Nana-HDR: A Non-attentive Non-autoregressive Hybrid Model for TTS

Shilun Lin, Wenchao Su, Li Meng, Fenglong Xie, Xinhui Li, Li Lu

Tencent, Beijing, China

{cirolin, shirosu, leemeng, fenglongxie, hiccupli, adolphlu}@tencent.com

Abstract

This paper presents Nana-HDR, a new **n**on-**a**ttentive **n**on-**a**utoregressive model with **h**ybrid Transformer-based **D**ense-fuse encoder and **R**NN-based decoder for TTS. It mainly consists of three parts: Firstly, a novel Dense-fuse encoder with dense connections between basic Transformer blocks for coarse feature fusion and a multi-head attention layer for fine feature fusion. Secondly, a single-layer non-autoregressive RNN-based decoder. Thirdly, a duration predictor instead of an attention model that connects the above hybrid encoder and decoder. Experiments indicate that Nana-HDR gives full play to the advantages of each component, such as strong text encoding ability of Transformer-based encoder, stateful decoding without being bothered by exposure bias and local information preference, and stable alignment provided by duration predictor. Due to these advantages, Nana-HDR achieves competitive performance in naturalness and robustness on two Mandarin corpora and shows potential on a small Spanish corpus of Blizzard Challenge 2021.
Index Terms: Speech synthesis, sequence-to-sequence model, Transformer, Recurrent Neural Network

1. Introduction

With advances in deep learning, the text-to-speech acoustic model that converts text into acoustic features gradually transits from Hidden Markov Models (HMMs) to Deep Neural Network (DNN) based ones. Sequence-to-sequence neural network with attention mechanism [1] is one of the most popular methods. The traditional synthesis process is simplified by merging the generation of linguistic feature and acoustic feature into a single network. Works such as Char2Wav [2], Deep Voice 3 [3] and Tacotron [4], make significant progress in generating highly natural speech close to human quality.

As an attention-based autoregressive model, Tacotron is able to generate human-like speech for in-domain text. However, it cannot handle out-domain situations robustly. For instance, the length of testing text is quite different from that of training text and the field of testing text is not included in the training set. The causes of the robustness issue can be roughly classified as follows: Firstly, there are no explicit restrictions on the soft attention mechanisms such as content-based attention [5] and location sensitive attention [6] to prevent skip, repetition and mispronunciation. Secondly, the model predicts a stop flag to judge whether the synthesis process is completed. Therefore, a wrong prediction can lead to serious failures, such as early cut-off and late stop. Finally, teacher forcing training induces a mismatch between training and inference, usually known as exposure bias [7]. And the local information preference on autoregressive decoder may weaken the dependence between predicted acoustic features and text conditions, which makes the model tend to produce bad cases [8]. Local information preference still exists due to autoregressive property, although teacher forcing is not applied [9].

Many efforts are made to solve the above problems. Methods in [10, 11, 12] improve the attention mechanism by introducing location information and monotonic constraints. These approaches are proved to be effective in improving the speed of convergence, the stability of feature generation and the robustness of long sentence synthesis. However, the improved attention mechanisms can not fundamentally solve the problem of attention failure. Fastspeech [13] relies on the duration predictor instead of attention model, which eliminates the robustness problems caused by attention failure and stop frame misprediction. As a feed-forward non-autoregressive model, Fastspeech can instantly convert text into acoustic features. However, there is still a gap between its synthetic quality and that of autoregressive models. For exposure bias and local information preference issues, the effect of exposure bias on the autoregressive decoder is reduced by adversarial training [14]. Inspired by InfoGAN [15], a recognizer is introduced to maximize the mutual information between predicted feature and text condition which reduces the impact of local information preference [8].

Tacotron and Fastspeech mentioned above have single type of encoder and decoder. Experiments demonstrate that, compared to the architecture with a single type of encoder and decoder, the hybrid architecture with Transformer-based encoder and RNN-based decoder achieves the best results on the English-to-French machine translation task [16]. The results confirm their intuition that Transformer-based encoder is good at text feature extraction and the stateful RNN-based decoder is beneficial for conditional generation.

Recently, the work in the field of natural language processing [17] further explores what BERT learn about the structure of language. The results indicate that intermediate layers of BERT encode a rich hierarchy of linguistic information, with surface features at the bottom, syntactic features in the middle and semantic features at the top. This conclusion is similar to the one obtained by visual analysis of Convolutional Neural Network (CNN), which shows that the features learned by CNN are also hierarchical [18]. The bottom layers learn the surface features like edges, corners and colors. Higher layers learn more semantic information such as faces. In the field of computer vision, the performance of various tasks can be improved by fusing features with different representation meanings. ResNet [19] and DenseNet [20] merge the features of different layers through bypass connections, which improves the performance of image classification while alleviating gradient vanishing. FCN [21], U-Net [22] and SegNet [23] fuse the features of different encoder layers in different ways, and then provide the fusion feature to the decoder, which effectively improve the accuracy of semantic segmentation. Because the features extracted by Transformer encoder and CNN have similar hierarchical characteristics, the feature fusion method for CNN may be transferred to Transformer encoder for the purpose of improving the performance of the TTS acoustic model.

As shown in Figure 1, a non-attentive non-autoregressive

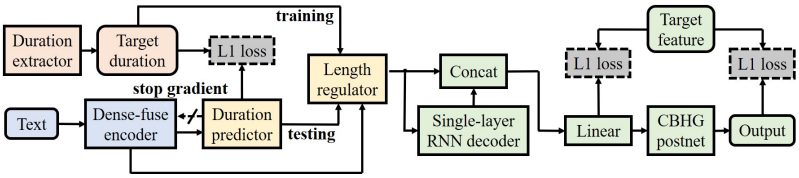

Figure 1: *Architecture of the proposed Nana-HDR.*

model with hybrid Transformer-based Dense-fuse encoder and RNN-based decoder (Nana-HDR) for TTS is proposed.

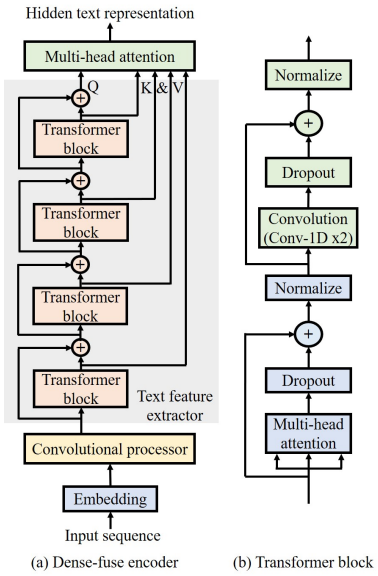

Figure 2: *Architecture of the proposed Dense-fuse encoder.*

2. Proposed Method

2.1. Dense-fuse encoder

The proposed Dense-fuse encoder is presented in Figure 2(a). Chinese Pinyin sequence with tone and prosody information or Spanish phoneme sequence is converted to the corresponding embedding sequence. Then, a convolutional processor with multiple Conv-1D layers are applied to preprocess the embedding sequence. For two reasons, sinusoidal positional embedding is not used. Firstly, the maximum length of the input sequence needs to be preset for calculating of the sinusoidal encoding table, which makes the model unable to handle arbitrarily long sentences. Secondly, using convolutional layers as input processor is able to capture implicit relative position information which has been proved to improve the performance of automatic speech recognition [24]. Next, the output of the processor with convolutional context is fed to the text feature extractor stacked by basic Transformer blocks. As shown in Figure 2(b), each Transformer block is composed of multi-head attention and convolution sub-modules. All sub-modules follow a strict computation flow: process, dropout, residual-add and layer normalize. Transformer blocks are densely connected through bypass connections. From the perspective of back propagation, the supervision signal from the top layer can be better transmitted back to the lower layer, which plays a similar role of deep supervision and makes the model easier to train. From the perspective of feature fusion, dense connections make the features of lower layer can be reused by higher layer. Since the features extracted by different layers of Transformer encoder have different representation meanings [17], intuitively such feature reuse can enhance the representation ability of the final encoding as it has been proven in DenseNet. Different from DenseNet which uses channel-wise concatenation, element-wise addition is applied for feature fusion which helps to maintain the computational complexity by keeping encoding dimension between blocks unchanged. In this way, features of different layers are treated indiscriminately which can be regarded as a coarse fusion. It is relatively rough to fuse features learned by transformer blocks through dense bypass-connections, because the model cannot control the proportion of these features.

In [25], attention mechanism is applied to output a set of combination weights over the style tokens. Inspired by this method, multi-head attention is introduced for fine fusion. As shown in Figure 2(a), the coarse fusion feature is used as the query (Q), and the output of each Transformer block is used as the key (K) and value (V) of an attention head. The final hidden text representation is obtained by combining outputs of different blocks with the weights learned by multi-head attention layer. Fine fusion can be regarded as a further adjustment of the coarse fusion feature by the model. The model can decide how much additional information the coarse fusion feature needs to obtain from each Transformer block by fine fusion.

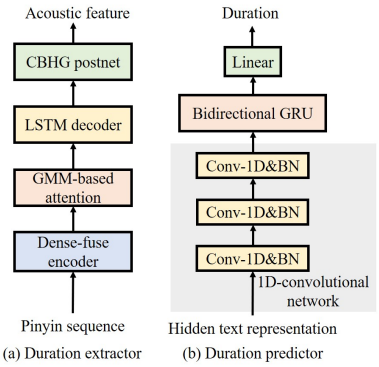

Figure 3: *Architecture of the duration extractor and predictor.*

2.2. Duration extractor and predictor

Before training Nana-HDR, an attention-based duration extractor is trained to align the training text and training audio, and then output the duration of the training text tokens (target duration). Duration of individual text token can be obtained by ASR force alignment. However, training an attention-based model and extracting target duration through ground truth aligned (GTA) mode is a relatively efficient and accurate way. The duration extractor (Figure 3(a)) is a sequence-to-sequence model composed of the above-mentioned Dense-fuse encoder, GMM-based attention model, a two-layers LSTM decoder and

a CBHG postnet. As shown in Figure 1, the extracted target duration is used to upsample the encoder outputs to match the length of target acoustic features during training. It is also used as label for the learning of the duration predictor.

The duration predictor is trained jointly as a part of Nana-HDR. The duration predictor (Figure 3(b)) processes the hidden text representation (output of Nana-HDR's Dense-fuse encoder) through a 1D-convolutional network to capture local position-related information. Considering that the duration is related to the global context information, a bidirectional RNN layer is added to the convolution network, and its output is linearly mapped to a scalar. During training, the gradient of duration predictor is prevented from back-propagating to the Dense-fuse encoder for the purpose of avoiding affecting its learning. During inference, duration predictor combined with length regulator are used to solve the problem of length mismatch between hidden text representation and acoustic feature to be generated.

2.3. Non-autoregressive single-layer RNN decoder

Decoder which keeps tracking the state information is beneficial for conditional generation [16]. As a stateful decoder, autoregressive RNN-based decoder is widely used in text-to-feature model. However, it suffers from exposure bias, local information preference and slow inference.

Scheduled sampling deals with the exposure bias by adjusting the drop teacher forcing frame rate empirically. For local information preference, it is mainly caused by the powerful RNN decoder with autoregressive dependency. Let z and x be conditional variable obtained from text and corresponding acoustic features to be decoded, the autoregressive conditional decoding process can be described as $p(x|z) = \prod_i^N p(x_i|z, x_{<i})$, where N is the number of acoustic frames in x. Since RNNs are universal function approximators and any joint distribution over x admits an autoregressive factorization, the RNN autoregressive decoding distribution can in theory represent any probability distribution. The information that can be modeled locally by decoding distribution $p(x|z)$ will be encoded locally without using information from z and only the remainder will be modeled using them [9]. Experiments in [26] show that weakening the autoregressive part of the model by dropout can encourage the conditional variables to be used. Using larger reduction factor (e.g. r=5) in Tacotron works in a similar way, which can alleviate local information preference to a certain extent. However, this method will introduce a problem: how to select the context conditional frame when using the duration-based model.

[8] further formalizes conditional autoregressive attention-based model Tacotron as a variational encoder-decoder. The goal of training is to maximize the sum of conditional likelihood of text t and acoustic features x pairs in the training set. For each pair, the conditional likelihood can be written as $\log p_\theta(x|t) = \sum_{i=1}^N \log p_\theta(x_i|x_{<i}, t)$. The distribution of the time-aligned latent variables (context vectors generated by attention model) c can be factorized as $\log p_\theta(c|x, t) = \sum_{i=1}^N \log p_\theta(c_i|x_{<i}, t)$ [27]. Then the conditional likelihood of each training pair can be written as $\log p_\theta(x|t) = \log \int_c p_\theta(x, c|t)dc = \sum_{i=1}^N \log \int_{c_i} p_\theta(x_i, c_i|x_{<i}, t)$. However, the integral over c is intractable to compute. Therefore, an encoder approximation is introduced. For a training time step i, encoder distribution $q_\phi(c_i|x_{<i}, t)$ is used to approximate the posterior distribution $p_\theta(c_i|x_{\leq i}, t)$, where ϕ and θ is the parameters of the encoder (with attention model) and the autoregressive decoder. The KL-divergence of the encoder approximation from the posterior distribution can be written as Eq.1. In Eq.2, since the KL-divergence term is always non-negative, the conditional likelihood $\log p_\theta(x_i|x_{<i}, t)$ equals the variational lower bound $\mathcal{L}(\theta, \phi, x, t)$ only when the KL-divergence $D_{KL}(q_\phi(c_i|x_{<i}, t) \parallel p_\theta(c_i|x_{\leq i}, t))$ equals 0. This means that c_i and x_i are conditionally independent. As summarized in [8], when $x_{<i}$ contains enough information to predict x_i, the model tends to reduce the dependence between c_i and x_i to maximize $\log p_\theta(x_i|x_{<i}, t)$. At the same time, it also reduces the dependence between text and acoustic features. Modeling dependency between the text and the predicted acoustic features insufficiently may lead to higher bad-case rate. For the duration-based model, the dependence between time-aligned latent variables (generated by duration predictor with length regulator) and acoustic features is further reduced because there is no attention model to explicitly connect c_i and $x_{<i}$. Introducing an auxiliary CTC recognizer to maximize the mutual information between the text and acoustic features is an optional way [8]. To alleviate these problems fundamentally, a non-autoregressive single-layer RNN is used as a decoder in this work. Due to the absence of autoregressive feedback, only hidden text representations are fed to the decoder which avoids the influence of local information preference, but also also puts forward a higher requirement for their representation ability. We believe that Dense-fuse encoder with strong text encoding ability can meet the requirement. In order to allow the text information to fully participate in the decoding process, the output of decoder and the expanded text representation are concatenated for final acoustic feature generation.

$$
\begin{aligned}
& D_{KL}(q_\phi(c_i|x_{<i}, t) \parallel p_\theta(c_i|x_{\leq i}, t)) \\
&= \mathbb{E}_{q_\phi(c_i|x_{<i},t)}[\log q_\phi(c_i|x_{<i}, t) - \log p_\theta(c_i|x_{\leq i}, t)] \\
&= \mathbb{E}_{q_\phi(c_i|x_{<i},t)}[\log q_\phi(c_i|x_{<i}, t) - \log p_\theta(x_i, c_i|x_{<i}, t) \\
&\quad + \log p_\theta(x_i|x_{<i}, t)] \\
&= \mathbb{E}_{q_\phi(c_i|x_{<i},t)}[\log q_\phi(c_i|x_{<i}, t) - \log p_\theta(x_i, c_i|x_{<i}, t)] \\
&\quad + \log p_\theta(x_i|x_{<i}, t) \\
& \log p_\theta(x_i|x_{<i}, t)
\end{aligned}
\tag{1}
$$

$$
\begin{aligned}
&= D_{KL}(q_\phi(c_i|x_{<i}, t) \parallel p_\theta(c_i|x_{\leq i}, t)) \\
&\quad + \mathbb{E}_{q_\phi(c_i|x_{<i},t)}[\log p_\theta(x_i, c_i|x_{<i}, t) - \log q_\phi(c_i|x_{<i}, t)] \\
&\geq \mathbb{E}_{q_\phi(c_i|x_{<i},t)}[\log p_\theta(x_i, c_i|x_{<i}, t) - \log q_\phi(c_i|x_{<i}, t)] \\
&= \mathcal{L}(\theta, \phi, x, t)
\end{aligned}
\tag{2}
$$

3. Experiments

3.1. Datasets

Experiments were conducted on two Mandarin corpora and a small Spanish corpus. The first one was recorded by a professional Chinese female speaker in studio quality. The transcription used in the recording covered multiple fields, with an average sentence length of 70 characters. The number of utterances used for training was 9600. The second one consisted of 12000 audio files extracted from an online audio-book which recorded by an actor with rich rhythm. The transcription was the corresponding novel, with an average sentence length of 86 characters. The last one was a 5 hours European Spanish corpus provided by the Blizzard Challenge 2021 organizing committee (without using external data). Text transcription was converted to phoneme-based one by professionals (manually labelled) to deal with the problem of synthesizing Spanish text with a small number of English words. This phoneme-based transcription will be made public. All recordings were sampled at 16kHz

with 16-bit quantization. Consistent with LPCNet [28], 18 Bark cepstral coefficients and 2 pitch parameters were extracted as the prediction targets of Nana-HDR. 100 utterances that have not appeared in the training set were used for the in-domain naturalness testing. In order to verify whether the model can cope with out-domain scenarios, 200 popular words with an average length of 5 characters and 50 long paragraphs with an average length of 1000 characters were used for robustness testing. The latter was selected from WeChat official account, covering the fields of politics, sports and so on. For Blizzard Challenge 2021, test text was provided by the organizing committee.

3.2. Model configuration

Nana-HDR was a sequence-to-sequence acoustic model with a Dense-fuse encoder, a duration predictor (with a length regulator) and a single-layer RNN-based decoder. The main components of the Dense-fuse encoder were: (1) A convolutional processor with three 256-dimensional Conv-1D layers whose kernel sizes were set to 3. (2) A text feature extractor with four Transformer blocks whose attention head numbers and hidden sizes were set to 4 and 256 respectively. (3) A 256-dimensional 4-head attention layer for fine feature fusion. Duration predictor was mainly composed of three 256-dimensional Conv-1D layers with kernel size of 3 and a 64-dimensional bidirectional GRU layer. The decoder was a non-autoregressive 512-dimensional unidirectional GRU layer. The postnet was a CBHG module with the same structure as the one in Tacotron. Nana-HDR was trained using the Adam optimiser [29]. L1 loss was used for acoustic feature (before and after postnet) and duration loss. The model was trained for 300,000 steps, with a learning rate of 0.0001 and a batch size of 32. LPCNet as a relatively lightweight neural vocoder is applied in this work. Different from [28], in addition to being input to the first GRU layer, conditional feature was also fed to the second GRU layer.

3.3. Results

To evaluate the naturalness of the proposed model, we conducted Mean Opinion Score (MOS) and Comparison Mean Opinion Score (CMOS) tests on two Mandarin corpora. For MOS tests, native Chinese speakers were invited to listen and score 125 audio. 100 test utterances synthesized by the corresponding model were mixed with 25 original recordings. Scores ranged from 1 to 5 with intervals of 0.5. For CMOS tests, the same listeners were asked to listen to the paired test utterances synthesized by two different models in random order and evaluate how the latter feels comparing to the former using a score in [-3, 3] with intervals of 1 (from much worse to much better). All listening tests were conducted in a quiet room with headphones. We evaluated the robustness by measuring the failure rate and the word error rate (WER). The failure was mainly identified by whether the synthesized audio ended early, repeated the same clip or contained meaningless clip which seriously affected the understanding of the content. The WER was measured by an ASR system described in [30]. Relevant audio samples were available on the accompanying web page[1]. For Blizzard Challenge 2021, we reported the results of SH and SS tasks.

3.3.1. Naturalness and robustness

For two Mandarin corpora, our Nana-HDR was compared with Tacotron and Fastspeech. Both models had the single type of encoder and decoder. All models were trained with the same number of iterations (300,000) to ensure their performance.

[1] https://linshilun.github.io/nanahdrsamples/nanahdr.html

Table 1 and Table 2 contain MOS and CMOS results. It can be seen that Nana-HDR filled the naturalness gap between Fastspeech and Tacotron on both corpora. Listeners preferred the results synthesized by Nana-HDR to those synthesized by the other two systems. The results indicated that our Nana-HDR has achieved competitive performance in naturalness. In the aspect of robustness, because of the bad attention alignments, Tacotron with GMM-based attention had overall failure rates of 1.5% and 2.8% on two corpora. There was no serious synthesis failure in non-attentive models. General speech recognition was performed for synthesis samples without serious failure. WERs are recorded in Table 3. The results indicate that Nana-HDR achieved the lowest WER whether trained with a studio quality corpus or with a challenging audio-book corpus.

Table 1: *MOS with 95% confidence intervals.*

Model	female	male
Fastspeech	4.10 ± 0.05	4.08 ± 0.06
Tacotron-GMMA	4.20 ± 0.04	4.13 ± 0.06
Nana-HDR	$\mathbf{4.22 \pm 0.04}$	$\mathbf{4.23 \pm 0.05}$
Ground truth	4.41 ± 0.04	4.37 ± 0.04

Table 2: *CMOS comparison, the p-value<0.01.*

Model	female	male
Nana-HDR	**0.000**	**0.000**
Tacotron-GMMA	-0.245	-0.311
Fastspeech	-0.336	-0.363

Table 3: *Word error rate of the neural TTS models.*

Model	female	male
Tacotron-GMMA	2.8%	4.1%
Fastspeech	2.7%	3.4%
Nana-HDR	**2.0%**	**2.1%**

For Blizzard Challenge 2021, the results are shown in the figure 4, 5, 6 and 7. The identifier of natural speech is R, and our system is C. There was a gap between the Nana-HDR and some other systems (and natural speech). We concluded that there were three possible reasons related to it. The first was that the training text did not contain prosody related information. The second was that a small corpus was used for training, so that the model might not be fully trained. The last was that the sampling rate of audio synthesized by our system is relatively low (System A, F, K, L and R is 48KHz. System D, E, G and J is 22KHz. System B, C and N is 16KHz).

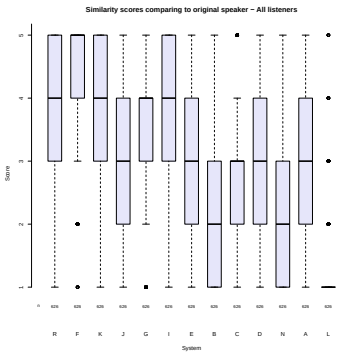

Figure 4: *Boxplot of similarity scores of submitted systems (SH).*

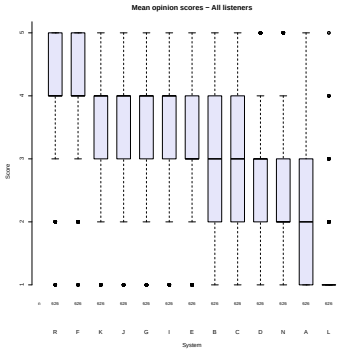

Figure 5: *Boxplot of MOS of submitted systems (SH).*

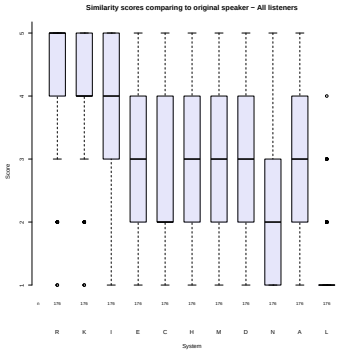

Figure 6: *Boxplot of similarity scores of submitted systems (SS).*

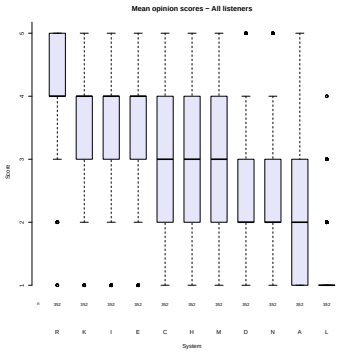

Figure 7: *Boxplot of MOS of submitted systems (SS).*

3.3.2. Ablation studies

We attributed three possible reasons why Nana-HDR can achieve good naturalness and robustness. First, the hybrid structure without attention model; secondly, the strong text feature extraction ability of the Dense-fuse encoder; thirdly, stable conditional generation ability of non-autoregressive stateful RNN decoder. The previous experimental results demonstrated the advantages of non-attentive structure with hybrid Transformer-based encoder and RNN-based decoder. Next, we conducted ablation studies to verify the effectiveness of Dense-fuse encoder and non-autoregressive RNN-based decoder in Nana-HDR. The results are shown in the table 4, 5 and 6.

First, it was found that the duration extractor often failed to align when feature fusions were removed from the Dense-fuse encoder. And the absence of fine feature fusion led to inaccurate duration extraction of long pause. Therefore, we used the duration extractor with the original Dense-fuse encoder in subsequent experiments. Removing feature fusions from Nana-HDR (Nana-HDR-Nofusion) resulted in -0.153 and -0.258 CMOS. The WERs on both corpora were also higher. Then, we replaced the Dense-fuse encoder with the commonly used RNN-based CBHG encoder (Nana-HDR-CBHGE). This reduced the MOS on two corpora by 1.7% and 5.2% respectively. In terms of CMOS, listeners preferred the results synthesized by the system without replacement. Using CBHG encoder also resulted in higher WERs on both corpora. We argued that the output of the Dense-fuse encoder obtained better representation ability which made Nana-HDR perform better. On the one hand, rich linguistic information helped to improve the naturalness. On the other hand, it was more suitable for the text encoding with stronger representation capability to be the only input of non-autoregressive RNN-based decoder.

Then we replaced the non-autoregressive RNN-based decoder with an autoregressive one (Nana-HDR-ARD) which led to significant degradation in naturalness, robustness and speed (reduced by 8.85x). On the challenging audio-book corpus, we found that there were obvious word skipping and mispronunciation in some in-domain sentences, which has not been seen in other experiments. We held that autoregressive decoder without frame reduction was more sensitive to the exposure bias and the local information preference, which increased the bad case rate.

Ablation studies demonstrated that Dense-fuse encoder and non-autoregressive RNN-based decoder were effective components of Nana-HDR.

Table 4: *MOS (ablation studies) with 95% confidence intervals.*

Model	female	male
Nana-HDR-Nofusion	4.20 ± 0.04	4.18 ± 0.05
Nana-HDR-CBHGE	4.15 ± 0.06	4.01 ± 0.06
Nana-HDR-ARD	4.11 ± 0.08	3.73 ± 0.07

Table 5: *CMOS comparison, the p-value<0.01.*

Model	female	male
Nana-HDR-Nofusion	-0.153	-0.258
Nana-HDR-CBHGE	-0.265	-0.471
Nana-HDR-ARD	-0.349	-0.694

Table 6: *Word error rate of models (ablation studies).*

Model	female	male
Nana-HDR-Nofusion	2.2%	2.4%
Nana-HDR-CBHGE	2.5%	3.3%
Nana-HDR-ARD	2.5%	4.4%

4. Conclusions

In this paper, we proposed Nana-HDR, a non-attentive non-autoregressive hybrid model for TTS. By fully exploiting the advantages of each component, Nana-HDR achieves competitive performance on two Mandarin corpora in both naturalness and robustness compared with Tacotron and Fastspeech. It also shows potential on a small Spanish corpus of Blizzard Challenge 2021. If the text can be annotated more richly (such as adding pause) and the training audio can be preprocessed pertinently, Nana-HDR may also achieve good performance.

5. References

[1] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. NIPS*, 2014, pp. 3104–3112.

[2] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, and Y. Bengio, "Char2wav: End-to-end speech synthesis," in *Proc. ICLR workshop*, 2017.

[3] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: Scaling text-to-speech with convolutional sequence learning," in *Proc. ICLR*, 2018.

[4] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, 2017, pp. 4006–4010.

[5] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. ICLR*, 2015.

[6] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *Proc. ICASSP*, 2018, pp. 4779–4783.

[7] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba, "Sequence level training with recurrent neural networks," in *Proc. ICLR*, 2016.

[8] P. Liu, X. Wu, S. Kang, G. Li, D. Su, and D. Yu, "Maximizing mutual information for tacotron," *arXiv preprint arXiv:1909.01145*, 2019.

[9] X. Chen, D. P. Kingma, T. Salimans, Y. Duan, P. Dhariwal, J. Schulman, I. Sutskever, and P. Abbeel, "Variational lossy autoencoder," in *Proc. ICLR*, 2017.

[10] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[11] J.-X. Zhang, Z.-H. Ling, and L.-R. Dai, "Forward attention in sequence-to-sequence acoustic modeling for speech synthesis," in *Proc. ICASSP*, 2018, pp. 4789–4793.

[12] E. Battenberg, R. Skerry-Ryan, S. Mariooryad, D. Stanton, D. Kao, M. Shannon, and T. Bagby, "Location-relative attention mechanisms for robust long-form speech synthesis," in *Proc. ICASSP*, 2020, pp. 6194–6198.

[13] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech: Fast, robust and controllable text to speech," in *Proc. NIPS*, 2019.

[14] H. Guo, F. K. Soong, L. He, and L. Xie, "A new gan-based end-to-end tts training algorithm," in *Proc. Interspeech*, 2019, pp. 1288–1292.

[15] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: interpretable representation learning by information maximizing generative adversarial nets," in *Proc. NIPS*, 2016, pp. 2180–2188.

[16] M. X. Chen, O. Firat, A. Bapna, M. Johnson, W. Macherey, G. Foster, L. Jones, M. Schuster, N. Shazeer, N. Parmar *et al.*, "The best of both worlds: Combining recent advances in neural machine translation," in *Proc. ACL*, 2018, pp. 76–86.

[17] G. Jawahar, B. Sagot, and D. Seddah, "What does bert learn about the structure of language?" in *Proc. ACL*, 2019, pp. 3651–3657.

[18] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. ECCV*, 2014, pp. 818–833.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. CVPR*, 2017, pp. 4700–4708.

[21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. CVPR*, 2015, pp. 3431–3440.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, 2015, pp. 234–241.

[23] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[24] A. Mohamed, D. Okhonko, and L. Zettlemoyer, "Transformers with convolutional context for asr," *arXiv preprint arXiv:1904.11660*, 2019.

[25] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *Proc. ICML*, 2018, pp. 5180–5189.

[26] S. Bowman, L. Vilnis, O. Vinyals, A. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," in *Proc. CoNLL*, 2016, pp. 10–21.

[27] S. Shankar and S. Sarawagi, "Posterior attention models for sequence to sequence learning," in *Proc. ICLR*, 2018.

[28] J.-M. Valin and J. Skoglund, "Lpcnet: Improving neural speech synthesis through linear prediction," in *Proc. ICASSP*, 2019, pp. 5891–5895.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.

[30] Y. Wang, A. Mohamed, D. Le, C. Liu, A. Xiao, J. Mahadeokar, H. Huang, A. Tjandra, X. Zhang, F. Zhang *et al.*, "Transformer-based acoustic modeling for hybrid speech recognition," in *Proc. ICASSP*, 2020, pp. 6874–6878.