ON THE IMPORTANCE OF MODELING AND ROBUSTNESS FOR DEEP NEURAL NETWORK FEATURE

Shuo-Yiin Chang[1,2] and Steven Wegmann[2]

[1] EECS Department, University of California-Berkeley, Berkeley, CA, USA
[2] International Computer Science Institute, Berkeley, CA, USA
{shuoyiin; swegmann}@icsi.berkeley.edu

ABSTRACT

A large body of research has shown that acoustic features for speech recognition can be learned from data using neural networks with multiple hidden layers (DNNs) and that these learned features are superior to standard features (e.g., MFCCs). However, this superiority is usually demonstrated when the data used to learn the features is very similar in character to the data used to test recognition performance. An open question is how well these learned features generalize to realistic data that is different in character to their training data. The ability of a feature representation to generalize to unfamiliar data is a highly desirable form of robustness. In this paper we investigate the robustness of two DNN-based feature sets to training/test mismatch using the ICSI meeting corpus. The experiments were performed under 3 training/test scenarios: (1) matched near-field (2) matched far-field and (3) the mismatched condition near-field training with far-field testing. The experiments leverage simulation and a novel sampling process that we have developed for diagnostic analysis within the HMM-based speech recognition framework. First, diagnostic analysis shows that a DNN-based feature representation that uses MFCC inputs (MFCC-DNN) is indeed superior to the corresponding MFCC baselines in the two matched scenarios where the source of recognition errors are from incorrect model, but the DNN-based features and MFCCs have nearly identical and poor performance in the mismatched scenario. Second, we show that a DNN-based feature representation that uses a more robust input, namely power normalized spectrum (PNS) and Gabor filters, performs nearly as well as the MFCC-DNN features in the matched scenarios and much better than MFCCs and MFCC-DNNs in the mismatched scenario.

Index terms **–** deep neural network, acoustic feature, robust speech recognition

1. INTRODUCTION

Neural networks have been used successfully for HMM-based speech recognition for more than two decades [1]. In that approach, network outputs were used as posteriors to derive emission probabilities for hidden Markov models (HMMs). Later, a number of researchers (e.g., [Hermansky et al.]) made use of network outputs as features for HMM observations (tandem) [2][3]. Both of these approaches have been used in more recent methods that have been designed to effectively incorporate a larger number of layers, and in particular have been successfully applied to automatic speech recognition (ASR) [4][5][6]. In a typical system, cepstral coefficients or short-term spectra are generated as input to a (deep) neural network [7][8]. While neural network trained features can effectively reduce word error rate (WER) on a matched testing set, they might be too specialized to their training set. The features would do best when tested on similar material as the training, but performance could degrade for a mismatched train-test condition. Here, we present a series of analysis experiments yielding insight into how the recognition errors are distributed in different train-test condition.

Unlike most research on neural network focusing on how to actually improve speech recognition accuracy [9][10][11] or on theoretical asymptotic results [12], we explore the scientific questions surrounding how these applications of neural networks improve speech recognition accuracy and why it fails for particular train-test conditions. In this paper, we first discover the basic mechanisms that neural network-based features use to substantially improve HMM-based speech recognition accuracy for matched near-field or far-field experiments. Second, we investigate the failings of MFCC based DNN for the mismatched train-test condition. Third, we explored the contribution of robust signal processing techniques prior to neural network training. To accomplish this, we employed the robust representation [13] that incorporated Gabor filtering and power normalized spectrum [14] prior to neural network training. The analyses of the improvement from DNN features based on this robust representation allow us to investigate the contribution of robust feature generation within the DNN framework.

The primary statistical tool for analysis experiments is a version of resampling introduced in [15]. We create pseudo speech data by simulation or stringing together samples of real speech segments. By manipulating the test data to include or remove specific statistical properties, we are able to perform an in-depth analysis of the performances in HMM framework.

2. FEATURES

The features explored in this paper are (1) MFCC (2) DNN feature based on MFCC input and (3) DNN feature based on robust representation (means were normalized per utterance before HMM training and testing for all the features in this paper). We are primarily focusing on the actual improvement of the two DNN features comparing to 39-d MFCC baseline.

For MFCC based DNN feature, we exploit a 4-hidden layer neural network structure with a bottleneck layer in the 3rd hidden layer. The bottleneck size is set to 25 while other hidden layers

each consist of 1600 neurons so that total number of parameters is about 3M. The network input is 9 successive frames of MFCC. The output layer consists of 43 context-independent phonetic targets. Restricted Boltzman machine (RBM) pre-training is used to initialize the parameters of the neural network. For back propagation following the pre-training, we begin with a learning rate of .008 and reduced the learning rate by factors of two once cross-validation indicated limited progress with each learning rate, and continued until cross-validation showed essentially no further progress. The final feature is taken from the 25-d bottleneck feature augmented with 39-d MFCCs, which is called MFCC-DNN in the following sections.

To investigate the contribution of robust feature extraction, we generate DNN feature based on a robust representation. The robust signal processing algorithm is taken from [13]. A general summary of how to compute the features is as follows: (1) Compute the power normalized spectrogram. (2) Convolve the spectrogram with each of the desired 2-dimensional filters. (3) Integrate the filter outputs using DNN.

We first generate the power normalized spectrum. To enhance insensitivity to noise, the power normalized spectrum modifies Mel spectrum in three aspects: (1) gammatone filter, (2) medium-duration bias subtraction and (3) power-law nonlinearity. First, the power normalized spectrum employs gammatone auditory filters derived from psychophysical observations of the auditory periphery instead of Mel filterbank. Second, subtraction of the medium-duration power bias is carried out, where the bias level calculation is based on the ratio of arithmetic mean and geometric mean (AM-GM ratio) of the medium duration power, which is motivated by a decrease of the noise power for a decreasing AM-GM ratio. Finally, a power nonlinearity with an exponent of 0.1 replaces the logarithm nonlinearity for compression because the output of the logarithm would be dominated by noise when the intensity of the input signal is low.

Power normalized spectrum is then processed by many Gabor filters yielding a high dimensional feature vector. To generate Gabor filters serving as model for spectro-temporal receptive fields (STRFs) [16][17], we multiply a complex sinusoid with a Hanning envelope. The complex sinusoid (with time modulation frequency ω_n and spectral modulation frequency ω_k) is represented as:

$$s(n,k) = \exp[i\omega_n(n-n_0) + i\omega_k(k-k_0)] \quad (1)$$

while the hanning envelope is given (with W_n and W_k denote window length) by

$$h(n,k) = [\frac{1}{2}(1-\cos(\frac{2\pi n}{W_n+1}))][\frac{1}{2}(1-\cos(\frac{2\pi k}{W_k+1}))] \quad (2)$$

By tuning parameters of spectral and temporal modulation frequency, Gabor functions have different extent and orientation for a given number of oscillations under the envelope used in this study. The 59 Gabor filters, which emphasize different temporal and spectral modulation frequencies, that are used are shown in Fig. 1. However, the filters with a large spectral extent result in high correlations between frequency channels. Hence, a subset of the possible combinations are used to avoid high correlations of feature components, resulting in an 814-dimensional feature. The feature incorporating the usage of power normalized spectrum and Gabor filtering is called PNS-Gabor. Finally, we applied PNS-Gabor feature as input for DNN training. The neural network configure is the same as MFCC-DNN but the hidden layer size is reduced to 450 for comparable number of parameters. The 25-d bottleneck feature trained from PNS-Gabor is concatenated with MFCCs and it is referred as PNS-Gabor DNN in the following discussion.

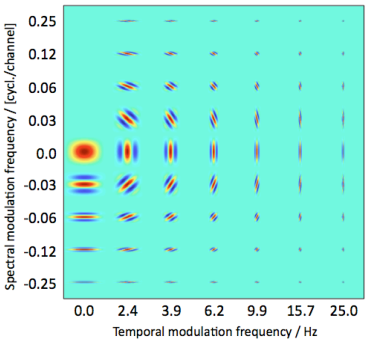

Figure 1. 2-D Gabor filters shown by temporal and spectral modulation frequencies

3. SIMULATION AND RESAMPLING METHOD

The diagnostic research described in [15] proposed simulation and a novel sampling process to generate pseudo test data that deviate from the HMM in a controlled fashion. These processes allow us to generate pseudo data that, at one extreme, agree with all of the model's assumptions, and at the other extreme, deviate from the model in exactly the way real data do. In between, we can precisely control the degree of data/model mismatch. By measuring recognition performance on this pseudo test data, we are able to quantify the effect of this controlled data/model residual on recognition accuracy. In our experiments, we employed the diagnostic process on the 4 levels of simulation and resampling: (1) simulation, (2) frame-resampling, (3) phone-resampling and (4) original test utterance.

3.1. Simulation

Given the generative property of HMM, we can simulate data directly from model where the simulated data matches all the assumptions of the model. These assumptions are (1) the observations are independent conditioned on the states and (2) the output distributions are stationary and can be modeled using Gaussian model. The independent assumption is clearly wrong, partly because of speech production mechanisms and partly because of different ways of feature extraction. For example, neural networks compute feature from a temporal context window and Gabor filters capture feature over a broad time interval.

To generate the test data by simulation, we start with the test transcriptions, and look up each word in the pronunciation dictionary to create phone transcriptions. We then use the state transitions and the output distribution associated with the states belonging to the triphones to generate the data.

3.1. Resampling

For frame-resampling, rather than simulating a pseudo frame from

the Gaussian model, we draw an actual speech frame from an urn filled with examples of relevant state. The resampling process is at random so that the resampled data respects independence assumptions. Specifically, we first use the training model to perform forced alignment on the training utterances, so that each speech frame is annotated with its most likely generating state. Next, we walk through this alignment, filling an urn for each state with its representative frames; at the end of this process, each urn is populated with frames representing its empirical distribution. To generate resampled data, we use the model to create a forced alignment of the test data, and then randomly sample a frame (with replacement) from the corresponding urn for each frame position. With the resampling process, the frames share the output distribution of real data instead of Gaussian distribution but satisfy the HMM independent assumption.

We can extend the idea of frame-resampling to the phone level by resampling phones. In the phone-resampling fashion, we place entire phone sequence of frames in the urns (the urn labels include triphone context), and then resample the concatenated frames of phone. Therefore, the phone-resampled data is dependent within phone region but independent across phone boundaries. Unlike perfect independent data created by simulation or frame-resampling, phone-resampling creates data partially respecting the independent assumption.

3. DATA AND MODEL

3.1. Data

Our experimental protocol, data, and model training are described in [18], which we briefly describe. We used a dataset of spontaneous meeting speech recorded at ICSI [19] where each spoken utterance was captured using near-field and far-field microphones. Our training set is based on the 20 hours of meeting data adapted from the SRI-ICSI meeting recognition system [20]. For the test set we used 1 hour of ICSI meetings drawn from the NIST RT eval sets; this was done to control the variability in the data and to avoid dominant speaker for the resampling experiments. The statistics of training and testing sets are reported in Table 1.

While near-field and far-field corpora were recoded in a parallel manner, both the time delay of physical distance and the systematic delays introduced by the recording software caused skew between two recordings. We detected and fixed the skew by calculating cross-correlation between two recordings [18]. Thus, the near-field and far-field recordings are completely parallel. We create alignments using near-field model and near-field data and use this alignment to perform all the resampling experiments.

Dataset	Speakers	Utterances	Time
Training	26	23729	20.4 (hrs)
Test	18	1063	57.9 (mins)

Table 1. Training and test statistics for near-field and far-field data

3.2. Model

The acoustic models are cross-word triphones modeled by a three-state HMM with a discrete linear transition structure (no skipping) and one diagonal Gaussian per state. While significantly better performance can be achieved with mixture models, the simplicity of a single component is preferable for our analysis; it highlights the performance differences between our experiments. The resulting triphone states are clustered using decision trees to 2500 tied states.

To build a parallel set of near-field and far-field acoustic models that shares the same state-tying, far-field models are trained using single-pass retraining from the final near-field models and the far-field data. Specifically, the E-step is performed using the near-field models and data, while the M-step and model updates use the far-field data. We use a 10K trigram language model [21] that was created from SRI for NIST RT evaluation. The perplexity of this meeting room LM is around 70 on our test set. 186 of 1063 test utterances containing OOV are removed.

4. RESULTS AND DISUSSION

In addition to the original test data, we created near-field and far-field test data by simulation, resampling frames and phones. The corresponding recognition models were used for decoding. All simulation/resampling results report the average results of 5 repeated experiments. The results of matched near-field and far-field experiments are reported in Table 2 and 3 respectively. The experiments of near-field training and far-field testing experiments are reported in Table 4.

Previous work on MFCC [15][18] with matched training/test has shown that recognition errors are dominated by incorrect independent assumptions. The observation still holds for deep neural network features as shown in Table 2. In particular, WERs are extremely low for simulated and frame-resampled data where independent assumption is satisfied by data. By comparing MFCC to DNN features, we observe that deep neural network trained features (both MFCC-DNN and PNS-Gabor DNN) consistently outperform MFCC. For simulated data, transforming MFCC with deep neural network reduces recognition errors by 73%. The improvement decreases as we introduce dependency (at phone-level). Thus, dependency in real data degrades improvement of deep neural network feature in HMM framework.

For matched far-field data, deep neural network trained features keep providing significant improvement as shown in Table 3. Thus, acoustic feature can be learned using deep neural networks even when training data is noisy. Also, since the difference of PNSGB-DNN and MFCC-DNN is negligible, it suggests that signal processing techniques prior to neural network training didn't contribute to extra improvement for matched far-field data.

For the mismatched case, we observe that MFCC-DNN and MFCC have nearly identical and poor performance. As the diagnostic experiments reported in Table 4, recognition errors from observation mismatch of MFCC-DNN is more pronounced for simulated and frame-resampled data where the WERs are 70.9% and 76.9% respectively. For simulated data, applying DNN transformation trained from near-field data to far-field data increase 65% WER relative to MFCC. The result indicates that

resampling	(1) MFCC	(2) MFCC-DNN		(3) PNS-Gabor DNN	
	WER	WER	Rel to (1)	WER	Rel to (1)
sim	1.5	0.4	73%	0.5	67%
frame	2.4	0.7	71%	0.8	67%
phone	28.6	12.7	56%	15.3	47%
original	44.7	33.9	24%	36.5	18%

Table 2. Results for matched near-field data

resampling	(1) MFCC	(2) MFCC-DNN		(3) PNS-Gabor DNN	
	WER	WER	Rel to (1)	WER	Rel to (1)
sim	1.8	0.5	72%	0.5	72%
frame	3.4	1.2	65%	1	71%
phone	45.5	33.5	26%	33.1	27%
original	71.4	62.8	12%	62.9	12%

Table 3. Results for matched far-field data

observation mismatch is a serious issue for MFCC-DNN. While MFCC-DNN is corrupted in the presence of serious mismatch, DNN based on PNS-Gabor performs significantly better than MFCC or MFCC-DNN for all the experiments in Table 4. For simulation, PNS-Gabor DNN reduced 43% WER relative to MFCC and 65% relative to MFCC-DNN. The results suggest that robust signal processing prior to DNN training is the key step for decreasing WER by avoiding specialization and generating more invariant features.

From our experiments, DNN features can effectively reduce recognition errors when training and test sets are matched whether they are both clean or both noisy; however, the transformation is not generalized enough to apply to realistic data with serious mismatch. Thus, robust signal processing is important for deep neural network feature extraction.

resampling	(1) MFCC	(2) MFCC-DNN		(3) PNS-Gabor DNN	
	WER	WER	Rel to (1)	WER	Rel to (1)
sim	43.0	70.9	-65%	24.5	43%
frame	59.9	76.9	-28%	43.4	28%
phone	80.6	80.8	0%	55.9	31%
original	84.7	83.6	1%	70.1	17%

Table 4. Results for mismatched scenario

Given the DNN results in mismatched scenario, we are curious about the difference of DNN transformations learned from near-field data and far-field data. It is infeasible to compare the parameters of two nets, so we train another net where we initialize it using final net trained from near-field data and then train the net using far-field data. Thus, the net starts at near-filed data trained weights and then eventually adapts to far-field data. The experiments allow us to compare the weights moving from near-field data to far-field data. We conduct the experiments for both MFCC and PNS-Gabor input. Applying the adapted nets in the mismatched case, WER is reduced from 83.6% to 65.8% for MFCC-DNN and from 70.1% to 65.3% for PNS-Gabor DNN. Figure 2 shows the initial and the final adapted weights for first hidden layer using MFCC (2a) and PNS-Gabor (2b) input where each data point consists of two variables: initial weight (*X-axis*) and the final weight (*Y-axis*). We observe that deviation from initial weight is more evident for MFCC input and the deviation for PNS-Gabor input is relative small. The root mean square deviation between initial weights and final weights for MFCC is 0.25 and 0.12 for PNS-Gabor input. Thus, the net based on PNS-Gabor is more invariant for different data. In particular, Figure 3 shows the initial and final weights learned by two hidden nodes of the first hidden layer for 9 frames of a MFCC (C1) and a PNS-Gabor input. The examples illustrate greater insensitivity to different data for the DNN transformation using PNS-Gabor input. Similar characteristic can be observed for the following layers in our experiments. As

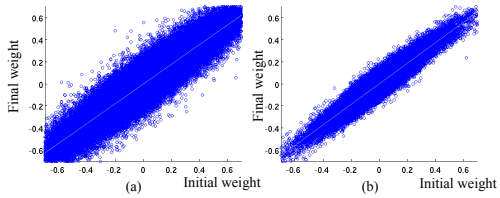

Figure 2. Initial weights and final weights of first hidden layer for MFCC input (a) and PNS-Gabor input (b)

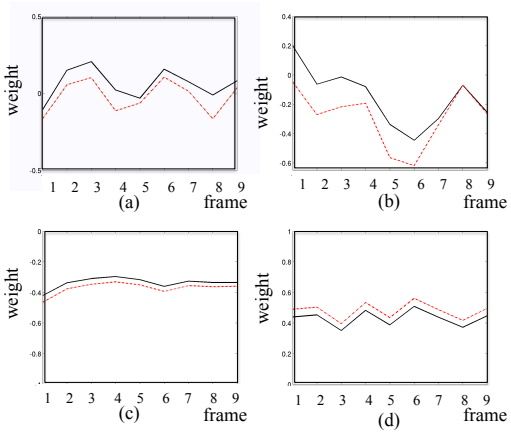

Figure 3. Initial weight (black line) and final weight (red dashed line) learned from 2 hidden nodes for 9 frames of a MFCC (C1) (above: (a) and (b)) and a PNS-Gabor input (below: (c) and (d))

both the PNS-Gabor input and the following neural network transformation is more invariant, it explains the reason that PSN-Gabor DNN is better than MFCC-DNN in mismatched scenario.

12. CONCLUSION

In this paper, we exploited the method of simulation and resampling to investigate the success and failings of deep neural network features in different train/test scenarios. Diagnosis shows that DNN-based feature representation is indeed superior to the corresponding MFCC in both two matched scenarios where the incorrect independent assumption of HMM dominates recognition errors. However, evidence from simulation and resampling experiments reveals that MFCC-DNN feature is easily specialized to training data and the observation mismatch dominates the source of recognition errors in mismatched scenario. On the other hand, DNN-based feature that uses PNS-Gabor, performs nearly identical as MFCC-DNN features in the matched scenarios and much better than MFCCs and MFCC-DNNs in mismatched scenario. Thus, modeling and robustness are the key steps to improve ASR performance using deep neural network.

12. ACKNOWLEDGEMENTS

The authors would like to thank to Nelson Morgan, Hari Parthasarathi, Dan Gillick, and Richard Stern for help with features and data sets, and Suman Ravuri for helpful comments. This material is based upon work supported by the National Science Foundation under Grant No. IIS-1450916.

12. REFERENCES

[1] H. Bourlard and N. Morgan, "Connectionist Speech Recognition: A Hybrid Approach", Kluwer Press, 1993

[2] H. Hermansky, D. P. W. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional HMM systems," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., 2000, vol. 3, pp. 1635–1638.

[3] Q. Zhu, B. Chen, N. Morgan, and A. Stolcke, "On using MLP features in LVCSR," in Proc. Interspeech, 2004, pp. 921–924

[4] A. Mohamed, G. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," Audio, Speech, and Language Processing, IEEE Transactions on, vol. 20, no. 1, pp. 14 –22, Jan. 2012.

[5] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, and T. N. Sainath, "Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups," Signal Processing Magazine, IEEE, vol. 29, no. 6, p. 8297, 2012.

[6] G. E. 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, p. 3042, 2012.

[7] O. Abdel-Hamid, L. Deng, and D. Yu. "Exploring convolutional neural network structures and optimization for speech recognition," Proc. Interspeech, 2013

[8] K. Vesely, M. Karafiat, and Frantisek Grezl, "Convolutive bottleneck network features for LVCSR," in Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on. IEEE, 2011 pp. 42–47.

[9] A. Graves, A. Mohamed, and G. Hinton, "Speech Recognition with Deep Recurrent Neural Networks.," in Proc. ICASSP. pp. 6645-6649, 2013.

[10] Hasim Sak, Andrew Senior, Franoise Beaufays, "Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling" Proc. Interspeech 2014

[11] "Li Deng, John C. Platt", Ensemble Deep Learning for Speech Recognition Proc. Interspeech 2014

[12] Bengio, Y. (2009). Learning deep architectures for AI. Foundations and Trends in Machine Learning, 2, 1–127. Also published as a book. Now Publishers, 2009

[13] S.Y. Chang, B. Meyer and N. Morgan "Spectro temporal features for noise-robust speech recognition using power-law nonlinearity and power-bias subtraction", Proc. ICASSP 2013

[14] C. Kim and R. M. Stern, "Feature extraction for robust speech recognition based on maximizing the sharpness of the power distribution and on power flooring", in Proc. ICASSP, pp. 4574–4577, 2010.

[15] D. Gillick, L. Gillick, and S. Wegmann, "Dont Multiply Lightly: Quantifying Problems with the Acoustic Model Assumptions in Speech Recognition," in Proceedings of ASRU. 2011, pp. 71–76, IEEE

[16] N. Mesgarani, and S. Shamma, "Speech Processing with a Cortical Representation of Audio", Proc. ICASSP 2011, May 2011, pp. 5872-5875.

[17] M. Kleinschmidt, "Localized spectro-temporal features for automatic speech recognition," in Proc. of Eurospeech, 2003, Sep 2003, pp. 2573–2576.

[18] H. 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 Proc, ICASSP pp. 6758-6762, 2013

[19] 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." Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003), 2003.

[20] 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.

[21] O. Cetin and A. Stolcke, "Language modeling in the ICSI-SRI Spring 2005 meeting speech recognition evaluation system," Tech. Rep., International Computer Science Institute, 2005.