Training Neural Speech Recognition Systems with Synthetic Speech Augmentation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Building an accurate automatic speech recognition (ASR) system requires a large dataset that contains many hours of labeled speech samples produced by a diverse set of speakers. The lack of such open free datasets is one of the main issues preventing advancements in ASR research. To address this problem, we propose to augment a natural speech dataset with synthetic speech. We train very large end-to-end neural speech recognition models using the LibriSpeech dataset augmented with synthetic speech. These new models achieve state of the art Word Error Rate (WER) for character-level based models without an external language model.

1 Introduction

There has been a large amount of success in using neural networks (NN) for automatic speech recognition (ASR). Classical ASR systems use complicated pipelines with many heavily engineered processing stages, including specialized input features, acoustic models, and Hidden Markov Models (HMMs). Deep NNs have traditionally been used for acoustic modeling (Weibel et al., 1989; Hinton et al., 2012). A key breakthrough occurred when a state of the art ASR system, Deep Speech, was built using an end-to-end deep learning approach (Hannun et al., 2014). This model replaced both acoustic modeling, and HMMs with very deep neural networks and was able to directly translate spectrograms into English text transcriptions. This end-to-end approach was extended in follow-up papers (Amodei et al., 2016; Collobert et al., 2016). Recent advances in ASR have further improved upon these models using even more advanced techniques such as replacing n-gram language models with a neural language model, usually in the form of a recurrent neural network (RNN) (Zeyer et al., 2018; Povey et al., 2018; Han et al., 2018).

As opposed to making neural networks more complex, we were interested in an orthogonal direction of study: whether we can improve quality by creating larger models. In order to train such large models, deep NNs require a vast quantity of data to be available. However the lack of a large public speech dataset blocked us from successfully building large NN models for ASR. We were inspired by recent work in improving translation systems using synthetic data (Sennrich et al., 2015), and as such, we decided to augment speech with synthetic data. [1] Furthermore, given the recent impressive improvement in neural speech synthesis models (van den Oord et al., 2016; Shen et al., 2018), it becomes cheap to generate high quality speech with varying prosody.

We show that by using synthetic speech created from a neural speech synthesis model, we can improve ASR performance compared to models trained using only LibriSpeech data (Panayotov et al., 2015). By naively increasing the depth of the model, we observe that the synthetic data allows us to achieve state of the art WER using a greedy decoder.

[1] Synthesized speech has previously been used to improve speech recognition for low-resource languages (Rygaard, 2015).

2 Synthetic Speech Dataset

We use the Tacotron-2 like model from the OpenSeq2Seq [2] toolkit (Kuchaiev et al., 2018) and add Global Style Tokens (GST) (Wang et al., 2018) to learn multiple speaker identities. Tacotron-2 with GST (T2-GST) was trained on the MAILABS English-US dataset (M-AILABS, 2018) with approximately 100 hours of audio recorded by 3 speakers. T2-GST was able to learn all 3 different speaking styles and different accents that they use to portray different characters.

Using the T2-GST model, we created a fully synthetic version of the LibriSpeech training audio. In order to produce audio, T2-GST requires a spectrogram used for the style and the audio transcription. For the transciption, we took the transcripts from the train-clean-100, train-clean-360, and train-other-500 LibriSpeech splits and randomly paired them with style spectrograms from the MAILABS English-US dataset. At the end, we had a dataset that was the same size as the original training portion of the LibriSpeech dataset but spoken in the tones of the speakers from the MAILABS dataset.

The audio from the T2-GST model could be further controlled by the amount of dropout in the prenet of the decoder. By decreasing the dropout rate, we found that we could slightly distort the audio. The main difference that we noticed was that the lower the dropout rate, the faster the resulting audio would sound. Using this observation, we further increased the size of the synthetic dataset. Thus, we used 46%, 48%, and 50% for the dropout and created a synthetic speech dataset that was 3 times as large as the LibriSpeech training dataset.

3 Training Speech Recognition with Synthetic Data

3.1 Neural Speech Recognition Models

Our speech recognition model is an end-to-end neural network that takes logarithmic mel-scale spectrograms as inputs and produces characters. We use a deep convolutional NN model, which we will further address as Wave2Letter+ (w2lp) [3] . It is based on Wav2Letter (Collobert et al., 2016) except:

- We use ReLU instead of Gated Linear Unit
- We use batch normalization instead of weight normalization
- We add residual connections between convolutional blocks
- We use Connectionist Temporal Classification loss instead of Auto Segmentation Criterion
- We use Layer-wise Adaptive Rate clipping (LARC) for gradient clipping

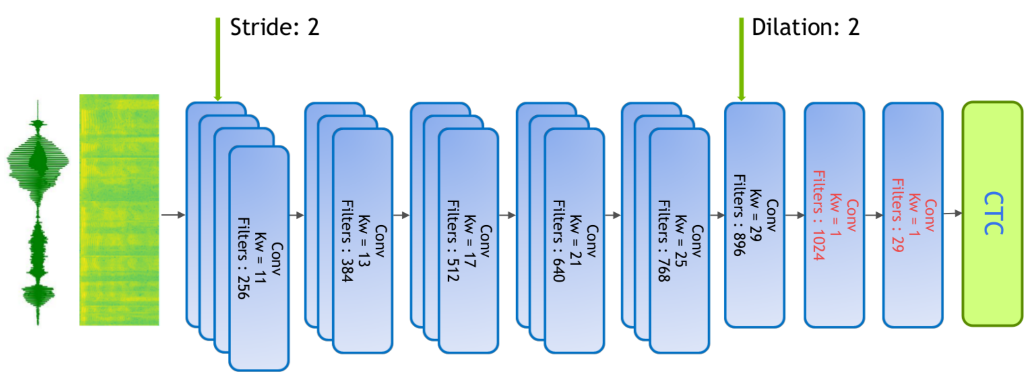

Figure 1: Base Wave2Letter+ model.

[2]We used OpenSeq2Seq both to create a synthetic speech dataset and build ASR.
[3]More model details: https://nvidia.github.io/OpenSeq2Seq/html/speech-recognition/wave2letter.html

The base Wave2Letter+ model has 19 convolutional layers:

- 1 pre-processing layer at the beginning of the network
- 15 layers consists of 5 blocks of 3 repeating convolutional layers
- 3 post-processing layers at the end of the network

We made the base model deeper and experimented with 24, 34, 44, and 54 layer networks. The 24 layer networks consists of 5 blocks of 4 repeating convolutional layers. The 34, 44, and 54 layer network consists of 10 blocks of 3, 4, and 5 repeating layers respectively.

3.2 Word Error Rate Improvement using Synthetic Augmentation

The training dataset was created by combining the synthetic data with the original training data from LibriSpeech. Despite the synthetic data being larger than the natural data, we found it most helpful to sample natural and synthetic data at a 50/50 rate.

Our best performing model, the 54 layer network, currently has a word error rate of 4.32% on test-clean and a word error rate of 14.08% on test-other by greedily choosing the most probable character at each step without any language model rescoring. To the best of our knowledge, the previous best performing model without using any language model achieves 4.87% on test-clean and 15.39% on test-other (Zeyer et al., 2018). The complete results can be found in Table 1. Models trained on the combined dataset outperform those trained on original LibriSpeech. The augmented dataset improves results on test-clean by 0.15, and 0.44 for the 24 and 34 layer models. For test-other, we see an improvement of 0.08, and 0.74 for the 24 and 34 layer models.

By using beam search with beam width 128 and the 4-gram OpenSLR [4] language model rescoring, we improved our WER on test-other to 12.21% on the 54 layer model which is better than previous public 4-gram language models and comparable to LSTM language models (Zeyer et al., 2018).

Model	Dataset Used	Dev		Test	
		Clean	Other	Clean	Other
attention-Zeyer et al.	LibriSpeech	4.87	14.37	4.87	15.39
w2lp-24	LibriSpeech	5.44	16.57	5.31	17.09
w2lp-24	Combined	5.12	16.25	5.16	17.01
w2lp-34	LibriSpeech	5.10	15.49	5.10	16.21
w2lp-34	Combined	4.60	14.98	4.66	15.47
w2lp-44	Combined	4.24	13.87	4.36	14.37
w2lp-54	Combined	4.32	13.74	4.32	14.08

Table 1: Greedy WER on LibriSpeech for Different Models and Datasets

3.3 How To Mix Natural and Synthetic Speech

We performed a number of additional experiments to find the best sampling ratio between synthetic data and LibriSpeech. We tested training on only LibriSpeech, a 50/50 split, a 33/66 split, and a pure synthetic dataset. All tests were done on the 34 layer model. The results are shown in Table 2.

Model	Sampling Ratio	Dev		Test	
	(Natural/Synthetic)	Clean	Other	Clean	Other
w2lp-34	Natural	5.10	15.49	5.10	16.21
w2lp-34	50/50	4.60	14.98	4.66	15.47
w2lp-34	33/66	4.91	15.18	4.81	15.81
w2lp-34	Synthetic	51.39	80.27	49.80	81.78

Table 2: Greedy WER for Different Ratios Between Natural and Synthetic Datasets

[4]The LM can be found here: www.openslr.org/11

Despite the larger amount of synthetic data, the synthetic dataset fails to capture the larger variety of LibriSpeech. We believe that this effect could be moderated if a speech synthesis model with larger speaker variety was used as opposed to the current 3 speaker speech synthesis model. A 50/50 split between the natural and synthetic seems to be a good ratio for our dataset.

3.4 Traditional Speech Augmentation vs Synthetic Speech

Adding synthetic data proved to be better regularization than standard regularization techniques. In addition to dropout which is employed for all models, OpenSeq2Seq supports speech augmentation such as adding noise and time stretching. Using these 3 techniques, we tested 4 additional models. We tested 2 larger dropout factors, and on top on this, we tested whether speech augmentation would improve performance. Since the dropout factor varies by layer, we multiply the local dropout keep probabilities by a global dropout keep factor. All tests were done on the 34 layer model. The tests and results are detailed in Table 3.

A slightly larger dropout resulted in minor improvement in WER. The effects of speech augmentation seem to be negligible or, in the case of large dropout, make WER worse. Adding synthetic data significantly outperforms all other methods of regularization.

Model	Dropout Keep Factor	Time Stretch Factor	Noise (dB)	Dev		Test	
				Clean	Other	Clean	Other
LibriSpeech	None	None	None	5.10	15.49	5.10	16.21
Dropout	0.9	None	None	5.01	15.15	5.15	15.70
Dropout + Aug	0.9	0.05	[-90, -60]	5.07	15.00	5.02	15.83
Dropout	0.75	None	None	5.46	15.77	5.39	16.62
Dropout + Aug	0.75	0.1	[-90, -60]	5.80	16.33	5.72	17.41
Combined	None	None	None	4.60	14.98	4.66	15.47

Table 3: Greedy WER on Using Different Regularization Techniques

4 Conclusion and Future Plans

Using synthetic data is an effective way to build large neural speech recognition systems. The synthetic data should be combined with the natural data in the correct ratio to obtain best results. With this method, we achieved a WER of 4.32% on test-clean and a WER of 14.08% on test-other using a greedy decoder. This is the current state of the art on character-level based greedy decoding. Furthermore, using a language model and a beam search width of 128, we get 12.21% WER on test-other.

For future studies, we are interested in creating a larger synthetic dataset with noise. For now, we have restricted ourselves to take transcriptions from the training subsets of LibriSpeech, but the speech synthesis models are general enough to accept any transcript. It would be interesting to scrape text from another source and add to the training set additional phrases not found in LibriSpeech.

Acknowledgments

We would like to thank Rafael Valle, Ryan Leary, Igor Gitman, Oleksii Kuchaiev and Ujval Kapasi for helpful conversation and advice throught this project.

References

Dario Amodei, Rishita Anubhai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Jingdong Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Erich Elsen, Jesse Engel, Linxi Fan, Christopher Fougner, Awni Y. Hannun, Billy Jun, Tony Han, Patrick LeGresley, Xiangang Li, Libby Lin, Sharan Narang, Andrew Y. Ng, Sherjil Ozair, Ryan Prenger, Sheng Qian, Jonathan Raiman, Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Chong Wang, Yi Wang, Zhiqian Wang, Bo Xiao, Yan Xie, Dani Yogatama, Jun Zhan, and Zhenyao Zhu. 2016. Deep speech 2 : End-to-end speech recognition in english and mandarin. In *ICML*.

Ronan Collobert, Christian Puhrsch, and Gabriel Synnaeve. 2016. Wav2letter: an end-to-end convnet-based speech recognition system. *arXiv preprint arXiv:1609.03193*.

Kyu J. Han, Akshay Chandrashekaran, Jungsuk Kim, and Ian R. Lane. 2018. The CAPIO 2017 conversational speech recognition system. *CoRR*, abs/1801.00059.

Awni Y. Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, and Andrew Y. Ng. 2014. Deep speech: Scaling up end-to-end speech recognition. *CoRR*, abs/1412.5567.

Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Brian Kingsbury, and Tara Sainath. 2012. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*.

Oleksii Kuchaiev, Boris Ginsburg, Igor Gitman, Vitaly Lavrukhin, Carl Case, and Paulius Micikevicius. 2018. Openseq2seq: extensible toolkit for distributed and mixed precision training of sequence-to-sequence models. `https://arxiv.org/abs/1805.10387`.

M-AILABS. 2018. The M-AILABS Speech Dataset. `http://www.m-ailabs.bayern/en/the-mailabs-speech-dataset/`. Accessed: 2018-10-09.

Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio. *CoRR*, abs/1609.03499.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 5206–5210. IEEE.

Daniel Povey, Gaofeng Cheng, Yiming Wang, Ke Li, Hainan Xu, Mahsa Yarmohammadi, and Sanjeev Khudanpur. 2018. Semi-orthogonal low-rank matrix factorization for deep neural networks. In *Interspeech*.

Luise Valentin Rygaard. 2015. Using synthesized speech to improve speech recognition for low-resource languages. `https://parasol.tamu.edu/dreu2015/Rygaard/report.pdf`.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *CoRR*, abs/1511.06709.

Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al. 2018. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4779–4783. IEEE.

Yuxuan Wang, Daisy Stanton, Yu Zhang, R. J. Skerry-Ryan, Eric Battenberg, Joel Shor, Ying Xiao, Fei Ren, Ye Jia, and Rif A. Saurous. 2018. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. *CoRR*, abs/1803.09017.

Alexander Weibel, Toshiyki Hanazawa, Geoffrey Hinton, Kiyohiro Shirano, and kevin Lang. 1989. A time-delay neural network architecture for isolated word recognition. *IEEE Trans. on Acoustics, Speech and Signal Processing*.

Albert Zeyer, Kazuki Irie, Ralf Schlüter, and Hermann Ney. 2018. Improved training of end-to-end attention models for speech recognition. *CoRR*, abs/1805.03294.