Unifying Speech Recognition and Generation with Machine Speech Chain

Andros Tjandra[1,2] Sakriani Sakti[1,2] Satoshi Nakamura[1,2]

[1] Nara Institute of Science and Technology, Japan

[2] RIKEN AIP, Japan

{andros.tjandra.ai6, ssakti, s-nakamura}@.is.naist.jp

1 Introduction

Human speech chain was first introduced by Denes et al. [2], describes the basic mechanism involved in speech communication when a spoken message travels from the speaker's mind to the listener's mind (Fig. 1). It consists of a speech production mechanism in which the speaker produces words and generates speech sound waves, transmits the speech waveform through a medium (i.e., air), and creates a speech perception process in a listener's auditory system to perceive what was said. Over the past few decades, tremendous research effort has struggled to understand the principles underlying natural speech communication. Many attempts have also been made to replicate human speech perception and production by machines to support natural modality in human-machine interactions. In this paper, we proposed a closed-loop between ASR and TTS model to construct a machine speech chain mechanism. By using similar sequence-to-sequence architecture and interchangeable source-target domain, it allows us to train both labeled and unlabeled data in a single loop. The ASR model transcribes the unlabeled speech utterances, then TTS reconstructs the speech waveform based on the predicted text from ASR. In the opposite case, the TTS reconstructs the speech features, then ASR reconstructs the text based on predicted speech from TTS. Based on our knowledge, this is the first deep learning model that integrates human speech perception and production behaviours.

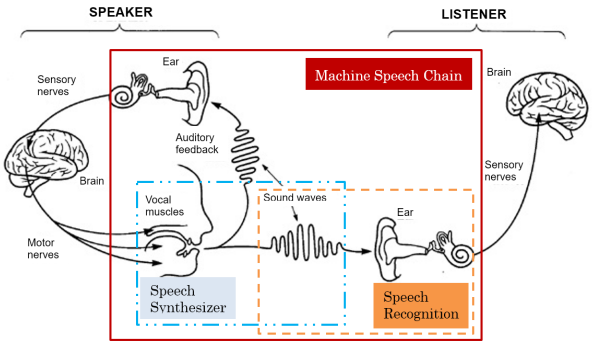

Figure 1: Speech chain [2] and related spoken language technologies.

2 Machine Speech Chain

In Fig. 2(a), we illustarted an overview of machine speech chain mechanism. It consists of a sequence-to-sequence ASR [1], a sequence-to-sequence TTS [3], and a loop connection from ASR to TTS and from TTS to ASR. The key idea is to jointly train both the ASR and TTS models. As mentioned above, the sequence-to-sequence model in closed-loop architecture allows us to train our model on the concatenation of both the labeled and unlabeled data. For supervised training with labeled data (speech-text pair data), both models can be trained independently by minimizing the loss between their predicted target sequence and the ground truth sequence. However, for unsupervised training with unlabeled data (speech only or text only), both models need to support each other through a connection.

To further clarify the learning process during unsupervised training, we unrolled the architecture as follows:

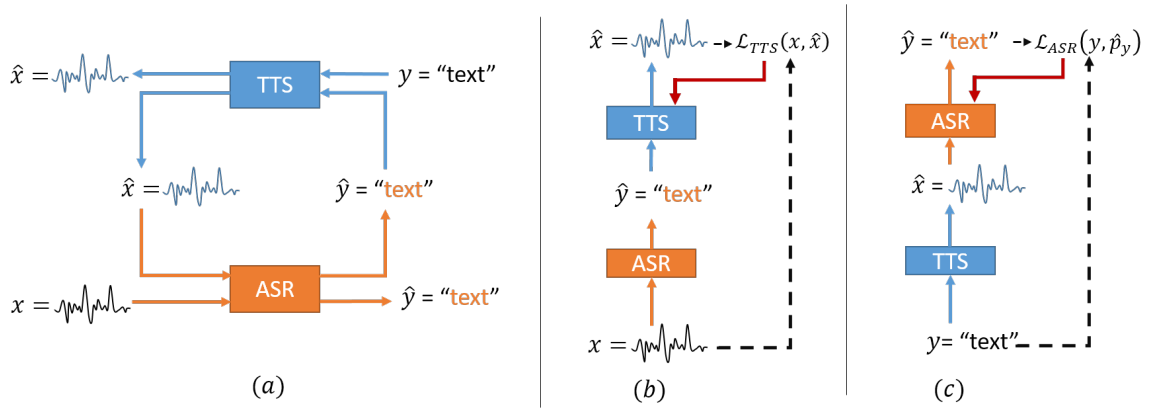

Figure 2: (a) Overview of machine speech chain architecture. Examples of unrolled process: (b) from ASR to TTS and (c) from TTS to ASR.

- **Unrolled process from ASR to TTS**
  Given the unlabeled speech features, ASR transcribes the unlabeled input speech, while TTS reconstructs the original speech waveform based on the output text from ASR. Fig. 2(b) illustrates the mechanism. We may also treat it as an autoencoder model, where the speech-to-text ASR serves as an encoder and the text-to-speech TTS as a decoder.

- **Unrolled process from TTS to ASR**
  Given only the text input, TTS generates speech waveform, while ASR also reconstructs the original text transcription given the synthesized speech. Fig. 2(c) illustrates the mechanism. Here, we may also treat it as another autoencoder model, where the text-to-speech TTS serves as an encoder and the speech-to-text ASR as a decoder.

3 Experiments

To gather a large single speaker speech dataset, we utilized Google TTS to generate a large set of speech waveform based on basic travel expression corpus (BTEC) English sentences. For training and development we used part of the BTEC1 dataset, and for testing we used the default BTEC test set. For supervised training on both the ASR and TTS models, we chose 10,000 speech utterances that were paired with their corresponding text. For our development set, we selected another 3000 speech utterances and paired them with corresponding text. For our test set, we used all 510 utterances from the BTEC default test set. For the unsupervised learning step, we chose 40,000 speech utterances just from BTEC1 and 40,000 text utterances from BTEC1.

3.1 Features Extraction

For the speech features, we used a log magnitude spectrogram extracted by short-time Fourier transform (STFT). We extracted the spectrogram with STFT (50-ms frame length, 12.5-ms frame shift, 2048-point FFT). After getting the spectrogram, we used the squared magnitude and a Mel-scale filterbank with 40 filters to extract the Mel-scale spectrogram. After getting the Mel-spectrogram, we squared the magnitude spectrogram features.We normalized each feature into 0 mean and unit variances. Our final set is comprised of 40 dims log Mel-spectrogram features and a 1025 dims log magnitude spectrogram. For the text, we converted all of the sentences into lowercase and tokenize them into a character sequence.

3.2 Model Details

Our ASR model is a encoder-decoder with an attention mechanism. On the encoder side, we used a log-Mel spectrogram as the input features, which are projected by a fully connected layer, processed by three stacked BiLSTM layers with 256 hidden units. On the decoder side, we use LSTM with 512 hidden

units, followed by an MLP attention and a softmax function.

Our TTS model hyperparameters are generally the same as the original Tacotron, except that we used LeakyReLU instead of ReLU for most of the parts. On the encoder sides, the CBHG used $K = 8$ different filter banks instead of 16 to reduce our GPU memory consumption. Our TTS predicted four consecutive frames in one time step to reduce the number of time steps in the decoding process.

3.3 Experiment Result

Table 1 shows our result on the single-speaker ASR and TTS experiments. For the ASR experiment, we generated best hypothesis with beam search (size= 5). We used a character error rate (CER) for evaluating the ASR model. For the TTS experiment, we reported the MSE between the predicted log Mel and the log magnitude spectrogram to the ground truth. We also report the accuracy of our model that predicted the last speech frame. We used different values for α and text decoding strategy for ASR (in the unsupervised learning stage) with a greedy search or a beam search.

Table 1: Experiment result for single-speaker test set.

Data	**Hyperparameters**			**ASR**	**TTS**	
	α	β	**gen. mode**	**CER (%)**	**Mel**	**Raw**
Paired (10k)	-	-	-	10.06	7.068	9.376
+ Unpaired (40k)	0.25	1	greedy	5.83	6.212	8.485
	0.5	1	greedy	5.75	6.247	8.418
	0.25	1	beam 5	5.44	6.243	8.441
	0.5	1	beam 5	5.77	6.201	8.435

The result show that after ASR and TTS models have been trained with a small paired dataset, they start to teach each other using unpaired data and generate useful feedback. Here we improved both ASR and TTS performance. Our ASR model reduced CER by 4.6% compared to the system that was only trained with labeled data. In addition to ASR, our TTS also decreased the MSE and the end of speech prediction accuracy.

4 Acknowledgements

Part of this work was supported by JSPS KAKENHI Grant Numbers JP17H06101 and JP17K00237.

References

[1] Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio. End-to-end attention-based large vocabulary speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pp. 4945–4949. IEEE, 2016.

[2] P.B. Denes and E. Pinson. *The Speech Chain*. Anchor books. Worth Publishers, 1993.

[3] Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, et al. Tacotron: Towards end-to-end speech synthesis. *arXiv preprint arXiv:1703.10135*, 2017.