Hybrid MLP/Structured-SVM Tandem Systems for Large Vocabulary and Robust ASR

Suman V. Ravuri[1,2]

[1]International Computer Science Institute, Berkeley, CA, USA
[2]EECS Department, University of California - Berkeley, Berkeley, CA, USA
ravuri@icsi.berkeley.edu

Abstract

Tandem systems based on multi-layer perceptrons (MLPs) have improved the performance of automatic speech recognition systems on both large vocabulary and noisy tasks. One potential problem of the standard Tandem approach, however, is that the MLPs generally used do not model temporal dynamics inherent in speech. In this work, we propose a hybrid MLP/Structured-SVM model, in which the parameters between the hidden layer and output layer and temporal transitions between output layers are modeled by a Structured-SVM. A Structured-SVM can be thought of as an extension to the classical binary support vector machine which can naturally classify "structures" such as sequences. Using this approach, we can identify sequences of phones in an utterance.

We try this model on two different corpora – Aurora2 and the large-vocabulary section of the ICSI meeting corpus – to investigate the model's performance in noisy conditions and on a large-vocabulary task. Compared to a difficult Tandem baseline in which the MLP is trained using 2nd-order optimization methods, the MLP/Structured-SVM system decreases WER in noisy conditions by 7.9% relative. On the large vocabulary corpus, the proposed system decreases WER by 1.1% absolute compared to the 2nd-order Tandem system.

Index Terms: Structured SVM, Noise Robustness, LVCSR, Hybrid Systems

1. Introduction and Related Work

Using multi-layer perceptrons (MLPs) to model acoustics has had a long history in automatic speech recognition (ASR). One approach that has proven successful, especially for feature combination, is the Tandem method [5]. In the Tandem approach, 9-15 frames of a base feature, such as PLP, MFCC, or more exotic features, are trained with a MLP on a phone recognition task. Processed log posteriors from the MLP are appended to standard features such as MFCCs, which are then used as input to an acoustic model.

While the context frames allow the MLP to model longer temporal regions, a possible problem with this approach is that the MLP does not explicitly model any temporal dynamics. In more traditional acoustic modeling, a number of researchers over the years have tried to extend the MLP model to handle time transitions. The "hybrid" ANN/HMM approach of [11] included HMM-style parameters between two consecutive output layers, and the model was trained using maximum likelihood (ML). With the renewed interest in neural networks using a "deep approach," a number of new discriminative training criteria has been proposed or adapted from HMM-GMM systems: MMI/MPE [24], boosted MMI [20], and scalable minimum Bayes risk [6]. While comparatively less research has focused on Tandem systems, there have been some notable efforts. [23] explored the use of recurrent neural networks to replace the MLP, using second-order Hessian-free optimization for training. [10] and [12] proposed a hybrid system consisting of a linear-chain conditional random field and a multi-layer perceptron, and improve upon the MLP baseline on a phone recognition task.

Much of the difficulty of augmenting an MLP-based systems with time transitions and using a purely discriminative model is that, empirically, the system can easily become overtrained. In the original hybrid ANN/HMM acoustic model and its successors trained on discriminative criteria, the probability of the phone given the input feature is divided by the probability of state to create an ersatz generative model. In the hybrid CRF-MLP approach of [12], very clever normalizations were used to combat what the authors call "a low entropy frame output."

In this work, we propose introducing temporal structure using the framework of Structured-SVMs, and in particular the Hidden Markov Support Vector Machine (HMSVM), first introduced in [1]. Figure 1 shows the proposed hybrid system. The architecture from the input layer to hidden layer is a multi-layer perceptron, while the parameters from the hidden layer to output layer, and those between output layers, are part of the Structured-SVM.

There are two reasons, one theoretical and empirical, to suggest that such a hybrid discriminatively-trained system may work. The theoretical reason is that generalization error of a support vector machine, with a couple modifications, also hold true for Structured-SVMs (see [17] for a example of this bound) provided that the VC-dimension is finite. Bounded inputs provide a finite VC-dimension, and hidden layers based on tanh or sigmoid (as used in this work) non-linearities are bounded. Thus, at first glance, the generalization result may allow us to circumvent the overtraining problem. Empirically, researchers working on other tasks such as part-of-speech tagging ([4]) have noted that Structured-SVMs seem to work well when input features are binary. As shown in Figure 2, the hidden units of a multi-layer perceptron with sigmoid units serve as an approximation to binary features.

Structured-SVMs have been successfully applied to other areas of automatic speech recognition. [26] used Structured-SVMs as a type of meta-learning algorithm to improve ASR results; using log-probabilities from competing HMM word hypothesis and a language model as input features, the authors used a Structured-SVM to improve inference and optimize segmentations for the Structured-SVM. This work was later extended to large-vocabulary tasks in [28] by focusing on subword units and adding a parameter for a prior.

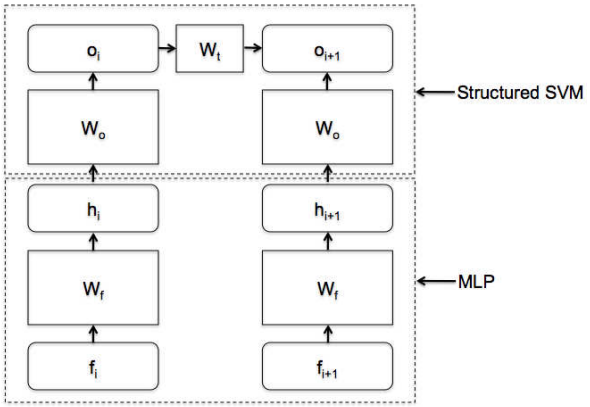

Figure 1: *Diagram of the Hybrid MLP/Structured-SVM Model for two consecutive frames. The parameters from the input features to the hidden units are those of a standard MLP, while the parameters from the hidden units to outputs, and time transitions, are trained using a Structured-SVM.*

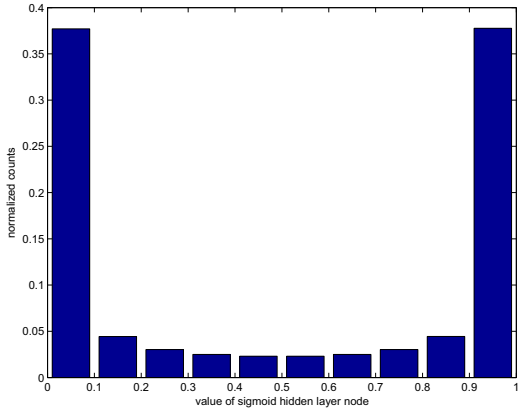

Figure 2: *Histogram for activation values for hidden layer nodes.*

In the following sections, we provide an overview of the Structured-SVM and the hybrid MLP/Structured-SVM Tandem system, and show results on noisy and large-vocabulary corpora.

2. Structured SVM

2.1. Model

Structured-SVMs can be thought of as an extension of the binary Support Vector Machine, in which the prediction is no longer a binary decision, but are more "complex" structured outputs such as multi-class labels, sequences, or trees. Although space constraints preclude us from giving a more general treatment, we refer readers to an excellent overview paper in [19], and outline the specific structure used in the work. We use a Hidden Markov Support Vector Machine, first introduced in [1]. Informally, the HMSVM includes the same temporal parameters as a HMM, but no normalization across exiting states is needed. Moreover, the output distribution of the HMM is replaced by a multi-class SVM. More formally, define n to be the length of an utterance, $\mathbf{h} = [\mathbf{h_1^T}|\dots|\mathbf{h_n^T}]^T$ the input features of the entire utterance, $\mathcal{P} \in \{1,\dots,k\}$ the output phone set , $\mathbf{y} \in \mathcal{P}^n$ the prediction, and $\mathbf{w}$ to be the model of the Hidden Markov SVM. $\mathbf{w}$ includes W_o and W_t, but the weights are stacked as follows to create a single vector:

$$w = [\mathbf{w_1^T}|\dots|\mathbf{w_k^T}|w_{11}|w_{12}|\dots|w_{kk}]^T$$

where

$$W_o = \begin{bmatrix} \mathbf{w_1^T} \\ \mathbf{w_2^T} \\ \dots \\ \mathbf{w_k^T} \end{bmatrix}$$

and

$$w_{ij} = [W_t]_{ij}$$

At test, the best prediction $\mathbf{y}$ is solved as follows:

$$\arg\max_{\mathbf{y}} \mathbf{w^T}\phi(\mathbf{h},\mathbf{y})$$

where

$$\phi(h,y) = \Big[\sum_i^N \mathbf{h_i^T} 1_{y_i=1} | \sum_i^N \mathbf{h_i^T} 1_{y_i=2} | \dots | \sum_i^N \mathbf{h_i^T} 1_{y_i=k} | \sum_{i=2}^N 1_{y_{i-1}=1,y_i=1} | \sum_{i=2}^N 1_{y_{i-1}=1,y_i=2} | \dots | \sum_{i=2}^N 1_{y_{i-1}=k,y_i=k} \Big]^T$$

Intuitively, $\phi(\mathbf{h},\mathbf{y})$ can be thought of as a feature function associated with the prediction $\mathbf{y}$, and the feature function "turns on" correct features to interact with the right parts of the model $\mathbf{w}$. In practice, the optimization problem is solved using a Viterbi algorithm.

At training time, we maximize the margin between the correct possible phone sequence and all incorrect ones. This leads to the constrained optimization problem:

$$\min_{\mathbf{w},\xi_i} \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \xi_i$$

subject to

$$\forall i \text{ s.t. } \hat{\mathbf{y}}_\mathbf{i} \neq \mathbf{y}_\mathbf{i}^*, \mathbf{w^T}(\phi(\mathbf{h},\mathbf{y}_\mathbf{i}^*) - \phi(\mathbf{h},\hat{\mathbf{y}}_\mathbf{i})) \geq 1 - \xi_i$$

The training objective is similar to the binary SVM, with one modification: there must be a margin between $\mathbf{y}_\mathbf{i}^*$ and all incorrect $\hat{\mathbf{y}}_\mathbf{i}$.

2.2. Training

The optimization problem in the previous subsection should give us pause, as the number of possible constraints is exponential in length of sequence. While cutting-plane training algorithms exist for reducing training complexity to linear in data size, optimization becomes increasingly more expensive as the size of dataset, and therefore the number of constraints, increases. Instead, we solve the problem in the primal. Rewriting the optimization problem and setting $\lambda = \frac{1}{C}$ yields:

$$\min_{\mathbf{w}} \frac{\lambda}{2}||\mathbf{w}||^2 + \sum_i \max(1 - \arg\max_{\hat{\mathbf{y}}_\mathbf{i} \neq \mathbf{y}_\mathbf{i}^*} \mathbf{w^T}(\phi(\mathbf{h},\mathbf{y}_\mathbf{i}^*) - \phi(\mathbf{h},\hat{\mathbf{y}}_\mathbf{i})), 0)$$

We use an extension of the PEGASOS algorithm [18] for Structured-SVMs. PEGASOS is a projected subgradient descent algorithm, and convergence is independent of training set size. While subgradients are extremely efficient to calculate (since for each sequence, the subgradient requires only a Viterbi output), we lose the ability to easily check for convergence. In practice, however, convergence can be monitored by checking performance on a held-out set.

2.3. Proposed Method

Parameter estimation for a hybrid MLP/Structured-SVM System consists of two steps: training a multi-layer perceptron model with $W_t = 0$, and then updating parameters W_o and W_t using a Structured-SVM approach. For the first stage, we train a standard multi-layer perceptron using a modified second-order method called Krylov Subspace Descent in [22], as we noted the best results (both for baseline Tandem systems and the hybrid model) using this method. In the second step, we propagate the input features to the hidden layer, and using the hidden layer as inputs, train the Structured-SVM using the PEGASOS algorithm.

During inference, we follow the approach in [27], and consider the model to be a conditional random field trained with large margin optimization, to generate frame-level posteriors. We also tried to compare against a more standard MLP-CRF hybrid system, but could not obtain results that matched the baseline Tandem results, likely because of the overtraining issue mentioned in [12]. We perform Karhunen-Loève Transform on the log posteriors per frame, and append those features to standard MFCCs.

3. Experimental Setup

In this study, we compared 13-dimensional perceptual linear prediction (PLP) features with first and second derivatives discriminatively trained either by a MLP or MLP/Structured-SVM, and processed using the Tandem approach. This feature is appended to a 13-dimensional MFCC with first and second derivatives in all the experiments.

We tested this hybrid model on Aurora2 [2] and the large-vocabulary section of the ICSI Meeting Corpus [3], to check the model's performance in noisy conditions and for a large-vocabulary task, respectively. The two subsections below provide more detail on the experimental setups.

3.1. Aurora2

The Aurora2 data set is a connected digit corpus which contains 8,440 sentences of clean training data and 70,070 sentences of clean and noisy test data. The test set comprises 10 different noises (subway, babble, car, exhibition, restaurant, street, airport, train-station, MIRS-filtered subway, and MIRS-filtered street) at 7 different noise levels (clean, 20dB, 15dB, 10dB, 5dB, 0dB, -5dB), totaling 70 different test scenarios, each containing 1,001 sentences. All systems were trained only on the clean training set but tested on the entire test set.

The parameters for the HTK decoder used for this experiment are the same as that for the standard Aurora2 setup described in [2]. The setup uses whole word HMMs with 16 states with a 3-Gaussian mixture with diagonal covariances per state; skips over states are not permitted in this model. This is the setup used in the ETSI standards competition. More details on this setup are available in [2].

3.2. ICSI Meeting Corpus

For the large vocabulary task, we use the spontaneous meeting portion of the ICSI meeting corpus [3], recorded with near-field microphones. The training set consists of 23,739 utterances – 20.4 hours – of speech across 26 speakers. The training set is based on meeting data used for adaptation in the SRI-ICSI meeting recognizer [16]. The test set comprises 58 minutes of speech, taken from ICSI meeting from the NIST Rich Transcription Evaluation Sets 2002 [13], 2004 [14], and 2005 [15].

We use HTK version 3.4 for MFCC calculation, acoustic modeling, and decoding (ICSI's feacalc is used for PLP features used for Tandem systems). The mel-cepstra are standard 13-dimensional features, including energy, with first and second derivatives, and the MFCCs are mean-normalized at the utterance level. We use HDecode with a wide beam search (300) for decoding. Decoded utterances are text normalized before NIST's sclite tool is used to calculate word error rate (WER).

The acoustic models use cross-word triphones and are trained using maximum likelihood. Each triphone is modeled by a three-state HMM with a discrete linear transition to prevent skipping. The output distribution for each state is modeled by a GMM with 8 components per mixture with diagonal covariance. Training roughly follows the standard recipe, in which monophone models are estimated from a "flat start", duplicated to form triphone models, clustered to 2,500 states, and then re-estimated.

4. Results

The details of every system reported in this Section are as follows:

- MLP: The MLP is a single hidden-layer multi-layer perceptron with 2,000 hidden units. This number was chosen as it gave the best results on both the Aurora2 and ICSI meeting corpora. The inputs to the MLP were 13-dimensional PLP features with first and second derivatives, and 9 frames of context were used per frame. The multi-layer perceptron is trained with Krylov Subspace Descent, as performance was better than similar networks trained with Hessian-free [8] or stochastic gradient descent. The neural network was trained with 8 sweeps through the data on Aurora2, and 20 on the ICSI Meeting Corpus.
- MLP/Structured-SVM: The hybrid structure also uses 2,000 hidden units. The Structured-SVM is trained with PEGASOS, extended for use with Structured-SVMs. Around 1,500 epochs were used Aurora2 data and about 3,000 epochs were used on the ICSI Meeting Corpus. A batch size of 128 was used (meaning that each epoch used 128 sequences, which constitutes 1.4% and 0.5% of Aurora2 and ICSI Meeting Corpus respectively). A λ of 0.25 was used for Aurora2, and 0.5 for the ICSI Meeting Corpus, although performance was similar for the two values.

4.1. Phone Recognition

When training both models, we held out around 10% of the training utterances – 800 for Aurora2 and 2,170 for the ICSI Meeting Corpus – to test for convergence of the MLP and MLP/Structured-SVM. Thus, we get error rates for a phone classification task from the held out data by using our systems. Tables 1 and 2 show the per-frame phone error rate on

Aurora2 and the ICSI Meeting Corpus, respectively. As expected, the MLP/Structured-SVM Model decreases phone error rate on the MLP baseline. For the smaller Aurora2 task, the MLP/Structured-SVM has a 22.7% relative improvement over the standard MLP baseline, while for the larger vocabulary task, the relative improvement was a more modest 8.4%.

System	MLP	MLP-SSVM
PER	10.28%	7.94%

Table 1: *Phone Error Rate on Cross-Validation Set of Aurora2 for both the multi-layer perceptron (MLP) and MLP/Structured-SVM (MLP-SSVM).*

System	MLP	MLP-SSVM
PER	39.91%	36.83%

Table 2: *Phone Error Rate on Cross-Validation Set of the ICSI Meeting Corpus for both the multi-layer perceptron (MLP) and MLP/Structured-SVM (MLP-SSVM).*

4.2. Speech Recognition

Typical results on the Aurora2 test set using the ETSI setup report accuracies (or mean accuracy) across the 10 noises at 7 noise conditions. Instead, we report word error rate (WER), as this is the standard metric used for ASR performance, and a reduction in errors typically corresponds fairly well to common costs of using a system (for instance, how often a system must back off to a human operator). Moreover, we average across noises and report scores for each noisy condition, to see how the system degrades as SNR decreases. We also include a "usable average" that calculates WER across all noises and conditions at SNRs of 10dB and higher. With the exception of the two cleanest conditions, all results are significant with a p-value of 0.02 using the differences of proportions significance test.

Table 3 shows the results for the different systems on Aurora2. In almost every condition, except for the noisiest case (-5dB), the hybrid system improves upon an standard MLP Tandem baseline, trained with second order methods, a difficult baseline.[1] In particular, the MLP/Structured-SVM system improves upon the MLP baseline by 7.9% relative. The best relative improvements seem to occur in the cleaner cases, and taper off with more mismatched conditions.

Table 4 shows results for the large vocabulary section of the ICSI meeting corpus. Including Tandem features to the standard MFCCs improves performance by 1.3% absolute over the MFCC baseline. Swapping those features with the MLP/Structured-SVM improves results by another 1.1%. All results are significant with a p-value of 0.05 using the differences of proportions significance test.

SNR	MFCC	MLP	MLP-SSVM
Clean	0.97%	0.54%	**0.50%**
20dB	5.99%	1.46%	**1.36%**
15dB	15.66%	3.85%	**3.48%**
10dB	36.62%	10.83%	**9.99%**
5dB	64.29%	28.75%	**27.44%**
0dB	84.66%	58.29%	**57.91%**
-5dB	92.21%	**84.20%**	85.18%
usable avg.	14.08%	4.10%	**3.83%**

Table 3: *Average WER for several systems under different noise conditions on the Aurora2 corpus. "Usable average" is the average WER across noise conditions with SNRs 10dB and above. Bold numbers indicate best performance. Note that, as mentioned before, MLP use the Krylov Subspace Descent optimization method.*

System	MFCC	MLP	MLP-SSVM
WER	33.2%	31.9%	**30.8%**

Table 4: *WER for several systems on the large vocabulary section of the ICSI meeting corpus. Note that, as mentioned before, MLP use the Krylov Subspace Descent optimization method.*

5. Conclusions and Future Work

In this work, we propose a hybrid MLP/Structured-SVM model, and show how to use a system in a "Tandem" approach. In both noisy and large-vocabulary tasks, the MLP/Structured-SVM improved upon a Tandem baseline trained with second-order methods.

There are a few ways in which the model could be improved. One way could be improving optimization. Currently, the model is trained in two stages: first, as standard MLP; and then as a standard Structured-SVM. The reasoning behind splitting optimization into two stages is that performing joint optimization would break the convexity of the Structured-SVM and the nice theoretical convergence properties of the Structured-SVM training algorithm. On the other hand, there is no reason to believe that the hidden units after MLP training are optimal for a Structured-SVM, and perhaps alternating between the two types of training phases could yield better results.

For actual modeling, it is by no means obvious that a HMSVM is the optimal Structured-SVM for the hybrid system. One simple extension would be to investigate as second- or third-order Markov parameters for improving performance; or perhaps another structure, such as a tree, would improve both phone and word recognition. For the multilayer perceptron, with the interest in deep architectures, it would be interesting to determine the optimal number of hidden layers for this hybrid approach.

Finally, given the resurgence in interest in "hybrid" systems for acoustic modeling, it is an interesting question if a MLP/Structured-SVM system could be a replacement for other types of ANN/HMM acoustic models. One unanswered question in this work is whether the current model could handle context-dependent triphones, and if not, what modifications need to be made to the model to handle that many states.

6. Acknowledgements

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship. I would like to thank N. Morgan, Oriol Vinyals, and Steven Wegmann for helpful and thought-provoking discussions about this work.

[1] The is among the best Tandem result in our lab, regardless if the MLP architecture is "shallow" or "deep". Please refer to [21] for comparison.

7. References

[1] Y. Altun, I. Tsochantaridis, T. Hofmann, "Hidden Markov Support Vector Machines." International Conference on Machine Learning (ICML), 2003.

[2] Hirsch, H.G. and Pearce, D., "The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions." ISCA ITRW ASR: Challenges for the Next Millennium. 2000.

[3] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters, "The ICSI meeting corpus." in Proceedings of International Conference on Acoustics, Speech, and Signal Processing, 2003.

[4] Joachims, T., "SVM-HMM for Sequence Tagging", `http://www.cs.cornell.edu/people/tj/svm_light/svm_hmm.html`

[5] H. Hermansky and D. P. W. Ellis and S. Sharma. "Tandem Connectionist Feature Extraction for Conventional HMM Systems." in Proceedings of International Conference on Acoustics, Speech, and Signal Processing, 2000.

[6] B. Kingsbury, T. N. Sainath, and H. Soltau, "Scalable minimum Bayes risk training of deep neural network acoustic models using distributed Hessian-free optimization," in Proc. INTERSPEECH, September 2012.

[7] Y. Konig, H. Bourlard, and N. Morgan. "REMAP: Recursive Estimation and Maximization of A Posteriori Probabilities - Application to Transition-Based Connectionist Speech Recognition". NIPS 1995.

[8] J. Martens. "Deep Learning via Hessian-Free Optimization." ICML 2010.

[9] A. Mohamed and D. Yu and L. Deng, "Investigation of Full-Sequence Training of Deep Belief Networks for Speech Recognition," Proc. Interspeech, 2010.

[10] J. Morris and E. Fossler-Lussier. "Crandem: Conditional random fields for word recognition." Proc. Interspeech, 2009.

[11] N. Morgan and H. Bourlard. "Continuous Speech Recognition: An Introduction to the Hybrid/Connectionist Approach." Signal Processing Magazine IEEE. Vol. 12, No. 3. 1995. p24-42.

[12] R. Prabhavalkar, P. Jyothi, W. Hartmann, J. Morris, and E. Fossler-Lussier. "Investigations into the Crandem Approach to Word Recognition". ACL Human Language Technologies. 2010.

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

[14] "Rt-04s evaluation data documentation," `http://www.itl.nist.gov/iad/mig/tests/rt/z2004-spring/eval/docs.html`.

[15] "Rich transcription spring 2005 evaluation," `http://www.itl.nist.gov/iad/mig/tests/rt/2005-spring/index.html`.

[16] A. Stolcke, X. Anguera, K. Boakye, O. Cetin, A. Janin, M. Magimai-Doss, C. Wooters, and J. Zheng, "The SRI-ICSI Spring 2007 Meeting and Lecture Recognition System," in Proceedings of the Second International Workshop on Classification of Events, Activities, and Relationships (CLEAR 2007) and the Fifth Rich Transcription 2007 Meeting Recognition (RT 2007), 2007.

[17] B. Taskar, C. Guestrin, and D. Koller. "Max-Margin Markov Networks." NIPS 2004.

[18] S. Shalev-Shwartz, Y. Singer, and N. Srebro. "Pegasos: Primal Estimated sub-GrAdient SOlver for SVM." ICML 2007.

[19] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large Margin Methods for Structured and Interdependent Output Variables." Journal of Machine Learning Research (JMLR), 6(Sep):1453-1484, 2005.

[20] K. Vesely, A. Ghoshal, L. Burget, D. Povey. "Sequence-discriminative training of deep neural networks." Proc. Interspeech 2013.

[21] O. Vinyals, and N. Morgan. "Deep vs. Wide: Depth on a Budget for Robust Speech Recognition." Proc. Interspeech 2013.

[22] O. Vinyals and D. Povey, "Krylov Subspace Descent for Deep Learning," in AISTATS, 2012.

[23] O. Vinyals, S. Ravuri, and D. Povey. "Revisiting Recurrent Neural Networks for Robust ASR". Proc. Interspeech 2012.

[24] G.Wang and K. C. Sim, "Sequential classification criteria for NNs in automatic speech recognition," in Proc. INTERSPEECH, August 2011, pp. 441444.

[25] S.J. Young, G. Evermann, MJF Gales, D. Kershaw, G. Moore, JJ Odell, DG Ollason, D. Povey, V. Valtchev, and PC Woodland, The HTK book version 3.4, 2006.

[26] Zhang, S.X., and Gales, M. , "Extending Noise Robust Structured Support Vector Machines to Larger Vocabulary Tasks," In Proceedings of ASRU, 2013. pp. 18-23.

[27] Zhang, S.X., and Gales, M. , "Structured SVMs for automatic speech recognition," IEEE Transactions on Audio, Speech and Language Processing, vol. 21, pp. 544555, 2013.

[28] Zhang, S.X., and Gales, M. , "Structured Support Vector Machines for Noise Robust Continuous Speech Recognition," In Proceedings of Interspeech, 2011. pp. 709-712.