INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

How Neural Network Depth Compensates for HMM Conditional Independence Assumptions in DNN-HMM Acoustic Models

Suman Ravuri[1,3], *Steven Wegmann*[2]

[1]International Computer Science Institute, Berkeley, CA, USA
[2]Semantic Machines, Inc.
[3] University of California - Berkeley, CA, USA
ravuri@icsi.berkeley.edu, swegmann@icsi.berkeley.edu

Abstract

While DNN-HMM acoustic models have replaced GMM-HMMs in the standard ASR pipeline due to performance improvements, one unrealistic assumption that remains in these models is the conditional independence assumption of the Hidden Markov Model (HMM). In this work, we explore the extent to which depth of neural networks helps compensate for these poor conditional independence assumptions. Using a bootstrap resampling framework that allows us to control the amount of data dependence in the test set while still using real observations from the data, we can determine how robust neural networks, and particularly deeper models, are to data dependence. Our conclusions are that if the data were to match the conditional independence assumptions of the HMM, there would be little benefit from using deeper models. It is only when data become more dependent that depth improves ASR performance. That performance substantially degrades, however, as the data becomes more realistic suggests that better temporal modeling is still needed for ASR.

Index Terms: Deep Learning, Acoustic Modeling, Bootstrap Resampling, Hidden Markov Models

1. Introduction

Neural Networks are a now-standard part of automatic speech recognition (ASR) pipelines, as its use in acoustic and language modeling have significantly improved performance of ASR systems. In particular, deep network networks (DNNs) are now ubiquitous in modern acoustic models since work in [1] demonstrated that replacing the Gaussian Mixture Model (GMM) of a GMM-Hidden Markov Model (HMM) acoustic model with a DNN substantially decreased word error rates. Since that time, more recent work has attempted to replace the HMM with recurrent neural models – such as RNNs and LSTMs [2, 3] – and some, such as CTC-trained RNNs and attention-based models [2, 4], obviate the need for a lexicon by training on spelling directly. While some of these efforts have been successful in improving the word error rate of state-of-the-art recognition systems, the relative improvement of replacing HMMs with recurrent models does not match that of swapping GMM-HMMs with DNN-HMMs.

This paper aims to understand why the DNN of the DNN-HMM acoustic model improves speech recognition performance so dramatically, and to determine what role depth plays. The purpose is to hopefully learn what types of problems neural networks solve and what still remain, so that future researchers can focus on the most pressing issues. This work adds to a growing body of literature on analyses of DNN-HMM acoustic models. [5] compared DNNs to GMMs in HMM-based systems on metrics such as phone error rate, noise robustness, and speaking rate, and concluded that DNNs are likely better frame estimators than GMMs. [6] measured the ASR performance after each step of MFCC processing, and also illustrated how layers learned auditory filters when given windowed raw time signals. [7] found that, on a phone recognition task, that hidden units of deeper layers encoded more specific phonemic information, and also removed seemingly uninformative properties such as gender. For Tandem [8] features, which share properties of DNNs in hybrid models, [9] showed that deep Tandem features were more somewhat robust to data dependence compared to MFCC features on metrics of phone and word error rates.

In this work, we make what seems at first glance to be a rather curious hypothesis: that neural networks, and particularly deep networks, help compensate for poor conditional independence assumptions in HMM-based acoustic models. Put another way, if our data were to match our assumptions of the statistical model, there would be little to be gained from using a deep neural network – and possibly one with many more parameters – instead of a shallower one, or even a Gaussian Mixture Model. It is only when data violate modeling assumptions of the HMM that deep neural networks exhibit better performance. This suggests that even beyond better frame accuracy, neural networks are able to compensate for poor conditional independence assumptions of Hidden Markov Models.

To test this hypothesis, we generate synthetic data that matches the conditional independence assumptions of the HMM, and slowly break those assumptions to determine how performance degrades as the data become more "realistic". The tool we use is the bootstrap of Bradley Efron ([10, 11]), first applied to ASR by [12]. A description of the method is outlined in the next section.

2. Synthetic Data Generation

2.1. Mathematical Setup

While implicitly researchers assume that the DNN-HMM is a single statistical model, in this work, we alternatively view the model as an HMM with log-linear observations. Input features to this model are last hidden layers of feedforward neural networks. The features themselves discriminatively trained on triphone state targets, and the log-linear observation model is specified as:

$$P(O_t = h|s_t) \propto \exp(Wh + b)$$

Copyright © 2016 ISCA

http://dx.doi.org/10.21437/Interspeech.2016-283

where h is the last hidden layer, and W and b are parameters of the model. W and b are never estimated explicitly; instead, up to a scaling constant, one can calculate $P(O_t = h|s_t)$ from the output of logistic regression layer of the neural network – which specifies the model for $P(s_t|O_t = h)$ – and the prior probability of states $P(s_t)$ – estimated from state alignments on training data – by applying Bayes' rule.

Ideally, with these acoustic and language models $P(O|S)$ and $P(W)$, respectively, we hope to have created a model $P_{model}(O, W) = P(O|S)P(W)$ that well represents the stochastic process of speech, specified by probability distribution $P_{true}(O, W)$. Naturally, we would like to check to what extent $P_{model}(O, W) \approx P_{true}(O, W)$, since, if the distributions are close, taking the Bayes decision at test time would theoretically achieve the (nearly) optimal word error rate. [1] As we only have samples of $P_{true}(O, W)$ and not direct access to this distribution, however, we cannot check out model directly.

Instead, we attempt to understand to what extent $P_{model}(O, W) \approx P_{true}(O, W)$ by creating synthetic data with distribution $P_{syn}(O, W)$ and calculating error metrics. If model assumptions are satisfied by the synthetic data, and the model is indeed close to true distribution, then:

$$E_{P_{syn}(O,W)}[L(W, \hat{W})] = E_{P_{true}(O,W)}[L(W, \hat{W})]$$

where $\hat{W}$ is hypothesized word sequence decoded using $P_{model}(O, W)$. We estimate this risk by calculating error on a test set. Mathematically, we make the approximation:

$$E_{P_{true}(O,W)}[L(W, \hat{W})] \approx \frac{1}{N} \sum_{O} L(W, \hat{W})$$

and

$$E_{P_{syn}(O,W)}[L(W, \hat{W})] \approx \frac{1}{N} \sum_{O_{syn}} L(W, \hat{W})$$

where O_{syn} are features generated from the synthetic test data. Of course, the converse is not true (i.e., equivalent risks do not imply that the probability model is correct), but highlighting the data/model mismatch will provide some intuition as to how our model is a poor representation of the underlying stochastic process of speech.

2.2. Bootstrap

Our goals for creating synthetic data are to control the independence assumptions of the simulated data while leaving other properties unchanged from the original data. Desiderata include using real acoustic observations, and a mechanism by which we can match, and then slowly break, the conditional independence assumptions of the HMM. We generate simulated data of the form:

$$P_{syn}(O, S) = P_{syn}(S) \prod_{i=1}^{N} P_{syn}(O_i|S_i)$$

where $S \sim P_{syn}(S)$ is the triphone state sequence and $O_i \sim P_{syn}(O_i|S_i)$ the observation given a particular state S_i. We use the test set alignment for the state sequence S (which can be considered a draw from the true distribution of state alignments $P_{true}(S)$), and bootstrap resampling to generate draws from $P_{syn}(O_i|S_i)$. Figure 1 illustrates an example of this process: using the alignment of an utterance from a simulation set to determine the underlying state sequence, features corresponding to particular phones from the utterance are binned. After phones from all utterances in the simulation set are binned, for each test utterance, the original features for a particular phone are discarded and a new one is drawn with replacement from the bins. Once observations for all phones are replaced, the resulting features, in this example, are conditionally independent at the phone level. This general form, however, also allows us to create data that match conditional independence assumptions of the model if we draw every frame from state bins. In this work, we generate synthetic data for frame, state, and phone[2] levels, and create 5 copies to estimate the standard error across sets. Unlike previous work [12, 14, 9], we do not sample at the word level, as a non-negligible percentage of words in the simulation set have few examples (in this dataset 8.5% and 13.2% of words are singletons or have fewer than 5 instances, respectively), and resampling from such a distribution does not give a reasonable estimate of the underlying probability distribution of the word, should it exist.

One final question is what data we should use for the simulation set. The two choices generally available are one disjoint from the training and test sets, and the other is the test set. Using the disjoint set allows us to draw from a much bigger dataset, but generally the speakers from this pool are different from the original test set, and thus makes the relative degradation from sampled to original data seem higher than in general.[3] Instead, we opt to sample from the test set, which allows for more direct comparisons between simulated and real data.

3. Experimental Setup

3.1. Data

For this experiment, we use the Switchboard 1 Release 2 corpus (LDC97S62) for training. This conversational speech recognition corpus comprises 300 hours of data over 542 speakers, 301 of whom are male. The test set for this experiment is the Switchboard portion of Hub5 '00 (LDC2002S09). The lexicon consists of 30K words, and the Kneser-Ney smoothed trigram language model (LM) used in this study is trained on the 3M words in 300-hour set, and contains 272K and 455K trigrams and bigrams, respectively.

3.2. Model

The recognizer for this experiment is Kaldi [16], and the setup roughly follows the s5b example (outlined in [17]). The baseline GMM-HMM system uses roughly 9,000 triphone states and 200K Gaussians, trained with maximum likelihood criterion. The features used are MFCCs with 7 frames of context, mean-normalized per speaker, projected down to 40 dimensions using linear discriminant analysis, with semi-tied covariance [18], and speaker-adaptively trained using a single feature-space maximum likelihood linear regression (FMLLR) per speaker. This feature is denoted as SAT.

The neural networks are trained with the these SAT features, with 11 frames of context, and are mean and variance normalized. The neural network features used in this work vary

[1] Calculating posterior expected loss may be unnecessary in most cases: if the posterior probability $P(W|O) > 0.5$, [13] showed that a MAP decision is equivalent to the Bayes decision rule.

[2] In Kaldi, each "phone" corresponds to pdf triple, to account for triphone context.

[3] As done in this work, one can partially control for speaker effects by using speaker-adapted features, but in preliminary work on the ICSI Meeting Corpus [15], we found higher relative degradation when using a disjoint independent set.

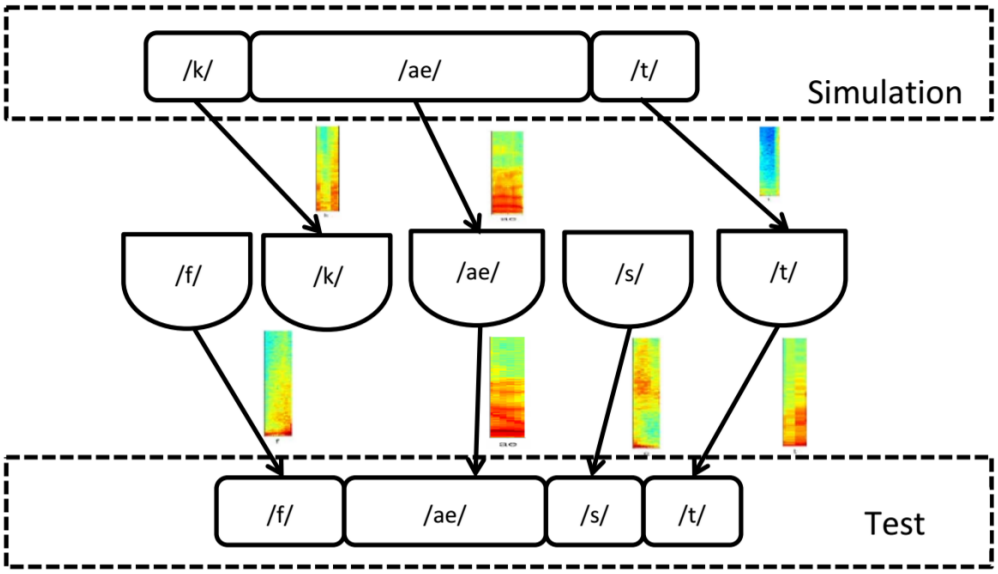

Figure 1: *Illustration of Bootstrap Resampling at the Phone Level. Inside the dashed boxes are alignments for the simulation and test sets. The urns in the middle pane correspond to the bins into which acoustic observations of particular phones are accumulated from the simulation set. Then, a synthetic test utterance is created by drawing, with replacement, observations from the bins.*

from 0 to 6 hidden layers (where 0 is simply logistic regression), and two types of architectures are tested: one where the size of each hidden layer, and one where the number of parameters, is held constant. For the first type of architecture, each hidden layer is of size 2,048, while for the second the number of parameters are constrained to be roughly 36M, with the hidden layer size equal for each layer. The number of parameters in the latter experiment were chosen from the best-performing neural network in the equal hidden layer size experiment. Table 1 shows the architecture used for the neural networks in the equal parameter experiment. These so-called "derived features" are denoted as NN-xHL in this and subsequent tables, where x is the number of hidden layers (HL), while the "-EP" designation refers to "equal parameter."

The neural networks are pertained using Restricted Boltzmann Machines (RBMs) [19] from initial weights drawn from a Gaussian distribution of mean 0 and standard deviation 0.01. The first Gaussian-Bernoulli layer is trained with learning rate 0.01 for 1 epoch, and following Bernoulli-Bernoulli layers are trained for 1 epoch with learning rate 0.04. The neural networks are then discriminatively trained using the cross-entropy training criterion, using alignments generated from the GMM-HMM system as labels. The networks are optimized with stochastic gradient descent, with 90% of the data used for training and remaining for cross-validation. The minibatch size is 256, and the learning rate schedule is initialized to 0.008, and halves when the cross-validation frame accuracy does not improve by at least 0.5%. Training terminates when the cross-validation accuracy fails to improve by more than 0.1%. Neural network trainings finished after 12-17 epochs.

	# hid units/layer	# parameters
NN-1HL-EP	3850	36.3M
NN-2HL-EP	2920	36.1M
NN-3HL-EP	2500	36.1M
NN-4HL-EP	2250	36.4M
NN-5HL-EP	2048	36.1M

Table 1: *The size of hidden layers and number of parameters for each feature type in equal parameter experiment (denoted (EP).*

Feature	frame	state	phone	original
GMM	2.7 (.05)	9.1 (.12)	15.7 (.08)	21.8
NN-0HL	3.7 (.05)	13.0 (.17)	21.9 (.16)	31.6
NN-1HL	2.6 (0.0)	7.8 (.13)	12.9 (.15)	18.7
NN-2HL	2.5 (.05)	7.2 (.11)	11.8 (.18)	16.4
NN-3HL	2.5 (.04)	7.1 (.10)	11.4 (.17)	15.7
NN-4HL	2.5 (.04)	7.0 (.10)	11.2 (.19)	15.2
NN-5HL	2.5 (0.0)	7.0 (.11)	11.2 (.23)	15.0
NN-6HL	2.5 (0.0)	7.1 (.15)	11.2 (.16)	15.1

Table 2: *Word Error Rate of various neural network features with equal-sized hidden layers for different types of resampled data, averaged across 5 runs. Numbers in parentheses indicate the standard error. GMM refers to Gaussian Mixture Model classifier, NN neural network, and HL hidden layer. Unless noted as GMM, the observation model is the log-linear model. Note that NN-0HL are simply SAT features. Data matches the conditional independence assumptions of the HMM model at the frame level, and becomes increasingly more dependent at the state level, phone level, and original data.*

4. Results

4.1. Equal Hidden Layer Size

Table 2 shows the results for various features as data becomes dependent (when read from left to right), where the neural network features have equal-sized hidden layers. Somewhat surprisingly, if the conditional independence assumptions of the Hidden Markov Model are matched, as they are for frame-level

resampled data, there is only a marginal improvement from replacing the SAT features and GMM classifier with neural network features and log-linear classifier. The results for SAT features using only a log-linear model – denoted NN-0HL – are uncompetitive, likely because it is a poorer frame classifier: frame accuracy on the test set was 31%, compared to 43-48% for other neural network features.[4] Moreover, using more than one hidden layer yielded at best modest – 4% relative – reduction in word error rate, and increasing the number of hidden layers to more than 2 does not improve performance. As data become more dependent, however, the improvement from using neural networks and deeper networks becomes more pronounced. At the state level, neural network features achieve better word error rates than standard speaker-adaptive features, and unlike for the frame-level resampled data, increasing depth beyond 1 hidden layer does improve results more substantially. In this case, using neural networks with more than 2 hidden layers yields only marginal improvements, and in fact is within the standard error of the best-performing feature. For phone-level resampled data, again increasing depth by a hidden layer improves word error rate results, while again word error rates for the neural network feature with 3 hidden layers are within a standard deviation away from the best-performing feature. Finally, for the original data, using 4 hidden layers yields an improvement, and using more provides little benefit (though one can improve results slightly with a 5-hidden-layer network). The general trend is that the "optimal" depth is deeper for more realistic data, which suggests that deeper models are more robust to violated conditional independence assumptions in the data. That said, the six- to seven-fold increase in word error rate suggests that hybrid models, even with deep neural networks, suffer from data/model mismatch.

4.2. Equal Number Neural Network Parameters

Table 3 shows the results for different neural network features if the number of parameters of the neural networks are equal. Compared to results in the previous section, the word error rates of the shallower networks decreases, such as for features with 1 hidden layer, or is roughly equivalent, for those with 3 hidden layers. The better performance of shallower networks for this experiment are not sufficient, however, to achieve the same performance as the deeper networks.[5] As a result, the overall message stays the same: if the data were to match the conditional independence assumptions of the HMM model, then there is little gain from using neural networks features compared to standard ones with a GMM classifier. As the data becomes more dependent, not only do neural network features improve performance compared to a GMM system, using deeper neural networks actually make the model more robust to data dependence.

Feature	frame	state	phone	original
GMM*	2.7 (.05)	9.1 (.12)	15.7 (.08)	21.8
NN-0HL-EP*	3.7 (.05)	13.0 (.17)	21.9 (.16)	31.6
NN-1HL-EP	2.6 (.05)	7.6 (.13)	12.5 (.17)	18.0
NN-2HL-EP	2.5 (.05)	7.2 (.10)	11.7 (.16)	16.3
NN-3HL-EP	2.5 (0.0)	7.1 (.10)	11.4 (.18)	15.7
NN-4HL-EP	2.5 (0.0)	7.0 (.11)	11.3 (.14)	15.2
NN-5HL-EP	2.5 (0.0)	7.0 (.11)	11.2 (.23)	15.0

Table 3: *Word Error Rate of various neural network features with equal number of parameters for different types of resampled data, averaged across 5 runs. Numbers in parentheses indicate the standard error. GMM refers to Gaussian Mixture Model classifier, NN neural network, and HL hidden layer. Unless noted as GMM, the observation model is the log-linear model. Note that NN-0HL are simply SAT features. The asterisks designate systems that do not use the same number of parameters. Unless noted as GMM, the observation model is the log-linear model. Note that NN-0HL are simply SAT features. Data matches the conditional independence assumptions of the HMM model at the frame level, and becomes increasingly more dependent at the state level, phone level, and original data.*

5. Conclusion

In this work, we studied to what extent neural networks in DNN-HMM hybrid systems, and particularly depth of those networks, compensated for incorrect assumptions in the HMM model. In viewing the hybrid system as a log-linear-HMM system with features from the last hidden layer of a trained DNN, we showed that neural networks indeed compensated for poor conditional independence assumptions in the HMM. If our data were to match our assumptions, there would be little benefit of using a neural network, much less a deep one. As the data became more dependent, however, the deeper networks were more robust to data/model mismatch than shallower ones or GMMs.

Despite these improvements, however, the six- to seven-fold increase in word error rate from synthetic to real data strongly suggests that our temporal modeling is still poor. One way to decrease this gap is to use sequence-discriminative training criteria such a (boosted) maximum mutual information [21, 22], minimum phone error [23], state-level minimum Bayes risk [24, 25], or large-margin methods [26]. These criteria, however, tend to ameliorate, rather than fix, unrealistic modeling assumptions. Newer acoustic models, such as simple recurrent neural networks and long short-term memory (LSTM) networks, have shown some promise, though the improvement relative to hybrid systems is quite less than that for replacing the GMM with the DNN. Why this occurs is an interesting and open problem, and one for which this current approach is ill-suited. Perhaps hypothesizing a more realistic generative model of speech, creating data based on that generative model, and comparing performance of better synthetic and real data for these newer acoustic models can shed some light on what problems remain in the system.

6. Reproducible Research

In an effort to make this research reproducible, the setup for these experiments will be made available at http://www.github.com/sumanvravuri.

7. Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1450916.

[4] While this explains results for synthetic data resampled at the frame level, an additional source of degradation is the violation of conditional independence assumptions. 10 of 11 frames of features are identical for consecutive frames.

[5] Although one could use a teacher-student framework such as [20] to possibly achieve performance equivalent to deeper networks. This model first trains a deep neural network, and then trains a shallow network to learn pre-softmax outputs of the deeper network. This approach was not studied in this work.

8. References

[1] G. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 30 –42, jan. 2012.

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

[3] H. Sak, A. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," in *Proc. Interspeech*, Dresden, Germany, 2015.

[4] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell," *CoRR*, vol. abs/1508.01211, 2015. [Online]. Available: http://arxiv.org/abs/1508.01211

[5] Y. Huang, D. Yu, C. Liu, and Y. Gong, "A comparative analytic study on the gaussian mixture and context dependent deep neural network hidden markov models," in *Interspeech 2014*, September 2014. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=230138

[6] Z. Tüske, P. Golik, R. Schlüter, and H. Ney, "Acoustic modeling with deep neural networks using raw time signal for lvcsr," in *Interspeech*, Singapore, Sep. 2014, pp. 890–894, iSCA best student paper award Interspeech 2014.

[7] T. Nagamine, M. Seltzer, and N. Mesgerani, "Exploring how deep neural networks form phonemic categories," in *Proc. Interspeech*, Dresden, Germany, 2015.

[8] H. Hermansky, D. P. W. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional HMM systems," *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, vol. 3, pp. 1635–1638, 2000. [Online]. Available: http://dx.doi.org/10.1109/icassp.2000.862024

[9] S. Ravuri and S. Wegmann, "How neural network features and depth modify the statistical properties of hmm acoustic models," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2016, March 21 - March 26, 2016, Shanghai, China*, 2016.

[10] B. Efron, "Bootstrap methods: Another look at the jackknife," *Annals of Statistics*, vol. 7, pp. 1–26, 1979.

[11] ——, *The Jackknife, the bootstrap and other resampling plans*, ser. CBMS-NSF Reg. Conf. Ser. Appl. Math. Philadelphia, PA: SIAM, 1982, lectures given at Bowling Green State Univ., June 1980. [Online]. Available: https://cds.cern.ch/record/98913

[12] D. Gillick, L. Gillick, and S. 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. [Online]. Available: http://dx.doi.org/10.1109/ASRU.2011.6163908

[13] R. Schlüter, M. Nußbaum-Thom, and H. Ney, "On the relation of bayes risk, word error, and word posteriors in asr," in *Interspeech*, Makuhari, Japan, Sep. 2010, pp. 230–233.

[14] S. H. K. Parthasarathi, S.-Y. Chang, J. Cohen, N. Morgan, and S. 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.

[15] 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 *Proc. Interspeech*, 2003, pp. 364–367.

[16] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011, iEEE Catalog No.: CFP11SRW-USB.

[17] K. Vesely, A. Ghoshal, L. Burget, and D. 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.

[18] M. J. F. Gales, "Semi-tied covariance matrices for hidden markov models," *IEEE Trans. Speech and Audio Processing*, vol. 7, no. 3, pp. 272–281, 1999. [Online]. Available: http://dx.doi.org/10.1109/89.759034

[19] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006. [Online]. Available: http://dx.doi.org/10.1162/neco.2006.18.7.1527

[20] L. J. Ba and R. Caurana, "Do deep nets really need to be deep?" *CoRR*, vol. abs/1312.6184, 2013. [Online]. Available: http://arxiv.org/abs/1312.6184

[21] 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*, vol. 11. IEEE, Apr. 1986, pp. 49–52. [Online]. Available: http://dx.doi.org/10.1109/icassp.1986.1169179

[22] D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. 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. [Online]. Available: http://dx.doi.org/10.1109/ICASSP.2008.4518545

[23] D. Povey and P. 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. [Online]. Available: http://dx.doi.org/10.1109/ICASSP.2002.5743665

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

[25] 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 *INTERSPEECH*. ISCA, 2012, pp. 10–13. [Online]. Available: http://dblp.uni-trier.de/db/conf/interspeech/interspeech2012.html

[26] S. V. Ravuri, "Hybrid dnn-latent structured SVM acoustic models for continuous speech recognition," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2015, Scottsdale, AZ, USA, December 13-17, 2015*. IEEE, 2015, pp. 37–44. [Online]. Available: http://dx.doi.org/10.1109/ASRU.2015.7404771