


## Article

# HierTTS: Expressive End-to-End Text-to-Waveform Using a Multi-Scale Hierarchical Variational Auto-Encoder

Zengqiang Shang <sup>1,2</sup> , Peiyang Shi <sup>1,2</sup>, Pengyuan Zhang <sup>1,2,\*</sup>, Li Wang <sup>1</sup> and Guangying Zhao <sup>1</sup>

<sup>1</sup> Key Laboratory of Speech Acoustics Content Understanding, Institute of Acoustics, Chinese Academy of Sciences, Beijing 100190, China

<sup>2</sup> University of Chinese Academy of Sciences, Beijing 100049, China

\* Correspondence: zhangpengyuan@hccl.ioa.ac.cn

**Abstract:** End-to-end text-to-speech (TTS) models that directly generate waveforms from text are gaining popularity. However, existing end-to-end models are still not natural enough in their prosodic expressiveness. Additionally, previous studies on improving the expressiveness of TTS have mainly focused on acoustic models. There is a lack of research on enhancing expressiveness in an end-to-end framework. Therefore, we propose HierTTS, a highly expressive end-to-end text-to-waveform generation model. It deeply couples the hierarchical properties of speech with hierarchical variational auto-encoders and models multi-scale latent variables, at the frame, phone, subword, word, and sentence levels. The hierarchical encoder encodes the speech signal from fine-grained features into coarse-grained latent variables. In contrast, the hierarchical decoder generates fine-grained features conditioned on the coarse-grained latent variables. We propose a staged KL-weighted annealing strategy to prevent hierarchical posterior collapse. Furthermore, we employ a hierarchical text encoder to extract linguistic information at different levels and act on both the encoder and the decoder. Experiments show that our model performs closer to natural speech in prosody expressiveness and has better generative diversity.

**Keywords:** text-to-speech; hierarchical VAE; expressive TTS; multi-scale; end-to-end



**Citation:** Shang, Z.; Shi, P.; Zhang, P.; Wang, L.; Zhao, G. HierTTS: Expressive End-to-End Text-to-Waveform Using a Multi-Scale Hierarchical Variational Auto-Encoder. *Appl. Sci.* **2023**, *13*, 868. <https://doi.org/10.3390/app13020868>

Academic Editors: Kai Yu, Yan Song and Ya Li

Received: 29 November 2022

Revised: 1 January 2023

Accepted: 5 January 2023

Published: 8 January 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Text-to-Speech (TTS) is a technique that takes text as an input and produces audible speech as an output. With the rapid development of machine learning over the years, the TTS technology has made remarkable progress in its goal of generating natural speech that is close to human speaking. However, there is still a huge gap in the diversity of the expressiveness of generated speech compared with natural speech. Therefore, improving the expressiveness of synthesized speech and enriching the diversity of expressiveness has become a hot topic.

Speech is a hierarchical system [1] that comprises phonemes, syllables, words, phrases, and sentences. Phonemes and syllables are defined by phonetic features; words, phrases and sentences are mainly defined by semantic knowledge. Due to the hierarchical properties of speech, many studies aim to use multi-scale information to enrich the expressiveness of speech synthesis. Sentence-level style vectors are first adopted in CHiVE [2], where linguistic information of the different scales is introduced to both encoding and decoding networks of a variational autoencoder (VAE). However, single scale style representation is not expressive enough. Researchers focus on using multi-scale style vectors later. Google [3] introduced phoneme and word-level latent variables to Tacotron2 to enhance generation diversity by sampling. HiMuV-TTS [4] investigated phoneme and sentence-level variables on FastSpeech. Moreover, the extracted style vectors are semantically related to the text. Randomly sampling from latent variables leads to unnatural prosody. Therefore, MsE-moTTS [5] was proposed to predict syllable and sentence-level style vectors from text for inference. Lei et al. [6] subsequently used hierarchical textual information to predict

multi-scale prosody information. However, there still remain some problems in the existing research on multi-scale style modeling. First, the style vectors of different scales are usually extracted in parallel. The dependence of different scale style vectors is not considered. Second, the style extraction and style prediction modules are not jointly optimized, which leads to a mismatch between training and inference. Finally, previous work on style has modeling mainly focused on acoustic models, resulting in poor sound quality. The frame, as the basic unit of speech acoustic features, is also an important scale level in the speech generation process because it connects phonetic information to the speech waveform signal generation process. New models, VITS and NaturalSpeech [7,8], utilize frame-level variables to implement end-to-end training of acoustic models and vocoder, which significantly improves synthetic sound quality

We believe that, for a specific speech segment, style vectors of different scales all describe the same style pattern, and the main difference is that coarser-grained vectors depict changes over a longer period of time, and finer-grained vectors enrich more detail. Therefore, in the process of style extraction [1], fine-grained style features should be extracted first, and then coarse-grained information is abstracted on top of the fine-grained information. The process of predicting is just the opposite [9]. First comes generating coarse-grained information to set the overall tone. Then, more fine-grained style information is predicted based on the coarse-grained style. In this way, styles of different scales can be coordinated and organized.

We can view linguistic, phonological, and acoustic features as the representations of speech at different scales. Therefore, we propose an expressive end-to-end text-to-waveform generation model called HierTTS. It considers the latent variables of frames, phonemes, syllables, words, and sentences and achieves highly expressive speech generation by mining multi-scale information from both modalities, text and speech. In addition, we employ a Gaussian distribution to model the latent variables at each scale. Gaussian sampling helps to enrich generation diversity. To prevent the posterior collapse of hierarchical VAE, we employed a staged Kullback–Leibler (KL) weighted annealing strategy.

Our contributions can be summarized as follows:

- A highly expressive TTS model which deeply couples the hierarchical properties of speech with hierarchical VAE.
- A staged KL-weighted annealing strategy that helps eliminating posterior collapse in hierarchical VAE.

## 2. Background

### *Hierarchical Variational Auto-Encoder*

The variational autoencoder is a neural network generative model [10]. Given a dataset  $X = [x_1, x_2, \dots, x_N]$ , VAE defines a prior distribution  $p(z)$  and models the joint distribution  $p(x, z)$  with  $p(z)p(x|z)$ , where  $p(x|z)$  is the decoder to generate data point  $x$  from latent variable  $z$ . Due to true posterior  $p(z|x)$  being usually intractable, VAE utilizes an encoder  $q(z|x)$  to approximate the posterior distribution. Generally, the prior distribution  $p(z)$  usually adopts the standard Gaussian distribution with independent components. VAE is optimized by minimizing reconstruction loss and KL divergence between prior distribution  $p(z)$  and the estimated posterior  $q(z|x)$  using resampling tricks.

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z|x)} \log p_\theta(x|z) - KL[q_\phi(z|x)||p(z)] \quad (1)$$

However, the assumption of a Gaussian distribution makes it difficult to model a more complex distribution. If the latent variable needs to model a more complex distribution, an ordinary VAE will lead to oversmoothing. Hierarchical variational autoencoders (HVAEs) [11,12] improve the complexity of both prior and posterior distributions by modeling hierarchically-dependent latent variables. A hierarchical VAE can be viewed as a series of VAEs stacked together. Apart from observed variables  $x$ , it contains hierarchical

latent variables  $\{z_1, z_2, \dots, z_L\}$ , where  $L$  represents the number of hierarchies. According to the chain rule, the joint distribution can be decomposed into:

$$p(x, z_1, \dots, z_L) = p(x | z_{\geq 1}) \prod_{\ell=1}^L p(z_\ell | z_{<\ell}) \tag{2}$$

The prior and estimated posterior distributions are  $p_\theta(z) = \prod_{\ell} p_\theta(z_\ell | z_{<\ell})$  and  $q_\phi(z | x) = \prod_{\ell} q_\phi(z_\ell | z_{<\ell}, x)$ , respectively. The conditional distribution of each level of prior and estimated posterior adopts the standard Gaussian distribution with independent components. Hierarchical VAEs extract the posterior distribution from the data based on a bottom-up path and follow a top-down path to generate the prior distribution and reconstruct samples. This architecture helps the model learn the hierarchy between latent variables efficiently. The evidence lower bound (ELBO) of HVAE are

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z|x)} \log p_\theta(x | z) - \sum_{\ell=1}^L \mathbb{E}_{q_\phi(z_{<\ell}|x)} [KL[q_\phi(z_\ell | x, z_{<\ell}) | p_\theta(z_\ell | z_{<\ell})]] \tag{3}$$

where  $q_\phi(z_\ell | x, z_{<\ell})$  and  $p_\theta(z_\ell | z_{<\ell})$  are the approximate posterior and prediction priors at the  $\ell$ th layer, respectively.

HVAE was first applied on the high-resolution image task, and then BVAE-TTS [13], and VARA-TTS [14] applies it to the acoustic model of speech synthesis. However, they only modeled the generation process from the phoneme to the Mel spectrogram. They did not consider the hierarchical characteristic of speech in the network structure’s design. In addition, due to the posterior collapse, some hidden variables in the hierarchical structure are not activated. The Mel spectrogram generated by BVAE-TTS is very blurred. Unlike these two approaches, we built hierarchical networks that directly generate waveforms. Furthermore, we closely combine the hierarchical properties of speech with hierarchical VAE to further improve prosodic expressiveness.

### 3. Method

#### 3.1. Overview

Considering the hierarchical properties of speech, we propose HierTTS, which deeply couples the hierarchical properties of speech into VAEs. The overall architecture is shown in Figure 1, which contains a hierarchical audioencoder (HAE), a hierarchical context encoder (HCE), and a hierarchical audiodecoder (HAD). HierTTS introduces five latent variables at different temporal resolutions—the sentence, word, subword, phoneme, and frame levels. First, HAE extracts the hierarchical latent variables from the linear spectrogram in a fine-to-coarse manner, and then the HAD reconstructs the speech waveform from coarse-to-fine leveraging-extracted hierarchical latent variables. To introduce text information, HCE obtains linguistic and phonological information at different scales from phoneme and character sequences and then injects it into each hierarchy of HAE and HAD. For the modeling of duration, we inject phoneme-scale durations at the phoneme-level encoder and reconstruct the durations using the phoneme decoder. Thus, the duration and waveform reconstruction share some of the hierarchical hidden variables, which facilitates learning more consistent prosody. The training goal of HierTTS is to maximize the lower bound of evidence (ELBO) for the marginal log-likelihood  $\log p_\theta(x, \mathcal{D} | c)$ .

$$\begin{aligned} ELBO = & \mathbb{E}_{q_\phi(z|x)} \log p_\theta(x, \mathcal{D} | z, c) - \mathbb{E}_{q_\phi(z_{<5}|x, \mathcal{D}, c)} [KL[q_\phi(z_5 | x, z_{<5}) | p_\theta(z_5 | z_{<5})]] \\ & - \sum_{\ell=1}^4 \mathbb{E}_{q_\phi(z_{<\ell}|x, \mathcal{D}, c)} [KL[q_\phi(z_\ell | x, \mathcal{D}, z_{<\ell}, c_\ell) | p_\theta(z_\ell | z_{<\ell}, c_\ell)]] \end{aligned} \tag{4}$$

where  $z_\ell$  and  $c_\ell$  denote the latent variables and linguistic features located at the  $\ell$ -th level, respectively. The first term represents the reconstruction loss, and the second and third terms denote the Kullback–Leibler (KL) divergence. Since the second term models the

frame-level latent variable, it does not conditionally depend on linguistic features. In the training phase, HAE learns the posterior distribution of the latent variables to guide the prior distribution in HAD. However, HAD directly generates a speech waveform using predicted prior at the inference. The specific structure of HierTTS is described in detail in Section 3.5.

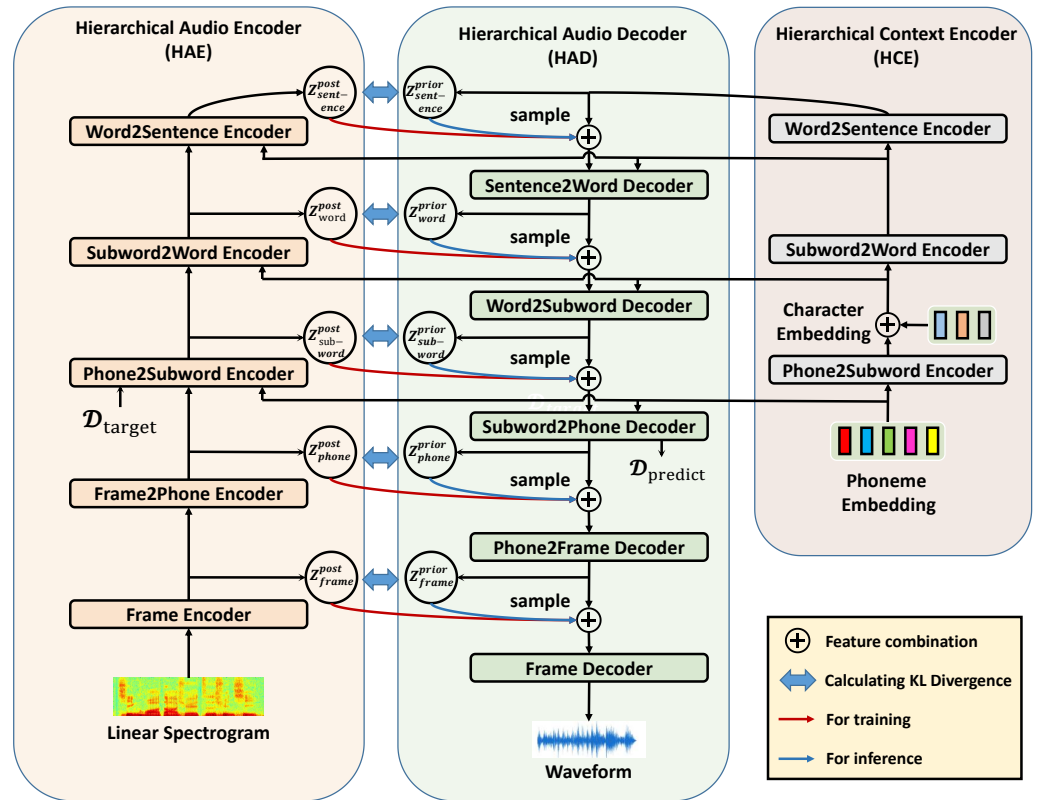


Figure 1. The architecture of HierTTS.

### 3.2. Reconstruction Losses

The reconstruction loss contains two parts: waveform reconstruction and duration reconstruction. For the waveform reconstruction loss, we use the multi-precision STFT loss [15] instead of calculating the waveform directly. It is equivalent to the sum of multiple spectrogram losses computed using various STFT parameter sets, consisting of a spectral convergence loss and a logarithmic STFT magnitude loss. It is defined as follows:

$$L_{sc}(s, \hat{s}) = \frac{\|s - \hat{s}\|_F}{\|s\|_F}, L_{mag}(s, \hat{s}) = \frac{1}{5} \|(\log(s) - \log(\hat{s}))\|_1 \quad (5)$$

$$L_{stft} = \frac{1}{M} \sum_{m=1}^M \mathbb{E}_{x, \hat{x}} [L_{sc}(s_m, \hat{s}_m) + L_{mag}(s_m, \hat{s}_m)] \quad (6)$$

where  $\|\cdot\|_F$  and  $\|\cdot\|_1$  represent the Frobenius and L1 norm, respectively.  $s$  denotes the number of elements in the spectrum, and  $M$  is the number of sets of STFT parameters chosen. Similarly to VITS and FastSpeech2s [16], we employ a window training strategy, only taking fragments of speech generated for calculating the waveform reconstruction loss.

For the duration reconstruction loss, we first get the phoneme duration as the target with the help of an external alignment tool and then optimize the duration reconstruction using the log-scale L2 loss.

$$L_{dur} = \left\| \log(D_{target}) - \log(D_{predict}) \right\|_2 \quad (7)$$

### 3.3. Adversarial Training

To further enhance the waveform generation, we introduce adversarial training in the time domain and frequency domain [17]. A multi-precision spectral discriminator (MRSD) is employed to discriminate speech from different levels of temporal and spectral precisions from different temporal and spectral accuracies. Additionally, a multi-period waveform discriminator (MPWD) is used to enhance the detailed adversarial modeling in the time domain. For the MPWD, the periodic components of the waveform are extracted at a set of prime intervals and used as input for each sub-discriminator. In this paper, we use the objective function of least squares [18] to perform adversarial training.

$$L_G = \frac{1}{K} \sum_{k=1}^K \mathbb{E}_{z,c} [(D_k(G(z,c)) - 1)^2] \quad (8)$$

$$L_D = \frac{1}{K} \sum_{k=1}^K \mathbb{E}_x [(D_k(x) - 1)^2] + \mathbb{E}_{z,c} [D_k(G(z,c))^2] \quad (9)$$

In these equations,  $D_k$  denotes the  $k$ -th sub-discriminator of MPWD and MRSD, and  $K$  denotes the total number of sub-discriminators.

### 3.4. Staged KL Weighted Annealing

Hierarchical VAEs are difficult to train [19,20], since the latent variables in the higher hierarchy tend to remain independent of the input data, which is the so-called phenomenon of posterior collapse. Eventually, there will be a tendency for the posterior distribution to fall back to the prior distribution. This problem is especially severe at the hierarchy furthest from the input: When training the HierTTS with a true variational lower bound, we found that the KL divergence at the sentence and word levels drops rapidly to zero, indicating that those upper hierarchy neurons may not be activated. Therefore, inspired by Beta-VAE [21], we propose the mechanism of staged KL weight annealing for HVAE training, where we set different weights for KL terms of different scales. Like a waterfall, to let the information gradually flow from the bottom latent variable to the top, we ensure a drop in KL weights in the adjacent hierarchy. The KL weight far away from the output is relatively higher. As shown in Figure 2, in the initial stage of training, we set lower weights for the overall KL items and increased the KL weights from fine-to-coarse step by step as the training progressed. The closed term of KL divergence at the  $\ell$ -th level will be

$$KL_\ell = \int \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}} \left[ \log \frac{\sigma_2}{\sigma_1} + \frac{(x-\mu_1)^2}{2\sigma_1^2} - \frac{(x-\mu_2)^2}{2\sigma_2^2} \right] dx \quad (10)$$

where  $\mu_1, \sigma_1^2$  and  $\mu_2, \sigma_2^2$  represents the parameters of  $\ell$ -th estimated posterior and predicted prior. After combining the VAE loss and GAN loss, the final loss of the training phase is

$$L_{\text{total}} = L_{dur} + L_{stft} + \sum_{\ell=0}^5 \beta_\ell * KL_\ell + L_{adv}(G) \quad (11)$$

where  $\beta_\ell$  denotes the penalty weight of the KL term at the  $\ell$ -th level.

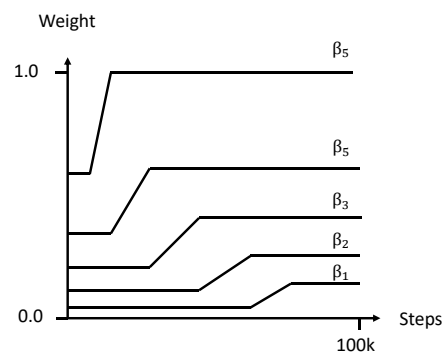


Figure 2. Weight changes of different scales.

### 3.5. Component of HierTTS

In this section, we will describe the components of HierTTS, including HAE, HAD, and HCE. It is worth mentioning HAE is only used in the training phase.

#### 3.5.1. HAE

HAE estimates the posterior distribution of hierarchical latent variables from fine to coarse. We first generate frame-level representations from the linear spectrogram. Then, frame-level representations are sequentially down-sampled into phoneme-level, subword-level, word-level, and sentence-level representations using hard alignment. The posterior distribution parameters of the corresponding hierarchy can be obtained with affine transformation.

HAE contains five components: the frame, Frame2Phone, Phone2Subword, Subword2Word, and Word2Sentence encoders. For the frame encoder, we use the non-causal WaveNet residual block in WaveGlow [22], Glow-TTS [23], and VITS. The WaveNet residual block consists of a dilated convolutional layer with gated activation units and skipped connections. Other encoders adopt a similar network structure to that shown in the left of Figure 3. It contains a bidirectional GRU and an attention-pooling module (AP). Finer representations and linguistic information are first sent to bidirectional GRU to extract temporally relevant contextual information. Then, AP is used to obtain coarser representations utilizing alignment information.

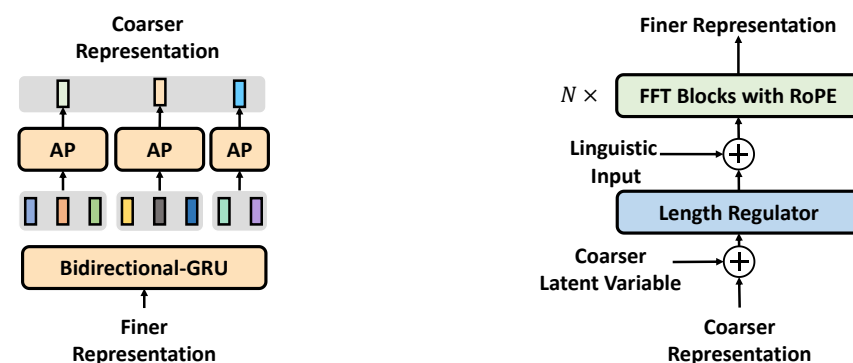


Figure 3. Components of fine-to-coarse and coarse-to-fine HAE.

#### 3.5.2. HAD

HAD generates the prior distribution of hierarchical latent variables from coarse to fine granularity utilizing multi-scale linguistic information and ultimately generates speech waveforms. We successively up-sample sentence-level representations to word-level, subword-level, phoneme-level, and frame-level representations with hard alignment information. Similarly, each level’s prior distribution parameters can be obtained by using the affine transformation on corresponding representations.

HAD consists of Sentence2Word, Word2Subword, Subword2Phone, Phone2-Frame, and frame decoders. For the frame decoder, we follow the vocoder structure of UnivNet [17]. It incorporates a MelGAN generator enhanced with location-variable convolution (LVC) where a noise sequence is used as input and a frame-level representation as the condition. All kernels of the LVC layer are directly predicted from the frame-level representation prediction using a residual module. Except for the frame decoder, the decoders in HAD adopt a similar structure. We adopt feed-forward transformer blocks [24] as the basic structure of a decoder. Rotary position embedding (RoPE) [25,26] is employed to provide relative location information. As shown in the right of Figure 3, coarse-grained representations are the first duplicates at a finer level. Then, the expanded features and linguistic information are fed into the FFT module to obtain fine-grained representations.

### 3.5.3. HCE

HCE is used to extract linguistic information at different granularity. Similarly to HAE, we rely on the structure on the left of Figure 3 to realize fine-to-coarse encoding except for replacing the bidirectional GRU with a linear layer. The obtained linguistic representations are input into each hierarchy of the HAE and HAD.

## 4. Experimental and Results

### 4.1. Experimental Setup

#### 4.1.1. Dataset

The experiment was implemented on an internal expressive Mandarin dataset, which contains 15 h of recordings from a male speaker, including 5 h of emotional data. The whole dataset contains 18,376 sentences. We randomly selected 17,776 sentences for training, 300 sentences for validation, and the remaining 300 sentences for testing.

The audios were unified to a 16,000 sampling rate and 16-bit PCM. The silence at the beginning and end of each sentence was removed. In addition, phone alignment was obtained by the MFA tool [27]. Subwords in Chinese mandarin correspond to Chinese characters. Using the pronunciation dictionary, the relationships between Chinese characters and phonemes can be determined. Subword-level information was provided from tiny-bert [28]. Additionally, we used the PKUseg word segmentation tool [29] to obtain word boundary information. In addition, we used the configuration of frame length 1024, frameshift 256, and window length 1024 to extract the linear spectrogram. We have made both our demo page and source code publicly available (source-code: <https://github.com/shang0712/HierTTS> (accessed on 4 January 2023); demo: <https://shang0712.github.io/HierTTS/> (accessed on 4 January 2023)).

#### 4.1.2. Training Setting

All models were trained with a batch size of 32 for 100,000 iterations. An AdamW optimizer with  $\beta_1 = 0.8$ ,  $\beta_2 = 0.99$  and weight decay  $\lambda = 0.01$  was adopted for training. The starting learning rate was  $2 \times 10^{-4}$ , and the period decay coefficient was set to  $0.999^{1/8}$ . Following previous work, we used a window generation training method. This method generates only part of the speech waveform at a time during training, which shortens the training time and reduces memory usage. Specifically, frame-level feature fragments of length 120 were fed to the frame decoder to generate the corresponding speech. We found that setting larger segment lengths may contribute to more stable performance in prosody generation.

### 4.2. Baselines

To further evaluate the performance of HierTTS, four baseline systems were employed for comparison: (1) FastSpeech 2 + HiFiGAN [30], (2) MultiGST [6] + HiFiGAN, (3) PortaSpeech [31] + HiFiGAN, (4) VITS [7]. MultiGST was proposed by Lei and Zhou et al., which employs multiple GSTs to extract style vectors at different scales and uses the hierarchical text information to predict the style vectors extracted during inference. For a fair

comparison, paragraph-level GST was removed, since our dataset was recorded sentence by sentence.

#### 4.3. Expressiveness

We relied on a 5-value mean opinion score (MOS) to evaluate the naturalness of speech synthesis systems. During the testing phase, 100 test samples were generated for each system, and each test sample was evaluated by 20 testers. Moreover, the text for generation contained a portion of sentences with significant emotional tendencies.

The listening test results are shown in Table 1. The recordings in our dataset contain rich prosodic variations. Among those baselines, PortaSpeech achieved the lowest MOS, since our dataset is a mix of emotional speech and reading; maybe it is difficult for PortaSpeech to fit training data with such rich prosodic variation. The MOS of FastSpeech2 is also low; its prosody is relatively flat, which may be the cause of the over-smoothing of FastSpeech2. MultiGST improves the naturalness of FastSpeech2, which suggests that leveraging semantic information helps learn variable prosody. However, since MultiGST only models prosody at the subword level and sentence level, the rhythm it produces is far from that of recordings. The speech generated by VITS also suffers from rhythm and stress errors. Our model achieves higher naturalness compared to all the baselines. HierTTS is very natural in pauses and stresses and close to the rhythmic performance of the recording.

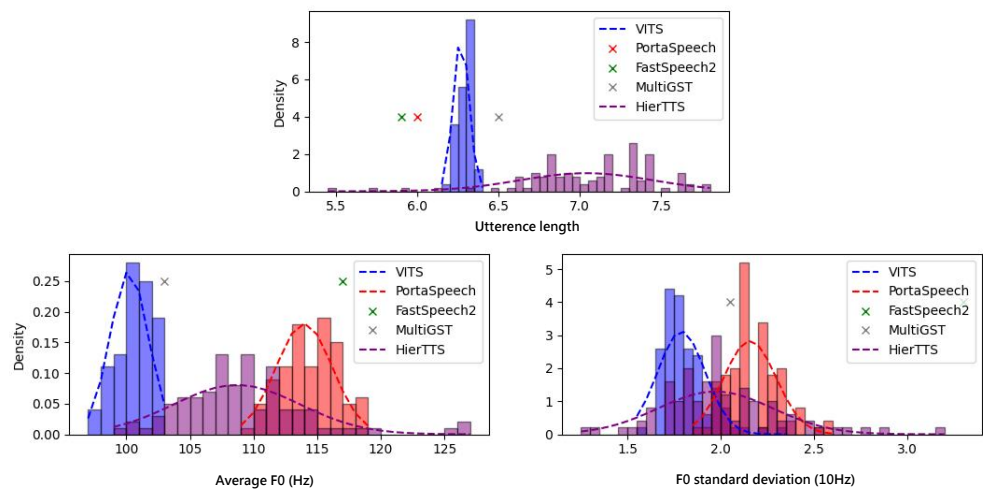
**Table 1.** MOS evaluation of HierTTS and baselines with 95% confidence intervals.

System	MOS
FastSpeech2 + HiFiGAN	3.84 ± 0.15
MultiGST + HiFiGAN	4.05 ± 0.13
PortaSpeech + HiFiGAN	3.06 ± 0.13
VITS	3.93 ± 0.11
HierTTS	4.32 ± 0.12
GroundTruth	4.56 ± 0.13

#### 4.4. Diversity Analysis

HierTTS has better generative diversity due to modeling five levels of latent variables. To verify the improvement of the model in generative diversity, we generated 100 samples using different systems for the same text (text to be synthesized: 淋过雨的空气, 疲倦了的伤心, 我记忆里的童话已经慢慢的融化。). Figure 4 shows example histograms of each prosody feature for the generated speech samples. For the duration, among the baseline models, only VITS has duration diversity. Other models can only generate a deterministic duration for the same text. However, compared to VITS, HierTTS is more diverse in duration. HierTTS can speak both quickly and slowly, but in VITS, durations tend to be aggregated to some centralized locations in the distribution. In terms of fundamental frequency diversity, PortaSpeech and VITS can generate diverse fundamental frequencies, and their fundamental frequency diversity is still inferior to HierTTS. As shown in the left and right panels of Figure 4, HierTTS covers the range of VITS and PortaSpeech. This means that VITS and Portaspeech cannot learn the completed distribution, and HierTTS can better fit the real prosodic distribution.





**Figure 4.** Example histograms of utterance length, utterance average pitch, and pitch standard deviation within an utterance. One-hundred samples were generated from each model.

#### 4.5. Ablation Experiments and Methodological Analysis

##### 4.5.1. Ablation Study

We implemented an ablation study to verify the effectiveness of different scales for HierTTS. We chose comparative MOS (CMOS) with seven points (from  $-3$  to  $3$ ) to evaluate the example by pairs. Results are summarized in Table 2: (1) By removing the HAE, HierTTS degrades into a simple cascaded TTS model, which brings a CMOS drop of 0.24. The performance loss is mainly manifested in unnatural rhythms and stress. For example, an unexpected pause occurs in the middle of a word, or there are multiple accents in a sentence. Additionally, some words are mispronounced. This may be because the text in the training data are sparse, and the generalization problem can be severe when the latent space is unconstrained, thereby affecting the pronunciation. For example, there are speech pauses within words or even multiple stresses within a sentence. In addition, some words are mispronounced. This may be because the text in the training data is sparse, and when the latent space is not constrained, the generalization issue will be serious, which will affect the pronunciation. (2) Removing the sentence-level latent variable brings a CMOS drop of 0.33. The sentence-level information encodes the overall prosody. After removing the sentence level, the prosody fluctuates too much in a sentence, which also leads to unnatural prosody. Occasionally, the speaking speed suddenly becomes fast or slow. (3) Removing word-level latent variables brings a CMOS drop of 0.22. After the word level is removed, the model loses the ability to recognize word boundaries, and the most direct impact is wrong pauses, especially for some nouns, such as names of people and places. (4) Removing the subword-level hidden variable brings a CMOS drop of 0.18. The rhythm of the generated speech is not coherent enough—e.g., it will be read word by word. (5) Removing staged weight annealing brings a CMOS drop of 0.42. KL divergence at the sentence and word level quickly drops to zero, and sampling over the corresponding hidden variable will not affect generated speech. Synthesized speech also suffers from serious prosody problems.

**Table 2.** CMOS evaluation of ablation studies.

System	CMOS
-HVAE	-0.24
-Sentence level	-0.33
-Word level	-0.22
-Subword level	-0.18
-Staged weight annealing	-0.42
HierTTS	0

#### 4.5.2. Visualization of Latent Space

In Figure 5, we visualize the distribution of the hidden variables. The hidden vectors of different emotions in the sentence-level hidden space are clustered together, and there are no obvious boundaries in the hidden spaces of several other scales. This indicates that the sentiment-related information is mainly encoded in the sentence-level hidden vectors. As shown on the right of Figure 5, HierTTS is able to infer sentimental tendencies from the text. It is worth mentioning that we do not enforce clustering among different emotions. Therefore, by sampling, HierTTS is able to generate speech rich in emotions at inference.

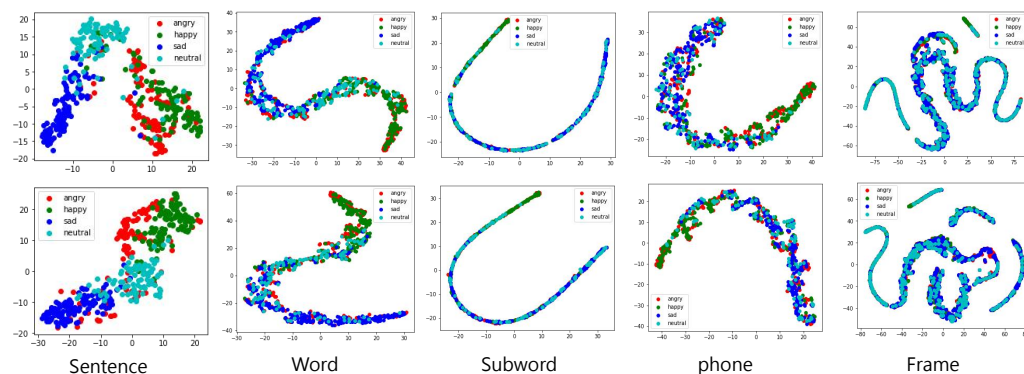


Figure 5. TSNE of priors (upper) and posteriors (lower).

## 5. Conclusions

This paper proposed HierTTS, an end-to-end text-to-waveform architecture designed to improve the expressiveness and generation diversity of text-to-speech systems. It models hierarchically dependent style variables and use hierarchical variational autoencoders to jointly optimize style extraction and style prediction. Experiments verified the improvements in naturalness and generation diversity. However, compared with recording, HierTTS still has room for improvement in sound quality. In the future, we will explore how to close the gap in sound quality in recordings and explore a more stable hierarchical generation neural network without the aid of staged KL weighted annealing.

**Author Contributions:** Conceptualization, Z.S., P.S., P.Z., L.W. and G.Z.; methodology, Z.S.; implementation Z.S.; validation, Z.S., P.S., P.Z., L.W. and G.Z.; formal analysis, Z.S.; writing—original draft preparation, Z.S.; supervision, P.S., L.W. and P.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is partially supported by the National Key Research and Development Program of China (No.2021YFC3320102).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Chen, W.; Xing, X.; Xu, X.; Pang, J.; Du, L. SpeechFormer: A Hierarchical Efficient Framework Incorporating the Characteristics of Speech. In Proceedings of the International Conference on Machine Learning, Boca Raton, FL, USA, 16–19 December 2019; pp. 3331–3340.
2. Kenter, T.; Wan, V.; Chan, C.A.; Clark, R.; Vit, J. CHiVE: Varying prosody in speech synthesis with a linguistically driven dynamic hierarchical conditional variational network. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 10–15 June 2019; pp. 3331–3340.
3. Sun, G.; Zhang, Y.; Weiss, R.J.; Cao, Y.; Zen, H.; Wu, Y. Fully-hierarchical fine-grained prosody modeling for interpretable speech synthesis. In Proceedings of the ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6264–6268.

4. Bae, J.S.; Yang, J.; Bak, T.J.; Joo, Y.S. Hierarchical and Multi-Scale Variational Autoencoder for Diverse and Natural Non-Autoregressive Text-to-Speech. *arXiv* **2022**, arXiv:2204.04004.
5. Lei, Y.; Yang, S.; Wang, X.; Xie, L. MsEmoTTS: Multi-scale emotion transfer, prediction, and control for emotional speech synthesis. *IEEE/ACM Trans. Audio Speech, Lang. Process.* **2022**, *30*, 853–864. [[CrossRef](#)]
6. Lei, S.; Zhou, Y.; Chen, L.; Hu, J.; Wu, Z.; Kang, S.; Meng, H.; Towards Multi-Scale Speaking Style Modelling with Hierarchical Context Information for Mandarin Speech Synthesis. *arXiv* **2022**, arXiv:2204.02743.
7. Kim, J.; Kong, J.; Son, J. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 5530–5540.
8. Tan, X.; Chen, J.; Liu, H.; Cong, J.; Zhang, C.; Liu, Y.; Wang, X.; Leng, Y.; Yi, Y.; He, L.; et al. NaturalSpeech: End-to-End Text to Speech Synthesis with Human-Level Quality. *arXiv* **2022**, arXiv:2205.04421.
9. Wu, J.; Hu, C.; Wang, Y.; Hu, X.; Zhu, J. A hierarchical recurrent neural network for symbolic melody generation. *IEEE Trans. Cybern.* **2019**, *50*, 2749–2757. [[CrossRef](#)] [[PubMed](#)]
10. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. *arXiv* **2013**, arXiv:1312.6114.
11. Sønderby, C.K.; Raiko, T.; Maaløe, L.; Sønderby, S.K.; Winther, O. Ladder variational autoencoders. In Proceedings of the 29th Annual Conference on Neural Information Processing Systems (NIPS 2016), Spain, Barcelona, 5–10 December 2016.
12. Vahdat, A.; Kautz, J. NVAE: A deep hierarchical variational autoencoder. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 19667–19679.
13. Lee, Y.; Shin, J.; Jung, K. Bidirectional variational inference for non-autoregressive text-to-speech. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
14. Liu, P.; Cao, Y.; Liu, S.; Hu, N.; Li, G.; Weng, C.; Su, D. Vara-tts: Non-autoregressive text-to-speech synthesis based on very deep vae with residual attention. *arXiv* **2021**, arXiv:2102.06431.
15. Yamamoto, R.; Song, E.; Kim, J.M. Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In Proceedings of the ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6199–6203.
16. Ren, Y.; Hu, C.; Tan, X.; Qin, T.; Zhao, S.; Zhao, Z.; Liu, T.Y. FastSpeech 2: Fast and High-Quality End-to-End Text to Speech. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
17. Jang, W.; Lim, D.; Yoon, J.; Kim, B.; Kim, J. UnivNet: A neural vocoder with multi-resolution spectrogram discriminators for high-fidelity waveform generation. *arXiv* **2021**, arXiv:2106.07889.
18. Mao, X.; Li, Q.; Xie, H.; Lau, R.Y.; Wang, Z.; Paul Smolley, S. Least squares generative adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2794–2802.
19. Pervez, A.; Gavves, E. Spectral smoothing unveils phase transitions in hierarchical variational autoencoders. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 8536–8545.
20. Pervez, A.; Gavves, E. Variance Reduction in Hierarchical Variational Autoencoders. In Proceedings of the International Conference on Machine Learning, Virtual, 13–18 July 2020.
21. Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. In Proceedings of the 5th International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
22. Prenger, R.; Valle, R.; Catanzaro, B. Waveglow: A flow-based generative network for speech synthesis. In Proceedings of the ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 3617–3621.
23. Kim, J.; Kim, S.; Kong, J.; Yoon, S. Glow-tts: A generative flow for text-to-speech via monotonic alignment search. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 8067–8077.
24. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the 30th Annual Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
25. Zhang, H.; Huang, Z.; Shang, Z.; Zhang, P.; Yan, Y. LinearSpeech: Parallel Text-to-Speech with Linear Complexity. In Proceedings of the Interspeech 2021, Brno, Czechia, 30 August–3 September 2021; pp. 4129–4133.
26. Su, J.; Lu, Y.; Pan, S.; Wen, B.; Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *arXiv* **2021**, arXiv:2104.09864.
27. McAuliffe, M.; Socolof, M.; Mihuc, S.; Wagner, M.; Sonderegger, M. Montreal Forced Aligner: Trainable Text-Speech Alignment Using Kaldi. In Proceedings of the Interspeech 2017, Stockholm, Sweden, 20–24 August 2017; pp. 498–502.
28. Jiao, X.; Yin, Y.; Shang, L.; Jiang, X.; Chen, X.; Li, L.; Wang, F.; Liu, Q. TinyBERT: Distilling BERT for Natural Language Understanding. In Proceedings of the Findings of the Association for Computational Linguistics, Online, 16–20 November 2020; pp. 4163–4174.
29. Luo, R.; Xu, J.; Zhang, Y.; Ren, X.; Sun, X. Pkuseg: A toolkit for multi-domain chinese word segmentation. *arXiv* **2019**, arXiv:1906.11455.

30. Kong, J.; Kim, J.; Bae, J. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 17022–17033.
31. Ren, Y.; Liu, J.; Zhao, Z. Portaspeech: Portable and high-quality generative text-to-speech. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 13963–13974.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.