CROSS-UTTERANCE CONDITIONED COHERENT SPEECH EDITING VIA BIASED TRAINING AND ENTIRE INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

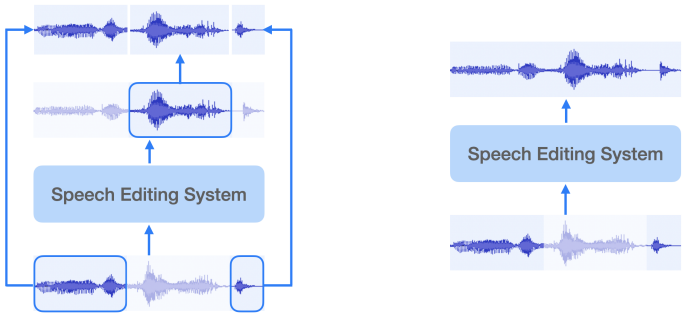
Text-based speech editing systems are developed to enable users to select, cut, copy, and paste speech based on the transcript. Existing state-of-the-art editing systems based on neural networks do partial inferences with no exception, that is, only generate new words that need to be replaced or inserted. This manner usually leads to the prosody of the edited part being inconsistent with the previous and subsequent speech and a failure to handle the alteration of intonation. To address these problems, we propose a cross-utterance conditioned coherent speech editing system, that first does the entire reasoning at inference time. Benefiting from a cross-utterance conditioned variational autoencoder, our proposed system can generate speech by utilizing speaker information, context, acoustic features, and the mel-spectrogram of unedited fragments from the original audio. Also, we apply biased training to concentrate more attention on the part that needs to be reconstructed throughout training. Experiments conducted on subjective and objective metrics demonstrate that our approach outperforms the partial inference method on various editing operations regarding naturalness and prosody consistency.

1 INTRODUCTION

Speech editing can be applied to a variety of areas with personalized voice needs and higher demands for speech naturalness, including video creation for social media, games, and movie dubbing. The traditional voice editing tool (Derry, 2012) allows users to pitch bend, denoise, modify volume, cut, copy, and paste waveform, among other features. Among these, standard voice editing work will be fairly cumbersome when the transcript of the audio that needs editing needs to be revised. Especially when new words that are not in the transcript occur, editors can only re-record the corresponding clips and then splice them with the original audio. According to this scheme, it can be expected that that variations of the recording environment may impact the background noise, while the changes in the speaker's condition could also lead difference to the loudness, pitch, and rhythm between the re-recorded voice clip and the original voice, further resulting in an unnatural hearing.

A promising neural-network-based audio editing technology is to synthesize speech according to text transcription and original audio. This system is capable of synthesizing speech that matches the tone and timbre of the original audio, according to the aligned transcription altered by content authors. As a result, rather than editing the original audio, editors could perhaps lessen their burden by altering the text transcription. Previous work (Moulines & Charpentier, 1990; Morise et al., 2016; Kawahara, 2006) based on digital signal processing (DSP) has partially overcome the problem of prosody mismatch created by directly concatenating the audio in different scenarios. Morrison et al. (2021) utilizes neural network to predict prosodic information and integrates the TD-PSOLA algorithm, denoising, and de-reverberation (Su et al., 2020) approaches to realize prosodic modification. Although the above systems support cut, copy, and paste operations, they cannot insert or replace a new word that doesn't exist in the voice data of the same speaker.

More recent research has applied text-to-speech (TTS) systems to synthesize the missing inserted word. VoCo (Jin et al., 2017) synthesizes the inserted word using a comparable TTS voice, then transforms it using the voice conversion (VC) model to fit the target speaker. EditSpeech (Tan et al.,



(a) Partial inference (b) Entire inference

Figure 1: Illustrations for different inference ways of speech editing system.

2021) proposes the partial inference and bidirectional fusion method to achieve smooth transitions at edit boundaries. CampNet (Wang et al., 2022) conducts mask-training on a context-aware neural network based on Transformer to improve the quality of the edited voice. Bai et al. (2022) suggests an alignment-aware acoustic and text pretraining method, which can be directly applied to speech editing by reconstructing masked acoustic signals through text input and acoustic text alignment. What's more, SpeechPainter(Borsos et al., 2022) leverages an auxiliary textual input to fill in gaps of up to one second in speech samples and generalize it to unseen speakers.

However, when applied to speech editing, all the existing methods (Jin et al., 2017; Tan et al., 2021; Wang et al., 2022; Bai et al., 2022) based on neural networks do partial inference instead of entire inference, as shown in Figure 1(a). Specifically, the input of existing systems is the waveform or mel-spectrogram of the segments that do not need editing. Although the direct output of the editing module is the complete waveform or mel-spectrogram corresponding to the edited transcripts, in order to improve the similarity with the original audio, the existing methods select only the segments that must be modified and then insert them back into the original waveform or mel-spectrogram. Although retaining the original audio as much as possible adheres to our intuition, it will also lead to the following potential problems,

1. Since partial inference artificially inserts the predicted acoustic characteristics of the editing area into the corresponding positions of the original waveform, the discontinuity near the boundary of the editing area is almost inevitable to a certain extent. Meanwhile, the output of the existing speech synthesis system based on partial inference is still the whole audio, including the context. Therefore, it will not spare time or resources compared with the entire inference.

2. When the transcript is modified, the tone and prosody could also change accordingly. That is, the audio corresponding to the altered text might not be intended to sound exactly like the original audio. A special example is when a general question sentence can be modified into a declarative sentence, partial inference will be difficult to deal with the mood change.

To address the issues raised above, we propose a cross-utterance conditioned coherent speech editing system. This text-based speech editing system applies the variational autoencoder with masked training to reconstruct the unmodified area of the original waveform with high fidelity. Therefore, the entire inference can replace partial inference, so as to avoid the incoherence of the junctions caused by splicing. Also, compared with the existing partial reasoning editing system, our method does not consume additional resources. This point can be intuitively accepted through Figure 1, where the framework of entire inference is more concise than partial inference.

Moreover, to ensure that the generated audio conforms to both the original audio features and the context after the edition, the variational autoencoder is conditioned on the semantic information of the context and audio features extracted from the original waveform. Apart from that, we set a bias to mask segments of the mel-spectrogram during training to enable the system to be more focused on the part that needs to be reconstructed. The subjective and objective results on a challenging dataset show that our proposed model can ensure a high degree of similarity with real audio, while the coherency of entire inference is significantly better than that of partial inference.

The rest of this paper is organized as follows. The background of the non-autoregressive TTS system and mask training are introduced in Section 2. Section 3 illustrates our proposed speech editing system. The experimental setup, results, and conclusion are presented in Sections 4, 5 and 6, respectively.

2 LITERATURE REVIEW

2.1 NON-AUTOREGRESSIVE TTS

Existing speech synthesis systems based on neural networks can be divided into two categories, including attention-based autoregressive (AR) systems and duration-based non-autoregressive (NAR) systems. Unlike AR systems, NAR systems can decode time series in parallel without internal dependency. Fastspeech 2 (Ren et al., 2021) is a representative of the NAR TTS system, which first predicts the duration and prosody information explicitly. Based on this system, Li et al. (2022) presents a cross-utterance conditional VAE component to estimate a posterior probability distribution of the latent prosody features for each phoneme by conditioning on acoustic parameters, speaker information, and context text features, further improving the expressiveness of prosody while maintaining high fidelity in synthesized speech.

2.2 MASK TRAINING

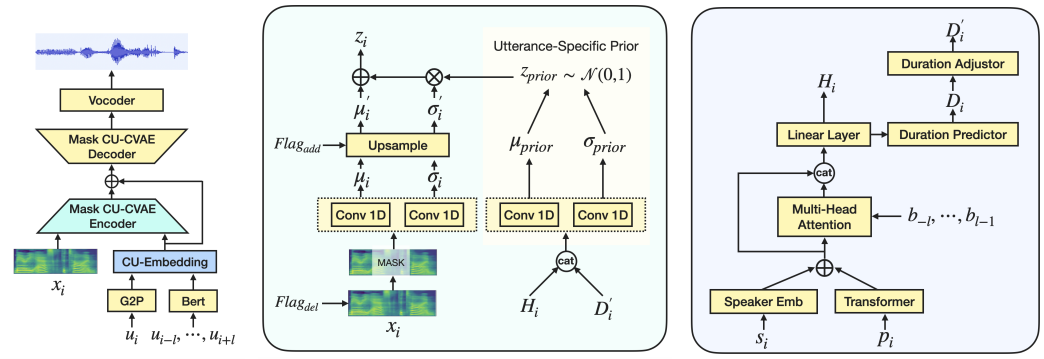
Masked signal modeling is a representation learning method that learns to understand and create, that is, masking a part of input signals and trying to predict these masked signals. This technology has been widely used in various fields and its effectiveness has been verified (Xie et al., 2022). In the natural language processing task, the pre-training language model BERT (Devlin et al., 2019), which is based on the mask language modeling task, has been proven to be widely extended to the downstream tasks, and has reshaped this field to a large extent. In the field of computer vision, MAE He et al. (2021) selected a high mask rate of 75% to add noise to the image, forcing the encoder to reconstruct the masked image by learning the semantic information in the image rather than simply surrounding pixels. In terms of speech recognition, wav2vec2.0 (Baevski et al., 2020) completes the modeling task by predicting the speech input of the hidden part in the potential space and defining the comparison task on the quantization of the potential representation. Similar to the BERT model, the speech editing model A^3T (Bai et al., 2022) uses phonemes and partially masked mel-spectrograms as inputs during training, which verifies that mask training can reconstruct mel-spectrograms with high quality.

3 OUR SYSTEM

Our proposed text-based speech editing system aims to synthesize the new audio that is consistent with the original audio rhythm and to truly restore the unmodified part of the audio, by virtue of the reconstruction ability of a variational autoencoder conditioned on context information. Figure 2(a) describes the model architecture, which takes the mel-spectrogram $\boldsymbol{x}_i$ extracted from the original waveform, current utterance $\boldsymbol{u}_i$, and l utterances before and after $\boldsymbol{u}_i$ as the input. Using an additional G2P conversion tool, the utterance $\boldsymbol{u}_i$ is translated into phonemes $\boldsymbol{p}_i$. Following Li et al. (2022), the $2l+1$ neighboring utterances into $2l$ pairs, i.e. $[(\boldsymbol{u}_{i-l}, \boldsymbol{u}_{i-l+1}), \cdots, (\boldsymbol{u}_{i+l-1}, \boldsymbol{u}_{i+l})]$, and use BERT to capture the cross-utterance information, yeilding $2l$ BERT embeddings $[\boldsymbol{b}_{-l}, \cdots, \boldsymbol{b}_{l-1}]$. Also, the start and end times of each phoneme can be extracted by Montreal forced alignment (McAuliffe et al., 2017). The following part details the design of our system and biased training.

3.1 MASK CU-ENHANCED CVAE

The mask CU-Enhanced CVAE module, as shown in Figure 2(b), is proposed to overcome the limitation that existing speech editing systems cannot restore the unmodified portion of audio and must splice the modified portion with the original mel-spectrogram or audio.



(a) Our system (b) Mask CU-CVAE encoder (c) CU-embedding module

Figure 2: The overall architecture of our system, consisting of the mask cross-utterance enhanced CVAE and cross-utterance embedding module, is integrated into and jointly optimized with the FastSpeech 2 system. $\oplus$ and $\otimes$ are element-wise addition and multiplication operations. (cat) is the concatenate operation.

3.1.1 Implementation of Text-based Speech Editing Operations

To start with, a text-based speech editing system supports the operations of deletion, insertion, and replacement. Without loss of generality, we can divide the original utterance transcript of the original speech as $[\boldsymbol{u}_a, \boldsymbol{u}_b, \boldsymbol{u}_c]$ and the modified utterance to be $[\boldsymbol{u}_a, \boldsymbol{u}_{b'}, \boldsymbol{u}_c]$, where $\boldsymbol{u}_{b'}$ is the modified segment and $\boldsymbol{u}_a, \boldsymbol{u}_c$ remain the same. Correspondingly, the phonemes translated by G2P can be denoted as $\boldsymbol{p}_i = [\boldsymbol{p}_a, \boldsymbol{p}_b, \boldsymbol{p}_c]$, with original speech's mel-spectrogram denoted as $\boldsymbol{x}_i = [\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{x}_c]$. For $i \in \{a, b, c\}$, $\boldsymbol{x}_i$ contains a sequence of frame-level mel-spectrogram. Since the replacement operation in editing can be regarded as deletion before addition, we can use two flags instead of three to indicate the place to delete and add the corresponding content, i.e., $Flag_{del}$ and $Flag_{add}$.

Deletion
The deletion procedure enables the user to eliminate a segment of the speech waveform which is associated with a set of certain words. The target utterance to be synthesized after deletion is $[\boldsymbol{u}_a, \boldsymbol{u}_c]$, where $\boldsymbol{u}_b$ is the part to be deleted. By comparing the utterance before and after editing, we can get the corresponding deletion indicator, which is further used to instruct the editing of mel-spectrogram

$$Flag_{del} = [\mathbf{0}_a, \mathbf{1}_b, \mathbf{0}_c].$$

Insertion and Replacement
Different from the deletion operation, the target synthesized speech after insertion or replacement is based on the edited utterance $[\boldsymbol{u}_a, \boldsymbol{u}_{b'}, \boldsymbol{u}_c]$, where $\boldsymbol{u}_{b'}$ is the content to replace $\boldsymbol{u}_b$. Noted that insertion process can considered as the special case where $\boldsymbol{u}_b = \boldsymbol{p}_b = \boldsymbol{x}_b = \varnothing$. Correspond to the deletion operation, we have the addition indicator

$$Flag_{add} = [\mathbf{0}_a, \mathbf{1}_{b'}, \mathbf{0}_c].$$

Based on $Flag_{del}$, the reference mel-spectrogram $[\boldsymbol{x}_a, \boldsymbol{x}_c]$ is sent into the Mask CU-Enhanced CVAE module since $\boldsymbol{x}_{b'}$ is to be generated. The mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\sigma}$ are learnt from two one-dimensional convolutions. Referring to $Flag_{add}$, 0 and 1s are added to the corresponding position of $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$, that is, $\hat{\boldsymbol{\mu}} = [\boldsymbol{\mu}_a, \mathbf{0}_{b'}, \boldsymbol{\mu}_c]$ and $\hat{\boldsymbol{\sigma}} = [\boldsymbol{\sigma}_a, \mathbf{1}_{b'}, \boldsymbol{\sigma}_c]$. This allows the speech generated by the editing area to be sampled from the utterance-specific prior, while the audio of the area that has no modification is sampled from the real audio and the utterance-specific prior. During the training process, actually the edited real audio is unavailable, so we can only mask certain audio segment and restore the same content to simulate the editing scenario, that is $b^{'} = b$.

3.1.2 Enhancement of Coherence and Prosody

We have introduced more mechanisms to ensure that the output of mask CU-CVAE module can further synthesize coherent and contextual audio. In order to make the editing boundary more fluent,

$\boldsymbol{\mu}^{'}$ and $\boldsymbol{\sigma}^{'}$ are further generated through one-dimensional convolution from $\hat{\boldsymbol{\mu}}$ and $\hat{\boldsymbol{\sigma}}$. At this time, the module can sample from the estimated prior and can be re-parameterized as

$$\boldsymbol{z} = \boldsymbol{\mu}^{'} \oplus \boldsymbol{\sigma}^{'} \otimes \boldsymbol{z}_{prior}$$

where $\oplus, \otimes$ are element-wise addition and multiplication operations, and $\boldsymbol{z}_{prior}$ is sampled from the learned utterance-specific prior, corresponding to Li et al. (2022). The re-parameterization is as follows,

$$\boldsymbol{z}_{prior} = \boldsymbol{\mu}_{prior} \oplus \boldsymbol{\sigma}_{prior} \otimes \boldsymbol{\epsilon}$$

where $\boldsymbol{\mu}_{prior}, \boldsymbol{\sigma}_{prior}$ are learned from the utterance-specific prior module, and $\boldsymbol{\epsilon}$ is sampled from the standard Gaussian $\mathcal{N}(0, 1)$. $\boldsymbol{H}_i$ is the output of CU-Embedding, as shown in Figur 2(c). The CU-embedding module encodes cross-utterance information into a sequence of mixture embeddings, with a pretained BERT to capture the contextual information $[\boldsymbol{b}_{-l}, \cdots, \boldsymbol{b}_{l-1}]$ of the first and last l utterances surrounding the current one $\boldsymbol{u}_i$, a Transformer encoder to encode the phoneme sequence, and a multi-head attention layer to capture contextual information. Also, an additional duration predictor takes $\boldsymbol{H}_i$ as inputs and predicts the duration $\boldsymbol{D}_i$ of each phoneme. In addition, in order to effectively utilize the duration information extracted from the original audio, similar to the method in Tan et al. (2021); Bai et al. (2022), we further adjust the phoneme duration of the edited area by multiplying it with the ratio of the original audio and the predicted audio duration of the unedited area to get $\boldsymbol{D}_i^{'}$. The estimated duration is rounded after the duration predictor and the adjustor.

Therefore, the ELBO objective can be expressed as

$$\begin{aligned}
\mathcal{L}(\boldsymbol{x} \mid \boldsymbol{H}, \boldsymbol{D}^{'}) = & \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{D}', \boldsymbol{H})}[\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{D}^{'}, \boldsymbol{H})] \\
& - \beta_1 \sum_{n=1}^{t} D_{\mathrm{KL}}^{'}\left(q_{\phi_1}\left(\boldsymbol{z}^n \mid \boldsymbol{z}_{prior}^n, \boldsymbol{x}\right) \| q_{\phi_2}\left(\boldsymbol{z}_{prior}^n \mid \boldsymbol{D}^{'}, \boldsymbol{H}\right)\right) \\
& - \beta_2 \sum_{n=1}^{t} D_{\mathrm{KL}}^{'}\left(q_{\phi_2}\left(\boldsymbol{z}_{prior}^n \mid \boldsymbol{D}^{'}, \boldsymbol{H}\right) \| p(\boldsymbol{z}_{prior}^n)\right),
\end{aligned}$$

where the index i which denotes the current instance, is omitted for simplicity. θ is the decoder module parameters, ϕ_1, ϕ_2 are two parts of mask CU-CVAE encoder ϕ to obtain $\boldsymbol{z}$ from $\boldsymbol{z}_{prior}, \boldsymbol{x}$ and $\boldsymbol{z}_{prior}$ from $\boldsymbol{D}^{'}, \boldsymbol{H}$ respectively, β_1, β_2 are two balance constants, $p(\boldsymbol{z}_{prior}^n)$ is chosen to be standard Gaussian $\mathcal{N}(0, 1)$. Meanwhile, $\boldsymbol{z}^n$ and $\boldsymbol{z}_{prior}^n$ represent the latent representation for the n-th phoneme, and $t = a + b^{'} + c$ is the length of the phoneme sequence.

3.2 Biased Training

To reconstruct the waveform of the given transcript for each masked phonemes, the commonly adapted loss function for acoustic model is the mean absolute error (MAE) between the reconstructed and original mel-spectrogram, most compute the loss only on masked segments, similar to BERT (Devlin et al., 2019). In the training process, the input reference mel-spectrogram only includes the unmasked part. To enable the system more focus on masked part, it is rational to increase the loss weight of this area.

However, although during training, we can only imitate the audio editing operation by reconstructing the mel-spectrogram, during inference process, the purpose is to synthesize naturally coherent audio with rhythm conforming to the modified text context. Therefore, setting the loss weight of the unmasked area to zero is not appropriate in the case of speech editing.

In this way, we expect to be able to balance the two purposes of approaching the original audio and the context of the newly modified transcript. In the experiment, we set the loss ratio of the masked and the unmasked parts to λ=1.5.

$$\mathcal{L}_{mel} = \frac{1}{\#\text{ of frames}}\left(\sum_{i \in \text{unmask}} |x_i^{pred} - x_i^{target}| + \lambda \sum_{i \in \text{mask}} |x_i^{pred} - x_i^{target}|\right)$$

The results of the subsequent experiments also show that, compared to other weight settings, increasing the weight of the loss function of the reconstructed mel-spectrogram of the masked part can make the synthesized sound more natural and coherent.

4 EXPERIMENTAL SETUP

4.1 DATASET

We conducted experiments on a multi-speaker dataset, LibriTTS. Both the train-clean-100 and train-clean-360 subsets were used, containing 245 hours of English audiobook from 1151 speakers (553 female speakers and 598 male speakers). The dataset (Zen et al., 2019) includes adjacent sentences from which context information can be extracted. We randomly select 90%,5%,5% data from datasets for train, valid and test set, respectively. All audio clips were re-sampled at 22.04 kHz.

4.2 CONFIGURATION DETAIL

The proposed Mask CU-CVAE TTS system was based on the framework of FastSpeech 2. In the CU-embedding module, a Transformer is utilized to learn the current utterance representation, where the dimension of phoneme embeddings and the size of the self-attention were both set to 256. Meanwhile, we used "BERT_BASE" configuration, including 12 Transformer blocks; each block has 12-head attention layers; and the hidden size is 768. Also, the BERT model and associated embeddings were fixed throughout TTS system training. Additionally, information about different speakers learned from a 256-dim embedding layer was added to the Transformer output.

In the Mask CU-enhanced CVAE module, four 1D-convolutional (1D-Conv) layers with kernel sizes of 1 were used to predict the mean and variance of 2-dim latent features. Meanwhile, an additional upsampling layer was applied to make the predicted sequence length consistent with the phoneme sequence length after editing, and also promote the naturalness of synthesized audio. We randomly select the part to be masked by taking a word instead of a phoneme as a unit to faithfully recreate the actual editing scenario. In addition, to balance the system's ability to learn and predict audio information, we set the shielding rate to 50%, which has been proven effective in Bai et al. (2022).

Then, the sampled latent feature was converted to a 256-dim vector by a linear layer. The length regulator in FastSpeech 2's duration predictor, which consisted of two 1D convolutional blocks with ReLU activation, followed by layer normalization and an additional linear layer to predict the length of each phoneme, was adapted to take in the outputs of the CU-embedding module. Each convolutional block was comprised of a 1D-Conv network with ReLU activation, followed by layer normalization and a dropout layer. Four feed-forward Transformer blocks were used by the decoder to transform hidden sequences into an 80-dim mel-spectrogram sequence.

Finally, the vocoder HifiGAN (Kong et al., 2020) was finetuned for 1200 steps on an open-sourced, pre-trained version of "UNIVERSAL_V1" to synthesize a waveform from the predicted mel-spectrogram.

4.3 EVALUATION METRICS

Both subjective and objective tests were conducted in order to measure the performance of our proposed method. First of all, 20 volunteers participated in a subjective listening test over 15 synthesized audios in which they were asked to assess the level of naturalness and similarity of speech samples using a 5-scale mean opinion score (MOS) evaluation. 95% confidence intervals and p-value were provided with the MOS results.

For the objective evaluation, F0 frame error (FFE) (Chu & Alwan, 2009) and mel-cepstral distortion (MCD) (Kubichek, 1993) were used to measure the reconstruction performance of different VAEs and different settings of loss weights. FFE was used to assess the accuracy of the F0 track reconstruction by combining the Gross Pitch Error (GPE) and the Voicing Decision Error (VDE). In detail,

$$\begin{aligned} FFE &= \frac{\#\text{ of error frames}}{\#\text{ of total frames}} \times 100\% \\ &= \frac{N_{U\to V} + N_{V\to U} + N_{F0E}}{N} \times 100\%. \end{aligned}$$

where $N_{U\to V}$ and $N_{V\to U}$ are the numbers of unvoiced/voiced frames classified as voiced/unvoiced frames, N is the number of the frames in the utterance, and N_{F0E} is number of frames for which

$$\left|\frac{F0_{i,\text{ estimated}}}{F0_{i,\text{ reference}}} - 1\right| > 20\%$$

where i is the frame number.

Besides, MCD evaluated the timbral distortion, computing from the first 13 MFCCs in our trials.

$$\text{MCD}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \frac{10\sqrt{2}}{\ln 10}\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2 \quad \text{(dB)}$$

where $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ are the MFCCs of original and reconstructed waveform. The coefficient is used to convert MCD units into decibels. The smaller the MCD between the synthesized and natural mel-spectrogram sequences, the closer it is to natural speech.

Moreover, word error rates (WER) from an automatic speech recognition model was also reported. Complementary to naturalness, the WER metric demonstrated the degree of intelligibility and consistency between synthetic and real speech. The attention-based encoder-decoder model utilized in this study, which was trained on Librispeech 960-hour data, is open-sourced[1].

5 RESULTS

This section presents a series of experiments for our proposed speech editing system. First, the naturalness and similarity of synthesized audio generated by EditSpeech (Tan et al., 2021) and our system via both partial and entire inference were evaluated. Next, an ablation study was performed to progressively show the influence of restrictions on context information in our system, based on MOS and reconstruction performance. At last, the effect of the degree of biased training on reconstruction performance was also investigated. Our audio examples are available on the demo page [2].

5.1 PARTIAL VS. ENTIRE INFERENCE

To investigate the performance of partial inference versus entire inference, experiments were conducted on the following systems: 1) *GT*, the ground truth audio; 2) *GT (Mel+HifiGAN)*, first convert the ground truth audio to the ground truth mel-spectrogram, and then convert it back to audio using HifiGAN vocoder; 3) *Wave_cut*, manually cut the modified region from the generated waveform, and insert it back to the original waveform; 4) *EditSpeech (Tan et al., 2021)*, using partial inference and bidirectional fusion to improve the prosody near boundaries; 5) *Our system (Mel_cut)*, cut the modified region from the generated mel-spectrogram, and insert it back to the original mel-spectrogram with a forced aligner; 6) *Our system*, regenerate a complete mel-spectrogram from the whole sentence to be edited, and then use HifiGAN vocoder to generate the complete waveform;

Note that since ground truth audios do not include the edited audio, *GT* and *GT (Mel+HifiGAN)* are used to evaluate the reconstruction performance. For the editing operations, we manually spliced the audio waveform, and the MOS similarity score of *Wave_cut* serves as an upper bound indicator.

According to the MOS scores on naturalness shown in Table 1, our model with entire inference achieved the highest score on all the editing operations. The gap in replacement was noticeable, as the speech editing models based on partial reasoning have difficulty dealing with intonation conversion. The score of "Mel_cut" in deletion was relatively low since "Mel_Cut" is highly dependent on the accuracy of MFA. Especially when short words were deleted, its performance could be worse than manually careful deletion based on waveform. "Wave_cut" had a relatively lower naturalness MOS score in insertion and replacement since it involves the insertion of new words, and there is disharmony between the original audio and the generated audio.

MOS scores on similarity suggested that the performance of our system based on entire inference was close to partial inference "Mel_cut" and surpassed EditSpeech in insertion and replacement. It was also close to "Wave_cut", which is served as an upper bound indicator of similarity, with the maximum difference around 0.2.

[1] http://bitly.ws/uMKv
[2] http://bitly.ws/uMFd

Method	Insert		Replace		Delete	
	Nat.	Sim.	Nat.	Sim.	Nat.	Sim.
Wave_cut	2.93	3.76	2.82	3.50	3.25	3.82
EditSpeech (Mel_cut)	2.35	3.21	2.47	3.36	2.82	**3.81**
Our system (Mel_cut)	3.11	**3.57**	2.97	3.41	2.82	**3.81**
Our system	**3.37**	3.56	**3.39**	**3.43**	**3.37**	3.67

Table 1: Subjective naturalness and similarity results on EditSpeech and our system using partial and entire inference. Note that since the deletion operation of EditSpeech, which can only do partial inference, is to combine segments of the real mel-spectrogram, there is no difference in the results of different editing systems using partial inference.

The p-value in Table 2 presented that the naturalness of our model using entire inference was obviously superior to "Mel_cut" and "Wave_cut", while there was no significant difference in similarity between entire and the two partial inference methods. The only exception was in the case of deletion, where the naturalness of our model using entire inference was not significantly different from that of the "Wave_cut". Table 3 also demonstrated the capability of our mask CU-enhanced CVAE module to reconstruct the mel-spectrogram. Since partial inference directly copied the real mel-spectrogram of the unedited area, it is reasonable that partial inference had better reconstruction performance on similarity and MCD(mel-cepstral distortion). Nevertheless, our system using entire inference still surpassed EditSpeech on FFE and WER.

Method	Insert		Replace		Delete		Reconstruct	
	Nat.	Sim.	Nat.	Sim.	Nat.	Sim.	Nat.	Sim.
Our system vs. Mel_cut	0.0662	0.793	0.0294	0.771	0.0168	0.298	0.0525	0.691
Our system vs. Wave_cut	0.0219	0.135	0.0163	0.287	0.369	0.310	0.0564	0.143

Table 2: The significance analysis of our system using entire inference vs. "Mel_cut" and "Wave_cut" on naturalness and similarity MOS scores.

Method	Nat.	Sim.	**FFE**	**MCD**	**WER**
GT	4.56	-	-	-	3.124
GT (Mel+HifiGAN)	4.39	4.68	0.170	4.651	3.887
EditSpeech	3.14	3.80	0.372	6.345	6.702
Our system (Mel_cut)	3.66	**3.91**	**0.326**	**5.957**	**5.174**
Our system	**3.90**	3.83	0.327	6.657	5.377

Table 3: EditSpeech and our system's reconstruction performance using partial and entire inference. Lower objective results imply better similarity to the original audio.

Method	**FFE**	**MCD**	**WER**
GT	-	-	3.124
GT (Mel+HifiGAN)	0.170	4.651	3.887
Baseline1	0.371	6.919	7.404
Baseline2	0.333	6.750	5.503
Baseline3	0.332	6.697	5.392
Our system	**0.327**	**6.657**	**5.377**

Table 4: Objective metrics of the reconstruction performance of our system with different VAEs.

5.2 Ablation Study

In this section, we investigate the performance impact of using different VAEs in our system. We compare the reconstruction performance and MOS scores of the synthesized audio among the following systems: 1-2) same as the above experimental settings; 3) *Baseline1*, use a fine-grained VAE instead of CU-CVAE; 4) *Baseline2*, use a CVAE without the context embeddings, i.e.$l = 0$; 5) *Baseline2*, use CU-CVAE with 2 neighbouring utterances, i.e.$l = 2$; 6) *Our system*, use CU-CVAE with 5 neighbouring utterances, i.e.$l = 5$.

As shown in Table 4 and 5,when semantic restriction of edited text and context embeddings were added in consecutively, both the MOS scores on naturalness of edited and reconstructed waveforms and objective scores on reconstruction performance progressively increased. Also, the reconstruction metrics in Table 4 suggest that using more cross-utterances can improve the reconstruction capability. These results indicated that the CU-embedding and mask CU-CVAE module played a crucial role in generating more coherent audio.

Method	Insert	Replace	Delete	Reconstruct
GT	-	-	-	4.56±0.06
GT (Mel+HifiGAN)	-	-	-	4.39±0.05
Baseline1	2.91±0.11	2.87±0.10	3.31±0.15	2.90±0.08
Baseline2	3.43±0.12	3.11±0.09	3.52±0.14	3.45±0.10
Our system	**3.93±0.10**	**3.43±0.13**	**4.29±0.12**	**3.90±0.10**

Table 5: Subjective results of editing performance of our system with different VAEs. Baseline1 adds a fine-grained VAE to FastSpeech 2. Baseline2 represents a CVAE editing system without context embeddings.

5.3 Degree of Biased Training

The ablation study in this section was conducted based on reconstruction performance. To investigate the effect of the different intensity of attention to the masked mel-spectrogram, our system trained with ratio=1:1,1:1.5,1:2,1:5, and 0:1 were evaluated, where ratio=1: 1 implied using the normal loss function of the reconstructed mel-spectrogram, treating the masked and unmasked areas equally, and ratio=0:1 is applied by the existing speech editing systems which all use partial inference, only focusing on the masked areas. The purpose of this experiment is to find a parameter in the above two scenarios, namely, common TTS overall reasoning and partial reasoning of existing speech editing systems, so that the generated edited audio can be coherent and have the prosody characteristics of the original audio at the same time.

Method	Loss ratio (unmasked:masked part)	**FFE**	**MCD**	**WER**
Our system	1:1	0.437	6.931	5.406
	1:1.5	0.327	**6.657**	**5.377**
	1:2	**0.326**	6.697	**5.377**
	1:5	0.434	6.824	5.525
	0:1	0.457	7.596	14.808

Table 6: Reconstruction performance of our system trained in different loss ratio.

Table 6 shows system trained with ratio=1:1.5 achieved the smallet MCD and WER results, with FFE score slightly higher than system trained with ratio=1:2. The results demonstrates that properly paying more attention to the masked mel-spectrogram can effectively improve the overall quality of generated audio.

6 Conclusion

In this paper, we propose a cross-utterance conditioned coherent speech editing system, which is the first text-based speech editing system that can entirely generate audio corresponding to the edited transcript. A variational autoencoder conditioned on speaker information, context, and audio prior is integrated into a high-quality text-to-speech model to ensure both the restoration and generation quality of audio. Experiments show that our proposed system has the ability to reconstruct the acoustic characteristics of original audio with high fidelity and that the prosody of the synthesized speech conforms to the context of the edited transcript.

7 Ethics Statement

The experiments in this paper were carried out under the assumption that the user of the model is the target speaker and has been approved by the speaker. However, when the model is generalized to unseen speakers, relevant components should be accompanied by speech editing models, including the protocol to ensure that the speaker agrees to execute the modification and the system to detect the edited speech.

REFERENCES

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html.

He Bai, Renjie Zheng, Junkun Chen, Mingbo Ma, Xintong Li, and Liang Huang. A^3t: Alignment-aware acoustic and text pretraining for speech synthesis and editing. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 1399–1411. PMLR, 2022. URL https://proceedings.mlr.press/v162/bai22d.html.

Zalan Borsos, Matthew Sharifi, and Marco Tagliasacchi. Speechpainter: Text-conditioned speech inpainting. In Hanseok Ko and John H. L. Hansen (eds.), *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, pp. 431–435. ISCA, 2022. doi: 10.21437/Interspeech.2022-194. URL https://doi.org/10.21437/Interspeech.2022-194.

Wei Chu and Abeer Alwan. Reducing F0 frame error of F0 tracking algorithms under noisy conditions with an unvoiced/voiced classification frontend. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2009, 19-24 April 2009, Taipei, Taiwan*, pp. 3969–3972. IEEE, 2009. doi: 10.1109/ICASSP.2009.4960497. URL https://doi.org/10.1109/ICASSP.2009.4960497.

Roger Derry. *PC audio editing with Adobe Audition 2.0: Broadcast, desktop and CD audio production*. Routledge, 2012.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423. URL https://doi.org/10.18653/v1/n19-1423.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. *CoRR*, abs/2111.06377, 2021. URL https://arxiv.org/abs/2111.06377.

Zeyu Jin, Gautham J. Mysore, Stephen DiVerdi, Jingwan Lu, and Adam Finkelstein. Voco: text-based insertion and replacement in audio narration. *ACM Trans. Graph.*, 36(4):96:1–96:13, 2017. doi: 10.1145/3072959.3073702. URL https://doi.org/10.1145/3072959.3073702.

Hideki Kawahara. Straight, exploitation of the other aspect of vocoder: Perceptually isomorphic decomposition of speech sounds. *Acoustical Science and Technology*, 27(6):349–353, 2006. doi: 10.1250/ast.27.349.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/c5d736809766d46260d816d8dbc9eb44-Abstract.html.

R. Kubichek. Mel-cepstral distance measure for objective speech quality assessment. In *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing*, volume 1, pp. 125–128 vol.1, 1993. doi: 10.1109/PACRIM.1993.407206.

Yang Li, Cheng Yu, Guangzhi Sun, Hua Jiang, Fanglei Sun, Weiqin Zu, Ying Wen, Yang Yang, and Jun Wang. Cross-utterance conditioned VAE for non-autoregressive text-to-speech. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pp. 391–400. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-long.30. URL https://doi.org/10.18653/v1/2022.acl-long.30.

Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. Montreal forced aligner: Trainable text-speech alignment using kaldi. In Francisco Lacerda (ed.), *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, pp. 498–502. ISCA, 2017. URL http://www.isca-speech.org/archive/Interspeech_2017/abstracts/1386.html.

Masanori Morise, Fumiya Yokomori, and Kenji Ozawa. WORLD: A vocoder-based high-quality speech synthesis system for real-time applications. *IEICE Trans. Inf. Syst.*, 99-D(7):1877–1884, 2016. doi: 10.1587/transinf.2015EDP7457. URL https://doi.org/10.1587/transinf.2015EDP7457.

Max Morrison, Lucas Rencker, Zeyu Jin, Nicholas J. Bryan, Juan Pablo Cáceres, and Bryan Pardo. Context-aware prosody correction for text-based speech editing. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*, pp. 7038–7042. IEEE, 2021. doi: 10.1109/ICASSP39728.2021.9414633. URL https://doi.org/10.1109/ICASSP39728.2021.9414633.

Eric Moulines and Francis Charpentier. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. *Speech Commun.*, 9(5-6):453–467, 1990. doi: 10.1016/0167-6393(90)90021-Z. URL https://doi.org/10.1016/0167-6393(90)90021-Z.

Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech 2: Fast and high-quality end-to-end text to speech. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=piLPYqxtWuA.

Jiaqi Su, Zeyu Jin, and Adam Finkelstein. Hifi-gan: High-fidelity denoising and dereverberation based on speech deep features in adversarial networks. In Helen Meng, Bo Xu, and Thomas Fang Zheng (eds.), *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pp. 4506–4510. ISCA, 2020. doi: 10.21437/Interspeech.2020-2143. URL https://doi.org/10.21437/Interspeech.2020-2143.

Daxin Tan, Liqun Deng, Yu Ting Yeung, Xin Jiang, Xiao Chen, and Tan Lee. Editspeech: A text based speech editing system using partial inference and bidirectional fusion. In *IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2021, Cartagena, Colombia, December 13-17, 2021*, pp. 626–633. IEEE, 2021. doi: 10.1109/ASRU51503.2021.9688051. URL https://doi.org/10.1109/ASRU51503.2021.9688051.

Tao Wang, Jiangyan Yi, Ruibo Fu, Jianhua Tao, and Zhengqi Wen. Campnet: Context-aware mask prediction for end-to-end text-based speech editing. *CoRR*, abs/2202.09950, 2022. URL https://arxiv.org/abs/2202.09950.

Zhenda Xie, Zigang Geng, Jingcheng Hu, Zheng Zhang, Han Hu, and Yue Cao. Revealing the dark secrets of masked image modeling. *CoRR*, abs/2205.13543, 2022. doi: 10.48550/arXiv.2205.13543. URL https://doi.org/10.48550/arXiv.2205.13543.

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J. Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. Libritts: A corpus derived from librispeech for text-to-speech. In Gernot Kubin and Zdravko Kacic (eds.), *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, pp. 1526–1530. ISCA, 2019. doi: 10.21437/Interspeech.2019-2441. URL https://doi.org/10.21437/Interspeech.2019-2441.