Audio Augmentation for Speech Recognition

Tom Ko[1], *Vijayaditya Peddinti*[2], *Daniel Povey*[2,3], *Sanjeev Khudanpur*[2,3]

[1]Huawei Noah's Ark Research Lab, Hong Kong, China
[2]Center for Language and Speech Processing &
[3]Human Language Technology Center of Excellence,
Johns Hopkins University, Baltimore, MD, 21218, USA
{tomkocse,dpovey}@gmail.com, {vijay.p,khudanpur}@jhu.edu

Abstract

Data augmentation is a common strategy adopted to increase the quantity of training data, avoid overfitting and improve robustness of the models. In this paper, we investigate audio-level speech augmentation methods which directly process the raw signal. The method we particularly recommend is to change the speed of the audio signal, producing 3 versions of the original signal with speed factors of 0.9, 1.0 and 1.1. The proposed technique has a low implementation cost, making it easy to adopt. We present results on 4 different LVCSR tasks with training data ranging from 100 hours to 1000 hours, to examine the effectiveness of audio augmentation in a variety of data scenarios. An average relative improvement of 4.3% was observed across the 4 tasks.

Index Terms: speech recognition, data augmentation, deep neural network

1. Introduction

Data augmentation is a common strategy adopted to increase the quantity of training data. In [1, 2], corrupting clean training speech with noise was found to improve the robustness of the speech recognizer against noisy speech. With deep neural network (DNN) based acoustic modeling, vocal tract length perturbation (VTLP) [3], has shown gains on the TIMIT phoneme recognition task. VTLP was further extended to large vocabulary continuous speech recognition (LVCSR) in [4]. In [5, 6] the use of data augmentation on low resource languages, where the amount of training data is comparatively small ($\sim$ 10 hrs), was investigated. In [5] multiple data augmentation schemes were combined.

In this paper we report experiments with audio speed perturbation. This emulates a combination of tempo perturbation and VTLP, but we show it to perform better than either of those two methods.

In our experiments on the Switchboard (SWB) benchmark task, a 6.7% relative improvement in WER was obtained using the proposed data augmentation method over a state of the art DNN setup [7]. We present results on 4 different LVCSR tasks, with training data ranging from 100 to 960 hours to show the applicability of the proposed method in various scenarios.

This paper is organized as follows. Section 2 introduces the speed perturbation technique, Section 3 describes the experimental setup, Section 4 discusses the results and conclusions are presented in Section 5.

2. Audio perturbation

In this section we describe a speed-perturbation technique for data augmentation and compare it with the existing augmentation technique VTLP [3]. Speed perturbation produces a warped time signal. Given an audio signal $x(t)$, time warping by a factor α gives the signal $x(\alpha t)$. It can be seen from the Fourier transform of $x(\alpha t)$, $\alpha^{-1}\hat{x}(\alpha^{-1}\omega)$, that the warping factor produces shifts in the frequency components of the $\hat{x}(\omega)$ by an amount proportional to frequency ω. In [8] it was shown that this corresponds approximately to a shift of the spectrum in the mel spectrogram, since the mel scale is approximately logarithmic. It can be seen that these changes in the mel spectrogram are similar to those produced using VTLP. However, unlike VTLP, speed perturbation results in a change in the duration of the signal which also affects the number of frames in the utterance.

Speed perturbation differs from VTLP in one other aspect. When the speed of the signal is reduced, i.e, for $\alpha < 1$, there is a shift in the signal energy towards lower frequencies. This results in FFT bins with close to zero energy at higher frequencies. This likely means that some of the higher Mel bins end up with very small energies. However this does not seem to cause a problem in practice.

In order to implement speed perturbation, we resample the signal using the *s*peed function of the *Sox* audio manipulation tool [9].

3. Experimental Setup

We report results on LVCSR tasks in English and Mandarin. Initial experiments are conducted on the 300 hour Switchboard (SWB) English conversational telephone speech task and the observations are validated with the Gale Mandarin data set. We also present results on the TedLIUM [10] and Librispeech [11] LVCSR tasks.

For the Switchnoard task, results are presented on the Hub5 '00 evaluation set. This contains 20 conversations from Switchboard (SWBD) and 20 conversations from CallHome English (CHE). The CallHome data tends to be harder to recognize, partly due to a greater prevalence of foreign-accented speech. In this paper, we present results on both of these subsets as well as the complete Hub5 '00 evaluation set.

3.1. Language Model

For the Switchboard task, we use SWB-1 Release 2 (LDC97S62) as the training set, together with the Mississippi

This work was partially supported by NSF Grant No IIA 0530118 and DARPA BOLT Contract No HR0011-12-C-0015.

State transcripts[1] and the 30Kword lexicon released with those transcripts. The lexicon contains pronunciations for all words and word fragments in the training data. We use the first 4K sentences (about 5 hrs) from the training set as the development set and Hub5 00 (LDC2002S09) data as a separate test set. A 4-gram language model (LM) is trained[2] on 3M words of the training transcripts, which is then interpolated with another trigram LM trained on 22M words of the Fisher English Part 1 (LDC2004T19) and Part 2 (LDC2005T19) transcripts.

For the Mandarin task, we use GALE Phase 2 Chinese Broadcast News Speech (LDC2013S08) and the associated transcripts (LDC2013T20). This data is split into a training set (about 104 hrs) and a test set (about 6 hrs). A trigram LM is trained[3] on 700K words of the training transcripts.

3.2. Acoustic model

Time-delay neural network (TDNN) based acoustic models [7] are used in our experiments. These models provide state of the art performance on various LVCSR tasks. Hence they provide a strong baseline to verify the gains due to the proposed data augmentation technique. This TDNN architecture has 4 hidden layers with layerwise temporal contexts of $[-2, 2]$, $\{-1, 2\}$, $\{-3, 3\}$ and $\{-7, 2\}$.

The TDNN uses the p-norm non-linearity [12]. This dimension reducing non-linearity is a generalization of the max-out nonlinearity. Given affine transform outputs $x(t)_{i,j}$ indexed by j at layer i and time t, the activations $y(t)_{i,k}$ are computed as shown in Equation 1, for a group size G and N p-norm units.

$$y(t)_{i,k} = (\sum_{j=kG}^{(k+1)G-1} |x(t)_{i,j}|^{p})^{\frac{1}{p}} \qquad (1)$$

$$for\ k \in [1, N]$$

A group size of 10 and 2-norm were used across all neural networks in our experiments, based on the observations in [12]. As the p-norm non-linearity has an unbounded output, which can lead to instabilities in training, each p-norm layer was followed by a normalization layer. This layer scales the input vector by its root mean square value. This layer is applied during both training and testing.

$$\sigma = \sqrt{\frac{1}{N} \sum_{k} y(t)_{i,k}^{2}}$$

$$h(t)_{i,k} = y(t)_{i,k} / \sigma \qquad (2)$$

Thus the layer scales down the p-norm outputs $y(t)_{i,k}$ to ensure that the vector $\mathbf{h(t)_i}$ has a norm of 1. p-norm layers with input dimension of 2750 were used.

3.2.1. Input features

Mel-frequency cepstral coefficients (MFCCs) ([13]), without cepstral truncation, were used as input to the neural network. 40 MFCCs were computed at each time index. The input MFCCs are provided to the neural network over a wide asymmetric temporal context. Different input temporal contexts were explored in this paper. 100 dimensional i-vectors were also provided as an input to the network, every time frame to perform instantaneous speaker adaptation of the network ([14]).

3.2.2. Training recipe

The paper follows the training recipe detailed in [12]. It uses greedy layer-wise supervised training, preconditioned stochastic gradient descent (SGD) updates, an exponentially decreasing learning rate schedule and *mixing-up*. Parallel training of the DNNs using up to 18 GPUs was done using the model averaging technique in [15].

The same TDNN architecture was used across all the experiments on the Switchboard task. However the number of training epochs was varied. The baseline TDNN without data augmentation was trained for 6 epochs. For TDNNs trained on augmented data due to increase in training data, the number of epochs was reduced to keep the overall training time similar to the baseline system.

3.3. VTLP based data augmentation

In [3] the VTLP warping factors for each utterance is randomly chosen from a range (e.g. $[0.9, 1.1]$). Using these sampled warping factors, improvement was reported on TIMIT phoneme recognition task. In [4], VTLP was used in large vocabulary continuous speech recognition (LVCSR) tasks, and an observation was made that selecting VTLP warping factors from a limited set of perturbation factors, was better.

In this paper, we follow the VTLP implementation in [4] with the exception that we use the same warping factors for all the speakers in the training set. Two sets of warping factors, $\{0.9, 1.0, 1.1\}$ and $\{0.9, 0.95, 1.0, 1.05, 1.1\}$, are used to create 3 and 5 copies of the original feature vectors, respectively. These two sets of training data were used to train two different DNN systems, which are tagged as 3-fold and 5-fold systems in the comparison.

3.4. Tempo perturbation based data augmentaion

Speech rate perturbation, where the speech rate of the audio was modified by randomly selected factor, was investigated in [6]. In speech rate modification, the tempo of the signal is modified while ensuring that the pitch and spectral evelope of the signal does not change. The WSOLA [16] based implementation in the *tempo* command of the *SoX* tool was used to achieve this perturbation.

Two additional copies of the original training data were created by modifying the tempo to 90% and 110% of the original rate. This creates a 3-fold training set, which is tagged as such in the comparison tables. Alignments of the tempo modified data are regenerated using the GMM-HMM system.

3.5. Speed perturbation based data augmentation

To modify the speed of a signal we just resample the signal. The *speed* function of *Sox* was used for this. Two additional copies of the original training data were created by modifying the speed to 90% and 110% of the original rate. This creates a 3-fold training set, which is tagged as such in the comparison tables. Due to the change in the length of the signal, the alignments for the speed perturbed data are regenerated using the GMM-HMM system.

4. Results and Discussion

Table 1 presents the results on the Switchboard LVCSR task. A relative improvement of 4.8% was observed on the total Hub5 '00 evaluation set, when using speed perturbed training data. Speed perturbation was found to be better than VTLP based

[1] Available from: http://www.isip.piconepress.com/

[2] Location in scripts: egs/swdb/s5c/run.sh

[3] Location in scriptsg: egs/gale_mandarin/s5/run.sh (revision 4970)

Table 1: *Results (% WER) for the baseline and speed-perturbed DNN systems on the subsets of the Hub5 00 evaluation set.*

System	Fold	Epochs	LM	SWB	CHE	Total
Baseline	1	6	fg	13.7	27.7	20.7
VTLP	3	2	fg	13.1	26.5	19.9
VTLP	5	2	fg	13.2	26.7	20.0
VTLP + time-warp	3	2	fg	13.3	26.8	20.1
Tempo-perturbed	3	2	fg	13.5	27.0	20.3
Speed-perturbed	3	2	fg	13.1	26.1	19.7
Speed-perturbed	3	6	fg	12.9	25.7	19.3

Table 2: *Results (% WER) for the baseline and speed-perturbed DNN systems on the GALE Mandarin test set.*

System	Fold	Epochs	LM	Pitch	Total
Baseline	1	6	tg	N	18.46
Baseline	1	12	tg	N	18.63
Speed-perturbed	3	2	tg	N	18.34
Speed-perturbed	3	6	tg	N	**18.09**
Baseline	1	6	tg	Y	17.51
Baseline	1	12	tg	Y	17.63
Speed-perturbed	3	2	tg	Y	17.56
Speed-perturbed	3	6	tg	Y	**17.16**

Table 3: Comparison of baseline and speed-perturbation on various LVCSR tasks with different amount of training data

LVCSR task	Hrs of training data	WER		Rel. improvement
		Baseline	Speed-perturbed	
GALE Mandarin	100 hrs	17.51	17.16	2.0
Tedlium	118 hrs	17.9	17.2	3.9
Switchboard	300 hrs	20.7	19.3	6.7
Librispeech	960 hrs	12.93	12.51	3.2
ASpIRE	5500 hrs	30.8	30.7	0.32

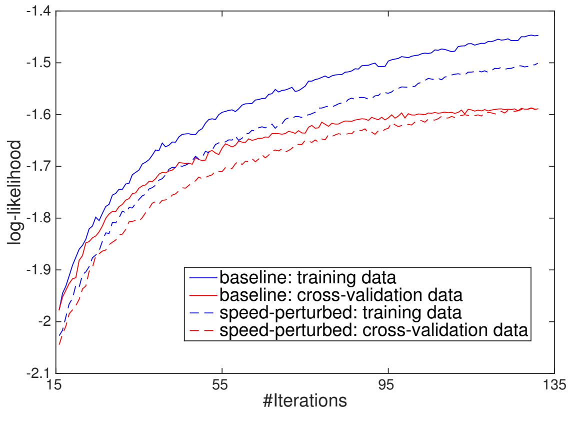

Figure 1: Average likelihood of training and cross-validation data across iterations

augmentation. As discussed before, speed perturbation emulates VTLP perturbation combined with time warping of the feature time indices. However, even a combination of VTLP and time-warping was not better than the speed perturbed system. The addition of time warping to VTLP was actually found to be detrimental. Additionally, we tried increasing the number of perturbation factors used in VTLP from 3 to 5; however, this seemed to be detrimental. We conclude that 3-fold augmentation of data is sufficient for VTLP systems.

Using tempo perturbation was beneficial compared to the baseline. However it was not better than either VTLP or speed perturbation. It is to be noted that tempo perturbation does not involve perturbation of the log Mel spectral envelopes; on the other hand both VTLP and speed perturbation involve some perturbation of these envelopes.

Figure 1 shows log-likelihood plots on training and cross-validation data, for baseline and speed perturbed systems. We found that using speed perturbed training data led to better generalization, as measured from the difference between frame likelihoods of training and validation data. DNNs being trained on speed perturbed data still had a training data likelihood which was lower than baseline systems. Hence we trained the speed-perturbed system for few more epochs. This was found to improve the results. A corresponding increase in the number of epochs for the baseline system deteriorated the performance (see Table 2).

Table 3 compares the performance improvement from speed perturbation across a variety of LVCSR tasks with a varying amount of training data. It can be seen that data augmentation was helpful on all the tasks irrespective of amount of training data. In the ASpIRE far field recognition task, however, the improvement was much less than the other tasks. This is a special case because in this task, the data was already augmented to create reverberant copies of training data. Speed perturbation was performed on the audio signals before convolving them with room impulse responses. The minimal gains seen in this task could be attributed to the fact that reverberation already created sufficient perturbation in the data in the baseline system.

From Table 2 we can see that the data augmentation techniques also helped in the case of the Gale Mandarin LVCSR task. Increasing the number of training epochs led to better WERs only in the case of speed-perturbed systems. Pitch and voicing features, when combined with MFCCs, were found to be helpful in many LVCSR tasks. We extracted these features [17] for both baseline and speed-perturbed systems. The gains due to data augmentation were consistent. Speed perturbation of the training data led to a relative improvement of 2% on this task.

5. Conclusions

In this paper we presented an audio augmentation technique with low implementation cost. Speed perturbation, which emulates both VTLP and tempo perturbation, is shown to give more WER improvement than either of those methods. The experiments were performed using state-of-the-art DNN systems, with training data ranging from 100 to 960 hours, including a task where pitch and voicing features were included. However, we saw very little improvement on the ASpIRE challenge, possibly because the data had already been augmented by simulated reverberation.

6. Acknowledgements

This work was partially supported by NSF Grant No IIA 0530118.

7. References

[1] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Y. Ng, "Deep speech: Scaling up end-to-end speech recognition," *CoRR, abs/1412.5567*, 2014.

[2] M. J. F. Gales, A. Ragni, H. AlDamarki, and C. Gautier, "Support vector machines for noise robust asr," in *ASRU*, 2009, pp. 205–210.

[3] N. Jaitly and G. E. Hinton, "Vocal tract length perturbation (VTLP) improves speech recognition," in *International Conference on Machine Learning (ICML) Workshop on Deep Learning for Audio, Speech, and Language Processing*, 2013.

[4] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic modeling," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 100–104.

[5] A. Ragni, K. M. Knill, S. P. Rath, and M. J. F. Gales, "Data augmentation for low resource languages," in *Interspeech*, 2014.

[6] N. Kanda, R. Takeda, and Y. Obuchi, "Elastic spectral distortion for low resource speech recognition with deep neural networks," in *ASRU*, 2013.

[7] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proceedings of INTERSPEECH*, 2015.

[8] J. Andén and S. Mallat, "Deep scattering spectrum," *Signal Processing, IEEE Transactions on*, vol. 62, no. 16, pp. 4114–4128, Aug 2014.

[9] *SoX, audio manipulation tool*, (accessed March 25, 2015). [Online]. Available: http://sox.sourceforge.net/

[10] A. Rousseau, P. Deléglise, and Y. Estève, "Ted-lium: an automatic speech recognition dedicated corpus." in *LREC*, 2012, pp. 125–129.

[11] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015.

[12] X. Zhang, J. Trmal, D. Povey, and S. Khudanpur, "Improving deep neural network acoustic models using generalized maxout networks," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, May 2014, pp. 215–219.

[13] S. B. Davis and P. Mermelstein, "Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[14] M. Karafiat, L. Burget, P. Matejka, O. Glembek, and J. Cernocky, in *2011 IEEE Workshop on Automatic Speech Recognition & Understanding*. IEEE, Dec., pp. 152–157.

[15] D. Povey, X. Zhang, and S. Khudanpur, "Parallel training of deep neural networks with natural gradient and parameter averaging," *CoRR*, vol. abs/1410.7455, 2014. [Online]. Available: http://arxiv.org/abs/1410.7455

[16] W. Verhelst and M. Roelands, "An overlap-add technique based on waveform similarity (wsola) for high quality time-scale modification of speech," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, April 1993, pp. 554–557 vol.2.

[17] P. Ghahremani, B. BabaAli, D. Povey, K. Riedhammer, J. Trmal, and S. Khudanpur, "A pitch extraction algorithm tuned for automatic speech recognition," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 2494–2498.