EURASIP Journal on Audio, Speech, and Music Processing

RESEARCH Open Access

Text-to-speech system for low-resource language using cross-lingual transfer learning and data augmentation

Zolzaya Byambadorj[1*], Ryota Nishimura[1], Altangerel Ayush[2], Kengo Ohta[3] and Norihide Kitaoka[4]

Abstract

Deep learning techniques are currently being applied in automated text-to-speech (TTS) systems, resulting in significant improvements in performance. However, these methods require large amounts of text-speech paired data for model training, and collecting this data is costly. Therefore, in this paper, we propose a single-speaker TTS system containing both a spectrogram prediction network and a neural vocoder for the target language, using only 30 min of target language text-speech paired data for training. We evaluate three approaches for training the spectrogram prediction models of our TTS system, which produce mel-spectrograms from the input phoneme sequence: (1) cross-lingual transfer learning, (2) data augmentation, and (3) a combination of the previous two methods. In the cross-lingual transfer learning method, we used two high-resource language datasets, English (24 h) and Japanese (10 h). We also used 30 min of target language data for training in all three approaches, and for generating the augmented data used for training in methods 2 and 3. We found that using both cross-lingual transfer learning and augmented data during training resulted in the most natural synthesized target speech output. We also compare single-speaker and multi-speaker training methods, using sequential and simultaneous training, respectively. The multi-speaker models were found to be more effective for constructing a single-speaker, low-resource TTS model. In addition, we trained two Parallel WaveGAN (PWG) neural vocoders, one using 13 h of our augmented data with 30 min of target language data and one using the entire 12 h of the original target language dataset. Our subjective AB preference test indicated that the neural vocoder trained with augmented data achieved almost the same perceived speech quality as the vocoder trained with the entire target language dataset. Overall, we found that our proposed TTS system consisting of a spectrogram prediction network and a PWG neural vocoder was able to achieve reasonable performance using only 30 min of target language training data. We also found that by using 3 h of target language data, for training the model and for generating augmented data, our proposed TTS model was able to achieve performance very similar to that of the baseline model, which was trained with 12 h of target language data .

Keywords: Speech synthesis, Text to speech, Transfer learning, Data augmentation, Low-resource language

1 Introduction

Deep learning techniques are now widely used in TTS systems due to their ability to generate higher quality synthesized speech than traditional methods. For example, recent end-to-end neural models such as Tacotron [1], Tacotron 2 [2], Deep Voice 3 [3], and Char2Wav [4] are all able to generate natural-sounding speech. However, these models require a large amount of paired text-speech data for training, as well as substantial processing power. Chung [5] found that the Tacotron model requires more than 10 h of training data to produce good synthesized speech. But collecting large amounts of speech data is expensive and time-consuming, which creates a significant hurdle when developing TTS systems for the world's many, less widely spoken languages. Thus, recent studies

*Correspondence: bb.zolzaya@gmail.com
[1] Department of Information Science and Intelligent Systems, Tokushima University, Tokushima, Japan
Full list of author information is available at the end of the article

have proposed a variety of techniques which can be used for TTS with low-resource languages. These techniques include:

Monolingual transfer learning: When there is only a small dataset of a particular type of speech available, such as the speech of an additional speaker, emotional speech data, and alternative speaking style data, a pre-trained model, trained using a large amount of a different type of speech data, can be used as a low-resource speech model by using transfer learning. Tits et al. [6] explored transfer learning for TTS with low-resource, emotional speech. After training their model with a large dataset, they fine-tuned it using a small, neutral speech dataset from a new speaker. They then adapted the resulting model by training it with a small, emotional speech dataset also created using the new speaker. Bollepalli et al. [7] used the same method as in [6], except that Lombard speech was used for transfer learning instead of emotional speech. They adapted a pre-trained TTS system using 2 h of normal speech data from a new speaker. They then adapted the normal speech model for the new speaker to a different speaking style from the same speaker, such as Lombard speech, using a transfer learning method. In studies [6] and [7], all of the datasets used were in the same language.

Cross-lingual transfer learning: Since large amounts of data are often unavailable for low-resource languages, most of the proposed approaches for TTS for these languages have used cross-lingual transfer learning to train their target language TTS systems. However, when using cross-lingual transfer learning, input space mismatches can occur. Chen et al. [8] developed TTS systems for low-resource languages and explored cross-lingual symbol mapping to improve the transfer of knowledge learned previously from a high-resource language dataset. Three methods for cross-lingual symbol mapping were evaluated, and two of these methods, which were denoted "Unified" and "Learned", achieved good results. Their proposed method "Learned" automatically mapped the relationship between source and target language linguistic symbols to transfer knowledge learned previously. To do this, they pre-trained an automatic speech recognition (ASR) system using the source language, then fixed the parameters of the pre-trained ASR system and concatenated their proposed Phonetic Transformation Network (PTN). They used the target language data in this stage, and PTN learned to find the possible target symbols given the ASR output, source symbols. Their results when using their proposed "Learned" method were no better than when using the "Unified" method, but were comparable. In this study, we used two high-resource languages, English and Japanese, and these datasets were used both sequentially and simultaneously when training the model. Therefore, in our approach, we used the "Unified" method. In other words, we converted the transcriptions of all of the utterances in each dataset into their phonetic transcriptions based on IPA, and we then created a unified symbol set to solve the input space mismatch problem.

Multi-speaker models: In addition, multi-speaker models have been used to reduce the amount of training data needed for TTS. Latorre et al. [9] have shown that multi-speaker models, which use a small amount of data from each speaker, are more effective than speaker-dependent models trained with more data. In [10], researchers investigated the effect on TTS performance of training a multi-speaker model using a speaker-imbalanced corpus. They found that simply combining all the available data from every speaker when training the multi-speaker model produced better results than using a speaker-dependent model. Gutkin et al. [11] constructed a multi-speaker, acoustic database using crowdsourcing, and then used it to bootstrap a statistical, parametric speech synthesis system. These studies all used multi-speaker datasets which were in the same language as the target speech.

Multilingual models: Since high-quality, multi-speaker data is generally unavailable in most low-resource languages, multilingual or multilingual/multi-speaker models can be used to address data availability issues. Yu et al. [12] proposed a multilingual bi-directional long short-term memory (BLSTM)-based speech synthesis method which transforms the input linguistic features into acoustic features. The input layer and hidden layers of the BLSTM were shared across different languages for speech synthesis of low-resource languages, but the output layer was not shared. The input feature vectors of different languages were combined to form a single, uniform representation of the input features. The shared hidden layers transform the uniform input features into an internal representation that can benefit low-resource TTS. Their proposed multilingual BLSTM-based speech synthesis method was able to more accurately predict acoustic features than a monolingual BLSTM. Li and Zen [13] built a long short-term memory (LSTM) recurrent neural network based, multi-language/multi-speaker (MLMS) statistical parametric speech synthesis system using six languages. Their proposed MLMS model achieved similar performance to that of conventional language-dependent and speaker-dependent models. They also demonstrated that adapting their proposed system to new languages using limited training data achieved better performance than building low-resource language models from scratch. Korte et al. [14] conducted experiments to compare the naturalness of speech from single-speaker models with speech from multilingual models when different amounts of the target speaker's data were used for training. They also compared the naturalness of speech from monolingual, multi-speaker models with speech from multilingual, multi-speaker models when larger amounts of non-target language training data were used. As a result, they demonstrated the effectiveness of using multilingual models to improve the naturalness of speech in low-resource language TTS systems, finding that the use of foreign language training data improved the quality of low-resource target language speech output. Their proposed multilingual model used a separate encoder for each language to represent language information. They found that this method of representing language information was more effective than using language embedding. Lee et al. [15] built bilingual, multi-speaker TTS models using two monolingual datasets to investigate how speech synthesis networks learn pronunciation from datasets of different languages. They noticed that two, learned phoneme embeddings were located close together when they had similar pronunciations. Therefore, based on this observation, they proposed a training framework to utilize phonetic information from a different language. They showed that pre-training a speech synthesis model using datasets from both high- and low-resource languages could enhance the performance of the TTS model with low-resource languages. Chen et al. [16] built a cross-lingual, bilingual TTS system with learned speaker embedding, using two monolingual, multi-speaker datasets. A speaker encoder model, trained with the English and Chinese datasets, was used to represent the latent structure the utterances of different speakers and language pronunciations. The learned speaker embedding extracted by the speaker encoder was then used to condition the spectrogram prediction network. They noted that the learned speaker embedding could represent the relationship between pronunciations across the two languages, even though English and Chinese have different phoneme sets. They observed that phonemes with similar pronunciations were inclined to remain closer to each other across the two languages than to the other phonemes.

Data augmentation: Data augmentation is widely used in ASR to produce additional synthetic training data [17], [18] and [19]. Recent speech synthesis studies have shown that data augmentation can also improve the performance of TTS models. Huybrechts et al. [20] built high-quality TTS models for expressive speech, to be used when only a very small amount of expressive speech data is available for a target speaker. First, they generated synthetic speech data from a source speaker to the target speaker in the desired speaking style using a voice conversion model. Second, they trained the TTS model using the generated synthetic speech data and the target speaker recordings. Then the pre-trained model was fine-tuned with non-synthetic data in order to focus on the actual target space more closely. Using both data augmentation and fine-tuning methods improved the signal quality, naturalness, and style adequacy of the synthetic speech without any drop in speaker similarity. Hwang et al. [21] proposed a TTS-driven data augmentation method to improve the quality of the output of a non-autoregressive (NAR) TTS system. First, they trained the source autoregressive (AR) TTS model using recorded speech data from a professional speaker. Then, text scripts were prepared for generating synthetic data using the source AR TTS model. After generating a large amount of synthetic data (179 h), this augmented corpus was used to train the target NAR TTS model. The proposed data augmentation method was effective and significantly improved the quality of the output of the NAR TTS system. Cooper et al. [22] investigated two speaker augmentation scenarios for a multi-speaker TTS model. The first speaker augmentation method creates “artificial” speakers by changing the speed of the original speech using a sound exchange audio manipulation tool (SoX) [23]. The second method uses low-quality data containing background noise and reverberation, which was collected for purposes other than TTS, such as ASR. This low-quality data consisted of four new ASR corpora which included speech in different dialects. They modified the postnet and encoder of the Tacotron model to support the additional channel and dialect factors. The channel represents a factor in the low-quality data jointly caused by the frequency characteristics of the recording equipment, noise and reverberation. Their modified Tacotron model trained with low-quality data improved the naturalness of the synthesized speech of speakers seen during training. They observed that the speaker augmentation method using low-quality data contributed to speech naturalness rather than speaker similarity and improved the quality of the synthetic speech for seen speakers. Liu et al. [24] built a bilingual TTS model for use when the amount of target language data was limited. They tried to solve the problems of accent carry-over and mispronunciation. Accent carry-over can occur during cross-lingual speech synthesis, so tone preservation mechanisms were used to address this. Mispronunciation during low-resource synthesis occurs when the synthesizer does not have enough examples to learn proper phonetization. They addressed this problem with data augmentation, using noise and speed perturbations to increase the target low-resource language dataset 10-fold. SoX was used for speed perturbation. Their experimental results demonstrated the significant potential of data augmentation for improving speech quality when working with extremely low-resource languages. Our proposed method uses a data augmentation method similar to that used in [22] and [24], but in these studies either two or four additional versions of the original utterances, respectively, were generated by changing the speed factor. Vehicle noise was then added to all of the utterances in [24]. In this study, we generated synthetic speech with a wider range of variation, creating 26 versions of each utterance from the original speech by changing the pitch and speed. While [22] and [24] increased the amount of training data 3-fold and 10-fold, respectively, using data augmentation we increased the amount of target language training data 27 times the size of the original dataset.

In previous studies of monolingual and cross-lingual transfer learning which appear in the literature, knowledge learned from a large amount of data was transferred and a pre-trained model was adapted with the specific type of speech data in the same language or speech data in another language. In this study, we are proposing a single-speaker TTS system for use in a low-resource scenario; therefore, we also used the same method proposed in previous studies, and trained a monolingual, single-speaker TTS model. But in our approach, two high-resource languages, English and Japanese, were used sequentially for pre-training to improve the transfer of linguistic knowledge. Both of the high-resource language corpora we used are publicly available and contain speech data from a different, single, female speaker. In contrast, our target Mongolian language dataset contains speech data from a single, male speaker. Therefore, our single-speaker TTS model was trained using a different single-speaker dataset at each training stage, e.g., during the pre-training and fine-tuning stages. The phonemes of the few Mongolian letters which are not contained in English are contained in Japanese, and vice versa. For example, the phonemes of the Cyrillic letters ‘ө’ and ‘ц’ are not contained in English, while the phonemes of the Cyrillic letters ‘у,’ ‘ө’ and ‘л’ are not contained in Japanese. In addition, the Mongolian language belongs to the Altaic family of languages. It has been suggested that Japanese is linguistically related to Altaic, as there are structural similarities and the pronunciations of the phonemes are very similar. English, on the other hand, is an Indo-European language. Therefore, we first used English, then Japanese, to train the pre-trained model, because it is generally more effective to train models using the less similar data first.

In previous studies involving multi-speaker models and multilingual models which appear in the literature, investigators built multi-speaker models using monolingual data in order to reduce the amount of training data needed, while multilingual models were trained using multilingual or multilingual, multi-speaker datasets. In contrast, our aim is to build a monolingual, single-speaker TTS model which can synthesize the Mongolian speech of the male speaker recorded in the target language dataset. But we believe that a multi-speaker model can be used as a component in its development. In this study, since we do not have multi-speaker data for the targeted low-resource language, we instead trained the multi-speaker model with multilingual data, using the same input representation (one speaker per language, with different speakers for each language), and then fine-tuned it to realize the proposed monolingual, single-speaker TTS model. In other words, we used a multi-speaker model and multilingual data to obtain a monolingual, single-speaker model.

Although we used transfer learning in two different situations (with both single- and multi-speaker models) to address the issue of the limited data, we found that 30 min of target language training data was insufficient for cross-lingual training. Therefore, we generated a training dataset which was 27-fold larger using data augmentation in order to solve the limited target language data issue, and this dataset was used to train both the single- and multi-speaker models. Since the augmented data can be considered to be from different speakers, it may therefore be more suitable for training a multi-speaker model.

In this paper, we propose a single-speaker TTS system for the low-resource language of Mongolian. The contributions of this paper are as follows:

1 We explore the TTS model’s performance after cross-lingual transfer learning using high- and low-resource language datasets. These datasets were used both sequentially and simultaneously during the training of the spectrogram prediction network.
2 We create a large amount of augmented data by changing some of the characteristics of a very small amount of original target language speech data, such as pitch and speed, and evaluate the TTS model’s performance when this augmented data is used for training.
3 We show how the performance of our low-resource language TTS model is enhanced by combining the previous two methods.
4 We investigate how much original target language data is needed when training the proposed TTS model in order to achieve the same results as the baseline model trained with a much larger amount of target language data.
5 We demonstrate how augmented data can also be used to train the neural vocoder, in addition to the spectrogram prediction network.

In the experiments related to contributions 1–3 described above, we tested two types of TTS models: single-speaker ($\mathrm{M_S}$) and multi-speaker ($\mathrm{M_M}$).

The rest of this paper is organized as follows. The architecture used to conduct our experiments, the input representation method used, the datasets used and all subjective evaluations we conducted are presented in Section 2. In Section 3, we describe each of the various methods we tested when building our proposed TTS system, our experiments, and the results of our subjective evaluations. Our conclusions are then presented in Section 4.

2 Experimental setup

2.1 TTS system

As mentioned previously, we tested single-speaker and multi-speaker TTS models, trained with high-resource and low-resource language datasets, sequentially or simultaneously, with or without augmented data, to obtain a single-speaker TTS system that is effective when only a limited amount of target language training data is available. Our base TTS system consists of three components: an x-vector speaker encoder, a Tacotron 2-based spectrogram prediction network, and PWG neural vocoder [25]. We adopted the original Tacotron 2 architecture, which consists of a bi-directional LSTM-based encoder and a unidirectional LSTM-based decoder with location sensitive attention, using the same hyperparameters as in [2], except for the addition of a loss function for guided attention loss [26], which supports faster convergence. Although a reduction factor (r), representing the number of frames to generate at each decoding step, was not used in [2], we used the reduction factor r = 1 for the single-speaker model, while the reduction factor r = 2 was used for the multi-speaker model to speed up the training process. Table 1 shows the hyper-parameters used in all models. We used a batch size of 32 for all of the models, except the pre-trained, multi-speaker models trained with both high-resource language datasets. The spectrogram prediction network was constructed using the open-source speech processing toolkit ESPnet [27]. We used a pre-trained x-vector [28] for speaker embedding, as provided by Kaldi. The speaker embeddings were concatenated with each encoder state. PWG is a non-autoregressive neural vocoder trained to minimize multi-resolution, short-time Fourier transform (STFT) loss and waveform domain adversarial loss. We used the public implementation[1] to train the PWG neural vocoder with augmented data created using a very small target language dataset and it was used to generate the waveform in all of our experiments. Figure 1 shows an overview of our base TTS system. The speaker embedding network in Fig. 1 is used to train a multi-speaker model. For the single-speaker model, we used the same network without speaker embedding. Although we used three monolingual, single-speaker datasets simultaneously for the multi-speaker model, we did not use the language identity.

Table 1 Hyper-parameters and network architectures

Feature extraction	
Sampling rate	22,050 Hz
Window size	46.4 ms (1,024 pt)
Shift size	11.6 ms (256 pt)
Acoustic feature	log-mel spectrogram 80 dim
Encoder	
# phoneme embedding dimension	512
# CNN layers	3
# CNN filters	512
CNN filter size	5
# BLSTM layer	1
# BLSTM units	512
Decoder	
# LSTM layers	2
# LSTM units	1024
# Prenet layers	2
# Prenet units	256
# Postnet layers	5
# Postnet filters	512
Postnet filter size	5
# Speaker embedding dimension	512
Attention	
# Dimensions in attention	128
# Filters in attention	32
Filter size in attention	31
Sigma in guided attention loss	0.4
Reduction factor (r)	1 (M_S) / 2 (M_M)
Optimization and minibatch	
Dropout rate	0.5
Zoneout rate	0.1
Learning rate	0.001
Optimization method	Adam with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-6}$
# Epoch	300 / 500 / 1000
Batch size	32 / 64

2.2 Input representation

We chose English and Japanese as our high-resource source languages and used Mongolian as the low-resource target language in our experiments. The pronunciation of some phonemes in the three languages are similar; therefore, learned phoneme embedding can be shared, improving the performance of our low-resource language TTS model. We created a unified symbol set to solve the input space mismatch between the source and target languages before training. The transcriptions of all of the utterances in the English and Mongolian datasets were converted into their phonetic transcriptions based on IPA. For Japanese, the transcripts of all of the utterances were first converted into Romaji using an online converter [29] and then converted from their Romaji representations into phonetic transcriptions based on IPA. Table 2 shows all of the phonemes used in each dataset. Since some phonemes in these three languages have the same pronunciations, there are overlapping phonemes in the source and target languages. On the other hand, some

[1] https://github.com/kan-bayashi/ParallelWaveGAN

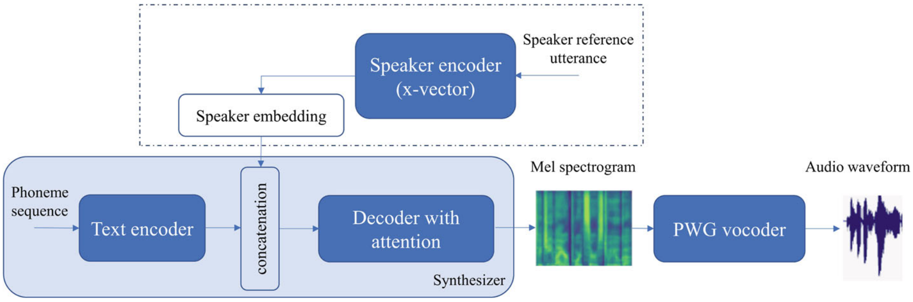


Fig. 1 Overview of the base TTS system. Speaker embedding is used to train the multi-speaker model, but is not used for training the single-speaker model

phonemes exist only in a particular source or target language. The number of phonemes which occurred only in the English language was greater than the number of phonemes that existed only in the target language. In contrast, all of the phonemes of the Japanese language are contained in Mongolian. Only one phoneme in the target language dataset, ‘ö,’ is not contained in either of the high-resource language datasets, while three phonemes, ‘l,’ ‘ʊ’ and ‘c’ are contained in the data of one of the high-resource languages. Therefore, to create the unified symbol set, the phonemes ‘ö,’ and ‘c’ were inserted into the English language dataset by replacing the phonemes that sound the most similar in the English source language dataset. We did not replace many phonemes, and each new phoneme replaced only one occurrence of the English language phonemes. This replacement is necessary when the source and target datasets are used sequentially during cross-lingual transfer learning.

2.3 Dataset

English and Japanese were selected as our high-resource source languages. As our English speech corpus, we used LJSpeech [30], a public domain dataset consisting of 13,100 utterances, with a total length of 24 h. Each audio file is a single-channel, 16-bit PCM WAV with a sampling rate of 22,050 Hz. For our Japanese speech corpus,

Table 2 Phonemes used in each dataset

#	English	Japanese	Mongolian	#	English	Japanese	Mongolian
1.	a (aɪ, aʊ)	a	a	21.	v	v	v
2.	b	b	b	22.	w	-	-
3.	d	d	d	23.	z	z	z
4.	e	e	e	24.	æ	-	-
5.	f	f	f	25.	ð	-	-
6.	g	g	g	26.	ŋ	ŋ	ŋ
7.	h	h	h	27.	ɑ	-	-
8.	i	i	i	28.	ɔ	-	-
9.	j	j (ja, jo, ju)	j (ja, jo, jʊ)	29.	ə	-	-
10.	k	k	k	30.	ɛ	-	-
11.	l	-	l	31.	ɜ	-	-
12.	m	m	m	32.	ɪ	-	-
13.	n	n	n	33.	ʃ	ʃ	ʃ
14.	o	o	o	34.	ʊ	-	ʊ
15.	p	p	p	35.	ʌ	-	-
16.	r	r	r	36.	ʒ	-	-
17.	s	s	s	37.	ʤ	ʤ	ʤ
18.	t	t	t	38.	ʧ	ʧ	ʧ
19.	u	u	u	39.	θ	-	-
20.	-	-	ö	40.	-	c	c

JSUT [31] was used. It is also a public dataset consisting of 7696 utterances, with a total length of 10 hours of paired text-speech data. We down-sampled each audio file in the corpus to 22,050 Hz. These source corpora feature the voices of different, single, female speakers.

We prepared a target speech corpus using part of a Mongolian language translation of the Bible, which was manually divided into individual sentences. The entire corpus consisted of 8183 short audio clips of a single, male speaker, with a total length of 12 h. Each audio file is a single-channel, 16-bit PCM WAV with a sampling rate of 22,050 Hz. We randomly selected 30 min of paired text-speech data, consisting of 307 utterances, to use as the target language dataset in our experiments. There are 35 letters in the Mongolian Cyrillic alphabet. We counted the number of occurrences of each letter in the 30 min of target language data before the transcriptions were converted into their phonetic representations, in order to explore how the number of occurrences of a letter affects the learning of its pronunciation. Table 3 lists the Mongolian letters contained in the 30-min and 12-h target language datasets with corresponding phonetic symbols, as well as the number of occurrences and the distribution of each letter. We sorted the list in descending order by the number of occurrences of the letters in the entire dataset. The letter 'щ' is not contained in the target language dataset because it is never used in the Mongolian language—only Russian loanwords contain this letter. Its pronunciation is identical to 'ш'; Russian loanwords which include the letter 'щ' will sometimes be spelled with 'ш.' In addition, the letters 'к' and 'ф' are also only used in foreign words, and thus appear infrequently. Also, the letter 'п' does not appear in the middle or at the end of a Mongolian word, but sometimes appears at the beginning of a word. Therefore, the four consonants 'к,' 'ф,' 'щ' and 'п' are called "special consonants" in Mongolian, and the number of occurrences of these letters is usually small. Although we randomly selected 30 min of target language data for training, the distribution of letters within this target language training data is almost the same as the distribution of letters in the entire 12 h of the target language dataset. The chart in Fig. 2 shows the distribution of each letter in both the 30 min and 12 h of target language data.

We used the entire 12 h of target language data to train the baseline TTS model (M-MN), which was used for a performance comparison with the proposed models. In addition, the baseline PWG neural vocoder (NV-MN) was also trained using the entire 12 h of target language data, for a performance comparison with the vocoder trained using augmented speech data (NV-DA).

2.4 Evaluation

We conducted an AB preference test to assess the quality of the output from the neural vocoders trained using the original target data (NV-MN) and augmented target data (NV-DA). Our subjects were asked to select the higher quality speech when comparing 15 speech samples

Table 3 Occurrences and distributions of Mongolian letters in the 30-min and 12-h target language datasets and IPA phonetic symbols

#	Letter	Phoneme	#Occurrences and distribution				#	Letter	Phoneme	#Occurrences and distribution			
			30 minutes		12 hours					30 minutes		12 hours	
1.	а	a	1870	11.00%	54403	11.33%	18.	м	m	372	2.19%	10349	2.15%
2.	н	n, ŋ	1378	8.11%	41565	8.65%	19.	в	v	234	1.38%	8175	1.70%
3.	э	e	1625	9.56%	38739	8.07%	20.	ч	ʧ	190	1.12%	6744	1.40%
4.	г	g	1030	6.06%	28254	5.88%	21.	з	z	218	1.28%	6691	1.39%
5.	р	r	1060	6.24%	26879	5.60%	22.	ы	i	148	0.87%	5109	1.06%
6.	и	i	686	4.04%	25673	5.34%	23.	ж	ʤ	259	1.52%	4890	1.02%
7.	д	d	827	4.86%	23810	4.96%	24.	ь	i	176	1.04%	4641	0.97%
8.	х	h	856	5.04%	22237	4.63%	25.	е	j	132	0.78%	4390	0.91%
9.	л	l	742	4.36%	21224	4.42%	26.	ш	ʃ	114	0.67%	3516	0.73%
10.	Y	u	684	4.02%	19969	4.16%	27.	ц	c	72	0.42%	2351	0.49%
11.	т	t	647	3.81%	19362	4.03%	28.	я	ja	27	0.16%	1765	0.37%
12.	о	o	805	4.74%	19019	3.96%	29.	ю	jʊ	17	0.10%	843	0.18%
13.	й	i	613	3.61%	18724	3.90%	30.	ё	jo	36	0.21%	727	0.15%
14.	у	ʊ	604	3.55%	17868	3.72%	31.	к	k	24	0.14%	591	0.12%
15.	ө	ö	532	3.13%	14145	2.94%	32.	ф	f	5	0.03%	471	0.10%
16.	с	s	495	2.91%	13438	2.80%	33.	п	p	1	0.006%	401	0.08%
17.	б	b	519	3.05%	13330	2.78%	34.	ъ	i	1	0.006%	26	0.005%

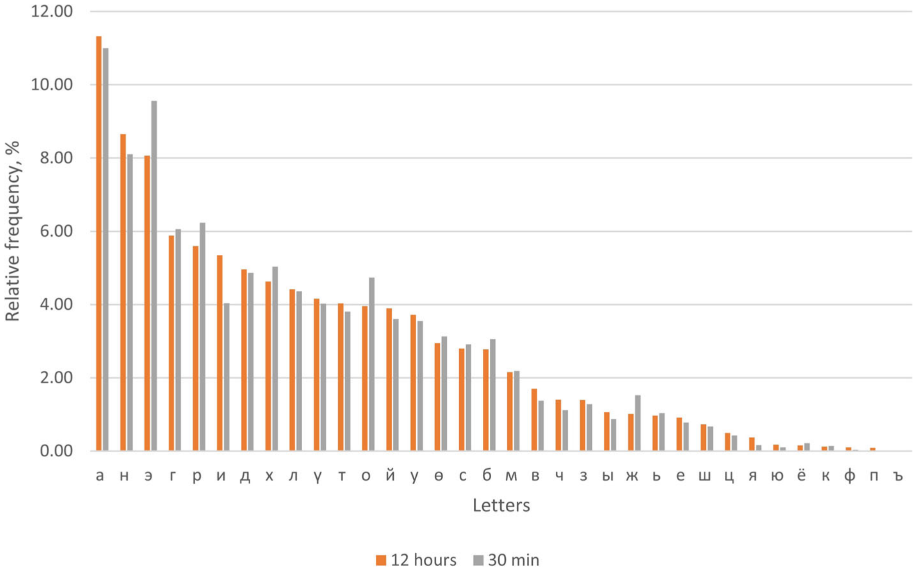


Fig. 2 Distribution of each letter in 30-min and 12-h target language datasets

generated by each vocoder. The results of this evaluation are shown in Table 5.

For the spectrogram prediction models, we conducted subjective naturalness and speaker similarity tests (Test-1 to Test-5). To evaluate the naturalness of the synthesized speech produced when using each TTS model, we conducted subjective tests using eight speech samples produced by each model which were not contained in the training dataset. We used the web-based MUltiple Stimuli with Hidden Reference and Anchor (webMUSHRA) test [32] to evaluate naturalness. All of the speech samples being evaluated are presented in one panel, and the samples within the panel are randomized. We created four separate naturalness test sets (Test-1, Test-2, Test-3, and Test-4, shown in Table 4), each containing eight stimulus panels. Each panel included a hidden reference and hidden anchors. In addition to the hidden reference and hidden anchors, Table 4 also shows all the systems that generated the speech samples included in each stimulus panel. Test-1 and Test-2 are naturalness evaluation tests of the single-speaker and multi-speaker models used in the cross-lingual transfer learning method explained in Section 3.2.1, in order to investigate the effect of the high resource language dataset. Test-3 is a comparison of all of the proposed single-speaker and multi-speaker models described in Section 3.2, conducted to determine the best performing method. Test-4, the final naturalness evaluation test, was conducted to compare the output of

Table 4 Systems used to generate the speech samples included in each stimulus panel of the MUSHRA subjective naturalness tests

Systems / Tests	MUSHRA subjective naturalness tests			
	Test-1	Test-2	Test-3	Test-4
Systems	◇ M_{SJ}-TL ◇ M_{SE10}-TL ◇ M_{SE24}-TL ◇ M_{SEJ}-TL	◇ M_{MJ}-TL ◇ M_{ME10}-TL ◇ M_{ME24}-TL ◇ M_{MEJ}-TL	◇ M_{M}-DA ◇ M_{SEJ}-TL ◇ M_{SEJ}-TL-DA ◇ M_{SEJ}-TL-DA_{D} ◇ M_{MEJ}-TL ◇ M_{MEJ}-TL-DA ◇ M_{MEJ}-TL-DA_{D} ◇ M-MN	◇ M_{MEJ}-TL-DA ◇ M_{MEJ}-TL-DA_{1hour} ◇ M_{MEJ}-TL-DA_{2hours} ◇ M_{MEJ}-TL-DA_{3hours} ◇ M-MN
Hidden reference	◇ Ground truth	◇ Ground truth	◇ Ground truth	◇ Ground truth
Hidden anchors	◇ M_{SEJ}-TL-DA ◇ M-MN	◇ M_{MEJ}-TL-DA ◇ M-MN	◇ M_{MEJ}-TL-DA_{3hours}	◇ M_{SEJ}-TL ◇ M_{MEJ}-TL

the models when using the best performing method (a combination of cross-lingual transfer learning and data augmentation, as described in Section 3.2.4), when different amounts of the target language data were used during training. All of the TTS systems shown in Table 4 are summarized in Table 7 at the end of Section 3. The results of these naturalness evaluations are shown in Figs. 5, 10, and 11.

A MUSHRA speaker similarity evaluation (Test-5) was also performed to compare the output of proposed multi-speaker model $\mathrm{M_{MEJ}}$-TL-DA, which uses a combination of transfer learning and data augmentation, with the ground truth Mongolian target speech data. Study participants also compared the output of the baseline M-MN model with the ground truth. They evaluated the similarity of eight speech samples generated from each of these two models, in comparison to the ground truth speech data, to assess their similarity to the original target language speech. The results of these comparisons are shown in Fig. 12. These comparisons were performed because, in addition to the small, target language dataset, two high-resource language datasets and augmented data were also used to build the basic single-speaker TTS system used in the proposed model.

Twenty-two subjects were asked to rate the naturalness and speaker similarity of the synthesized audio, and twenty-nine subjects were asked to rate the quality of the output from the neural vocoders. All of the subjects who participated in the subjective naturalness, similarity, and quality tests were native Mongolian speakers. Speech samples generated by each of these models and vocoders are publicly available[2].

3 Methods and results

3.1 PWG neural vocoder results

In this study, we proposed a TTS system containing both a spectrogram prediction network and a neural vocoder, for use when only a small amount of target data is available. To evaluate the effectiveness of training the vocoder with augmented data, we trained a PWG neural vocoder with 13 h of our augmented data and 30 min of original target language data (NV-DA), while the baseline vocoder was trained with 12 h of original target language data (NV-MN). We then performed an AB preference test to compare the output of the two PWG vocoders, as evaluated by twenty-nine, native Mongolian speaking subjects. We used the same M-MN baseline model used by the spectrogram prediction model for both of the vocoders. Listeners had the option of selecting "no preference" if the difference between the synthesized speech pairs was too difficult to distinguish. The test results in Table 5 show that the quality of the synthesized speech generated by the two vocoders was almost the same, as it was difficult for the listeners to distinguish the difference. Therefore, the PWG neural vocoder trained with augmented data was used to generate the waveform in all of the following experiments investigating the best method of training the spectrogram prediction network, as described in Section 3.2.

Table 5 Results of AB preference test on vocoders trained with original (NV-MN) and augmented (NV-DA) data

NV-MN (baseline)	NV-DA	No preference
21.61%	16.09%	62.30%

3.2 Spectrogram prediction models and results

3.2.1 Proposed method 1: Cross-lingual transfer learning

We trained the TTS model for our target language by transferring knowledge from our source languages in two ways. First, we used the source and target language datasets sequentially to train the TTS model without speaker embedding. To obtain pre-trained models, we first trained the TTS models using only the English (E) or Japanese (J) source language datasets, each of which contains speech data from a different, single, female speaker. The English speech dataset is more than twice as long as the Japanese dataset. Therefore, in addition to model pre-trained with the entire English dataset (E24), we also pre-trained a model using randomly selected English speech data equal in size to the Japanese dataset (E10) to determine the effect of using different proportions of the high-resource languages. The model which was pre-trained with the entire English source language dataset (E24) was also adapted by training it again with the Japanese source language dataset, creating a fourth pre-trained model (EJ). These four TTS models (E10, E24, J and EJ), pre-trained with the high-resource language datasets, were trained again using the target language dataset, as in [6], [7] and [8], which in this study consisted of Mongolian language data. All of the datasets were recorded using the voice of a single male or female speaker. Therefore, we denote our single-speaker, sequentially-trained, cross-lingual transfer learning models as $\mathrm{M_{SE10}}$-TL, $\mathrm{M_{SE24}}$-TL, $\mathrm{M_{SJ}}$-TL and $\mathrm{M_{SEJ}}$-TL. The training flow diagrams for these models are shown in Fig. 3.

In order to evaluate the effectiveness of multi-speaker training, the source and target language datasets were also used to simultaneously train a second set of pre-trained TTS models with speaker embedding as the conditioned feature. In other words, we pre-trained three multi-speaker TTS models using bilingual datasets as follows: one using the Japanese source language dataset with the target language dataset, one using 10 h of the English source language dataset with the target language dataset, and one using 24 h of the English source language dataset with the target language dataset. One multi-speaker TTS model was also pre-trained using trilingual

[2] https://zolzaya-byambadorj.github.io/tts/

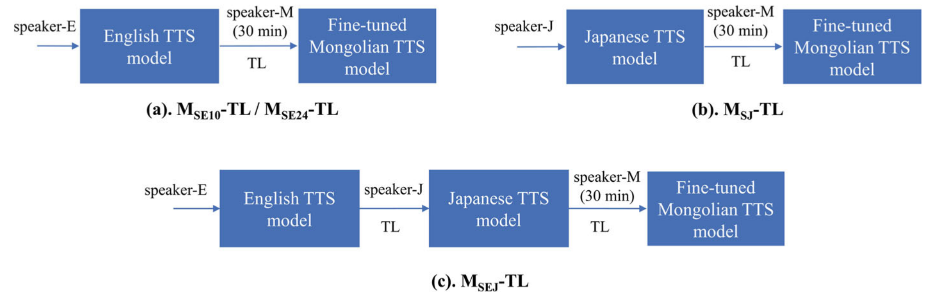

speaker-E
English TTS model
speaker-M (30 min)
TL
Fine-tuned Mongolian TTS model
(a). $\mathrm{M_{SE10}}$-TL / $\mathrm{M_{SE24}}$-TL
speaker-J
Japanese TTS model
speaker-M (30 min)
TL
Fine-tuned Mongolian TTS model
(b). $\mathrm{M_{SJ}}$-TL
speaker-E
English TTS model
speaker-J
TL
Japanese TTS model
speaker-M (30 min)
TL
Fine-tuned Mongolian TTS model
(c). $\mathrm{M_{SEJ}}$-TL

Fig. 3 Training flow diagrams for our single-speaker TTS models. Transfer learning from the source languages to the target language is used, where **a** are models using only different amounts of the English dataset, **b** is a model using only the Japanese dataset, and **c** is a model using the entire datasets of both high-resource languages

datasets (the entire, high-resource language datasets of both English and Japanese, along with 30 min of the target language dataset), with each dataset containing speech data from a different, single speaker. These four, pre-trained, multi-speaker TTS models were then fine-tuned using the same target language Mongolian dataset used to train the pre-trained multi-speaker models, as in [15]. The four multi-speaker, simultaneously-trained, cross-lingual transfer learning models were denoted as $\mathrm{M_{ME10}}$-TL, $\mathrm{M_{ME24}}$-TL, $\mathrm{M_{MJ}}$-TL, and $\mathrm{M_{MEJ}}$-TL. The training flow diagrams for these models are shown in Fig. 4.

All pre-trained TTS models shown in Figs. 3 and 4 were trained for 300 epochs, and the final models, fine-tuned with the target language dataset, were trained for 1000 epochs. We compared the performance of the four single-speaker and four multi-speaker models to understand how each training approach, i.e., using each high-resource language dataset separately, or both high-resource language datasets, either sequentially or simultaneously, affects the quality of the TTS system's output. We found that using both high-resource languages datasets simultaneously improved the performance of both the single-speaker and multi-speaker models. The results of our comparison are shown in Fig. 5 in Section 3.2.2. Based on these results, we used both high-resource language datasets for model training when using the transfer learning method with data augmentation, as described in the following section.

3.2.2 *Test-1 and Test-2: Cross-lingual transfer learning method*

In these experiments, we investigated the effects of training the models with high-resource languages on low-resource language TTS performance. Figure 5 shows the boxplots of the MUSHRA subjective naturalness scores for the single-speaker and multi-speaker TTS models described in Section 3.2.1. Native Mongolian speaking subjects performed these naturalness evaluations. As we expected, when using the same amount of data from each of the high-resource languages, the effect of Japanese language training on low-resource target language TTS performance was more beneficial than English language training for both the single-speaker and multi-speaker models. We think this is because, as explained in Section 1, Japanese and Mongolian are more similar than English and Mongolian. However, we can see that when the entire English language dataset was used for training, the performance of both the single-speaker and multi-speaker models was better than when using the Japanese high-resource language dataset. A reason for

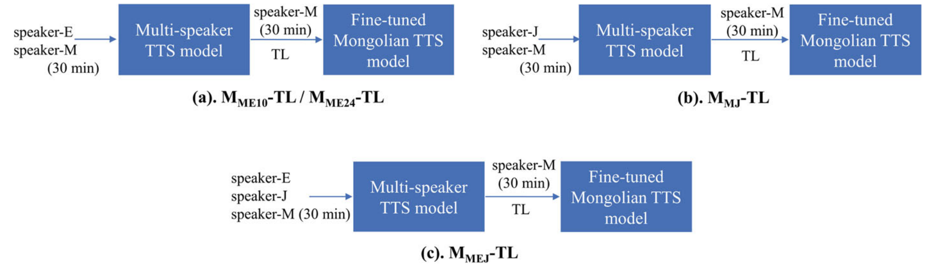


Fig. 4 Training flow diagrams for our multi-speaker TTS models. Transfer learning from the source languages to the target language is used, where **a** are models using the different amounts of the English dataset and the Mongolian dataset, **b** is a model using the Japanese and Mongolian datasets, and **c** is a model using the entire datasets of both high-resource languages and the Mongolian dataset

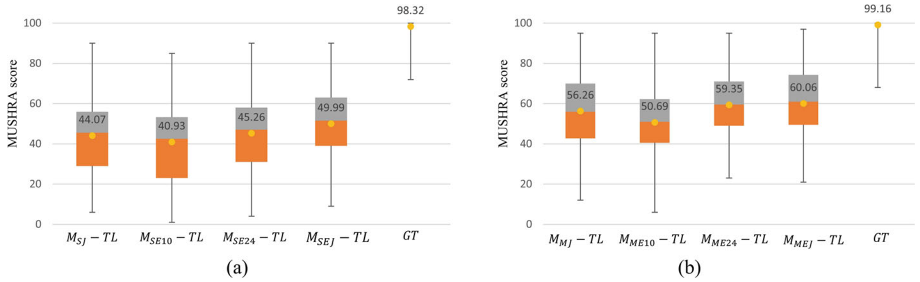


Fig. 5 MUSHRA naturalness scores for single-speaker and multi-speaker models trained using cross-lingual transfer learning, where (**a**) are single-speaker models trained with one high-resource language or both and (**b**) are multi-speaker models trained with one high-resource language or both. $M_{\text{SJ(Japanese)}}$, $M_{\text{SE10(English,10hours)}}$, $M_{\text{SE24(English,24hours)}}$, $M_{\text{SEJ(EnglishandJapanese)}}$: sequentially trained single-speaker models. $M_{\text{MJ(Japanese)}}$, $M_{\text{ME10(English,10hours)}}$, $M_{\text{ME24(English,24hours)}}$, $M_{\text{MEJ(EnglishandJapanese)}}$: simultaneously trained multi-speaker models

this could be the size of the datasets. The English speech dataset is more than twice as long as the Japanese dataset. Using both high-resource language datasets improved the performance of both the single-speaker and multi-speaker models more than using only one high-resource language dataset. Therefore, we used both high-resource language datasets to train the single-speaker and multi-speaker models using the transfer learning method in the rest of the experiments. In addition, as shown in Fig. 5, the performance of the multi-speaker models (M_{MJ}-TL, M_{ME10}-TL, M_{ME24}-TL and M_{MEJ}-TL) was better than the performance of the corresponding single-speaker models (M_{SJ}-TL, M_{SE10}-TL, M_{SE24}-TL and M_{SEJ}-TL), even though MUSHRA naturalness evaluations (a) and (b) were conducted separately. Among the multi-speaker models trained with only one high-resource language dataset during the pre-training stage (M_{MJ}-TL, M_{ME10}-TL and M_{ME24}-TL), when the target language dataset was not used in the pre-training stage, the operation of the models was almost the same as that of the corresponding single-speaker models, except for the use of speaker embedding. This suggests that using target language data when training the pre-trained model improves the performance of the TTS model. Note that we used only 30 min of the target language data to train the models shown in Fig. 5. The results for these models are low because the use of only 30 min of target language data for pre-training and fine-tuning the models is insufficient for generating good quality speech when using cross-lingual transfer learning. Studies [6] and [15] also showed that the amount of target data used affects the performance of the final fine-tuned model. But we can see from these experiments that the use of high-resource languages helps the models learn to synthesize speech in the low-resource target language.

3.2.3 *Proposed method 2: Data augmentation*

Data augmentation is a method commonly used to address the problem of insufficient data. We used a basic audio data augmentation method which involves altering the pitch and speed of the original speech data, generating synthetic data from the original samples. We changed the pitch and speed of only 30 min of the original target language data using the SoX tool [23] to synthetically generate a large amount of data with a wide range of variation, while using the same transcriptions as the original samples. The number of semitones of shift when changing the pitch was between −2.5 and 2.5, at steps of 0.5. The ratio of the speed of the augmented speech to the speed of the original speech was within the range of 0.7 to 1.55 times the speed of the original speech, at steps of 0.05, but no augmented data was generated at 1.05 times the original speed. The SoX tool shifts the full spectrum, not just the pitch, therefore all formants are also modified. We generated 26 different versions of 30 min of the original target language data, as shown in Table 6 of Section 3.2.5, creating a total of 13 hours of augmented target language data. We then trained a multi-speaker TTS model with both the augmented data and 30 min of the original target language data, treating it as a multi-speaker dataset. We also used the x-vectors for each virtual speaker generated during data augmentation. A single-speaker TTS model was also trained with the same data. We denoted these single-speaker and multi-speaker data augmentation models as M_S-DA and M_M-DA, respectively, and their training flow diagrams are shown in Fig. 6. Both models were trained with the augmented data for 500 epochs. The augmented data was also used to train the PWG neural vocoder, which was designated NV-DA.

3.2.4 *Proposed method 3: Combination of cross-lingual transfer learning and data augmentation*

We then created two additional TTS models by training the single-speaker and multi-speaker models M_{SEJ}-TL-DA and M_{MEJ}-TL-DA with the two high-resource language datasets, the original target language dataset and the augmented data. These two models are almost the same as M_{SEJ}-TL and M_{MEJ}-TL, described in Section 3.2.1, except that augmented data was also used for training.

Table 6 Number of semitones of pitch shift (PF), or ratio of speed of the new speech to speed of the original speech (SF), used when generating augmented data from the original data, for each virtual speaker

# Speaker	Pitch or speed factor		# Speaker	Pitch or speed factor	
1	−2.5	PF	14	0.85	SF
2	−2.0		15	0.9	
3	−1.5		16	0.95	
4	−1.0		17	1.1	
5	−0.5		18	1.15	
6	0.5		19	1.2	
7	1.0		20	1.25	
8	1.5		21	1.3	
9	2.0		22	1.35	
10	2.5		23	1.4	
11	0.7	SF	24	1.45	
12	0.75		25	1.5	
13	0.8		26	1.55	

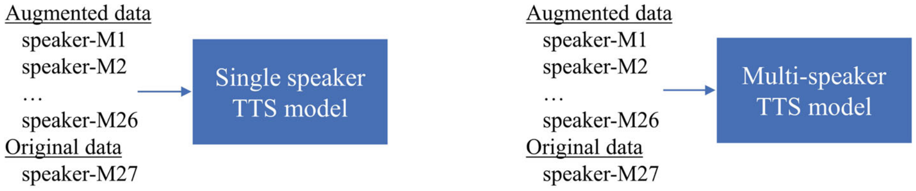
PF pitch factor, *SF* speed factor

The single-speaker model pre-trained using both high-resource languages datasets simultaneously was fine-tuned using augmented data and then fine-tuned again using the original target language data. The pre-trained multi-speaker model was trained using the trilingual datasets. During pre-training of the multi-speaker model, the two source language datasets are single-speaker datasets, while the target language dataset contains both original and augmented target data; thus, it can be considered a multi-speaker dataset with 27 "speakers". The pre-trained multi-speaker TTS model was then fine-tuned using the original target language dataset.

Both the pre-trained single-speaker and pre-trained multi-speaker models were trained for 300 epochs using the high-resource language datasets, and the pre-trained single-speaker model was also fine-tuned by training it with augmented data for 500 epochs. The final models were both fine-tuned by training each model for 1000 epochs using the original target language dataset. Training flow diagrams for the single-speaker and multi-speaker models are shown in Fig. 7.

3.2.5 *Proposed method 4: Combination of cross-lingual transfer learning and data augmentation with additional fine tuning*

For these TTS models, we added additional fine-tuning steps, using some of the augmented target language data to further improve the models' gradual adaptation to the target language. We used t-SNE [33] to visualize the x-vectors extracted from the real and virtual speakers' speech, as shown in Fig. 8. Table 6 shows the identity of each virtual speaker generated by changing the pitch and speed factors of the original speech data, where the identity of the real speaker is 27. The x-vectors extracted from virtual speakers 1, 2, 11, 12, 13, 14, 15, 21, 22, 23, 24, 25, and 26 were judged to be farther away from the x-vectors of the real speaker. Therefore, the first set of augmented data contained these 13 copies of the 30 min of the original target language data, which sounded very different from the original target language speech. The second set of augmented data contained 7 copies (from virtual speakers 3, 9, 10, 17, 18, 19 and 20) of the original data which sounded more similar to the original target speaker's voice than the augmented speech in the first set. The x-vectors extracted from virtual speakers 4, 5, 6, 7, 8, and 16 were closest to the x-vectors of the real speaker. Therefore, the third set of augmented data contained these 6 copies of the original data, which sounded the most similar to the target speaker's actual voice. We used these three sets of the augmented data to fine-tune the pre-trained model sequentially. The single speaker model, which was pre-trained with the two high-resource language datasets, was then sequentially fine-tuned using the same three sets of augmented data. Finally, the pre-trained, single-speaker model was fine-tuned with the original target language dataset. For the multi-speaker model, the pre-trained multi-speaker model was trained with the trilingual datasets (two high-resource language datasets plus the original and augmented target language datasets) and then sequentially fine-tuned using the three sets of



(a). M_S-DA (b). M_M-DA

Fig. 6 Method used to train TTS models with augmented target language data, where **a** is a single-speaker model and **b** is a multi-speaker model

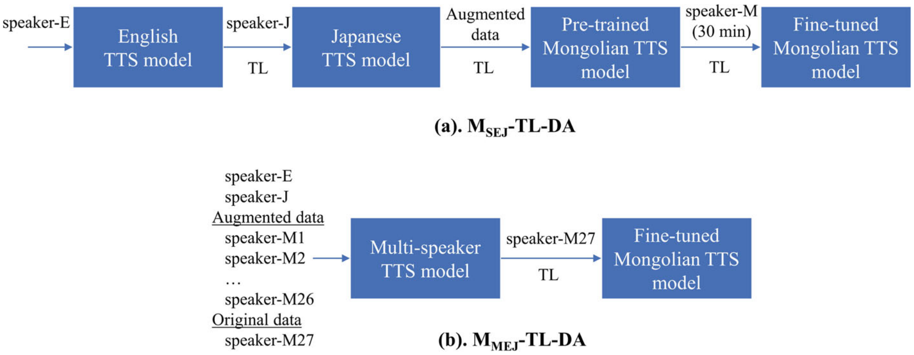


Fig. 7 Methods used to train TTS models using cross-lingual transfer learning and augmented data, where **a** is a single-speaker model and **b** is a multi-speaker model

augmented data. The model was then fine-tuned again using the original target language dataset. We denoted these single-speaker and multi-speaker models as M_{SEJ}-TL-DA$_{D}$ and M_{MEJ}-TL-DA$_{D}$, respectively. Training flow diagrams for these models are shown in Fig. 9. Both the pre-trained single-speaker and pre-trained multi-speaker models were trained for 300 epochs using the high-resource language datasets. We then further trained both of these fine-tuned models for 500 epochs at each fine-tuning step, using the sets of augmented data sequentially, before a final 500 epochs of training using the original target language data.

All proposed systems described in Section 3.2 are summarized in Table 7 at the end of Section 3.

3.2.6 *Test-3: All proposed methods*

Figure 10 shows the boxplots of the MUSHRA subjective naturalness scores, as rated by native Mongolian speakers,

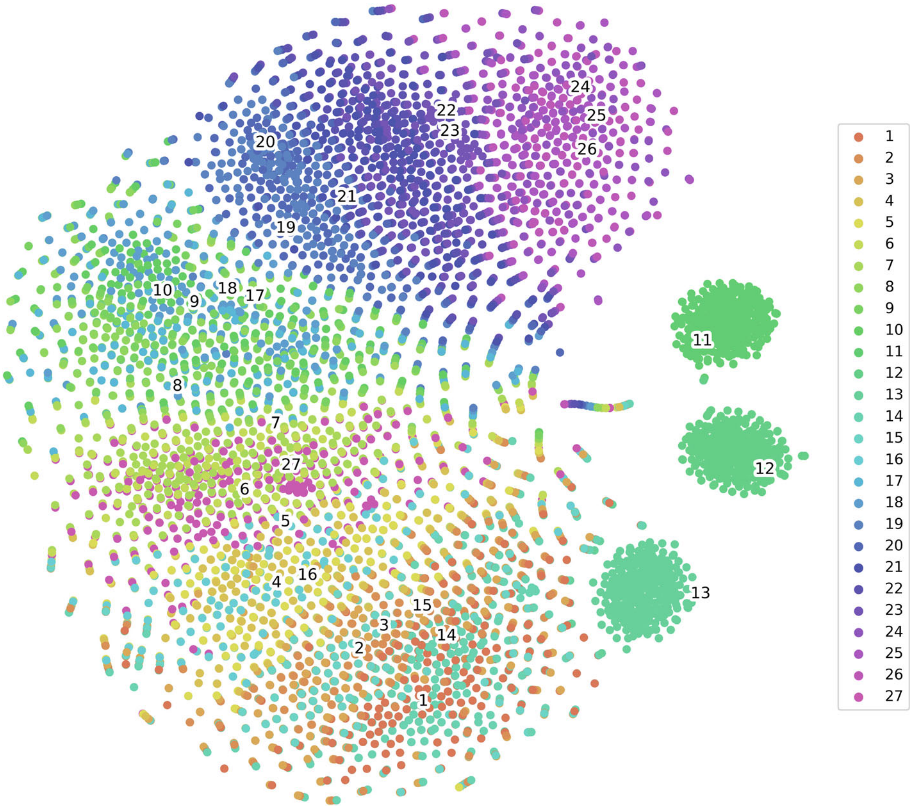


Fig. 8 t-SNE visualization of x-vectors extracted from the speech of the real and virtual speakers, where Speaker 27 is the real speaker

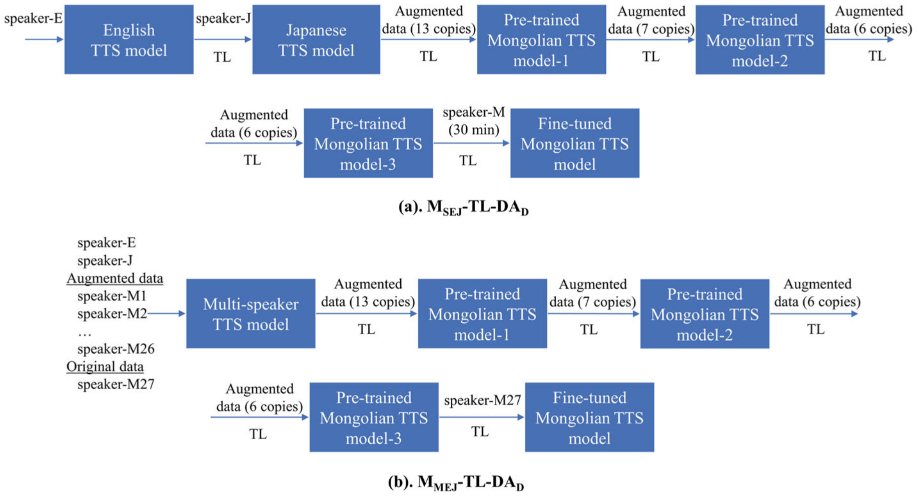

speaker-E
English TTS model
speaker-J
TL
Japanese TTS model
Augmented data (13 copies)
TL
Pre-trained Mongolian TTS model-1
Augmented data (7 copies)
TL
Pre-trained Mongolian TTS model-2
Augmented data (6 copies)
TL
Augmented data (6 copies)
TL
Pre-trained Mongolian TTS model-3
speaker-M (30 min)
TL
Fine-tuned Mongolian TTS model
(a). M_{SEJ}-TL-DA_D
speaker-E
speaker-J
Augmented data
speaker-M1
speaker-M2
…
speaker-M26
Original data
speaker-M27
Multi-speaker TTS model
Augmented data (13 copies)
TL
Pre-trained Mongolian TTS model-1
Augmented data (7 copies)
TL
Pre-trained Mongolian TTS model-2
Augmented data (6 copies)
TL
Augmented data (6 copies)
TL
Pre-trained Mongolian TTS model-3
speaker-M27
TL
Fine-tuned Mongolian TTS model
(b). M_{MEJ}-TL-DA_D

Fig. 9 Methods used to train TTS models using cross-lingual transfer learning and augmented data with additional fine-tuning steps, where **a** is a single speaker model and **b** is a multi-speaker model

for all of the proposed single-speaker and multi-speaker models described in Sections 3.2. As discussed in Section 3.2.1, we found that using both high-resource language datasets simultaneously improved the performance of both the single-speaker and multi-speaker models. Therefore, we used both high-resource language datasets for model training when using the transfer learning method with data augmentation, described in Sections 3.2.4 and 3.2.5. All of the proposed models evaluated in Fig. 10 were trained using only 30 min of original target language data.

The amount of augmented data (DA) used for training these models is almost same as the amount of original target language training data used for the baseline single-speaker model M-MN, which achieved the best results in our experiment. The augmented data was used to train both single-speaker and multi-speaker TTS models, but the single-speaker model (M_S-DA) was a failure because it could not synthesize intelligible speech. Therefore, we did not ask the study participants to rate this model. Although the naturalness score of the multi-speaker model trained with augmented data (M_M-DA) was lower than the scores of most of the other models, it was able to learn how to synthesize intelligible speech using only augmented data and 30 min of the original target language data.

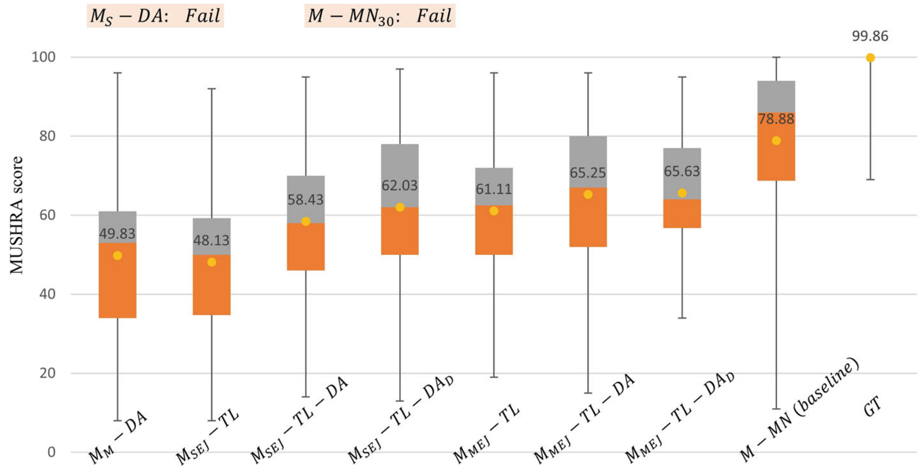


Fig. 10 MUSHRA naturalness scores for all single-speaker and multi-speaker models. M-MN: TTS model trained with 12 h of target language data; M-MN_{30}: TTS model trained from scratch with only 30 min of target language data; M_{SEJ}: sequentially trained single-speaker model; M_{MEJ}: simultaneously trained multi-speaker model; TL: cross-lingual transfer learning; DA: data augmentation; DA_D: data augmentation method with additional fine-tuning

This suggests that adding speakers could improve training of multi-speaker models, since the augmented data can be considered to be multi-speaker data. Also note that although the M_{SEJ}-TL model was trained using both high-resource language datasets and the original target speech data, the M_M-DA model received a higher score, even though the M_M-DA model was unable to learn the pronunciations of some of the letters which appeared very few times in the 30 min of original target language data, which was used to generate the augmented data. For example, the letters 'ц,' 'к,' 'ф' and 'п,' which could not be synthesized by the M_M-DA model, occurred less than 100 times in the 30 min of the target language data. But some letters, such as 'ъ' and 'я,' which also occurred less than 100 times in the data, could be learned by this model. This is because the phonetic notations of the letters 'ю,' 'я,' 'ё' are a combination of phonemes. Although these letters appear infrequently in the 30 min of original target language data before the transcriptions were converted into their phonetic representations, each phoneme contained in the phoneme notations of these letters occurred more than 100 times in the 30 min of original target language data. Furthermore, the phonetic notation of some letters, such as 'ъ,' is the same as the phonetic notation of some other letters, such as 'ь,' 'ы,' 'й' and 'и' because these letters have the same pronunciation. Therefore, although these letters only occurred a few times in the data, since we used phonetic representations the pronunciations of these letters could still be learned. We also observed that some letters which occurred less than 200 times in the data could not be synthesized clearly. However, if the transcript to be converted into speech does not include these particular, low-frequency letters, the synthesized speech created using the M_M-DA model sounds very reasonable. Thus, in general, the performance of the multi-speaker model using augmented data (M_M-DA) shows that the use of augmented data can improve the performance of TTS models.

Regarding the models trained using the cross-lingual transfer learning method, the pronunciations of the letters that only occurred a few times in the target language data could be learned from the high-resource language datasets, since there are overlapping phonemes in the source and target languages. Therefore, learned phoneme embeddings are shared by the different languages. The model trained with only 30 min of target language data from scratch (M-MN_{30}) could not synthesize intelligible speech. However, the performance of the single-speaker and multi-speaker transfer learning models (M_{SEJ}-TL and M_{MEJ}-TL) shows that the cross-lingual transfer learning approach improves TTS model performance when only a small amount of target data is available. On the other hand, single-speaker model M_{SEJ}-TL was trained using data from three languages sequentially, while multi-speaker model M_{MEJ}-TL was trained using data from three languages simultaneously. As a result, the naturalness score of multi-speaker model M_{MEJ}-TL is higher than that of single-speaker model M_{SEJ}-TL. This suggests that adding languages could also improve the training of multi-speaker models.

Each of the proposed methods, i.e., using only transfer learning or only data augmentation, were capable of improving the performance of the TTS model. Therefore, unsurprisingly, we can also see in Fig. 10 that a combination of both the transfer learning and data augmentation methods improved both single-speaker (M_{SEJ}-TL-DA) and multi-speaker (M_{MEJ}-TL-DA) model performance. As mentioned previously, adding speakers or adding languages each improved the performance of the multi-speaker models. In the case of the model M_{MEJ}-TL-DA, we added both languages and speakers simultaneously. As a result, the performance of the multi-speaker model with data augmentation (M_{MEJ}-TL-DA) was superior to that of the single-speaker model with data augmentation (M_{SEJ}-TL-DA), the multi-speaker model without data augmentation (M_{MEJ}-TL) and the multi-speaker, single-language model with data augmentation (M_M-DA). We can also see that the performance of TTS models M_{SEJ}-TL-DA_D and M_{MEJ}-TL-DA_D improved slightly when fine-tuning steps that included the use of augmented data were added. Related works [17], [22] and [24] have used data augmentation-generated synthetic speech created by changing the speed and tempo of the original speech within a relatively narrower range of variation, compared to the augmentation method used in our study. In other words, the differences between the synthetic and real data used in previous studies were not as great as in our approach. In contrast, we generated our synthetic speech using a wider range of variation, and 26 versions of the data were generated from the original speech. As a result of this wider variation, the speech of some of the virtual speakers is very different from the speech of the real speaker, while some is very similar to the real speaker's speech. Therefore, gradual fine-tuning as part of a multi-stage process may yield further gains in performance. On the other hand, although the naturalness score of single-speaker model M_{SEJ}-TL-DA_D is lower than that of multi-speaker model M_{MEJ}-TL-DA_D, we observed that the effect of the additional fine-tuning steps using augmented data was greater on the single-speaker model than on the multi-speaker model, when their naturalness scores are compared with those of the corresponding single- and multi-speaker models M_{SEJ}-TL-DA and M_{MEJ}-TL-DA. We suspect this may occur because the single-speaker model "discovers" each new speaker at each training stage, while the multi-speaker model encounters all of the speakers during the first training stage; thus, gradual fine-tuning may have been more effective for the single-speaker model and less effective for the multi-speaker model.

Finally, the results shown in Fig. 10 indicate that the performance of the multi-speaker ($\mathrm{M_M}$/$\mathrm{M_{MEJ}}$) TTS models was superior to that of the single speaker ($\mathrm{M_S}$/$\mathrm{M_{SEJ}}$) TTS models. In other words, multi-speaker models were effective as intermediate models when constructing a single-speaker, low-resource TTS model. The score of the proposed $\mathrm{M_{MEJ}}$-TL-$\mathrm{DA_D}$ model was higher than the scores of the other models trained with a limited amount of target language data, but lower than the score of the baseline M-MN model.

3.2.7 Test-4: The size of the target language training data

We also wanted to know the minimum amount of original target language training data that was needed to obtain a model with the same performance as the baseline model. Therefore, we increased the 30 min of original target language data to 1, 2, or 3 h of training data. We selected this data randomly, and then created 26 different versions of augmented data using the same amounts of original target language data (1, 2, or 3 h) as described above. Although the proposed multi-speaker model with additional fine-tuning ($\mathrm{M_{MEJ}}$-TL-$\mathrm{DA_D}$ in Fig. 10) achieved the best performance, we chose the proposed multi-speaker model utilizing a combination of cross-lingual transfer learning and data augmentation (the $\mathrm{M_{MEJ}}$-TL-DA model described in Section 3.2.4) because it is less time-consuming to train and has almost similar performance to the best-performing model. We trained it with these various amounts of target language data, and with the additional augmented training data created using this extra target language data. The performance of these variously trained models, and the baseline model, were then compared based on the naturalness of the output speech, which was measured using a MUSHRA test. Figure 11 shows the boxplots of the naturalness scores for these models. The performance of the models improved as the amount of original target language data increased. Three hours of target language data were sufficient to cover variations in pronunciation, and fluctuations in the speakers' voices were enhanced using data augmentation. Furthermore, the multi-speaker model was able to capture the features of the original voices more accurately than the single-speaker model. Therefore, the naturalness score of the proposed model trained with three hours of the original target language data was similar to the score of the baseline model trained with 12 h of target language data. We also observed that the proposed models trained with two and 3 h of the original target language data synthesized very clear, good quality speech, while the baseline model synthesized slightly more nuanced speech. Therefore, the native Mongolian speaking subjects may have preferred the output of the baseline model.

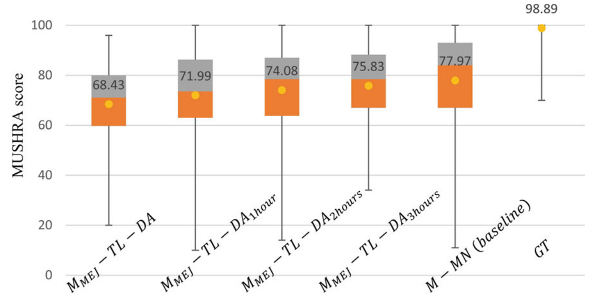


Fig. 11 MUSHRA naturalness scores for baseline and proposed multi-speaker model trained with various amounts of target language data

3.2.8 Test-5: Speaker similarity

A MUSHRA speaker similarity evaluation was performed on the output of the $\mathrm{M_{MEJ}}$-TL-DA and baseline models. Note that we again chose the proposed multi-speaker model $\mathrm{M_{MEJ}}$-TL-DA, which utilizes a combination of cross-lingual transfer learning and data augmentation (as described in Section 3.2.4), instead of the best performing model $\mathrm{M_{MEJ}}$-TL-$\mathrm{DA_D}$, for our speaker similarity evaluation test. We asked our native Mongolian speaking subjects to evaluate speech samples generated by our proposed and the baseline models in comparison to the ground truth of the original Mongolian speaker. The subjects were asked to, "Please rate the speaker similarity of each speech sample in comparison to the reference sample, on a scale of between 0 (definitely different) to 100 (definitely the same)." The results are shown in Fig. 12. Our goal in this study was to obtain a model whose performance is the same or similar to that of the baseline

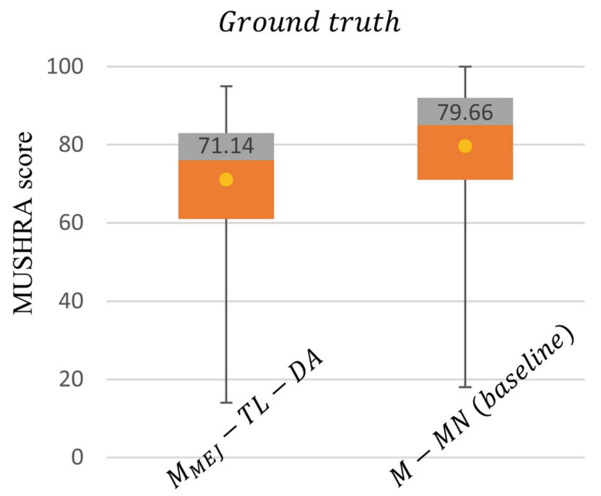


Fig. 12 Speaker similarity comparison of speech samples generated using our proposed model and using the baseline model, in relation to the ground truth

Table 7 Summarization of all systems tested

#	System	Training stage 1	Training stage 2	Training stage 3	Training stage 4	Training stage 5	Training stage 6
Spectrogram prediction models							
1.	M-MN (baseline)	MN12h	-	-	-	-	-
2.	M_{SJ}-TL	JP	MN30	-	-	-	-
3.	M_{SE10}-TL	EN10	MN30	-	-	-	-
4.	M_{SE24}-TL	EN24	MN30	-	-	-	-
5.	M_{SEJ}-TL	EN24	JP	MN30	-	-	-
6.	M_{S}-DA	AD_{30} + MN30	-	-	-	-	-
7.	M_{SEJ}-TL-DA	EN24	JP	AD_{30}	MN30	-	-
8.	M_{SEJ}-TL-DA_{D}	EN24	JP	AD_{30}-set1	AD_{30}-set2	AD_{30}-set3	MN30
9.	M_{MJ}-TL	JP + MN30	MN30	-	-	-	-
10.	M_{ME10}-TL	EN10 + MN30	MN30	-	-	-	-
11.	M_{ME24}-TL	EN24 + MN30	MN30	-	-	-	-
12.	M_{MEJ}-TL	EN24 + JP + MN30	MN30	-	-	-	-
13.	M_{M}-DA	AD_{30} + MN30	-	-	-	-	-
14.	M_{MEJ}-TL-DA	EN24 + JP + AD_{30} + MN30	MN30	-	-	-	-
15.	M_{MEJ}-TL-DA_{D}	EN24 + JP + AD_{30} + MN30	AD_{30}-set1	AD_{30}-set2	AD_{30}-set3	MN30	-
16.	M_{MEJ}-TL-DA_{1hour}	EN24 + JP + AD_{1h} + MN1h	MN1h	-	-	-	-
17.	M_{MEJ}-TL-DA_{2hours}	EN24 + JP + AD_{2h} + MN2h	MN2h	-	-	-	-
18.	M_{MEJ}-TL-DA_{3hours}	EN24 + JP + AD_{3h} + MN3h	MN3h	-	-	-	-
Neural vocoders							
19.	NV-MN (baseline)	MN12h	-	-	-	-	-
20.	NV-DA	AD_{30} + MN30	-	-	-	-	-

Table 7 Summarization of all systems tested *(Continued)*

Model type

M_{SXXX} = Single-speaker TTS model

M_{MXXX} = Multi-speaker TTS model

NV = neural vocoder

Method used for model training

TL = Cross-lingual transfer learning

DA = Data augmentation

TL-DA = Cross-lingual transfer learning and data augmentation

TL-DA_D = Cross-lingual transfer learning and data augmentation with additional fine-tuning

Databases used for training stages

EN10 = 10 hours of the English dataset

EN24 = 24 hours of the English dataset

JP = 10 hours of the Japanese dataset

MN12h = 12 hours of the target language dataset

MN30 = 30 minutes of the target language data

MN1h = 1 hour of the target language data

MN2h = 2 hours of the target language data

MN3h = 3 hours of the target language data

AD_{30} = augmented data generated from 30 minutes of the target language data

AD_{30}-set1 = the first set of the augmented data generated from 30 minutes of the target language data

AD_{30}-set2 = the second set of the augmented data generated from 30 minutes of the target language data

AD_{30}-set3 = the third set of the augmented data generated from 30 minutes of the target language data

AD_{1h} = augmented data generated from 1 hour of the target language data

AD_{2h} = augmented data generated from 2 hours of the target language data

AD_{3h} = augmented data generated from 3 hours of the target language data model in a low resource scenario. The baseline model was trained with more than 10 h of target language data, while our best performing proposed models fine-tuned pre-trained models trained with two high-resource languages and augmented data. Therefore, we wanted to know how training TTS model with the high-resource language data and augmented data affect speaker similarity between the speech samples generated by our proposed model and the ground truth. The results of our evaluation show that speaker similarity of the speech samples generated by our proposed model to the ground truth was slightly lower when using cross-lingual training and augmented data. But the similarity score of our proposed model was only slightly lower than the similarity score of the baseline model, despite the proposed model using far less original target language data for training.

Finally, we have summarized all of systems evaluated in this study in Table 7.

4 Conclusion

In this paper, we proposed a TTS system containing both a spectrogram prediction network and a neural vocoder, for use when only a small amount of target data is available. We compare the performance of various TTS models and found that multi-speaker models were effective as intermediate models when constructing a single-speaker, low-resource TTS model. We trained some models using only transfer learning and some using only data augmentation, to evaluate how each method affected the naturalness of the speech output by the TTS model. We found that training the TTS model using both cross-lingual transfer learning and data augmentation improved performance, reducing the gap between our low-resource model and the baseline M-MN model, which was trained with a much larger amount (12 h) of original target speech data. We then tried adding additional fine-tuning steps using augmented data and the original target language data, which slightly improved the performance of our proposed model. Although the naturalness and speaker similarity scores for our proposed model using both cross-lingual transfer learning and data augmentation was very reasonable, we also investigated increasing the amount of original target language data used for training. By increasing the amount of original target language data used for model training from 30 min to 3 h, our proposed model using both cross-lingual transfer learning and data augmentation achieved performance very close to that of the baseline model.

We also trained the PWG vocoder using augmented data generated from 30 min of the original target language data. As a result, our proposed method achieved almost the same speech quality as the vocoder trained with the entire 12 h of target language data. As a result, our proposed TTS system, consisting of a spectrogram prediction network and a PWG neural vocoder, was able to achieve almost equivalent performance to the baseline model using only 3 h of original target language training data, and reasonable performance using only 30 min of original target language training data.

In future work, we will investigate other TTS approaches for use with low-resource languages, to see if we can outperform our baseline TTS model trained with a large Mongolian dataset.

Abbreviations

TTS: Text-to-speech; MOS: Mean opinion score; PWG: Parallel WaveGAN; BLSTM: Bi-directional long short-term memory; LSTM: Long short-term memory; STFT: short-time Fourier transform; IPA: International Phonetic alphabet; Sox: Sound eXchange; PCM: Pulse-Code Modulation; TL: Transfer learning; DA: Data augmentation; t-SNE: t-Distributed Stochastic Neighbor Embedding; MUSHRA: Multiple stimuli with hidden reference and anchor

Acknowledgements

Not applicable.

Authors' contributions

ZB implemented the methods and conducted the experiments under the supervision of NK, RN, and KO. AA conducted the subjective test. ZB and NK were involved in the writing of the paper. All the authors read and approved the final manuscript.

Funding

Not applicable.

Availability of data and materials

LJSpeech and JSUT datasets are publicly available at: https://keithito.com/LJ-Speech-Dataset/ and https://sites.google.com/site/shinnosuketakamichi/publication/jsut, respectively. The Mongolian dataset used and/or analyzed during the current study is available from the corresponding author on reasonable request.

Declarations

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Department of Information Science and Intelligent Systems, Tokushima University, Tokushima, Japan. [2]Department of Information Technology, Mongolian University of Science and Technology, Ulaanbaatar, Mongolia. [3]Department of Creative Technology Engineering, National Institute of Technology, Anan College, Tokushima, Japan. [4]Department of Computer Science and Engineering, Toyohashi University of Technology, Toyohashi, Japan.

Received: 15 May 2021 Accepted: 7 October 2021
Published online: 04 December 2021

References

1. Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, R. A. Saurous, in *Interspeech 2017: 20-24 August 2017; Stockholm*. Tacotron: Towards end-to-end speech synthesis (ISCA, 2017), pp. 4006–4010
2. J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, Y. Wu, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 15-20 April 2018; Canada*. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions (IEEE, 2018), pp. 4779–4783
3. W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, J. Miller, in *6th International Conference on Learning Representations (ICLR): April 30-May 3, 2018; Vancouver, Canada*. Deep voice 3: Scaling

text-to-speech with convolutional sequence learning (ICLR, 2018), pp. 1094–1099

4. J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, Y. Bengio, in *5th International Conference on Learning Representations (ICLR): 24-26 April 2017; Toulon, France*. Char2wav: End-to-end speech synthesis (ICLR, 2017)
5. Y. A. Chung, Y. Wang, W. N. Hsu, Y. Zhang, R. Skerry-Ryan, in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 12-17 May 2019; Brighton, UK*. Semi-supervised training for improving data efficiency in end-to-end speech synthesis (IEEE, 2019), pp. 6940–6944
6. N. Tits, K. El Haddad, T. Dutoit, in *Proceedings of SAI Intelligent Systems Conference: 5-6 September 2019; London, United Kingdom*. Exploring transfer learning for low resource emotional tts (Springer, 2019), pp. 52–60
7. B. Bollepalli, L. Juvela, P. Alku, in *Interspeech 2019: 15-19 September 2019; Graz, Austria*. Lombard speech synthesis using transfer learning in a tacotron text-to-speech system (ISCA, 2019), pp. 2833–2837
8. Y. J. Chen, T. Tu, C. C. Yeh, H. Y. Lee, in *Interspeech 2019: 15-19 September 2019; Graz, Austria*. End-to-end text-to-speech for low-resource languages by cross-lingual transfer learning (ISCA, 2019), pp. 2075–2079
9. J. Latorre, J. Lachowicz, J. Lorenzo-Trueba, T. Merritt, T. Drugman, S. Ronanki, V. Klimkov, in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 12-17 May 2019; Brighton, UK*. Effect of data reduction on sequence-to-sequence neural tts (IEEE, 2019), pp. 7075–7079
10. H. T. Luong, X. Wang, J. Yamagishi, N. Nishizawa, in *Interspeech 2019: 15-19 September 2019; Graz, Austria*. Training multi-speaker neural text-to-speech systems using speaker-imbalanced speech corpora (ISCA, 2019), pp. 1303–1307
11. A. Gutkin, L. Ha, M. Jansche, K. Pipatsrisawat, R. Sproat, in *10th Edition of the Language Resources and Evaluation Conference: 23-28 May 2016, Portorož (Slovenia)*. Tts for low resource languages: A bangla synthesizer (ELRA, 2016), pp. 2005–2010
12. Q. Yu, P. Liu, Z. Wu, S. K. Ang, H. Meng, L. Cai, in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 20-25 March 2016; Shanghai, China*. Learning cross-lingual information with multilingual blstm for speech synthesis of low-resource languages (IEEE, 2016), pp. 5545–5549
13. B. Li, H. Zen, in *Interspeech 2016: 8-12 Sep 2016, San Francisco*. Multi-language multi-speaker acoustic modeling for lstm-rnn based statistical parametric speech synthesis (ISCA, 2016), pp. 2468–2472
14. M. de Korte, J. Kim, E. Klabbers, in *Interspeech 2020: 25-29 October 2020; Shanghai, China*. Efficient neural speech synthesis for low-resource languages through multilingual modeling (ISCA, 2020), pp. 2967–2971
15. Y. Lee, S. Shon, T. Kim, Learning pronunciation from a foreign language in speech synthesis networks (2020). arXiv:1811.09364
16. M. Chen, M. Chen, S. Liang, J. Ma, L. Chen, S. Wang, J. Xiao, in *Interspeech 2019: 15-19 September 2019; Graz, Austria*. Cross-Lingual, Multi-Speaker Text-To-Speech Synthesis Using Neural Speaker Embedding (ISCA, 2019), pp. 2105–2109
17. T. Ko, V. Peddinti, D. Povey, S. Khudanpur, in *Interspeech 2015: 6-10 September 2015; Dresden, Germany*. Audio augmentation for speech recognition (ISCA, 2015), pp. 3586–3589
18. Y. Zhou, C. Xiong, R. Socher, Improved regularization techniques for end-to-end speech recognition (2017). arXiv:1712.07108
19. M. Geng, X. Xie, S. Liu, J. Yu, S. Hu, X. Liu, H. Meng, in *Interspeech 2020: 25-29 October 2020; Shanghai, China*. Investigation of data augmentation techniques for disordered speech recognition (ISCA, 2020), pp. 696–700
20. G. Huybrechts, T. Merritt, G. Comini, B. Perz, R. Shah, J. Lorenzo-Trueba, in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 6-11 June 2021; Toronto, Ontario*. Low-resource expressive text-to-speech using data augmentation (IEEE, 2021), pp. 6593–6597
21. M. J. Hwang, R. Yamamoto, E. Song, J. M. Kim, in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 6-11 June 2021; Toronto, Ontario*. Tts-by-tts: Tts-driven data augmentation for fast and high-quality speech synthesis (IEEE, 2021), pp. 6598–6602
22. E. Cooper, C.-I. Lai, Y. Yasuda, J. Yamagishi, in *Interspeech 2020: 25-29 October 2020; Shanghai, China*. Can Speaker Augmentation Improve Multi-Speaker End-to-End TTS? (ISCA, 2020), pp. 3979–3983
23. SoX: Sound eXchange audio manipulation tool. http://sox.sourceforge.net. Accessed Aug 2021
24. R. Liu, X. Wen, C. Lu, X. Chen, in *Interspeech 2020: 25-29 October 2020; Shanghai, China*. Tone learning in low-resource bilingual tts (ISCA, 2020), pp. 2952–2956
25. R. Yamamoto, E. Song, J. M. Kim, in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 4-8 May 2020; Barcelona*. Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram (IEEE, 2020), pp. 6199–6203
26. H. Tachibana, K. Uenoyama, S. Aihara, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 15-20 April 2018; Canada*. Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention (IEEE, 2018), pp. 4784–4788
27. T. Hayashi, R. Yamamoto, K. Inoue, T. Yoshimura, S. Watanabe, T. Toda, K. Takeda, Y. Zhang, X. Tan, in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 4-8 May 2020; Barcelona*. Espnet-tts: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit (IEEE, 2020), pp. 7654–7658
28. D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, S. Khudanpur, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP): 15-20 April 2018; Canada*. X-vectors: Robust dnn embeddings for speaker recognition (IEEE, 2018), pp. 5329–5333
29. NihongoDera. https://nihongodera.com/. Accessed Aug 2021
30. K. Ito, The LJ speech dataset. https://keithito.com/LJ-Speech-Dataset/. Accessed Aug 2021
31. R. Sonobe, S. Takamichi, H. Saruwatari, JSUT corpus: free large-scale Japanese speech corpus for end-to-end speech synthesis (2017). arXiv:1711.00354
32. M. Schoeffler, S. Bartoschek, F. R. Stöter, M. Roess, S. Westphal, B. Edler, J. Herre, webmushra — a comprehensive framework for web-based listening tests. J. Open Res. Softw. **6**(1), 8 (2018)
33. L. van der Maaten, G. Hinton, Visualizing data using t-sne. J. Mach. Learn. Res. **9**(86), 2579–2605 (2008)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.