JOINT AND ADVERSARIAL TRAINING WITH ASR FOR EXPRESSIVE SPEECH SYNTHESIS

Kaili Zhang[1,†], *Cheng Gong* [1,†], *Wenhuan Lu*[*], *Longbiao Wang*[*], *Jianguo Wei, Dawei Liu*

Tianjin Key Laboratory of Cognitive Computing and Application,
College of Intelligence and Computing, Tianjin University, Tianjin, China
{kellyzhang, gongchengcheng, wenhuan, longbiao_wang, jianguo, liudave}@tju.edu.cn

ABSTRACT

Style modeling is an important issue and has been proposed in expressive speech synthesis. In existing unsupervised methods, the style encoder extracts the latent representation from the reference audio as style information. However, the style information extracted from the style encoder will entangle some content information, which will cause conflicts with the real input content, and the synthesized speech will be influenced. In this study, we propose to alleviate the entanglement problem by integrating Text-To-Speech (TTS) model and Automatic Speech Recognition (ASR) model with a share layer network for joint training, and using ASR adversarial training to eliminate the content information in the style information. At the same time, we propose an adaptive adversarial weight learning strategy to prevent the model from collapsing. The objective evaluation using word error rate(WER) demonstrates that our method can effectively alleviate the entanglement between style and content information. Subjective evaluation indicates that the method improves the quality of synthesized speech and enhances the ability of style transfer compared with the baseline models.

Index Terms— Expressive speech synthesis, style modeling, style disentanglement, automatic speech recognition

1. INTRODUCTION

In recent years, with the development of deep neural networks, end-to-end speech synthesis has made rapid progress, and the speech naturalness has closed to real humans [1, 2, 3, 4, 5]. However, the synthesized speech of most existing methods are less expressiveness. The listeners often feel dissatisfied and bored because of no emotional resonance. At present, more and more applications require highly expressive synthesized speech, such as audiobooks, newsreaders, and conversational assistants. With the increasing importance of human-computer interaction, the research on expressive speech synthesis is very promising.

Recent research on expressive speech synthesis is to learn latent representation of prosody and global style from reference audios in an unsupervised way, and then combine this latent representation with text vector to achieve speaking style transfer and control [6, 7, 8, 9, 10, 11, 12, 13]. These works [6, 10] designed to make the style of the synthesized audio imitate the style of the reference audio. Specifically, the style embedding is extracted from the style encoder, and this style embedding implicitly contains acoustic information such as rhythm, duration, pitch, energy, etc. However, the style embedding also entangles some content information, which leads to the degradation of the synthesized speech quality. During model training stage, the input text content is the same as the reference audio content, which causes the style encoder to encode some content information in the reference audio. But at the inference stage, when the input text content is different with the reference audio content, the decoder will get the content information from the style embedding, which will conflict with the real input text information. So the generated speech will be influenced and there will be problems with wrong words, missing words and fuzzy words. This phenomenon is called content leakage. Therefore, the content leakage problem has a significant negative impact on synthesized speech quality.

There are many research methods in recent research to disentangle style information and content information [14, 15, 16, 17]. The first method is to add auxiliary tasks for model training. In work [15], the authors added ASR guided tasks for model training. They trained the TTS model with unpaired input text and reference utterance, used the word error rate of the pre-trained end-to-end ASR model as an additional learning target of the TTS model, and prevented the reference encoder from encoding any text information. The second method is to use adversarial training. In work [14], the authors minimized mutual information between the style and the content. In work [16], the authors adopted a pairwise training procedure to enforce the model to map from one text to two different reference audios. They realized the disentanglement of style and other factors by introducing reconstruction loss and style loss. The third method is to use

[†]Indicates equal contribution. * Marks the corresponding author.

information bottleneck. In work [17], the prosody embedding is performed down-sampling and up-sampling to force the model to pay attention to style information.

Inspired by the auxiliary ASR tasks and adversarial training formulation, we proposed combining the task of ASR with the idea of adversarial training to prevent the style encoder from encoding the content information to eliminate the content information in the style embedding. Our TTS model extends the Tacotron [1] model by adding a style encoder and a share layer. The contributions of this paper are summarized as follows:

1. We use a share layer to integrate TTS and ASR tasks into one network. The experiment results demonstrate that the share layer is suitable for learning style information.

2. We use adversarial training to prevent the style encoder from encoding content information, and achieve the elimination of the content information in the style information.

3. We propose an adaptive adversarial weight learning strategy to prevent the model from collapsing in adversarial training.

The rest of this paper is organized as follows. In Section 2, The structure of proposed model and training strategy are introduced. Section 3 shows the experimental settings and evaluation results. The conclusion and future work are presented in Section 4.

2. PROPOSED MODEL

In this section, we mainly introduce our proposed expressive speech synthesis model based on joint and adversarial training with ASR model as shown in Fig. 1. First we talk about a pre-train TTS model, then introduce a pre-train ASR model—Listen, Attend and Spell (LAS) [18], finally introduce the details of joint training of ASR and TTS.

2.1. Pre-train TTS Model

Our TTS model extends the Tacotron model by adding a style encoder and a share layer. The structure of the modified Tacotron model is shown in Fig. 1(a), the same architecture and hyperparameters for reference encoder as [10], which consists of six 2D convolutional layers followed by a GRU layer. And the last GRU state passes through a fully connected layer to generate a 128-dimensional style embedding. Then style embedding concatenates with text embedding as the input of the decoder to synthesize the audio of the desired speaking style. The structure of the share layer is a BLSTM structure, which serves as a bridge to integrate TTS and ASR tasks into one network during joint training.

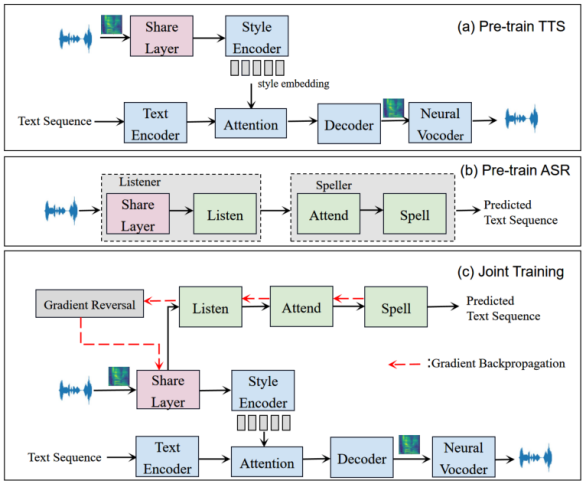

Fig. 1. The framework of the proposed approach. (a) Pre-train TTS model. (b) Pre-train ASR model. (c) Joint training of ASR and TTS model.

2.2. Pre-train ASR Model

LAS is a sequence-to-sequence ASR model with attention, which directly converts audio sequences into text sequences. The model structure is shown in Fig. 1(b). The LAS model mainly includes two sub-modules: Listener and Speller. The listener is an encoder, and the speller is a decoder based on the attention mechanism. The structure of the share layer is a BLSTM, which is a part of the listener, and it is the same as the shared layer in the modified Tacotron. The listener is used to extract the high-dimensional features of input audio sequences. Speller is an RNN network, whose function is to convert high-dimensional features from the listener into character sequences.

$$h = Listen(x) \tag{1}$$

$$\mathrm{P}(y \mid x) = AttendAndSpell(h, y) \tag{2}$$

x is the input acoustic feature sequence. Listener encodes x as high-dimensional hidden features h, and $\mathrm{P}(y \mid x)$ represent speller decodes the high-dimensional hidden features h as a probability distribution of output characters y.

2.3. Joint training of TTS and ASR

We first pre-trained a LAS model, and then pre-trained a TTS model. In the joint training stage, the model structure is shown in Fig. 1(c). We add the pre-trained LAS model to the pre-trained TTS model, and then continue to train the two models jointly, keeping the LAS model parameters fixed during the joint training process. The training algorithm of our proposed model is given in Algorithm 1. We first input the mel-spectrogram of the reference audio into a share layer,

and the output of this share layer is called share embedding which is used as the input of the LAS model. In this ASR task, we perform gradient reversal in an adversarial training formulation during gradient backpropagation, in order to make the reference audio not well recognized by the ASR model. This share embedding is used as the input of the style encoder, and this process reduces the information regarding the content in the extracted style embedding. Finally, the style embedding concatenates with the text vector from text encoder at each decoder step, as inputs of decoder. During joint training of TTS and ASR model, the model loss consists of reconstruction loss from TTS task and adversarial loss from ASR task.

Reconstruction loss: In the training process, the goal of reconstruction loss is to force the synthesized spectrum to be close to the real spectrum. So we define the reconstruction loss as:

$$\mathrm{L}_{recon} = L_{tts} = L_{mse}(y, \hat{y}) + L_{mse}(z, \hat{z}) \tag{3}$$

$\hat{y}$ is the generated mel-spectrogram, $\hat{z}$ is the generated linear spectrogram, y and z are learning goals. y is real mel spectrogram, and z is real linear spectrogram. L_{mse} is the mean square error function.

Adversarial Loss: We take the loss of the ASR task as our adversarial loss. In the model training stage, the parameters of the ASR model M_{asr} are kept fixed, and the negative gradients are used for adversarial training. So we define the adversarial loss as:

$$\mathrm{L}_{adv} = L_{asr} = L_{CE}(x, \hat{x}) \tag{4}$$

x is the learning target, $\hat{x}$ is the predicted character sequence, and L_{CE} the cross-entropy loss function.

As shown in Algorithm 1. During the training stage, we need to continuously update the model M_{tts} to minimize reconstruction loss L_{recon} and maximize adversarial loss L_{adv}. So we define the total loss as:

$$\mathrm{L}_{total} = L_{recon} - \frac{1}{a + L_{adv}} * L_{adv} \tag{5}$$

We set a=20 according to the experiment results. The reason why we define total loss as formula (5) is to prevent the model from collapsing, and we will further analyze the total loss according to the experiment results in Section 3.2.1.

Algorithm 1 Pseudocode for the proposed model training

Input: Paired text and speech $(x^i, y^i, z^i)_{i=1}^N$, where x^i denotes character sequence, y^i denotes mel spectrogram, and z^i denotes linear spectrogram.
Pre-trained ASR model M_{asr} with $(x^i, y^i, z^i)_{i=1}^N$
Pre-trained TTS model M_{tts} with $(x^i, y^i, z^i)_{i=1}^N$
Output: Synthesized speech with desired speaking style.
1: Fixed parameters of the pre-trained ASR model.
2: **for** $t = 0, ..., num\_iter$ **do**
3: $x^i, y^i \leftarrow$ training data.
4: $L_{total} = L_{recon} - \frac{1}{a+L_{adv}} * L_{adv}$.
5: Update M_{tts} to maximize L_{adv}.
6: Update M_{tts} to minimize L_{recon}.
7: **end for**

3. EXPERIMENTS

3.1. Experimental setup

In the experiments, we trained the ASR model using the VCTK corpus [19], and trained TTS model using a subset of the Blizzard Challenge 2013 (Blizzard2013) corpus [20]. The subset contains 29679 text-audio pairs in 5 kinds of speaking styles, and 29479 pairs are used for training and 200 pairs for testing. We used a WaveRNN [21] vocoder to generate waveforms from the predicted mel-spectrograms. We trained the vocoder with the ground-truth mel-spectrograms from the Blizzard2013. For the pre-trained LAS model, the learning rate is 0.01 and 100 epochs are trained. For the pre-trained TTS model, the learning rate is 0.001 and 200 epochs are trained. During joint training, the pre-trained LAS model is added to the pre-trained TTS model, and the TTS model continues to be trained for 10 epochs. We recommend that readers listen to the examples on our demo page. Please visit `https://zklyu.github.io/demo/`.

3.2. Results and Evaluations

Comparative study: We compared our proposed model with the baseline models using subjective listening tests and objective metrics. We implemented two baseline systems and the proposed model, as summarized next.

• Modified Tacotron model: It is the same as our pre-trained TTS model.

• ASR_guide [15]: In the training process, the model makes the generated speech recognized well by the ASR, and the ASR task is used to continuously guide the TTS training to make the synthesized speech clearer.

• Our proposed method: We combine the pre-trained ASR model with the pre-trained TTS model. And we perform ASR adversarial training to make the reference audio not well recognized by the ASR, aiming to eliminate the content information in the style information. We also propose an adaptive adversarial weight learning strategy to prevent the model from collapsing.

3.2.1. Objective Evaluation

To verify that our model can effectively alleviate the entanglement between style and content information, we used the word information lost (WIL) and WER [22] as objective evaluation. If the value is smaller, it means that the synthesized

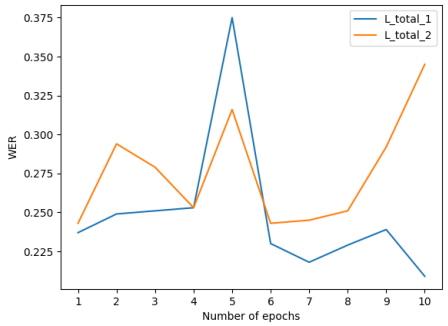

Fig. 2. As the number of epochs increases, different L_{total} correspond WER.

Table 1. WER and WIL for each model synthesized speech.

Model	Modified Tacotron	ASR_guide	Our proposed
WER	21.6%	25.5%	20.9%
WIL	33.9%	39.1%	32.1%

speech has fewer wrong words, missing words, and fuzzy words. We randomly chose 60 samples from test data and synthesized 60 audios for each model. We called the Sphinx API to identify the synthesized speech of different models. As we can see in Table 1, by comparison, our proposed method's WER and WIL values are smaller. This result proves that our proposed method is better than baseline models.

Analysis of the total loss: We selected the combination weight of reconstruction loss and adversarial loss according to WER in the experiments, beacuse that we expect that the synthesized speech has fewer missing words and fuzzy words. The experimental results of two different combination weight are shown in Fig. 2. L_total_1 is introduced in formula (5) in Section 2.3, and $L\_total\_2 = L_{recon} - L_{adv}$. We found that when the total loss is L_total_2, there is a high WER, and the model collapses at the end. At the same time, we found that when the L_{adv} has a smaller weight, there is a smaller WER, and the quality of the synthesized speech is better. Therefore, we use an adaptive adversarial weight learning strategy to prevent the model from collapsing.

3.2.2. Subjective Evaluation

We conducted two subjective listening tests, a Mean Opinion Score (MOS) test and an ABX preference test. In the MOS evaluation, we randomly selected fifteen samples for each model from above 60 synthesized audios. We invited twelve listeners using earphones to listen and evaluate the naturalness of each sample on a five-point scale in intervals of 0.5. The higher the MOS score, the better speech quality of the given samples. The results are shown in Table 2. Our proposed method has a higher naturalness than baselines, which shows that we have eliminated some content information by adding the ASR task and improved audio quality.

Table 2. Speech naturalness for expressive TTS. (95% confidence interval)

Model	MOS
Ground Truth	4.56± 0.206
Modified Tacotron Model	3.59± 0.299
ASR_guide	3.87± 0.294
Our proposed	3.92± 0.308

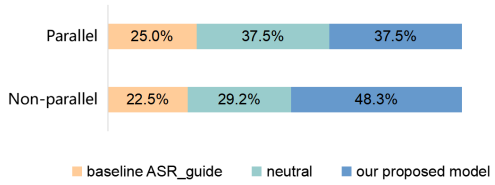

Fig. 3. ABX results for parallel and non-parallel style transfer.

ABX preference test assesses the style similarity between the reference and the synthesized audios. We randomly selected ten pairs of paired text and reference audios for parallel style transfer from the test dataset. For non-parallel style transfer, we chose ten pairs of unpaired text and reference audios. We invited twelve listeners and made them to evaluate which synthesized audio's speaking style is more similar to the reference audio's style. Regarding the results of ABX in Fig. 3. We can see that our proposed model is better than the baseline models in terms of parallel and non-parallel style transfer, which further proves that our proposed model not only can effectively alleviate the entanglement between style and content information, but also enhance style transfer performance of the model.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a novel method to effectively alleviate the entanglement between style and content information, by integrating TTS model and ASR model with a share layer network for joint training, and performing ASR adversarial training to prevent the style encoder from encoding content information. And we proposed an adaptive adversarial weight learning strategy to prevent the model from collapsing. The experimental results proved the validity of our proposed model. In the future, we will continue to consider other features disentangle methods, especially for fine-grained style modeling.

5. ACKNOWLEDGEMENTS

Thanks to the National Key R&D Program of China (No. 2020YFC2004103), National Natural Science Foundation of China (No. 61876131, U1936102).

6. REFERENCES

[1] Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, et al., "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, 2017, pp. 4006–4010.

[2] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, et al., "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[3] Wei Ping, Kainan Peng, Andrew Gibiansky, Sercan O Arik, et al., "Deep voice 3: Scaling text-to-speech with convolutional sequence learning," in *6th International Conference on Learning Representations*. ICLR, 2018.

[4] Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, et al., "Fastspeech 2: Fast and high-quality end-to-end text to speech," in *arXiv preprint*. arXiv:2006.04558, 2020.

[5] Cheng Gong, Longbiao Wang, Ju Zhang, Shaotong Guo, Yuguang Wang, et al., "TacoLPCNet: Fast and Stable TTS by Conditioning LPCNet on Mel Spectrogram Predictions," in *Proc. Interspeech 2021*, 2021, pp. 111–115.

[6] Yuxuan Wang, Daisy Stanton, Yu Zhang, RJ-Skerry Ryan, Eric Battenberg, et al., "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *International Conference on Machine Learning*. PMLR, 2018, pp. 5180–5189.

[7] Ya-Jie Zhang, Shifeng Pan, Lei He, and Zhen-Hua Ling, "Learning latent representations for style control and transfer in end-to-end speech synthesis," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6945–6949.

[8] Younggun Lee and Taesu Kim, "Robust and fine-grained prosody control of end-to-end speech synthesis," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5911–5915.

[9] Daisy Stanton, Yuxuan Wang, and RJ Skerry-Ryan, "Predicting expressive speaking style from text in end-to-end speech synthesis," in *Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 595–602.

[10] RJ Skerry-Ryan, Eric Battenberg, Ying Xiao, et al., "Towards end-to-end prosody transfer for expressive speech synthesis with tacotron," in *international conference on machine learning*. PMLR, 2018, pp. 4693–4702.

[11] Guangzhi Sun, Yu Zhang, Ron J Weiss, Yuan Cao, Heiga Zen, et al., "Fully-hierarchical fine-grained prosody modeling for interpretable speech synthesis," in *international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2020, pp. 6264–6268.

[12] Guangzhi Sun, Yu Zhang, Ron J Weiss, Yuan Cao, Heiga Zen, et al., "Generating diverse and natural text-to-speech samples using a quantized fine-grained vae and autoregressive prosody prior," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6699–6703.

[13] Cheng Gong, Longbiao Wang, et al., "Improving naturalness and controllability of sequence-to-sequence speech synthesis by learning local prosody representations," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5724–5728.

[14] Ting-Yao Hu, Ashish Shrivastava, Oncel Tuzel, et al., "Unsupervised style and content separation by minimizing mutual information for speech synthesis," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 3267–3271.

[15] Da-Rong Liu, Chi-Yu Yang, et al., "Improving unsupervised style transfer in end-to-end speech synthesis with end-to-end speech recognition," in *Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 640–647.

[16] Shuang Ma, Daniel Mcduff, and Yale Song, "Neural tts stylization with adversarial and collaborative games," in *International Conference on Learning Representations*. ICLR, 2018.

[17] Xudong Dai, Cheng Gong, Longbiao Wang, et al., "Information Sieve: Content Leakage Reduction in End-to-End Prosody Transfer for Expressive Speech Synthesis," in *Proc. Interspeech 2021*, 2021, pp. 131–135.

[18] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.

[19] Junichi Yamagishi, Christophe Veaux, Kirsten MacDonald, et al., "Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit (version 0.92)," 2019.

[20] Aimilios Chalamandaris, Pirros Tsiakoulis, Sotiris Karabetsos, et al., "The ilsp/innoetics text-to-speech system for the blizzard challenge 2013," in *Blizzard Challenge Workshop*. Citeseer, 2013.

[21] Nal Kalchbrenner, Erich Elsen, Karen Simonyan, Seb Noury, et al., "Efficient neural audio synthesis," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2410–2419.

[22] Andrew Cameron Morris, Viktoria Maier, and Phil Green, "From wer and ril to mer and wil: improved evaluation measures for connected speech recognition," in *Eighth International Conference on Spoken Language Processing*, 2004.