\intercal \phi(h, W)$$

where α comprises the model parameters, $\mathbf{h} = (h_n^{(1)}, \ldots, h_n^{(t)})$ constitutes the acoustic observations in the form of the sequence of final hidden layer activations, and feature function $\phi(h, W) \in \mathbb{R}^n$ encodes information about the features and the structure of the model.

In the concrete example of a hidden Markov Support Vector Machine (HMSVM) [18], α includes α_k, defining the hyperplane associated with class k, and $\alpha_{i,j}$, parameterizing the state transitions, while $\phi(\mathbf{x}, \mathbf{y})$ effectively defines a hidden Markov model via indicator functions that select the appropriate terms from α:

$$\alpha = \begin{bmatrix} \alpha_{\mathbf{1}} \\ \cdots \\ \alpha_{\mathbf{k}} \\ \alpha_{11} \\ \alpha_{12} \\ \cdots \\ \alpha_{kk} \end{bmatrix} \quad \phi(\mathbf{x}, \mathbf{y}) = \begin{bmatrix} \sum_i^N x_i 1_{y_i=1} \\ \cdots \\ \sum_i^N x_i 1_{y_i=k} \\ \sum_{i=2}^N 1_{y_{i-1}=1, y_i=1} \\ \sum_{i=2}^N 1_{y_{i-1}=1, y_i=2} \\ \cdots \\ \sum_{i=2}^N 1_{y_{i-1}=k, y_i=k} \end{bmatrix}$$

[2] One can also make similar claims about rectified linear units, assuming that the norm of each row of the pre-nonlinearity weight matrix W in calculation Wx is bounded. The less-discussed aspect of dropout constraints exactly this norm.

Thus, to obtain the decoding score for labeling $\mathbf{y} = (4, 5, 6)$ of input features $\mathbf{x} = (x_1, x_2, x_3)$, one simply performs the dot product [3]:

$$\alpha^\intercal \phi(\mathbf{x}, (4, 5, 6)) = \alpha_4^\intercal x_1 + \alpha_5^\intercal x_2 + \alpha_6^\intercal x_3 + \alpha_{4,5} + \alpha_{5,6}$$

For more examples of this formalism, please see [13].

SVMs (and structured extensions) seek to minimize:

$$\min_{\alpha \in \mathcal{A}} E_{P_{true}(W,O)}[\mathcal{L}(\hat{W}, W)]$$
$$\text{where} \quad \hat{W} = \operatorname*{argmax}_{\mathbf{W}} \alpha^\intercal \phi(\mathbf{h}, \mathbf{W})$$

by making the following three approximations: replacing true with empirical risk $\mathcal{R} \approx \frac{1}{N}\sum_i \mathcal{L}(\hat{W}, W)$, upper-bounding the loss $\mathcal{L}(\hat{W}, W)$ which is not differentiable with respect to α with sub-differentiable hinge loss $[\mathcal{L}(\hat{W}, W) + \alpha^\intercal(\phi(h, \hat{W}_i) - \phi(h, W_i^*))]_+$[4], and keeping $\|\alpha\|$ small to limit generalization error. This leads to the (margin-rescaled) Structured SVM:

$$\min_{\alpha, \xi} \frac{\lambda}{2}||\alpha||^2 + 1^\intercal \xi$$
$$\text{s.t. } \forall i, \hat{W}_i \neq W_i$$
$$\alpha^\intercal(\phi(h, W_i) - \phi(h, \hat{W}_i)) \geq \mathcal{L}(W_i, \hat{W}_i) - \xi_i$$

Unfortunately, the current form of the equation cannot be applied to acoustic model training, as updating parameters requires a state-level alignment; resorting to a frame-level loss based on fixed state-level alignment does not provide a good reflection of word error rate.

Instead, we propose an extension to the Structured SVM which includes latent variables to describe the alignment. Note that the optimal alignment l_i^* can be calculated as:

$$l_i^* = \operatorname*{argmax}_{l_i \in \mathcal{W}_i^*} \alpha^\intercal \phi(h, W_i, l_i)$$

where $l_i \in \mathcal{W}_i$ is the set of alignments associated with the reference word sequence W_i. If loss is based on words, then this form is equivalent to the Latent Structural SVM first proposed in [19]. We will also consider other units for loss (denoted below as y_i instead of W_i) which depart from the formalism in that work.

Incorporating latent variables and recasting the constrained optimization as unconstrained:

$$L(\alpha, \theta) = \sum_i [\max_{\hat{y}_i, \hat{l}_i, \hat{W}_i} (\mathcal{L}(y_i, \hat{y}_i) + \alpha^\intercal \phi(h, \hat{W}_i, \hat{l}_i)) - \max_{l_i \in \mathcal{W}_i} \alpha^\intercal \phi(h, W_i, l_i)]_+ + \frac{\lambda}{2}||\alpha||^2$$

[3] Computing $\operatorname{argmax}_{\mathbf{y}} \alpha^\intercal \phi(\mathbf{x}, \mathbf{y})$ additionally requires efficient inference

[4] $[x]_+ = \max(x, 0)$

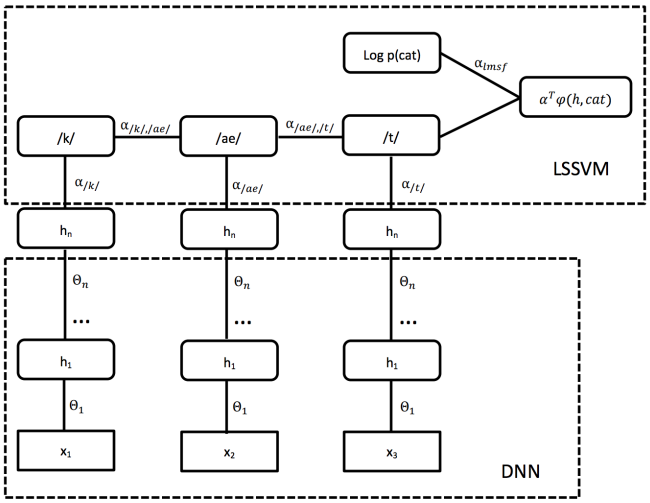

Fig. 1. *The figure represents the decode score for the word "cat" using monophone states.*

Here α consists of parameters of the output layer of the DNN, the language model scaling factor, and transition model parameters. ϕ corresponds to the structure of the log-linear model. In addition, we would also like to update the other parameters of the deep neural network θ using gradient descent. Fig. 1 illustrates a simplified model using monophone states. Algorithm 1 shows the full training procedure, employing stochastic sub-gradient descent for optimization.

This hybrid DNN-LSSVM model exhibits two nice properties, one practical and the other more theoretical. The computational advantage is that the sub-gradient is sparse – unlike gradients for other discriminative training criteria. To see this, define the loss-augmented alignment as $\phi(h, \bar{W}_i, \bar{l}_i) = \operatorname{argmax}_{\hat{y}_i, \hat{l}_i, \hat{W}_i}(\mathcal{L}(y_i, \hat{y}_i) + \alpha^\intercal \phi(h, \hat{W}_i, \hat{l}_i))$ and $\phi(h, W_i, l_i^*) = \operatorname{argmax}_{l_i \in \mathcal{W}_i} \alpha^\intercal \phi(h, W_i, l_i)$. Then if $\mathcal{L}(y_i^*, y_i) + \alpha^\intercal \phi(h, \bar{W}_i, \bar{l}_i) \geq \alpha^\intercal \phi(h, W_i, l_i^*)$, the subgradient (omitting, for clarity, the L2 penalty on α) is:

$$\nabla_\alpha L = \phi(h, \bar{W}_i, \bar{l}_i) - \phi(h, W_i, l_i^*)$$

otherwise it is 0.

For a particular frame, the error vector backpropagating through the output and earlier layers of the DNN contains only two non-zero values. Although not pursued in this work, for systems with a large number of context-dependent triphones (such as a recent state-of-the-art recognizer with 32k outputs[20]), exploiting the sparsity of this model could lead to large speed improvements: backpropagation of the error signal through the output layer requires subtraction of $2|H|$ vectors instead of a multiplication of $|O| \times |H|$ matrix with a $|H|$ vector (where $|O|$ and $|H|$ are the number of output and hidden units in the last hidden layer, respectively).

A more theoretical observation is that the boosting parameter b in boosted MMI is equivalent to L2 regularization pa-

rameter in the Latent SSVM framework. To see this, note that in the boosted optimization problem:

$$\min_{\alpha,\xi} \frac{\lambda}{2}||\alpha||^2 + 1^\intercal \xi$$
$$\text{s.t. } \forall i, \hat{W}_i \neq W_i$$
$$\alpha^\intercal(\phi(h, W_i, l_i^*) - \phi(h, \hat{W}_i, \hat{l}_i)) \geq b\mathcal{L}(y_i, \hat{y}_i) - \xi_i$$

dividing the constraints by b and making a transformation of variables $\alpha' = \frac{\alpha}{b}$ and $\xi' = \frac{\xi}{b}$ leads to the equivalent optimization problem:

$$\min_{\alpha',\xi'} \frac{b\lambda}{2}||\alpha'||^2 + 1^\intercal \xi'$$
$$\text{s.t. } \forall i, \hat{W}_i \neq W_i$$
$$\alpha^\intercal(\phi(h, W_i, l_i^*) - \phi(h, \hat{W}_i, \hat{l}_i)) \geq \mathcal{L}(y_i, \hat{y}_i) - \xi_i'$$

Since we use different regularization parameters for the Structured SVM parameters α and DNN parameters θ, we will use the boosting parameter b for the Latent Structured SVM parameters, and regularization constant λ for DNN parameters.

Algorithm 1 DNN-Latent SSVM training algorithm

1: Split training set T into K batches of size N, denoted $T_0 \dots T_k$
2: Set initial learning rate to β_0 and learning rate decrease to γ
3: **for** each i in $0 \dots k-1$ **do**
4: $\beta = \gamma^i \beta_0$
5: Calculate $l^* = \text{argmax}_{l \in \mathcal{Y}} \alpha^\intercal \phi(h, y_i^*, l)$ for each utterance in T_i
6: Generate N-Best List.
7: Calculate $\hat{y}_i, \hat{l}_i = \text{argmax}_{y_i, l_i} \mathcal{L}(y_i^*, y_i) + \alpha^\intercal \phi(h, y_i, l_i)$ for each i in batch
8: **if** $\mathcal{L}(y_i^*, y_i) + \alpha^\intercal \phi(h, \hat{y}_i, \hat{l}_i) \geq \alpha^\intercal \phi(h, y_i^*, l_i^*)$ **then**
9: $\nabla_{\alpha^{(t)}} L = \lambda\alpha + \frac{1}{k}\sum_{i=1}^{k}(\phi(h, \hat{y}_i, \hat{l}_i) - \phi(h, y_i^*, l_i^*))$
10: $\alpha^{(t+1)} \leftarrow \alpha^{(t)} - \beta \nabla_{\alpha^{(t)}} L$
11: $\theta^{(t+1)} \leftarrow \theta^{(t)} - \beta \nabla_{\theta^{(t)}} L$, where $\nabla_{\theta^{(t)}} L$ is the gradient with respect to the neural network parameters
12: **end if**
13: **end for**

3.1. Experiments

Given this framework, we would like to study five problems. The first is to determine which units of loss – frame-level, state-level, phone-level, or word-level – give us the best recognition. The latter three losses are measured as the number of substitutions plus deletions plus insertions, and do not need a raw accuracy approximation since we only need one loss-augmented alignment. Our second question is to understand how sensitive the models are to the boosting/regularization parameter. Third, since loss-augmented inference $\max_{\hat{y},\hat{W},\hat{l}} \mathcal{L}(y, \hat{y}) + \alpha^\intercal \phi(h, \hat{W}, \hat{l})$ currently uses an N-best list for search, we would like to see how the size of the N-best list affects recognition performance. Fourth, in initial experiments, we discovered that convergence of this model requires fewer utterances than other sequence-discriminative training criteria, which we wish to quantify. Finally, we would like to evaluate performance on an independent test without extra parameter tuning.

3.2. Connection to boosted MMI

Since the proposed method is not the first margin-inspired one, we would like to connect the SVM criterion to the more familiar boosted MMI. The analysis is similar to [17]. As a setup, define $G(\beta; \mathcal{B}) \equiv \log_\beta \sum_{b \in \mathcal{B}} \beta^b$ for $\beta > 1$. Note that $G(\beta; \mathcal{B}) \geq \max_{b \in \mathcal{B}} b$, is monotonically decreasing for increasing β, and $G(\beta; \mathcal{B}) \to \max_{b \in \mathcal{B}} b$ as $\beta \to \infty$. Also, note that raw phone accuracy is related to its loss by $\mathcal{L}(l^*, l) = |l| - \mathcal{P}(l^*, l)$, where $|l^*|$ is the number of frames. Defining $d\phi(h, l, l^*) \equiv \phi(h, l) - \phi(h, l^*)$, the boosted MMI criterion for one utterance in Structured SVM notation is:

$$\text{argmax}_{\alpha \in \mathcal{A}} \log \frac{\exp(\alpha^\intercal \phi(h, l^*))}{\sum_l \exp(\alpha^\intercal \phi(h, l) - b|l^*| + b\mathcal{L}(l^*, l))}$$
$$= \text{argmin}_{\alpha \in \mathcal{A}} \log(1 + \sum_{l \neq l^*} \exp(\alpha^\intercal d\phi(h, l, l^*) + b\mathcal{L}(l^*, l))))$$

Changing the bases of the natural logarithm and e to $\log_\beta$ and β, respectively, adding L2-regularization, and taking the limit as $\beta \to \infty$ recovers SVM criterion.

4. EXPERIMENTAL SETUP

4.1. Data and Language Model

We use the spontaneous portion of the ICSI meeting corpus [21], recorded with near-field microphones. The training set consists of 23,739 utterances – 20.4 hours – across 26 speakers. The training set is based on meeting data used for adaptation in the SRI-ICSI meeting recognizer [22]. The test set comprises 58 minutes of speech, taken from ICSI meetings portions of the NIST Rich Transcription Evaluation Sets 2002 [23], 2004 [24], and 2005 [25]. Previous work [2, 26, 27, 28] use this setup with an HTK recognizer, as described in [26].

4.2. Recognition System

We created a new Kaldi [29] recipe, adapted from the Switchboard System, to create relatively strong baseline systems, which we will make publicly available to encourage reproducible research.

GMM-HMM systems were trained using best-performing parameters of 2500 states and 40k Gaussians. Models were

initially trained on MFCC features with first and second derivatives. Then the GMM-HMM system was retrained using LDA+MLLT features, akin to the Switchboard setup. Finally, speaker-adaptive training (SAT) was performed using per-speaker feature-space maximum likelihood linear regression (fMLLR) transforms, which we refer to as LDA+MLLT+SAT.

Alignments from the GMM-HMM systems and the LDA+MLLT+SAT system were used to train the DNN models, using a 6-hidden-layer neural network with 2048 hidden units per layer, as these parameters produced the best results. Restricted Boltzmann Machine (RBM) pretraining [30] was performed until the final hidden layer, with each hidden layer using a sigmoid nonlinearity. Then cross-entropy training was performed using alignments from the GMM-HMM systems, which converged after 15 epochs.

We then updated the cross-entropy-trained DNN using four sequence-discriminative training models: MMI, boosted MMI, MPE, and sMBR. Some effort was made to ensure that each baseline sequence-discriminative training system was tuned for optimal performance. Each system converged after 3 epochs, with lattices regenerated after the first epoch of training. Neither more epochs of training, nor more lattice regeneration, produced better results on this corpus. For MMI, and bMMI, frames were dropped according to the standard recipe, and a boosting value of 0.05 gave best results. We also performed some initial experiments with L2-regularization, but this gave no benefit on the sequence-discriminative training systems. We also tuned learning rate, but found the optimal parameter to be the standard 0.00001.

For most experiments, the DNN-Latent Structured SVM system was trained on one sweep through the data, except for convergence and testing on independent test sets which were trained on two sweeps. The L2-regularization parameter on the weights was set to 0.0001. Since alignments in this framework are regenerated after every batch, we found that a much higher learning rate of 0.0002 could be used. Due to the aggressive step size, some utterances with poor alignments caused a temporary high bias to the silence phone: removing alignments which contained 1.5 seconds more silence than the "loss-augmented alignment" fixed this problem. This occurred for fewer than 1% of the utterances. Batch size was set to 512 utterances (after which alignments and N-best lists were generated for the following batch), and learning rate decay was set to 0.98, so that the learning rate at the end of the epoch was roughly half that at the beginning. N-best lists were generated from lattices using a unigram language model, akin to other sequence-discriminative training criteria, and the "loss-augmented alignment" was searched via an N-best list of size 1000 unless otherwise noted. DNN-Latent Structured SVM training used initial parameters from the cross-entropy trained DNN-HMM system.

We use a trigram language model (LM) [22] that was trained at SRI by interpolating a number of source LMs; these consisted of webtext and the transcripts of the following corpora: Switchboard, meetings (CMU, ICSI, and NIST), Fisher, Hub4-LM96, and TDT4. We renormalized the language model after removing words not present in the training dictionary. The perplexity of this meeting room LM is around 70 on our test set. To be compatible with the SRI LM, we use the SRI pronunciation dictionary, which includes two extra phones compared to the CMU phone set – "puh" and "pum" – to model hesitations.

5. RESULTS

Table 2 shows the the effect of loss and boosting parameters on ASR performance. In nearly every case (except for word loss with boosting parameter 1) the proposed systems beat the other sequence-discriminative training approaches, shown in Table 1. In particular, the best frame-level loss based system reduces error by 2.6% absolute compared to a cross-entropy trained baseline system, compared to 1.7% absolute with a state-level MBR trained system. The relative improvements of the system are in line with a comparative study of sequence-discriminative trained systems in [31].

Somewhat surprisingly, frame-level loss seems to outperform other types of loss, albeit by a small margin. Of the remaining loss units, phone-level loss seemed to perform the best, although the differences between phone, state, and word level loss are fairly small.

CE	MMI	bMMI	MPE	sMBR
22.7	21.3	21.2	21.1	21.0

Table 1. *Word Error Rates for baseline systems. CE refers to cross-entropy, MMI maximum mutual information, bMMI boosted MMI, MPE minimum phone error, and sMBR state-level minimum Bayes risk.*

Loss/Boost	1	3	5	7	9
frame	20.2	20.3	**20.1**	20.4	20.5
state	20.7	20.7	20.6	20.7	20.7
phone	20.7	**20.3**	20.5	20.5	20.5
word	21.2	20.6	20.6	20.6	20.6

Table 2. *Effect of loss unit and boosting parameter onthe performance of DNN-Latent Structured SVM systems.* $\lambda = 0.0001$*, size of the N-best list is 1000.*

Table 3 shows the effect of the size of the N-best list for the best-performing of the frame-loss and phone-loss models. The optimal size seems to be about 1000, although increasing or decreasing the size of the N-best list by a factor of two seems not to make much difference.

Table 4 shows the effect of updating the transition model in the best model for each type of loss unit. In this case, we

N-best size	100	500	1000	2000
frame	20.2	20.6	**20.1**	20.3
phone	20.7	20.5	**20.3**	20.5

Table 3. *Effect of N-best list size on word error rate. For the frame model the boosting parameter is 5, while for phone it is 3.*

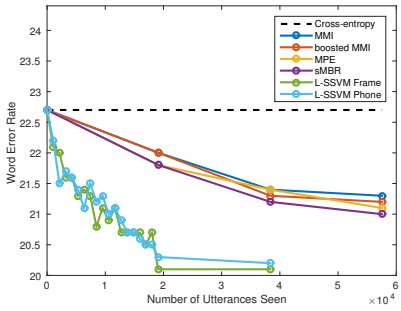

Fig. 2. *Word Error Rate vs. Number of training utterances seen for different sequence-discriminative training criteria*

do not normalize the probabilities from the outgoing states to sum to 1. This necessitated a change in the weighted FST composition algorithm, as FSTs are composed under a log semiring in the standard recipe, under the assumption that the language and HMM models are roughly probabilities. For updating the time transitions, we instead use a tropical semiring, which generally produced ASR results that were the same or 0.1% worse than graphs produced with a log semiring. In any case, updating the time transitions seems not to have a material effect either way. For loss and phone-level units, the results were the same, while results were slightly better for word and slightly worse for state. It is likely that updating transition parameters does not improve recognition results

Update time transitions?	No	Yes
frame	**20.1**	**20.1**
state	**20.6**	20.9
phone	**20.3**	**20.3**
word	20.6	**20.4**

Table 4. *The effect of updating phone model temporal parameters on word error rate. The boosting parameter is 5 for the frame model and 3 for the phone model.*

In initial experiments comparing the latent SSVM, which updated alignments after every batch, to a regular SSVM whose alignments were updated only after each epoch, we found that the latent SSVM converged to a better model after seeing fewer training utterances. Figure 2 compares two LSSVM systems to the standard sequence-discriminative training criteria. We note that the proposed model needs $33 - 66\%$ fewer utterances to converge, although with an N-best list of size 1000, processing time per utterance seems to be roughly 50% longer than standard systems.

Finally, given that these models were implicitly tuned on the test set, we wanted to determine their performance on an independent test set. We compared the best frame-level and phone-level loss models to standard sequence-discriminative systems on the dev and eval portions of the AMI meeting corpus under the individual headset microphone (IHM) condition. Each set consists of roughly 8 hours of speech; more details can be found at [32]. As is shown in Table 5, the latent Structured SVM models outperform the sequence-discriminative training criteria, and the results are statistically significant with $p < 0.001$ using a signed test for paired outcomes. The boosted MMI system is not included here as results on the AMI Dev and Test Sets were not better than those from the cross-entropy model.

	AMI Dev	AMI Eval
CE	37.2	42.6
MMI	36.0	41.3
MPE	35.0	39.8
sMBR	35.0	39.9
LSSVM-frame	34.6	39.1
LSSVM-phone	**34.5**	**38.9**

Table 5. $\lambda = 0.0001$, *for frame, boosting parameter is 5, while for phone, it is 3.*

6. CONCLUSION

In this work, we have proposed hybrid DNN-Latent Structured SVM acoustic models. These systems outperform strong sequence-discriminative trained baselines, while often requiring fewer than half the utterances to converge.

Some directions for future research include comparing our method on a larger task to see if both the performance and sample complexity generalize. Initial results using Kaldi AMI Setup seem to match those on the ICSI meeting corpus, but more work is needed. Second, currently, the "loss-augmented alignment" in the training algorithm requires both lattice generation and an N-best list, the latter of which seems to increase the processing time per utterance roughly 50% compared to that for extant sequence-discriminative training criteria. Future work will include methods for faster search.

Caveats aside, DNN-Latent Structured SVM acoustic models seem to offer a promising alternative to sequence-discriminative training criteria. Moreover, this framework is not specific to the DNN-HMM paradigm, and could be used with other acoustic models such as the LSTM, or another approximately log-linear model, such as [33].

7. ACKNOWLEDGMENTS

The author would like to gratefully acknowledge Nelson Morgan, Dan Ellis, Andreas Stolcke, and Steven Wegmann for helpful discussions on the system and experimental setup. This work is supported by Downs-Ravuri Family Fellowship for Wayward Children and Spouses.

8. REFERENCES

[1] Vaibhava Goel and William J. Byrne, "Minimum bayes-risk automatic speech recognition.," *Computer Speech Language*, vol. 14, no. 2, pp. 115–135, 2000.

[2] Dan Gillick, Larry Gillick, and Steven Wegmann, "Don't multiply lightly: Quantifying problems with the acoustic model assumptions in speech recognition," in *2011 IEEE Workshop on Automatic Speech Recognition & Understanding, ASRU 2011, Waikoloa, HI, USA, December 11-15, 2011*, 2011, pp. 71–76.

[3] Georg Heigold, *A log-linear discriminative modeling framework for speech recognition*, Ph.D. thesis, Aachen, 2010, Zsfassung in dt. und engl. Sprache; Aachen, Techn. Hochsch., Diss., 2010.

[4] Tamir Hazan, Joseph Keshet, and David A. McAllester, "Direct loss minimization for structured prediction," in *Advances in Neural Information Processing Systems 23*, J.D. Lafferty, C.K.I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, Eds., pp. 1594–1602. Curran Associates, Inc., 2010.

[5] David McAllester, "Generalization bounds and consistency for structured labeling in predicting structured data," 2007.

[6] Daniel Povey and Philip C. Woodland, "Minimum phone error and i-smoothing for improved discriminative training," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2002, May 13-17 2002, Orlando, Florida, USA*, 2002, pp. 105–108.

[7] Matthew Gibson and Thomas Hain, "Hypothesis spaces for minimum bayes risk training in large vocabulary speech recognition," in *In Proc. Interspeech*, 2006, pp. 2–4.

[8] Brian Kingsbury, Tara N. Sainath, and Hagen Soltau, "Scalable minimum bayes risk training of deep neural network acoustic models using distributed hessian-free optimization.," in *INTERSPEECH*. 2012, pp. 10–13, ISCA.

[9] 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*. Apr. 1986, vol. 11, pp. 49–52, IEEE.

[10] Daniel Povey, Dimitri Kanevsky, Brian Kingsbury, Bhuvana Ramabhadran, George Saon, and Karthik Visweswariah, "Boosted MMI for model and feature-space discriminative training," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2008, March 30 - April 4, 2008, Caesars Palace, Las Vegas, Nevada, USA*, 2008, pp. 4057–4060.

[11] Fei Sha and Lawrence K. Saul, "Large margin gaussian mixture modeling for phonetic classification and recognition," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing, ICASSP 2006, Toulouse, France, May 14-19, 2006*, 2006, pp. 265–268.

[12] Ben Taskar, Carlos Guestrin, and Daphne Koller, "Max-margin markov networks," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L.K. Saul, and B. Schölkopf, Eds., pp. 25–32. MIT Press, 2004.

[13] Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun, "Large margin methods for structured and interdependent output variables," *Journal of Machine Learning Research*, vol. 6, pp. 1453–1484, 2005.

[14] S.-X. Zhang and M. J. F. Gales, "Structured support vector machines for noise robust continuous speech recognition," in *Proc. Interspeech*, Florence, Italy, 2011, pp. 989–992.

[15] S.-X. Zhang, Chaojun Liu, Kaisheng Yao, and Yifan Gong, "Deep neural support vector machines for speech recognition," in *Proc. ICASSP*, April 2015.

[16] Hao Tang, Kevin Gimpel, and Karen Livescu, "A comparison of training approaches for discriminative segmental models," in *Proc. Interspeech*, Singapore, 2014.

[17] Georg Heigold, Thomas Deselaers, Ralf Schlüter, and Hermann Ney, "Modified mmi/mpe: A direct evaluation of the margin in speech recognition," in *Proceedings of the 25th International Conference on Machine Learning*, New York, NY, USA, 2008, ICML '08, pp. 384–391, ACM.

[18] Y. Altun, I. Tsochantaridis, and T. Hofmann, "Hidden Markov support vector machines," in *Proceedings of the International Conference on Machine Learning*, 2003.

[19] Chun-Nam John Yu and Thorsten Joachims, "Learning structural svms with latent variables," in *Proceedings of the 26th Annual International Conference on Machine Learning*, New York, NY, USA, 2009, ICML '09, pp. 1169–1176, ACM.

[20] George Saon, Hong-Kwang Jeff Kuo, Steven J. Rennie, and Michael Picheny, "The IBM 2015 english conversational telephone speech recognition system," *CoRR*, vol. abs/1505.05899, 2015.

[21] Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, and Chuck Wooters, "The icsi meeting corpus," in *Proc. Interspeech*, 2003, pp. 364–367.

[22] Oliver Cetin and Andreas Stolcke, "Language modeling in the icsi-sri spring 2005 meeting speech recognition evaluation system," Tech. Rep., International Computer Science Institute, 2005.

[23] "Rt-2002 evaluation plan, `http://www.itl.nist.gov/iad/mig/tests/rt/2002/docs/rt02_eval_plan_v3.pdf.`,".

[24] "Rt-04s evaluation data documentation, `http://www.itl.nist.gov/iad/mig/tests/rt/2004-spring/eval/docs.html`,

[25] "Rt-05s evaluation data documentation, `http://www.itl.nist.gov/iad/mig/tests/rt/2005-spring/eval/docs.html`,

[26] Sree Hari Krishnan Parthasarathi, Shuo-Yiin Chang, Jordan Cohen, Nelson Morgan, and Steven Wegmann, "The blame game in meeting room asr: An analysis of feature versus model errors in noisy and mismatched conditions," in *ICASSP'13*, 2013, pp. 6758–6762.

[27] Suman V. Ravuri, "Hybrid mlp/structured-svm tandem systems for large vocabulary and robust ASR," in *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*, 2014, pp. 2729–2733.

[28] S.-Y. Chang and Steven Wegmann, "On the importance of modeling and robustness for deep neural network feature," April.

[29] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. Dec. 2011, IEEE Signal Processing Society, IEEE Catalog No.: CFP11SRW-USB.

[30] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, July 2006.

[31] Karel Vesely, Arnab Ghoshal, Lukas Burget, and Daniel Povey, "Sequence-discriminative training of deep neural networks," in *INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France, August 25-29, 2013*, 2013, pp. 2345–2349.

[32] Jean Carletta, "Unleashing the killer corpus: experiences in creating the multi-everything ami meeting corpus," *Language Resources and Evaluation*, vol. 41, no. 2, pp. 181–190, 2007.

[33] Andrew L. Maas, Ziang Xie, Dan Jurafsky, and Andrew Y. Ng, "Lexicon-free conversational speech recognition with neural networks," in *North American Chapter of the Association for Computational Linguistics*, Singapore, 2015.