

SPEECH-TO-SPEECH TRANSLATION BETWEEN UNTRANSCRIBED UNKNOWN LANGUAGES

*Andros Tjandra*¹, *Sakriani Sakti*^{1,2}, *Satoshi Nakamura*^{1,2}

¹Nara Institute of Science and Technology, Japan

²RIKEN, Center for Advanced Intelligence Project AIP, Japan

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

ABSTRACT

In this paper, we explore a method for training speech-to-speech translation tasks without any transcription or linguistic supervision. Our proposed method consists of two steps: First, we train and generate discrete representation with unsupervised term discovery with a discrete quantized autoencoder. Second, we train a sequence-to-sequence model that directly maps the source language speech to the target languages discrete representation. Our proposed method can directly generate target speech without any auxiliary or pre-training steps with a source or target transcription. To the best of our knowledge, this is the first work that performed pure speech-to-speech translation between untranscribed unknown languages.

Index Terms— speech translation, sequence-to-sequence, zero-resource modeling, unit discovery, autoencoder

1. INTRODUCTION

Information exchanges among different countries continue to increase. International travelers for tourism, emigration, or foreign study are becoming increasingly diverse, heightening the need for devising a means to offer effective interaction among people who speak different languages. Since automatic spoken-to-speech translation (S2ST) provides an opportunity for people to communicate in their own languages, it significantly overcomes language barriers and closes cross-cultural gaps.

Many researchers have been developing a S2ST system over the past several decades. A traditional approach in S2ST systems requires effort to construct several components, including automatic speech recognition (ASR), machine translation (MT), and text-to-speech (TTS) synthesis, all of which are trained and tuned independently. Given speech input, ASR processes and transforms speech into text in the source language, MT transforms the source language text to corresponding text in the target language, and finally TTS generates speech from the text in the target language. Significant progress has been made and various commercial speech translation systems are already available for several language pairs. However, more than 6000 languages, spoken by 350 million people, have not been covered yet. Critically, over half of the world’s languages actually have no written form; they are only spoken.

Recently, end-to-end deep learning frameworks have shown impressive performances on many sequence-related tasks, such as ASR, MT, and TTS [1, 2, 3]. Their architecture commonly uses an attentional-based encoder-decoder mechanism, which allows the model to learn the alignments between the source and the target sequence, that can perform end-to-end mapping tasks of different

modalities. Many complicated hand-engineered models can also be simplified by letting neural networks find their way to map from input to output spaces. Thus, the approach provides the possibility of learning a direct mapping between the variable-length of the source and the target sequences that are often not known a priori. Several works extended the sequence-to-sequence models coverage by directly performing end-to-end speech translation using only a single neural network architecture instead of separately focusing on its components (ASR, MT, and TTS).

Although the first feasibility was shown by Duong et al. [4], they focused on the alignment between the speech in the source language and the text in the target language because their speech-to-word model did not yield any useful output. The first full-fledged end-to-end attentional-based speech-to-text translation system was successfully performed by Bérard et al. on a small French-English synthetic corpus [5]. But their performance was only compared with statistical MT systems. Weiss et al. [6] demonstrated that end-to-end speech-to-text models on Spanish-English language pairs outperformed neural cascade models. Kano et al. then proved that this approach is possible for distant language pairs such as Japanese-to-English translation [7]. Similar to the model by Weiss et al. [6], although it does not explicitly transcribe the speech into text in the source language, it also doesn't require supervision from the groundtruth of the source language transcription during training. However, most of these works remain limited to speech-to-text translation and require text transcription in the target language.

Recently, Jia et al. [8] proposed the deep learning model that is trained end-to-end, which learns to map speech spectrograms into target spectrograms in another language that corresponds to the translated content (in a different canonical voice). Unfortunately, since training without auxiliary losses leads to extremely poor performance, they provided a solution by integrating auxiliary decoder networks to predict phoneme sequences that correspond to the source and/or target speech. Despite much progress in direct speech translation research, no completely direct speech-to-speech translation has been achieved without any text transcription in source and target languages, as well as during training and decoding, has not been achieved yet. Therefore, it remains difficult to scale-up the existing approach to unknown languages without written forms or transcription data available.

On the other hand, there has been a project that held by speech community to push toward developing unsupervised, data-driven systems that are less reliant on linguistic expertise. Zero resource modeling is an approach where completely unsupervised techniques can learn the elements of a languages speech hierarchy solely from untranscribed audio data. This means that only spoken audio data are available in a specific language, but transcriptions, annota-

tions, and prior knowledge for it are all unavailable. The Zero Resource Speech Challenge series [9, 10, 11] was constructed to progress incrementally toward a system that learns an end-to-end spoken dialog (SD) system in an unknown language from scratch just using information available to language learning infants. The ZeroSpeech 2019 [11] challenge confronts the problem of constructing a speech synthesizer without any text or phonetic labels: TTS without T. It is a continuation of the subword unit discovery track of ZeroSpeech 2015 and 2017 [9, 10]. 19 systems were submitted, but few studies proposed end-to-end frameworks [12, 13, 14]. Among these proposed systems, the vector quantized variational autoencoder (VQ-VAE) approach provides a better performance of naturalness based on mean opinion score (MOS) on the generated speech and character error rate after human transcription of the speech synthesis. Further details of the results are available: www.zerospeech.com/2019/results.html.

In this paper, we take a step beyond the task of the current ZeroSpeech 2019 and propose a method for training speech to speech translation tasks without any transcription or linguistic supervision. Instead of only discovering subword units and synthesizing them within a certain language, our approach discovers subword units that are directly translated to another language. Our proposed method consists of two steps: (1) we train and generate discrete representation with unsupervised term discovery, which is also based on a discrete quantized autoencoder; (2) we train a sequence-to-sequence model to directly map the source language speech to the target language discrete representation. Our proposed method can directly generate target speech without any auxiliary or pre-training steps with source or target transcription. To the best of our knowledge, this is the first work that performed pure speech-to-speech translation between untranscribed unknown languages.

2. UNSUPERVISED UNIT DISCOVERY WITH VQ-VAE

A speech signal can be disentangled into independent factors of variation such as contexts and speaking styles. In a speech domain, we assume the context has a similar property with phonemes or subwords, which are represented with a limited set of discrete symbols. Therefore, to capture the context without any supervision, we use a generative model named a vector quantized variational autoencoder (VQ-VAE) [15] to extract the discrete symbols. There are several distinctions between a VQ-VAE with a normal autoencoder [16] and a normal variational autoencoder (VAE) [17]. The VQ-VAE encoder maps the input features to a limited number of discrete latent variables, and a standard VAE encoder maps the input features into continuous latent variables. Therefore, a VQ-VAE encoder has many-to-one mappings due to restricting the representation to the nearest codebook vector, and the standard VAE encoder has one-to-one mapping between the input and latent variables.

We illustrate the VQ-VAE model in Fig. 1 and define $\mathbf{E} = [e_1, \dots, e_K] \in \mathbb{R}^{K \times D_e}$ as a collection of codebook vectors and $\mathbf{V} = [v_1, \dots, v_L] \in \mathbb{R}^{L \times D_v}$. During the encoding step, input x is such speech features as MFCC or mel-spectrogram and input x 's speaker identity is denoted by $s \in \{1, \dots, L\}$. In Fig. 2, we show the details for the residual block inside the encoder and decoder modules. Encoder $q_\theta(y|x)$ generates discrete latent variable $y \in \{1, \dots, K\}$ (y can also be represented as a one-hot vector). To transform a continuous representation into a discrete random variable, the encoder first produces intermediate continuous representation $z \in \mathbb{R}^{D_e}$. Later, we find which codebook has a minimum distance between z and a

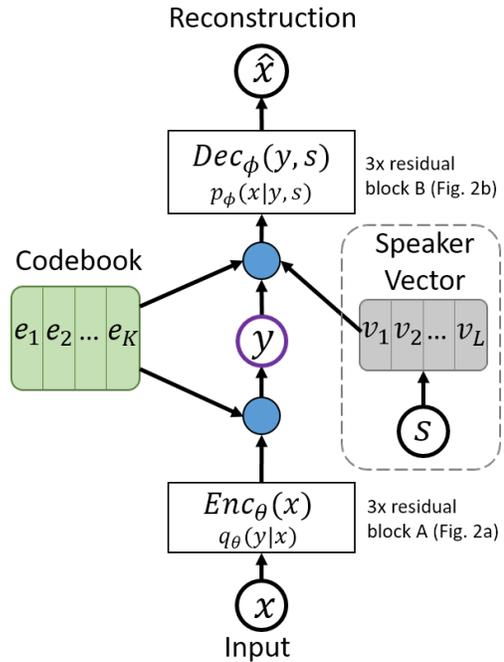


Fig. 1. VQ-VAE for unsupervised unit discovery consists of several parts: encoder $\text{Enc}_\theta^{VQ}(x) = q_\theta(y|x)$, decoder $\text{Dec}_\phi^{VQ}(y, s) = p_\phi(x|y, s)$, codebooks $\mathbf{E} = [e_1, \dots, e_K]$, and (optional) speaker embedding $\mathbf{V} = [v_1, \dots, v_L]$.

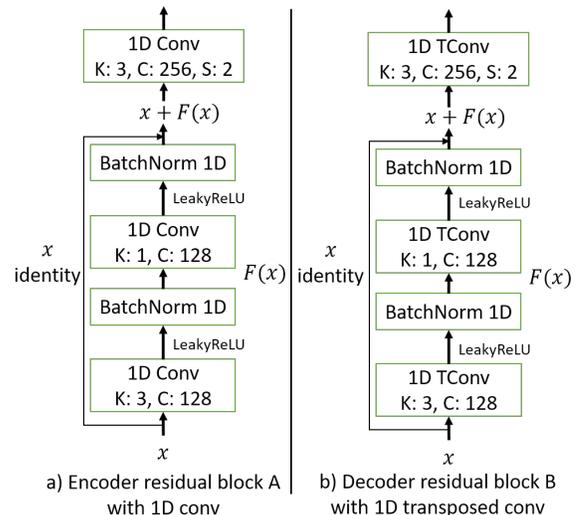


Fig. 2. Building block inside VQ-VAE encoder and decoder: a) Encoder residual block and 1D convolution with stride 2 to down-sample input sequence length; b) Decoder residual block and 1D transposed convolution with stride 2 to upsample codebook back to original input length.

vector in \mathbf{E} . Mathematically, we formulate the operation:

$$q_\theta(y = c|x) = \begin{cases} 1 & \text{if } c = \operatorname{argmin}_i \operatorname{Dist}(z, e_i) \\ 0 & \text{else} \end{cases} \quad (1)$$

$$e_c = \mathbb{E}_{q_\theta(y|c)}[\mathbf{E}] \quad (2)$$

$$= \sum_{i=1}^K q_\theta(y = i|x) e_i. \quad (3)$$

where $\operatorname{Dist}(\cdot, \cdot) : \mathbb{R}^{D_e} \times \mathbb{R}^{D_e} \rightarrow \mathbb{R}$ is a function to calculate the distance between two vectors. In this paper, we define $\operatorname{Dist}(a, b) = \|a - b\|_2$ as the L2-norm distance.

After we find closest codebook index $c \in \{1, \dots, K\}$, we substitute intermediate variable z with corresponding codebook vector e_c . To reconstruct the input data, decoder $p_\phi(x|y, s)$ reads codebook vector e_c and speaker embedding v_s and generates reconstruction \hat{x} .

The following is the learning objective for VQ-VAE:

$$\mathcal{L}_{VQ} = -\log p_\phi(x|y, s) + \gamma \|z - \operatorname{sg}(e_c)\|_2^2, \quad (4)$$

where function $\operatorname{sg}(\cdot)$ stops the gradient, defined as:

$$x = \operatorname{sg}(x) \quad (5)$$

$$\frac{\partial \operatorname{sg}(x)}{\partial x} = 0. \quad (6)$$

The first term is a negative log-likelihood to measure the reconstruction loss between original input x and reconstruction \hat{x} to optimize encoder parameters θ and decoder parameters ϕ . The second term minimizes the distance between intermediate representation z and nearest codebook e_c , but the gradient is only back-propagated into encoder parameters θ as commitment loss. Such commitment loss can be scaled with additional hyperparameters γ . To update the codebook vectors, we use an exponential moving average (EMA) [18]. With an EMA update rule for training codebook \mathbf{E} , the model has a more stable result during the training process and avoids the posterior collapse issue [19].

3. SEQUENCE-TO-SEQUENCE FROM SPEECH TO CODEBOOK

Our speech-to-speech translation model is built based on an attention sequence-to-sequence (seq2seq) framework [20, 21]. Assume paired source sequence $\mathbf{X} = [x_1, \dots, x_S]$ and target sequence $\mathbf{Y} = [y_1, \dots, y_T]$. A sequence-to-sequence model directly learns mapping $P_\psi(\mathbf{X}|\mathbf{Y})$, parameterized by ψ parameters. In this paper, we specify $\mathbf{X} \in \mathbb{R}^{S \times D_s}$ to represent such speech features as MFCC or mel-spectrogram and $\mathbf{Y} = [y_1, \dots, y_T] \in \{1, \dots, K\}$ to represent codebook \mathbf{E} indices. Fig. 3 illustrates a seq2seq model with an attention mechanism. Inside a seq2seq model, there are three different components:

1. The encoder module reads all the sequence speech features and represents them with $h^E = [h_1^E, \dots, h_S^E] \in \mathbb{R}^{S \times M}$ where $h^E = \operatorname{Enc}_\psi^{S2S}(\mathbf{X})$.
2. The attention module assists the decoder to find which part of the encoder contains related information for current decoding state [20]. Given decoder state $h_t^D \in \mathbb{R}^n$, the attention modules generate attention $a_t \in \mathbb{R}^S$ and context $c_t \in \mathbb{R}^N$:

$$a_t[s] = \frac{\exp(\operatorname{Score}(h_s^E, h_t^D))}{\sum_{s=1}^S \exp(\operatorname{Score}(h_s^E, h_t^D))} \quad (7)$$

$$c_t = \sum_{s=1}^S a_t[s] h_s^E, \quad (8)$$

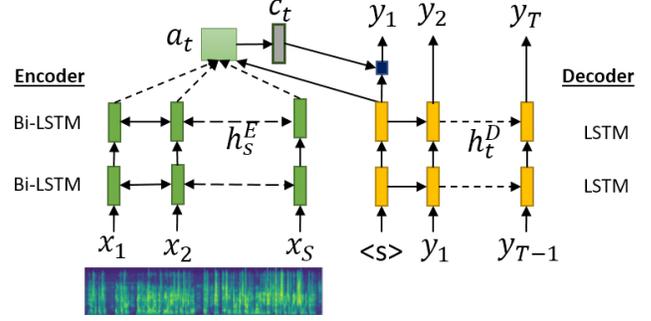


Fig. 3. Sequence-to-sequence model with attention mechanism. Here encoder input is speech features $\mathbf{X} = [x_1, \dots, x_S]$, and decoder predicts codebook index y_t for each time-step.

where function $\operatorname{Score}(\cdot, \cdot) : \mathbb{R}^M \times \mathbb{R}^N \rightarrow \mathbb{R}$ predicts the relevancy value between the encoder and decoder states. Many Score functions exist, including dot-product [22], MLP [20] or modified MLP with history [23].

3. The decoder module predicts class probability $p_t = \operatorname{Dec}_\psi^{S2S}(y_t|c_t, \mathbf{Y}_{<t}, h_t^D)$ over K different classes (depending on codebook \mathbf{E} size) given context c_t , previous information $\mathbf{Y}_{<t}$, and current decoder state h_t^D .

To train a seq2seq model, the most common objective is to minimize the negative log-likelihood over the correct class:

$$\mathcal{L}_{NLL} = -\frac{1}{T} \sum_{t=1}^T \sum_{k=1}^K \mathbb{1}(y_t = k) * \log p_t[y = k], \quad (9)$$

where $p_t[y = k]$ is the predicted probability on the k -th class and time-step t .

4. CODEBOOK INVERTER

A codebook inverter is an module that synthesizes the corresponding speech utterance from a sequence of the codebook index. Its input, which is a sequence of codebook embedding, is $[\mathbf{E}[y_1], \dots, \mathbf{E}[y_{T_Y}]]$, and the output target is a sequence of speech representation (e.g., linear magnitude spectrogram) $\mathbf{X}^R = [\mathbf{X}_1^R, \dots, \mathbf{X}_{T_x}^R]$.

We illustrate our codebook inverter architecture in Fig. 4. Our codebook inverter is composed of several residual 1D blocks, followed by stacked bidirectional LSTMs [24], and finally another several residual 1D blocks. Fig. 5 shows the details inside the block. Under certain circumstances, codebook sequence length T_Y might be shorter than T_X because VQ-VAE encoder $q_\theta(y|x)$ has convolution with a stride larger than 1. Therefore, to align the codebook sequence with the speech representation target sequence, we duplicate each codebook $\mathbf{E}[y_t]$ into r copies side-by-side where $r = T_X/T_Y$. To train a codebook inverter, we set the objective function:

$$\mathcal{L}_{INV} = \|\mathbf{X}^R - \hat{\mathbf{X}}^R\|_2 \quad (10)$$

to minimize the L2-norm between predicted spectrogram $\hat{\mathbf{X}}^R = \operatorname{Inv}_\rho([\mathbf{E}[y_1], \dots, \mathbf{E}[y_{T_Y}]])$ and groundtruth spectrogram \mathbf{X}^R . We defined Inv_ρ as the inverter parameterized by ρ . In the inference stage, we used Griffin-Lim [27] to reconstruct the phase from the spectrogram and applied an inverse short-term Fourier transform (STFT) to invert it into a speech waveform.

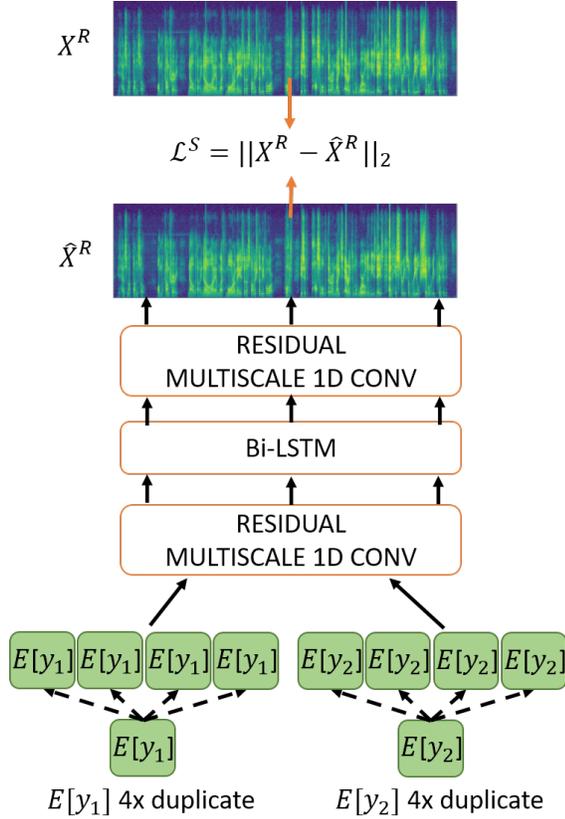


Fig. 4. Codebook inverter: given codebook sequence $[E[y_1], \dots, E[y_{T_Y}]]$, we predict corresponding linear magnitude spectrogram $\hat{\mathbf{X}}^R = [x_1^R, \dots, x_{T_X}^R]$. If the lengths between T_Y and T_X are different, we consecutively duplicate each codebook by r -times.

5. TRAINING AND INFERENCE

In this section, we explain our proposed method in detail and step-by-step. To train our proposed model, we setup three different modules: VQ-VAE (Section 2), a speech-to-codebook seq2seq (Section 3), and a codebook inverter (Section 4). Fig. 6 shows which modules are trained in each step. Initially, we defined $\{\mathbf{X}_{src}^M, \mathbf{X}_{tgt}^M\}$ as paired parallel speech, \mathbf{X}_{src}^M is the MFCC features from the source language, and \mathbf{X}_{tgt}^M is the MFCC features from the target language. \mathbf{Y}_{tgt} is the codebook sequences generated by VQ-VAE encoder $\text{Enc}_\theta(x)$ given \mathbf{X}_{tgt}^M as the input. $\hat{\mathbf{X}}_{tgt}^R$ is the predicted linear spectrogram of the target language. \mathcal{L}_{VQ} , \mathcal{L}_{INV} , and \mathcal{L}_{NLL} are calculated by the formula in Eqs. 4, 10, and 9.

1. First, we trained the VQ-VAE model on target language MFCC \mathbf{X}_{tgt}^M . We also trained the codebook inverter to predict corresponding linear spectrogram $\hat{\mathbf{X}}_{tgt}^R$.
2. Second, we trained the seq2seq model from the source language speech to the target language codebook. Given a paired parallel MFCC from source and target languages $\{\mathbf{X}_{src}^M, \mathbf{X}_{tgt}^M\}$, we extracted codebook sequence $\mathbf{Y}_{tgt} = \text{Enc}_\theta^{VQ}(\mathbf{X}_{tgt}^M)$ from the VQ-VAE encoder. Later, we trained the seq2seq translation model to predict $\hat{\mathbf{Y}}_{tgt} = \text{Seq2Seq}(\mathbf{X}_{src}^M)$ and minimize loss \mathcal{L}_{NLL} between \mathbf{X}_{src}^M and \mathbf{X}_{src}^M .

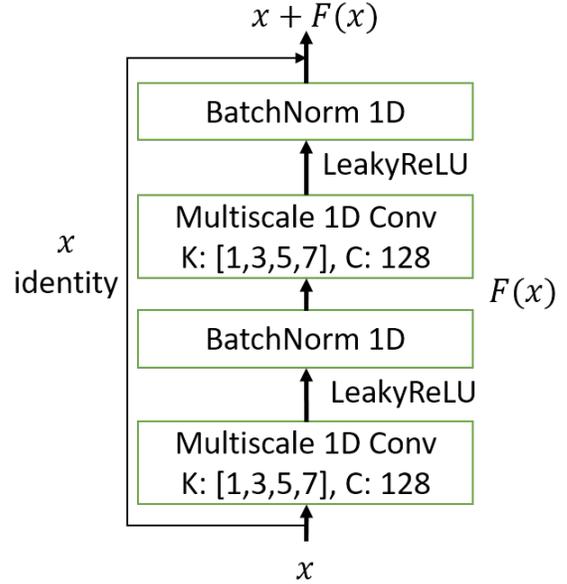


Fig. 5. Residual 1D block combines multiscale 1D convolution with different kernel size and “SAME” padding, LeakyReLU [25] activation function, and batch normalization [26].

3. In the inference step, given source language speech \mathbf{X}_{src}^M , we decoded a target language codebook index sequence $\hat{\mathbf{Y}}_{tgt} = \text{Seq2Seq}_v(\mathbf{X}_{src}^M)$ and synthesized it into target language speech $\hat{\mathbf{X}}_{tgt}^R = \text{Inverter}(\hat{\mathbf{Y}}_{tgt})$.

6. EXPERIMENTAL SETUP

6.1. Dataset

In this paper, we ran our experiment based on the Basic Travel Expression (BTEC) corpus [28, 29] that has several language pairs. We chose two tasks: French-to-English and Japanese-to-English. For both language pairs, we used the BTEC1 set that consisted of 162,318 training sentences and 510 test sentences. Since the speech utterances for the sentences are unavailable, we generated sentences with Google text-to-speech API for all languages pairs. Even though the lack of natural speech dataset in this paper, VQ-VAE and codebook inverter can be applied and has shown a great performance on multispeaker natural speech [14, 13]. Some papers [30, 31, 32] also show the performance improvement from the synthetic dataset can be carried over to the real dataset.

6.2. Speech Feature Extraction

For the source language and target language speech utterances, we represented the speech utterances with mel-frequency cepstral coefficients (MFCCs) with 13 dimensions $+\Delta + \Delta^2$ (total 39 dimensions). For the target language speech utterances, we also generated a linear magnitude spectrogram with 1025 dimensions for the codebook inverter (Section 4) training target. For each frame, we extracted the MFCCs and the linear magnitude spectrogram with a 25-millisecond-sized window and 10-millisecond time-steps. We extracted both the MFCC and the linear magnitude spectrogram with Librosa [33] library.

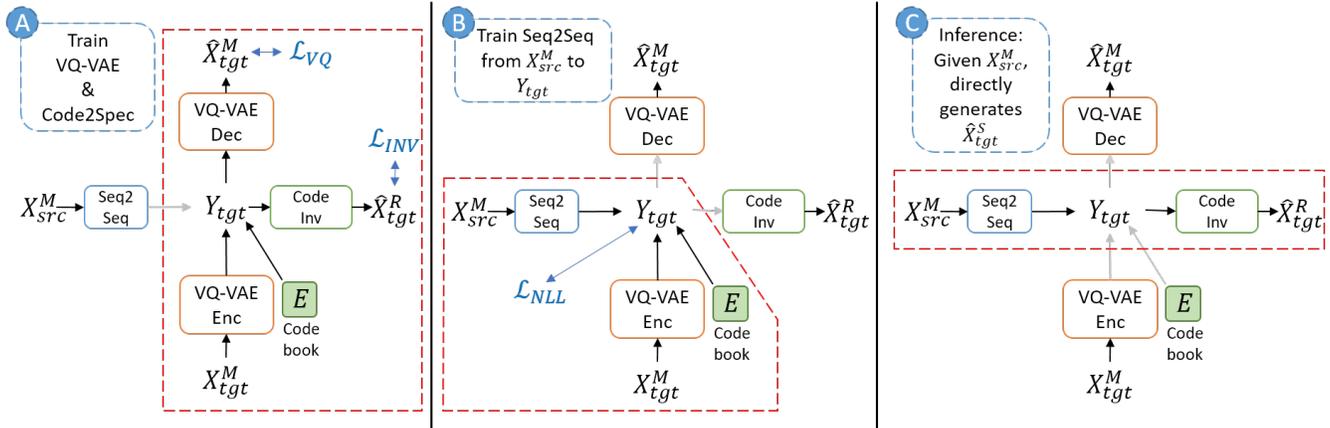


Fig. 6. a) Train VQ-VAE to represent continuous MFCC vectors with codebook sequence and train codebook inverter to generate a linear magnitude spectrogram based on generated codebook sequence; b) Train a seq2seq model from source language MFCC to target language codebook. c) In inference stage, seq2seq model takes source language MFCC and predicts codebook sequences, and then codebook inverter generates target language speech representation.

6.3. Evaluation

For an objective evaluation of the target speech utterances, currently there is no standard method can be used to measure translation quality directly on the speech utterances. Therefore, we utilized a pre-trained ASR on the English BTEC dataset and the generated transcription for our evaluation. For the ASR architecture, the encoder module has three stacked Bi-LSTMs with 512 hidden units, and the decoder has one LSTM with 512 hidden units. For the attention module, we utilized MLP attention with multiscale location history [23]. For the output unit, we used a word-level token from the English transcription. Because there is a performance gap between the ASR and the ground truth cause by imperfect transcription, we assume the metric (calculated based on the ASR transcription) is the lowerbound for the related translation model. We utilized two metrics to evaluate the translation performance from the transcribed text: BLEU scores [34] and METEOR [35] with a Multeval toolkit [36]. Our pre-trained ASR model resulted in a 2.84% WER, a 94.9 BLEU, and a 69.1 METEOR on English speech utterances from the BTEC test set, and we set those scores as the groundtruth topline scores.

7. RESULTS AND DISCUSSION

In this section, we present our experimental result and followed by the discussion.

7.1. Baseline

For the baseline translation task, we modified the Tacotron [3] model by changing the source input from a one-hot character embedding into a continuous vector. Basically, we changed the embedding layer in the encoder layer with a linear projection layer. Therefore, this model directly translated the source language MFCC to a target language mel-spectrogram. However, this approach did not converge at all and produced no audible speech. [8] also observed a similar result with a similar scenario.

Table 1. Our experiment results based on BTEC French-English speech-to-speech translation:

Model (FR-EN)		BLEU	METEOR
Baseline			
Tacotron with MFCC input		-	-
Proposed Speech2Code			
Codebook	Time Reduction		
32	4	19.4	19.1
32	8	23.8	22.2
32	12	23.2	22.1
64	4	16.1	16.9
64	8	24.4	22.9
64	12	25.0	23.2
128	4	16.9	17.4
128	8	23.3	22.1
128	12	24.2	21.9
Topline (Cascade ASR ->TTS)		47.4	41.2

7.2. Topline with Cascade ASR-TTS

In this paper, we set the topline performance by using the cascade of ASR and TTS system. First, we train the ASR system by using the source language MFCC as the input and target language character transcription. Second, we train a TTS based on Tacotron [3] to generate a speech from the target language characters to the target language speech representation.

7.3. French-to-English by Speech to Codebook

Table 1 shows our experimental result on various hyperparameters across different codebook sizes and time-reductions. We tried several hyperparameters, including codebook size and time-reduction factor. Our best performance was produced by codebook of 64 and a time-reduction factor of 12 with a score of 25.0 BLEU and 23.2 METEOR.

Table 2. Our experiment results based on BTEC Japanese-English speech-to-speech translation.

Model (JA-EN)		BLEU	METEOR
Baseline			
Tacotron with MFCC source		-	-
Proposed Speech2Code			
Codebook	Time Reduction		
32	4	14.8	15
32	8	14.2	15.6
32	12	16	16
64	4	10.8	12.1
64	8	14.2	14.7
64	12	14.7	14.8
128	4	11.9	13.5
128	8	15.3	15.3
128	12	14.9	14.5
Topline (Cascade ASR ->TTS)		37.4	32.8

7.4. Japanese-to-English by Speech to Codebook

Table 2 shows our experimental result on various hyperparameters across different codebook sizes and time-reductions. We tried several hyperparameters, including codebook size and time-reduction factor. Our best performance was produced by a codebook of 128 and a time-reduction factor 8 with a score of 15.3 BLEU and 15.3 METEOR.

7.5. Transcription Example and Discussion

In Table 3, we provide some transcriptions example from the ground-truth, our proposed speech-to-code and topline cascade ASR-TTS models. In the first result, all models translation contains similar meaning with the ground truth. In the second result, all models still maintain a similar semantic with the ground-truth. However, compared to the topline, the speech-to-code does not produce the additional translation for “as soon as he comes in”. In the third result, our proposed method can only translate the beginning of the sentence correctly and produce incorrect result in the latter part. From the transcription result, the missing part and arbitrary transcription in the latter half might be interesting to be investigated in the future.

For further information and translation samples, our reader could refer to:

<https://sp2code-translation-v1.netlify.com/>.

8. CONCLUSION

In this paper, we proposed a novel approach for training a speech-to-speech translation between two languages without any transcription. First, we trained a discrete quantized autoencoder to generate a discrete representation from the target speech features. Second, we trained a sequence-to-sequence model to predict the codebook sequence given the source speech representation. This method is applicable to any type of language, with or without a written form because the target speech representations are trained and generated unsupervisedly. Based on our experiment result, our model can perform a direct speech-to-speech translation on French-English and Japanese-English.

Table 3. Transcription example between the ground truth, our proposed Speech2Code, and topline (Cascade ASR-TTS) model.

Model	Transcription Result
Groundtruth	how long are you going to stay
Speech2Code FR-EN	how long are you going to stay
Speech2Code JA-EN	how long will it take
Topline FR-EN	how long are you staying
Topline JA-EN	how long are you staying
Groundtruth	please tell him to call me as soon as he comes in
Speech2Code FR-EN	please tell him to call me back
Speech2Code JA-EN	please tell him that i called
Topline FR-EN	please tell her to call me and check it
Topline JA-EN	please ask him to call me as soon as possible
Groundtruth	i would like a balcony seat please
Speech2Code FR-EN	i would like to have this film please
Speech2Code JA-EN	i would like a seat near the seat
Topline FR-EN	i would like a balcony seat please
Topline JA-EN	i would like a balcony seat

9. ACKNOWLEDGMENTS

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

10. REFERENCES

- [1] Jan Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio, “Attention-based models for speech recognition,” in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, 2015, pp. 577–585.
- [2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, “Neural machine translation by jointly learning to align and translate,” *CoRR*, vol. abs/1409.0473, 2014.
- [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.
- [4] Long Duong, Antonios Anastasopoulos, David Chiang, Steven Bird, and Trevor Cohn, “An attentional model for speech translation without transcription,” in *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, 2016, pp. 949–959.
- [5] Alexandre Bérard, Olivier Pietquin, Christophe Servan, and Laurent Besacier, “Listen and translate: A proof of concept for end-to-end speech-to-text translation,” *CoRR*, vol. abs/1612.01744, 2016.

- [6] Ron J. Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen, "Sequence-to-sequence models can directly translate foreign speech," in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, 2017, pp. 2625–2629.
- [7] Takatomo Kano, Sakriani Sakti, and Satoshi Nakamura, "Structured-based curriculum learning for end-to-end english-japanese speech translation," in *Proc. Interspeech 2017*, 2017, pp. 2630–2634.
- [8] Ye Jia, Ron J Weiss, Fadi Biadsy, Wolfgang Macherey, Melvin Johnson, Zhifeng Chen, and Yonghui Wu, "Direct speech-to-speech translation with a sequence-to-sequence model," *arXiv preprint arXiv:1904.06037*, 2019.
- [9] Maarten Versteegh, Roland Thiollere, Thomas Schatz, Xuan Nga Cao, Xavier Anguera, Aren Jansen, and Emmanuel Dupoux, "The zero resource speech challenge 2015," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015, pp. 3169–3173.
- [10] Ewan Dunbar, Xuan Nga Cao, Juan Benjumea, Julien Karadayi, Mathieu Bernard, Laurent Besacier, Xavier Anguera, and Emmanuel Dupoux, "The zero resource speech challenge 2017," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 323–330.
- [11] Ewan Dunbar, Robin Algayres, Julien Karadayi, Mathieu Bernard, Juan Benjumea, Xuan-Nga Cao, Lucie Miskic, Charlotte Dugrain, Lucas Ondel, Alan W. Black, Laurent Besacier, Sakriani Sakti, and Emmanuel Dupoux, "The zero resource speech challenge 2019: TTS without T," *CoRR*, vol. abs/1904.11469, 2019.
- [12] Andy T. Liu, Po-chun Hsu, and Hung-yi Lee, "Unsupervised end-to-end learning of discrete linguistic units for voice conversion," *CoRR*, vol. abs/1905.11563, 2019.
- [13] Suhee Cho, Yeonjung Hong, Yookyung Shin, and Youngsun Cho, "VQVAE with speaker adversarial training," <https://github.com/Suhee05/Zerospeech2019>, 2019.
- [14] Andros Tjandra, Berrak Sisman, Mingyang Zhang, Sakriani Sakti, Haizhou Li, and Satoshi Nakamura, "VQVAE unsupervised unit discovery and multi-scale code2spec inverter for zerospeech challenge 2019," *CoRR*, vol. abs/1905.11449, 2019.
- [15] Aaron van den Oord, Oriol Vinyals, et al., "Neural discrete representation learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 6306–6315.
- [16] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 1096–1103.
- [17] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [18] Lukasz Kaiser, Samy Bengio, Aurko Roy, Ashish Vaswani, Niki Parmar, Jakob Uszkoreit, and Noam Shazeer, "Fast decoding in sequence models using discrete latent variables," in *International Conference on Machine Learning*, 2018, pp. 2395–2404.
- [19] Aurko Roy, Ashish Vaswani, Niki Parmar, and Arvind Nee-lakantan, "Towards a better understanding of vector quantized autoencoders," *OpenReview*, 2018.
- [20] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.
- [21] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.
- [22] Thang Luong, Hieu Pham, and Christopher D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, Sept. 2015, pp. 1412–1421, Association for Computational Linguistics.
- [23] Andros Tjandra, Sakriani Sakti, and Satoshi Nakamura, "Multi-scale alignment and contextual history for attention mechanism in sequence-to-sequence model," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 648–655.
- [24] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [25] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint arXiv:1505.00853*, 2015.
- [26] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.
- [27] Daniel Griffin and Jae Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.
- [28] Genichiro Kikui, Eiichiro Sumita, Toshiyuki Takezawa, and Seiichi Yamamoto, "Creating corpora for speech-to-speech translation," in *Eighth European Conference on Speech Communication and Technology*, 2003.
- [29] Gen-ichiro Kikui, Seiichi Yamamoto, Toshiyuki Takezawa, and Eiichiro Sumita, "Comparative study on corpora for speech translation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1674–1682, 2006.
- [30] Andros Tjandra, Sakriani Sakti, and Satoshi Nakamura, "Listening while speaking: Speech chain by deep learning," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 301–308.
- [31] Andros Tjandra, Sakriani Sakti, and Satoshi Nakamura, "Machine speech chain with one-shot speaker adaptation," *Proc. Interspeech 2018*, pp. 887–891, 2018.
- [32] A. Tjandra, S. Sakti, and S. Nakamura, "End-to-end feedback loss in speech chain framework via straight-through estimator," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 6281–6285.
- [33] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto, "librosa: Audio and music signal analysis in Python," 2015.
- [34] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

- [35] Satyanjee Banerjee and Alon Lavie, “METEOR: An automatic metric for MT evaluation with improved correlation with human judgments,” in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, Ann Arbor, Michigan, June 2005, pp. 65–72, Association for Computational Linguistics.
- [36] Jonathan H Clark, Chris Dyer, Alon Lavie, and Noah A Smith, “Better hypothesis testing for statistical machine translation: Controlling for optimizer instability,” in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*. Association for Computational Linguistics, 2011, pp. 176–181.