DISTILLING SEQUENCE-TO-SEQUENCE VOICE CONVERSION MODELS FOR STREAMING CONVERSION APPLICATIONS

Kou Tanaka, Hirokazu Kameoka, Takuhiro Kaneko, Shogo Seki

NTT Communication Science Laboratories, NTT Corporation, Japan

2022 IEEE Spoken Language Technology Workshop (SLT) | 979-8-3503-9690-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/SLT54892.2023.10023432

ABSTRACT

This paper describes a method for distilling a recurrent-based sequence-to-sequence (S2S) voice conversion (VC) model. Although the performance of recent VCs is becoming higher quality, streaming conversion is still a challenge when considering practical applications. To achieve streaming VC, the conversion model needs a streamable structure, a causal layer rather than a non-causal layer. Motivated by this constraint and recent advances in S2S learning, we apply the teacher-student framework to recurrent-based S2S-VC models. A major challenge is how to minimize degradation due to the use of causal layers which masks future input information. Experimental evaluations show that except for male-to-female speaker conversion, our approach is able to maintain the teacher model's performance in terms of subjective evaluations despite the streamable student model structure. Audio samples can be accessed on `http://www.kecl.ntt.co.jp/people/tanaka.ko/projects/dists2svc`.

***Index Terms*—** Voice conversion, sequence-to-sequence learning, distillation, streaming conversion.

1. INTRODUCTION

Voice conversion (VC) technology, which converts one speaking style to another without changing linguistic information, has been applied for various tasks; speaker conversion [1, 2], singing conversion [3, 4], assistive systems [5, 6] to overcome speech and hearing impairments, and pronunciation and accent conversions [7] in language learning. Recently, the conversion quality has been improved thanks to sequence-to-sequence (S2S) learning approaches [2, 4, 6, 8–10]. An encoder-decoder structure, including an attention mechanism, makes it possible to learn conversion rules that reflect the long-term dependencies of input and output sequences. However, the conversion process is executable after we obtain the entire sentence because the encoder contains non-causal layers, as shown in Fig. 1. Considering practical use, waiting for the end of a speech is a major barrier against the smooth speech communication in daily life. Unlike other speech signal processing, such as text-to-speech synthesis, the effect of delayed auditory feedback [11], a well-known problem leading to unnatural speech, must also be considered since the converted speech is fed back to the auditory system. The VC task requires handling streaming processing with low latency, such as several dozen ms.

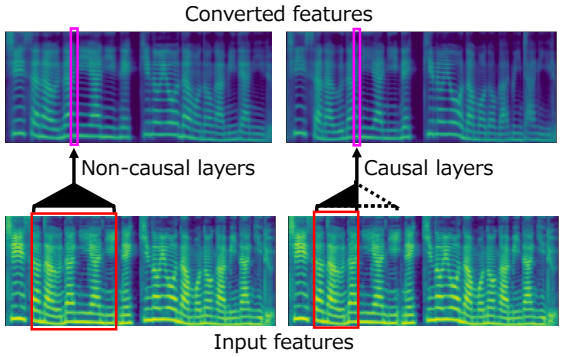

Fig. 1. Non-causal (left) and causal (right) layers. The red and pink boxes show the segments of the input feature and the frames of the converted feature. The causal layers do not take future frames as input, contrary to the non-causal layers. Real-time conversion of streaming speech data increases latency by the amount of look-ahead.

In the zero/minimal look-ahead scenario within the S2S learning, achieving low latency streaming processing with a recurrent-based approach appears possible rather than convolution/transformer-based [12, 13]. A recurrent neural network (RNN) can hold time-series information internally in a layer, while a convolution/transformer-based model requires a deep architecture to capture such information. Unfortunately, in the preliminary experiments, we confirmed that just replacing non-causal layers of the encoder with causal layers would result in converted speech with significant delay, as shown in Fig. 2. Note that this phenomenon has been reported in a speech recognition task [14]. Trying to train a model from scratch without any guides, uni-directional RNN may have been trained to generate the current output after considering inputs a little further ahead to capture the time-frequency structure accurately. During the model training, the time delay is offset by the attention mechanism. However, in the test time on real-time streaming VC, this time delay is

978-1-6654-7189-3/22/$31.00 ©2023 IEEE

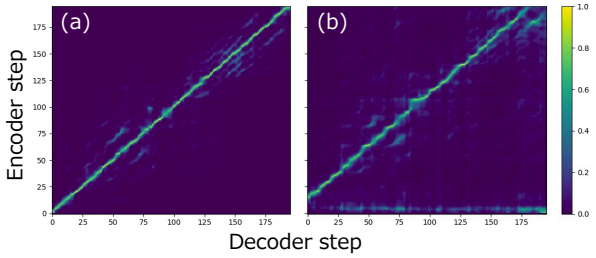

Fig. 2. Attention matrices in reconstruction scenarios; target speech is the same as source speech. a) is generated by the standard encoder-decoder model, and b) is generated by the encoder-decoder model replacing non-causal layers with causal layers. Attention matrix b) is automatically shifted by around 300 ms. This shifted attention directly increases the latency of real-time streaming VC.

not canceled and increases the latency because we force the attention matrix to be diagonal.

Motivated by the real-time streaming VC, the recent advanced VC, and our preliminary experiments, we apply knowledge distillation approaches, known as teacher-student frameworks, to a recurrent-based S2S-VC model as the first step. A major challenge is minimizing degradation due to introducing the causal structure instead of the non-causal structure, namely degradation caused by masking future input information. Therefore, we investigate several knowledge distillation approaches mentioned in Sec. 2.3. One is to copy some model parameters from the teacher model to the student model and fix them during training. The second is an attention distillation using Kullback-Leibler divergence (KLD). The last is to share some model parameters between teacher and student models and update them during training, known as joint training. Experimental evaluations show that except for male-to-female speaker conversion, the first approach is able to maintain the teacher model's performance in terms of subjective evaluations despite the streamable model structure.

2. STRUCTURE AND OBJECTIVE FUNCTION

The system overview is shown in Fig. 3. We employ a S2S model architecture where the source speech parameters are used as input and the target speech parameters are generated as output. All we need for training is a parallel corpus of input and output paired speech. We introduce pre-trained speaker encoder [15] to control output speaker styles and pre-trained neural vocoder [16] to generate speech waveform from speech parameters.

As the speech parameters, we extract 80-dimensional log-Mel spectrogram features $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_i]$ and $\boldsymbol{Y} = [\boldsymbol{y}_1, \cdots, \boldsymbol{y}_j]$ over a range of 80-7600 Hz from the given source and target speech signals sampled at 16 kHz. The requirements for Short-Time Fourier Transform are the same

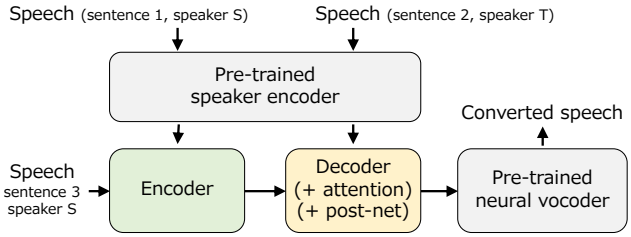

Fig. 3. System overview of teacher and student models.

as reported in [12]; a Hanning window, 64 ms frame length, eight ms frameshift, and 1024-point Fast Fourier Transform. Since shorter sequences make it easier to train each model, we shortened the length of the sequence by creating subframes containing two time-frames.

2.1. Speaker Encoder [15]

The network consists three LSTM layers with 768 unis and a fully connected layer with a 256-dimensional linear projection to get a 256-dimensional speaker vector $\boldsymbol{c}$ from an input log-Mel spectrogram. To train the speaker encoder, we used *Softmax* criterion reported in [15] and a speech database containing over 100k files uttered by thousands of speaker. Other training condition, such as mini-batch size and parameter initialization, is the same as in [15]. The resulting Top-1 accuracies are 99.98 % and 99.65 % for training data and evaluation data. During S2S model training, to avoid leaking context information to the S2S modules, the speaker vector $\boldsymbol{c}$ is extracted from different sentences from the source and target speech of the S2S model.

2.2. Teacher Model

Inspired by [17], an input log-Mel spectrogram $\boldsymbol{X}$ and a source speaker vector $\boldsymbol{c}_x$ is passed into an encoder network f_{Enc} is composed of three 1D non-causal convolutional layers each containing 512 filters with the kernel size of 5, followed by a batch normalization [18] layer and a rectified linear unit (ReLU) activation. The output of the final convolutional layer is passed into a single bi-directional long short term memory (LSTM) layer containing 512 units to generate encoded features $\boldsymbol{Z}_x$, as follows:

$$\boldsymbol{Z}_x = f_{\mathrm{Enc}}(\boldsymbol{X}, \boldsymbol{c}_x). \tag{1}$$

A decoder network f_{Dec} is the same as reported in [17], except for changing the LSTM unit size from 1024 to 512 [1] and taking the target speaker vector $\boldsymbol{c}_y$ as input. The decoder involving the attention mechanism f_{Att} and post-net f_{Post} predicts an attention matrix $\boldsymbol{A}$, the output log-Mel spectrogram $\hat{\boldsymbol{Y}} = [\hat{\boldsymbol{y}}_1, \cdots, \hat{\boldsymbol{y}}_j]$, and its improved $\hat{\boldsymbol{Y}}_{post}$ from $\boldsymbol{Z}_x$,

[1] We confirmed no degradation in the preliminary objective experiments.

Table 1. A list of investigated approaches. Post-net f_{Post} is omitted for brevity.

System	Encoder	Decoder	Fix	Attention mechanism	Skip	Other explanation
Baseline (related with Fig. 2 a)	f_{Enc}	f_{Dec}		f_{Att}		$\mathcal{L}_{\mathrm{Teacher}}$
Preliminary experiment (related with Fig. 2 b)	f_{DistEnc}	f_{DistDec}		f_{DistAtt}		$\mathcal{L}_{\mathrm{Student}}$
Approach-1	f_{DistEnc}	$f^{\star}_{\mathrm{Dec}}$	✓		✓	$\mathcal{L}_{\mathrm{Student}}$ + copied attention matrix
Approach-2	f_{DistEnc}	$f^{\star}_{\mathrm{Dec}}$	✓		✓	$\mathcal{L}_{\mathrm{Student}}$ + self-supervised learning
Approach-3	f_{DistEnc}	$f^{\star}_{\mathrm{DistDec}}$			✓	$\mathcal{L}_{\mathrm{Student}}$ + copied attention matrix
Approach-4	f_{DistEnc}	f_{DistDec}		f_{DistAtt}		$\mathcal{L}_{\mathrm{Student}}$ + $\mathcal{L}_{\mathrm{AttKLD}}$
Approach-5 (excluded)	f_{DistEnc}	f_{DistDec}		f_{DistAtt}		$\mathcal{L}_{\mathrm{Student}}$ + $\mathcal{L}_{\mathrm{Jnt}}$

$\tilde{\boldsymbol{Y}} = [\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{j-1}]$, and $\boldsymbol{c}_x$, where $\boldsymbol{y}_0$ indicates a zero vector, known as *start token*, as follows:

$$\hat{\boldsymbol{Y}}_{post}, \hat{\boldsymbol{Y}}, \boldsymbol{A} = f_{\mathrm{Dec}}(\boldsymbol{Z}_x, \tilde{\boldsymbol{Y}}, \boldsymbol{c}_y). \tag{2}$$

Finally, we minimize the objective function $\mathcal{L}_{\mathrm{Teacher}}$ to train the encoder f_{Enc} and the decoder f_{Dec}:

$$\mathcal{L}_{\mathrm{Teacher}} = ||\hat{\boldsymbol{Y}} - \boldsymbol{Y}||_2, \tag{3}$$

where $||\hat{\boldsymbol{Y}}_{post} - \boldsymbol{Y}||_2$ and *stop token loss* [17] are also used but omitted for brevity of the latter section.

2.3. Student Model

The structure of an encoder network f_{DistEnc} is the same as that of the teacher encoder f_{Enc}, except for streamable structure replacing the non-causal convolutions and bi-directional LSTM with the causal convolutions and uni-directional LSTM. The structure of a decoder network f_{DistDec} is the same as that of the teacher decoder f_{Dec}. A list of investigated approaches is shown in Table 1.

2.3.1. Approach-1 & -2: use a teacher decoder as a student decoder

In Approach-1 and -2, only the student encoder f_{DistEnc} is trained, because the teacher decoder is already streamable. In these approaches, we initialize the model parameters of the student decoder by those of the teacher decoder and fix them during training. To do so, we have two training schemes. Approach-1 uses the source and target parallel speech dataset for the student encoder training. To handle different lengths of speech sequence, we skip the attention mechanism f_{Att} but bring the attention matrix $\boldsymbol{A}$ generated by the teacher model as follows:

$$\boldsymbol{Z}'_x = f_{\mathrm{DistEnc}}(\boldsymbol{X}, \boldsymbol{c}_x), \tag{4}$$

$$\boldsymbol{Y}'_{post}, \boldsymbol{Y}' = f^{\star}_{\mathrm{Dec}}(\boldsymbol{Z}'_x \boldsymbol{A}, \tilde{\boldsymbol{Y}}, \boldsymbol{c}_y), \tag{5}$$

$$\mathcal{L}_{\mathrm{Student}} = ||\boldsymbol{Y}' - \boldsymbol{Y}||_2, \tag{6}$$

where $\star$ indicates skipping the attention mechanism.

On the other hand, Approach-2 uses the target speech $\boldsymbol{Y}$ as the source speech, so it is self-supervised learning, like an auto-encoder, as follows:

$$\boldsymbol{Z}'_y = f_{\mathrm{DistEnc}}(\boldsymbol{Y}, \boldsymbol{c}_y), \tag{7}$$

$$\boldsymbol{Y}'_{post}, \boldsymbol{Y}' = f^{\star}_{\mathrm{Dec}}(\boldsymbol{Z}'_y, \tilde{\boldsymbol{Y}}, \boldsymbol{c}_y). \tag{8}$$

It is assumed that if the teacher model is successfully trained, the input of the decoder will be disentangled features; the context information $\boldsymbol{Z}'_y$ extracted by the encoder and the speaker style $\boldsymbol{c}_y$ extracted by the speaker encoder. Therefore, even if it is self-supervised learning, we expect the student's encoder to be trained as a contextual information extractor that removes speaking style rather than simple compression.

2.3.2. Approach-3 & -4: train a student decoder

We also investigate schemes in which the student encoder f_{DistEnc} and decoder f_{DistDec} are trained. As mentioned in Sec. 1, we confirmed in the preliminary experiment that an approach of replacing the non-causal encoder with a causal encoder and training all parameters resulted in the shifted attention matrices. Therefore, Approach-3 and -4 take the attention matrix generated by the teacher model as a guide to train the student model. In Approach-3, the given attention matrix is used for changing the input sequence length as follows:

$$\boldsymbol{Y}'_{post}, \boldsymbol{Y}' = f^{\star}_{\mathrm{DistDec}}(\boldsymbol{Z}'_x \boldsymbol{A}, \tilde{\boldsymbol{Y}}, \boldsymbol{c}_y). \tag{9}$$

On the other hand, Approach-4 aims to train the attention mechanism f_{DistAtt}, which is contained in the student decoder f_{DistDec}. Each column of the attention matrix gives a probability distribution that describes the relationship between the output of the decoder at each step and the source feature sequence $\boldsymbol{Z}_x$. To guide the attention mechanism training, we introduce KLD between the attention matrix $\boldsymbol{A}$ generated by the teacher and that $\boldsymbol{A}'$ generated by the student model. This approach is known as *attention distillation*. The

KLD loss $\mathcal{L}_{\mathrm{AttKLD}}$ is the following;

$$\boldsymbol{Y}'_{post}, \boldsymbol{Y}', \boldsymbol{A}' = f_{\mathrm{DistDec}}(\boldsymbol{Z}'_x, \tilde{\boldsymbol{Y}}, \boldsymbol{c}_y), \tag{10}$$

$$\mathcal{L}_{\mathrm{AttKLD}} = \sum\nolimits_j \mathcal{D}_{\mathrm{KL}}(\boldsymbol{a}_j || \boldsymbol{a}'_j), \tag{11}$$

where $\boldsymbol{a}_j$ and $\boldsymbol{a}'_j$ denote j-th column of the attention matrices $\boldsymbol{A}$ and $\boldsymbol{A}'$.

2.3.3. Approach-5: jointly train non-causal and causal encoders

In Approach-5, we do not use any guides or model parameters from the teacher model but train the non-causal encoder f_{JntEnc} jointly with the causal encoder. Therefore, in addition to Eq. (4) and (10), we introduce the following formulation:

$$\ddot{\boldsymbol{Z}}_x = f_{\mathrm{JntEnc}}(\boldsymbol{X}, \boldsymbol{c}_x), \tag{12}$$

$$\ddot{\boldsymbol{Y}}_{post}, \ddot{\boldsymbol{Y}}, \ddot{\boldsymbol{A}} = f_{\mathrm{DistDec}}(\ddot{\boldsymbol{Z}}_x, \tilde{\boldsymbol{Y}}, \boldsymbol{c}_y), \tag{13}$$

$$\mathcal{L}_{\mathrm{Jnt}} = ||\ddot{\boldsymbol{Y}} - \boldsymbol{Y}||_2, \tag{14}$$

where the structure of f_{JntEnc} is the same as that of the teacher encoder f_{Enc}. From another point of view, we share and update the decoder between the teacher and student models.

Unfortunately, we confirmed the phenomenon of attention shifting, similar to Fig. 2b. One possible reason is that the non-causal encoders were unintentionally trained to match the characteristics of the causal encoders, as opposed to knowledge distillation from the non-causal encoders to the causal encoder, which is our aim. In order to achieve low latency streaming VC, Approach-5 was excluded from the experimental evaluations.

2.4. Neural Vocoder

In the proposed model, any neural vocoders by conditioning on Mel spectrograms as input can be used. In the experiment, HiFi-GAN [16] is used as the neural vocoder. We confirmed that *V2 setting* described in [16] is able to generate waveforms faster than real-time for streaming applications under the eight ms frameshift condition.

3. EXPERIMENTS

3.1. Experimental Conditions

To demonstrate the performance of each method, we conducted experimental evaluations using a phonetically balanced Japanese speech parallel dataset [19] consisting of utterances by two professional male speakers (*mht* and *msh*) and two professional female speakers (*fym* and *ftk*). The speeches were recorded in a quiet room with minimal reverberation. To train each VC model, we used 450 sentences (speech section of around 0.5 hours) of each speaker. Statistics of speech section length are shown in Table 2. To evaluate the performance, we used 53 sentences from each speaker. For the evaluations, we conducted intra-gender pairs, *mht-msh* and *fym-ftk*, and cross-gender pairs, *mht-ftk* and *fym-msh*.

Table 2. Speech section length [sec] over the training dataset.

Speaker		Avg.	Min.	Max.
Source	*mht*	4.18	1.48	8.53
	fym	4.11	1.47	8.31
Target	*msh*	3.93	1.42	7.66
	ftk	4.48	1.45	8.74

All of the encoder-decoder are trained 100k iterations. The learning rate and the exponential decay rate for the first moment for Adam [20] were set at 0.0002 and 0.9 after 4k step of warmup. The mini-batch size was 16. All of encoder-decoder models were trained on *many-to-many* condition, which is four-speaker input and four-speaker output. During test inference time, we forced the attention matrix to be diagonal because real-time streaming VC does not allow to change the speaking rate.

3.2. Objective Evaluation

As the objective evaluation metrics, we used Mel-cepstral distortion (MCD) [dB], root mean square error of F_0 (F_0 RMSE) [Hz], and character error rate (CER) [%]. We used dynamic time warping to get the alignment between the converted sample and the reference sample. To calculate the MCD, we calculated 1-24 order Mel-cepstrum. For VC model evaluation, we extracted Mel-cepstrum and F_0 from the converted speech waveform synthesized by the neural vocoder because the VC model generates Mel-spectrogram. The CER was calculated by the Transformer-based ASR model trained on the corpus of spontaneous Japanese (CSJ) [21], which was provided by ESPnet [22]. MCD and F_0 RMSE reflect speaker, prosody, and phonetic content similarities, and CER represents the intelligibility and correlates to naturalness [23]. Note that MCD and F_0 RMSE are less sensitive to speech discontinuity than CER because MCD and F_0 RMSE are calculated frame-by-frame while the CER is calculated by ASR model that takes the sequential information into account. The performances of neural vocoder, MCD, F_0 RMSE, and CER of the re-synthesized speech given the ground-truth Mel-spectrogram, were 2.99, 19.50, and 10.6, respectively.

The objective evaluation results are shown in Table 3. First, we focus on the results of Baseline. The result shows that the Baseline's performance depends on who the target speaker is rather than the gender pairs. The conversion into the speaker *ftk* seems to be more difficult than the conversion into the speaker *msh*.

Compared to Baseline, the performance of Approach-1,

Table 3. Objective evaluation results on intra-gender (upper) and cross-gender (lower) pairs. The lower the value the better the performance.

System	*mht-msh*			*fym-ftk*		
	MCD	F_0 RMSE	CER	MCD	F_0 RMSE	CER
Baseline	5.11 ± 0.08	18.66 ± 1.18	15.8	6.01 ± 0.08	27.49 ± 1.88	15.1
Approach-1	5.61 ± 0.08	21.31 ± 1.40	22.1	6.39 ± 0.09	31.90 ± 2.53	19.8
Approach-2	**5.51** ± 0.08	**19.09** ± 1.31	**17.0**	**6.19** ± 0.08	**26.32** ± 1.82	**14.2**
Approach-3	**5.44** ± 0.08	21.10 ± 1.43	18.9	6.43 ± 0.08	34.57 ± 2.54	16.8
Approach-4	5.73 ± 0.09	25.47 ± 1.48	29.6	6.55 ± 0.09	39.90 ± 2.89	26.1
System	*mht-ftk*			*fym-msh*		
	MCD	F_0 RMSE	CER	MCD	F_0 RMSE	CER
Baseline	5.95 ± 0.09	28.49 ± 1.58	15.0	5.13 ± 0.08	18.30 ± 1.27	16.7
Approach-1	**6.38** ± 0.09	34.87 ± 2.66	21.2	5.66 ± 0.08	**21.23** ± 1.55	19.6
Approach-2	**6.34** ± 0.06	**28.43** ± 1.56	**15.0**	5.82 ± 0.08	22.08 ± 1.74	**15.7**
Approach-3	**6.37** ± 0.08	36.13 ± 2.54	16.4	**5.55** ± 0.09	**20.08** ± 1.41	17.5
Approach-4	6.55 ± 0.08	41.01 ± 2.94	32.9	5.79 ± 0.08	24.14 ± 1.44	26.0

-3, and -4 was degraded. However, the F_0 RMSE and CER of Approach-2 were comparable to the Baseline. MCD of Approach-2 was also better than Approach-1, -3, and -4, except for *mht-ftk* pair. This result indicates that to distill the S2S-VC model, a self-supervised learning with a pre-trained decoder works better than the training approach using parallel data and attention matrices. One possible reason is that since the Baseline architecture includes the bi-directional RNN containing the backward RNN, strange attention shifting, which is acceptable for bi-directional RNN but not for uni-directional RNN, may occur. Moreover, although the parallel data is recorded carefully, it is very challenging to be exactly parallel in several terms such as pause position and length, pronunciation error, and different intonation. Even if the attention mechanism makes it possible to reduce the degradation caused by the recording mistake, self-supervised learning, which is under the perfect parallel data condition, seems to be better. However, it is essential to note that the self-supervised learning approach has a gap during training and inference processes because training data never see the speaker conversion setting, in which the input speaker is different from the target speaker. In the preliminary experiments, we also trained a simple auto-encoder. The result is the reconstruction of the input speech, and speaker conversion is a failure [2]. In the next section, we will confirm whether fixing the decoder achieves the speaker conversion.

[2]In AutoVC [24], the authors have reported that the training of simple auto-encoder to convert the speaker characteristics is very sensitive to the data setting and model architecture.

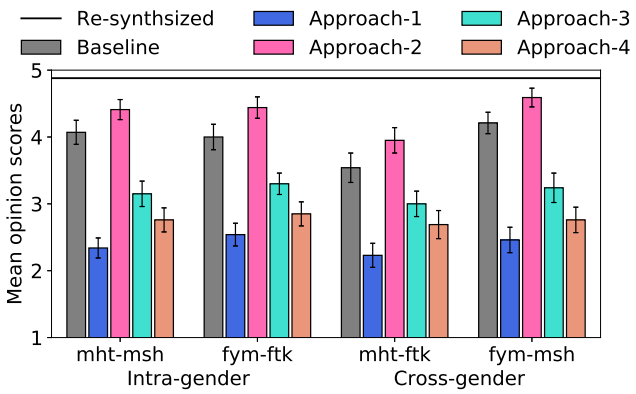

Fig. 4. Subjective evaluation results on sound quality. The higher the value the better the sound quality. Re-synthesized indicates the speech synthesized from the ground-truth Mel-spectrogram.

3.3. Subjective Evaluation

Next, we conducted subjective evaluation tests on sound quality and speaker similarity to check the perceptual quality. For sound quality, each subject listened to each sample and rated the sound quality on a 5-point scale: 5 for excellent, 4 for good, 3 for fair, 2 for poor, and 1 for bad. For speaker similarity, each subject listened to pairs of the target sample and the converted sample to judge whether the presented samples were produced by the same speaker with confidence on a 4-point scale: 4 for same (sure), 3 for same (not sure), 2 for different (not sure), and 1 for different (sure). 10 native Japanese speakers participated in each subjective evaluation. Each system was evaluated over 250 times.

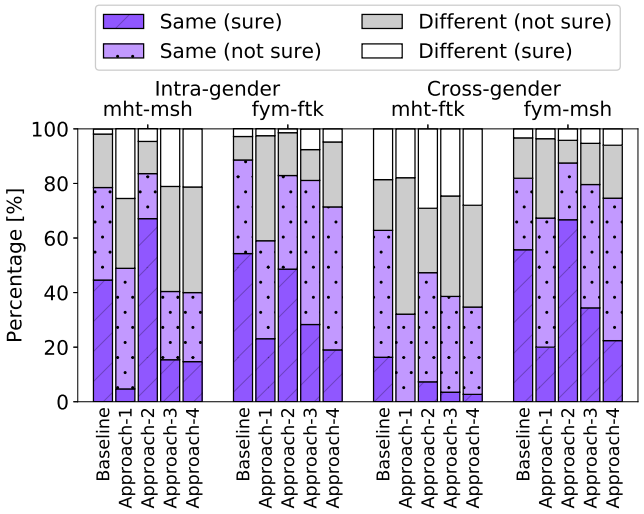

Fig. 5. Subjective evaluation results on speaker similarity. The higher the rate of *Same* the better the performance.

The subjective evaluation results on sound quality and speaker similarity are shown in Fig. 4 and 5. First, we focus on the results of Baseline. Although the MCD, F_0 RMSE, and CER of *mht-ftk* pair were similar to those of *fym-ftk* pair, the subjective evaluation results of *mht-ftk* pair were the worst. Given that the auto-regressive model also models something like phoneme duration, a significant change in the speaking rate may affect conversion performance. However, the speaking rate of the speaker *mht* was similar to the speaking rate of the speaker *fym*, as shown in Table 2. This result showed that the Baseline's performance depended on whether it was male-to-female speaker conversion or not, rather than the speaking rate changes.

Compared to Baseline, the performance of Approach-1, -3, and -4 was degraded, similar to the objective evaluation results. However, the sound quality of Approach-2 overcame that of Baseline, thanks to self-supervised learning, which was under the perfect parallel data condition. On the other hand, the speaker similarity of the *mht-ftk* pair was degraded. Note that there was no significant difference on the speaker similarity of the *fym-ftk* pair. This result indicated that the self-supervised learning approach might be more likely to be affected than the S2S learning approaches required the paired speech data. To tackle this limitation, we will work on it in future work.

4. CONCLUSIONS

Toward real-time streaming VC applications, this paper describes a method for distilling a recurrent-based S2S-VC model. The experimental results revealed that using a decoder, which is trained in advance by using parallel speech data, is a good constraint for the encoder training. Moreover, self-supervised learning with a fixed decoder can maintain the teacher model's performance in terms of subjective evaluations despite the streamable student model structure, except for male-to-female speaker conversion. In future work, we will work on reducing the impact of male-to-female speaker conversion in the self-supervised learning.

5. ACKNOWLEDGMENTS

This work was supported by JST CREST Grant Number JPMJCR19A3, Japan.

6. REFERENCES

[1] Aaron Van Den Oord, Oriol Vinyals, et al., "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[2] J. Zhang, Z. Ling, L. Liu, Y. Jiang, and L. Dai, "Sequence-to-sequence acoustic modeling for voice conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 3, pp. 631–644, 2019.

[3] Kazuhiro Kobayashi, Tomoki Toda, Graham Neubig, Sakriani Sakti, and Satoshi Nakamura, "Statistical singing voice conversion with direct waveform modification based on the spectrum differential," in *INTERSPEECH*, 2014, pp. 2514–2518.

[4] Rafael Valle, Jason Li, Ryan Prenger, and Bryan Catanzaro, "Mellotron: Multispeaker expressive voice synthesis by conditioning on rhythm, pitch and global style tokens," in *ICASSP*, 2020, pp. 6189–6193.

[5] Kou Tanaka, Tomoki Toda, Graham Neubig, Sakriani Sakti, and Satoshi Nakamura, "A hybrid approach to electrolaryngeal speech enhancement based on noise reduction and statistical excitation generation," *IEICE Transactions on Information and Systems*, vol. 97, no. 6, pp. 1429–1437, 2014.

[6] Fadi Biadsy, Ron J Weiss, Pedro J Moreno, Dimitri Kanevsky, and Ye Jia, "Parrotron: An end-to-end speech-to-speech conversion model and its applications to hearing-impaired speech and speech separation," in *INTERSPEECH*, 2019, pp. 4115–4119.

[7] Daniel Felps, Heather Bortfeld, and Ricardo Gutierrez-Osuna, "Foreign accent conversion in computer assisted pronunciation training," *Speech Communication*, vol. 51, no. 10, pp. 920–932, 2009.

[8] Kou Tanaka, Hirokazu Kameoka, Takuhiro Kaneko, and Nobukatsu Hojo, "AttS2S-VC: Sequence-to-sequence voice conversion with attention and context preservation mechanisms," in *ICASSP*, 2019, pp. 6805–6809.

[9] Hirokazu Kameoka, Kou Tanaka, Damian Kwaśny, Takuhiro Kaneko, and Nobukatsu Hojo, "ConvS2S-VC: Fully convolutional sequence-to-sequence voice conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1849–1863, 2020.

[10] Hirokazu Kameoka, Wen-Chin Huang, Kou Tanaka, Takuhiro Kaneko, Nobukatsu Hojo, and Tomoki Toda, "Many-to-many voice transformer network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 656–670, 2020.

[11] Aubrey J Yates, "Delayed auditory feedback," *Psychological bulletin*, vol. 60, no. 3, pp. 213, 1963.

[12] Hirokazu Kameoka, Kou Tanaka, and Takuhiro Kaneko, "FastS2S-VC: Streaming non-autoregressive sequence-to-sequence voice conversion," *arXiv preprint arXiv:2104.06900*, 2021.

[13] Xie Chen, Yu Wu, Zhenghao Wang, Shujie Liu, and Jinyu Li, "Developing real-time streaming transformer transducer for speech recognition on large-scale dataset," in *ICASSP*, 2021, pp. 5904–5908.

[14] Gakuto Kurata and George Saon, "Knowledge distillation from offline to streaming RNN transducer for end-to-end speech recognition," in *INTERSPEECH*, 2020, pp. 2117–2121.

[15] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno, "Generalized end-to-end loss for speaker verification," in *ICASSP*, 2018, pp. 4879–4883.

[16] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17022–17033, 2020.

[17] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al., "Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions," in *ICASSP*, 2018, pp. 4779–4783.

[18] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015, pp. 448–456.

[19] Akira Kurematsu, Kazuya Takeda, Yoshinori Sagisaka, Shigeru Katagiri, Hisao Kuwabara, and Kiyohiro Shikano, "ATR japanese speech database as a tool of speech recognition and synthesis," *Speech Communication*, vol. 9, no. 4, pp. 357–363, 1990.

[20] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[21] Kikuo Maekawa, "Corpus of spontaneous japanese: Its design and evaluation," in *Workshop on Spontaneous Speech Processing and Recognition*, 2003.

[22] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai, "ESPnet: End-to-end speech processing toolkit," in *INTERSPEECH*, 2018, pp. 2207–2211.

[23] Rohan Kumar Das, Tomi Kinnunen, Wen-Chin Huang, Zhenhua Ling, Junichi Yamagishi, Yi Zhao, Xiaohai Tian, and Tomoki Toda, "Predictions of subjective ratings and spoofing assessments of voice conversion challenge 2020 submissions," in *Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge*, 2020, pp. 99–120.

[24] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson, "AutoVC: Zero-shot voice style transfer with only autoencoder loss," in *ICML*, 2019, pp. 5210–5219.