

FluentEditor2: Text-based Speech Editing by Modeling Multi-Scale Acoustic and Prosody Consistency

Rui Liu, *Member, IEEE*, Jiatian Xi, Ziyue Jiang and Haizhou Li *Fellow, IEEE*

Abstract—Text-based speech editing (TSE) allows users to edit speech by modifying the corresponding text directly without altering the original recording. Current TSE techniques often focus on minimizing discrepancies between generated speech and reference within edited regions during training to achieve fluent TSE performance. However, the generated speech in the edited region should maintain acoustic and prosodic consistency with the unedited region and the original speech at both the local and global levels. To maintain speech fluency, we propose a new fluency speech editing scheme based on our previous *FluentEditor* model, termed *FluentEditor2*, by modeling the multi-scale acoustic and prosody consistency training criterion in TSE training. Specifically, for local acoustic consistency, we propose *hierarchical local acoustic smoothness constraint* to align the acoustic properties of speech frames, phonemes, and words at the boundary between the generated speech in the edited region and the speech in the unedited region. For global prosody consistency, we propose *contrastive global prosody consistency constraint* to keep the speech in the edited region consistent with the prosody of the original utterance. Extensive experiments on the VCTK and LibriTTS datasets show that *FluentEditor2* surpasses existing neural networks-based TSE methods, including Editspeech, Campnet, A³T, FluentSpeech, and our Fluenteditor, in both subjective and objective. Ablation studies further highlight the contributions of each module to the overall effectiveness of the system. Speech demos are available at: <https://github.com/Ai-S2-Lab/FluentEditor2>.

Index Terms—Speech Editing, Speech Fluency, Multi-Scale, Acoustic and Prosody Consistency

I. INTRODUCTION

TEXT-BASED SPEECH EDITING (TSE) [1] allows for modification of the audio by editing the transcript rather than the audio itself [1]. With the rapid development of the

The research by Rui Liu was funded by the Young Scientists Fund (No. 62206136) and the General Program (No. 62476146) of the National Natural Science Foundation of China, Guangdong Provincial Key Laboratory of Human Digital Twin (No. 2022B121201 0004), and the “Inner Mongolia Science and Technology Achievement Transfer and Transformation Demonstration Zone, University Collaborative Innovation Base, and University Entrepreneurship Training Base” Construction Project (Supercomputing Power Project) (No.21300-231510). The research by Haizhou Li was partly supported by Internal Project Fund from Shenzhen Research Institute of Big Data (Grant No. T00120220002), and Shenzhen Science and Technology Research Fund (Fundamental Research Key Project Grant No. JCYJ20220818103001002).

Rui Liu and Jiatian Xi are with the Department of Computer Science, Inner Mongolia University, Hohhot 010021, China. (e-mail: liurui_imu@163.com).

Ziyue Jiang is with the Zhejiang University, Hangzhou, China. (e-mail: ziyuejiang@zju.edu.cn).

Haizhou Li is with School of Data Science, The Chinese University of Hong Kong, Shenzhen 518172, China. He is also with University of Bremen, Faculty 3 Computer Science / Mathematics, Enrique-Schmidt-Str. 5 Cartesium, 28359 Bremen, Germany (e-mail: haizhouli@cuhk.edu.cn).

internet, audio-related media sharing has become a prevalent activity in our daily lives. Note that TSE can bring great convenience to the audio generation process and be applied to a variety of areas with personalized voice needs, including video creation for social media, games, and movie dubbing [2], [3].

The traditional TSE approach follows the following formal definition: Given a reference audio sample A_{ref} along with its corresponding textual transcript T_{ref} , aligned using a Montreal Forced Aligner (MFA) [4], the TSE task involves converting the edited text $T_{\text{edited_text}}$ into the corresponding audio sequence $A_{\text{edited_audio}}$. The objective is to ensure the final edited audio A_{edited} remains imperceptibly close to the original A_{ref} , preserving the naturalness and fluidity of the speech. During inference, the resulting audio segment is combined with the unedited portion of the reference audio, resulting in a natural and fluent output [5]. Please note that the critical challenge for TSE lies in ensuring a fluent transition [6] at the editing boundaries between A_{ref} and A_{edited} . In other words, fluent transitions are essential to avoid perceptible artifacts that may distract the listener, especially in the editing region. Discontinuities at the boundary can degrade the overall audio quality. Therefore, it is crucial to achieve smooth transitions at the boundary between the editing area and the non-editing area to maintain high-quality audio output [7].

In recent years, many efforts have been made to build neural networks-based TSE models inspired by neural text-to-speech (TTS) models. For example, in EditSpeech [8], the authors divided the given speech utterance into “to-modify” and “non-modify” regions according to the edited text and speech-text alignment, and generated the new “modified” speech frames using a duration based auto-regressive neural TTS conditioned on the “non-modify” frames. The CampNet [9] conducted mask training on a context-aware neural network based on Transformer to improve the quality of the edited voice. A³T [10] suggested 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. More recently, the diffusion model has gradually become the backbone of the neural networks-based TSE with remarkable results. For example, EdiTTS [11], [12] takes the diffusion-based TTS model as the backbone and proposes a score-based TSE methodology for fine-grained pitch and content editing. FluentSpeech [5] proposes a context-aware diffusion model that iteratively refines the modified mel-spectrogram with the guidance of

context features.

However, during training, existing TSE approaches primarily rely on constraining the Euclidean Distance [13] between the predicted and ground truth mel-spectrogram to ensure the naturalness of the edited speech. While they incorporate contextual information [5], [9] to mitigate the over-smoothing problem, their objective functions are not specifically designed to ensure a more refined and fluent speech output [14], [15]. Therefore, it is imperative to seek out more efficient speech fluency modeling in order to achieve seamless and fluent transitions in the editing boundaries.

In this study, we identify two significant challenges that must be addressed to model speech editing fluency effectively, which are ***Local Acoustic Consistency*** and ***Global Prosody Consistency***. These two challenges are called **multi-scale consistency**.

- 1) ***Local Acoustic Consistency***: The smoothness of the transition between the edited region and its neighboring segments must resemble real concatenation points. More important, such smoothness should be expressed in the hierarchical structure [16], which includes frames, phonemes, and words, of the acoustic signals [17]. This ensures natural transitions throughout the edited speech.
- 2) ***Global Prosody Consistency***: The prosodic style of the synthesized audio in the edited region should remain consistent with the overall prosody of the original utterance [18], [19].

To address these challenges, we propose *FluentEditor2*, a novel speech editing framework that introduces new training criteria for the multi-scale acoustic and prosody consistency. Specifically, *FluentEditor2* introduces two major innovations: 1) ***Hierarchical Local Acoustic Smoothness Consistency Constraint*** (\mathcal{L}_{HLAC}), which computes the concatenation costs [20] at multiple granularities (frame, phoneme, and word), ensuring that the *acoustic information* at the editing boundaries closely resembles real concatenation points across these hierarchical levels. Note that the concatenation cost is inspired by the traditional unit selection TTS [20]–[24]. We know that the main purpose of unit selection TTS is to stitch the selected multiple acoustic units into a smooth continuous speech fragment [21], [23], [24]. The concatenation cost is the key to achieving this goal, ensuring a smooth transition between different acoustic units at concatenation points or boundaries [22], [25]. Note that to achieve the hierarchical constraint of \mathcal{L}_{HLAC} at multiple granularities (frame, phoneme, and word), we adopt the new word-level masking strategy instead of the random frame masking [5], [9] in traditional TSE methods. 2) ***Contrastive Global Prosody Consistency Constraint*** (\mathcal{L}_{CGPC}), which employs a contrastive learning mechanism [26] to ensure that the prosodic features of the edited region remain consistent with the surrounding context of the original utterance while distinguishing it from other utterances in the same batch.

Subjective and objective evaluations on the VCTK [27] and LibriTTS [12] datasets show that *FluentEditor2* significantly outperforms the previous *FluentEditor* and all state-of-the-art TSE baselines in both acoustic and prosody consistency

while maintaining a fluency performance that is nearly indistinguishable from natural speech.

While this work shares a similar motivation with our previous *FluentEditor* work [6] in terms of acoustic and prosody consistency modeling, it is different in many ways. 1) For local acoustic consistency, *FluentEditor* just considers the frame-level consistency constraint, while overlooks the phoneme- and word-level consistency; 2) To support the *Hierarchical Local Acoustic Smoothness Consistency Constraint*, *FluentEditor2* adopt the new word-level masking strategy, while *FluentEditor* employ the random frame masking; 3) For global prosody consistency, *FluentEditor* just adopt Mean Squared Error (MSE) [28] loss to pull close the prosody feature between the edited region and the original speech, while *FluentEditor2* propose a novel contrastive learning-based loss term to implement a stronger constraint; 4) From an experimental perspective, *FluentEditor2* conduct a more comprehensive and detailed subjective and objective experiment on more benchmarking datasets, including VCTK and LibriTTS. In a nutshell, the main contributions of this work can be summarized as follows:

- We propose a novel fluent TSE scheme, termed *FluentEditor2*. To the best of our knowledge, this is the first attempt to consider the multi-scale acoustic and prosody consistency training constraint for TSE task.
- We propose *Hierarchical Local Acoustic Smoothness Consistency* (\mathcal{L}_{HLAC}) and *Contrastive Global Prosody Consistency* (\mathcal{L}_{CGPC}) constraints to conduct the fluency modeling for TSE. Note that the \mathcal{L}_{HLAC} takes into account multiple granularities such as frames, phonemes, and words simultaneously.
- Comprehensive subjective and objective experimental validate that the proposed *FluentEditor2* outperforms all advanced TSE baselines in terms of naturalness and fluency.

The rest of this paper is organized as follows. Section II reviews related work. Section III introduces the proposed method. Section IV describes the experiments and setup in detail. Section V presents the experimental results and discusses the performance. Finally, the paper concludes in Section VI.

II. RELATED WORKS

A. Speech Editing

Traditional speech editing methods often rely on digital signal processing (DSP) techniques applied directly to audio [29]–[32]. To simplify user interaction, text-based speech editing (TSE) systems were developed, allowing users to insert, delete, or replace speech by modifying the text directly. However, these methods often struggle to match the prosody of the edited regions with the surrounding speech. To address this, Morrison [7] introduced a method that predicts duration and pitch based on surrounding context and uses the TD-PSOLA algorithm [33] for prosody modification. While this improves fluency, it cannot generate new words not found in the original transcript. More recently, neural TTS approaches have shown promise [1], [34]–[37]. Editspeech [8] preserves

coherent prosody by employing bidirectional neural models, while CampNet [9] uses cross-attention to better model text-audio relationships, though it suffers from slow convergence. A³T [10] improves speech quality with alignment-aware pretraining integrated into Conformer models [38], [39]. Diffusion models [40]–[42] have emerged as a powerful tool for TSE. EdiTTS [11] and FluentSpeech [5] leverage diffusion-based techniques to refine prosody and smooth transitions in edited speech. AttentionStitch [43] combines pre-trained TTS models with attention mechanisms to enhance boundary stitching.

Despite these advancements, many current approaches focus on modifying the underlying model architectures, often overlooking the importance of designing specific loss functions that directly target prosodic fluency and coherence. This work identifies two significant challenges that are Local Acoustic Consistency and Global Prosody Consistency and proposes two new Fluency-Aware Training criterions.

B. Multi-Scale Acoustic Attributes

The speech signal is structured by different basic units, from fine to coarse, which are frames, phonemes and words [44]. This natural structure is unique to speech and contains extensive paralinguistic information such as fluency, articulation, prolongation and rhythm [45]. To this end, most works are explored toward hierarchical multi-granularity learning [46] to improve system performance. For example, in speech emotion recognition, SpeechFormer [47] introduced a hierarchical efficient framework that incorporates speech characteristics for better emotion recognition, leveraging multi-scale feature extraction across different linguistic levels. In speech synthesis, MsEmoTTS [48] also leveraged multi-scale emotional control to enhance expressiveness, enabling more precise and nuanced emotion representation. Hono et al. [49] proposed a hierarchical multi-grained generative model to enhance expressiveness. Li et al. [50] focused on multi-scale style control for improved synthesis. Ren et al. [51] introduced PortaSpeech, a lightweight, high-quality generative model. It proposed a linguistic encoder with mixture alignment combining hard inter-word alignment and soft intra-word alignment, which explicitly extracts Word-to-Phoneme semantic information. Lei et al. [52] developed MSStyleTTS for enhancing coherence using multi-scale style features. More recently, Jiang et al. [53] investigated hierarchical prosody modeling for zero-shot synthesis, improving nuanced prosodic control.

Inspired by the above works, we consider the hierarchical acoustic smoothing consistency in the TSE task. We believe that speech frames, phonemes, and words at the splice between edited and unedited speech should exhibit acoustic smoothness, and such hierarchical speech modeling is helpful for better learning speech fluency.

C. Contrastive Learning

1) Basic knowledge of contrastive learning: Contrastive learning [54]–[56] aims to train an encoder to produce similar representations for data instances that are alike while ensuring

that representations for dissimilar instances remain as distinct as possible [54]. This framework is grounded in the idea that by learning to differentiate between similar and dissimilar data, the encoder can capture meaningful patterns in the data space [54].

Specifically, Chen et al. [55] suggested that training the relative relationships within a sample space—specifically, contrastive relationships between pairs of data—is sufficient to represent the vectors in that space. The contrastive loss function encourages the model to push similar samples closer and dissimilar samples farther apart. The loss function is defined as:

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(z_i, z_k)/\tau)} \quad (1)$$

where $\text{sim}(z_i, z_j)$ represents the cosine similarity between two latent vectors z_i and z_j , and τ is a temperature parameter controlling the sharpness of the distribution. The indicator function $\mathbb{1}_{[k \neq i]}$ ensures that only distinct pairs are considered, excluding self-pairs.

In summary, contrastive learning provides a framework to train encoders to learn discriminative representations by maximizing the similarity between similar data instances and minimizing the similarity between dissimilar ones. This is achieved through the optimization of a contrastive loss function, which drives the model to capture the underlying structure of the data space.

2) Applications of contrastive learning in speech tasks: Contrastive learning has achieved considerable success in computer vision, particularly with the introduction of the SimCLR framework [55], which learns visual representations by maximizing the similarity between positive pairs. SimCLRV2 [57] further improves performance by employing advanced data augmentation techniques. In the field of speech processing, contrastive learning is still an emerging area of research, but it has shown promising results. Early works such as [58]–[61] have demonstrated that contrastive learning can be effective in tasks like speech representation learning, adversarial training, and data augmentation for speech models. More recent advancements have focused on learning style representations from text data. For instance, [62] explores using contrastive learning to derive style representations from large text corpora. Similarly, CALM [63] leverages contrastive learning to optimize the correlation between stylistic text features, enhancing the generation of stylistically consistent speech. Furthermore, DCTTS [64] introduces a discrete diffusion model with contrastive learning to improve alignment between text and speech, which in turn enhances sampling rates and the overall quality of synthesized speech.

In this work, we introduce contrastive learning into the TSE task, which forces the prosodic features of the edited regional speech to be close to the global prosody of the original speech and far from other samples.

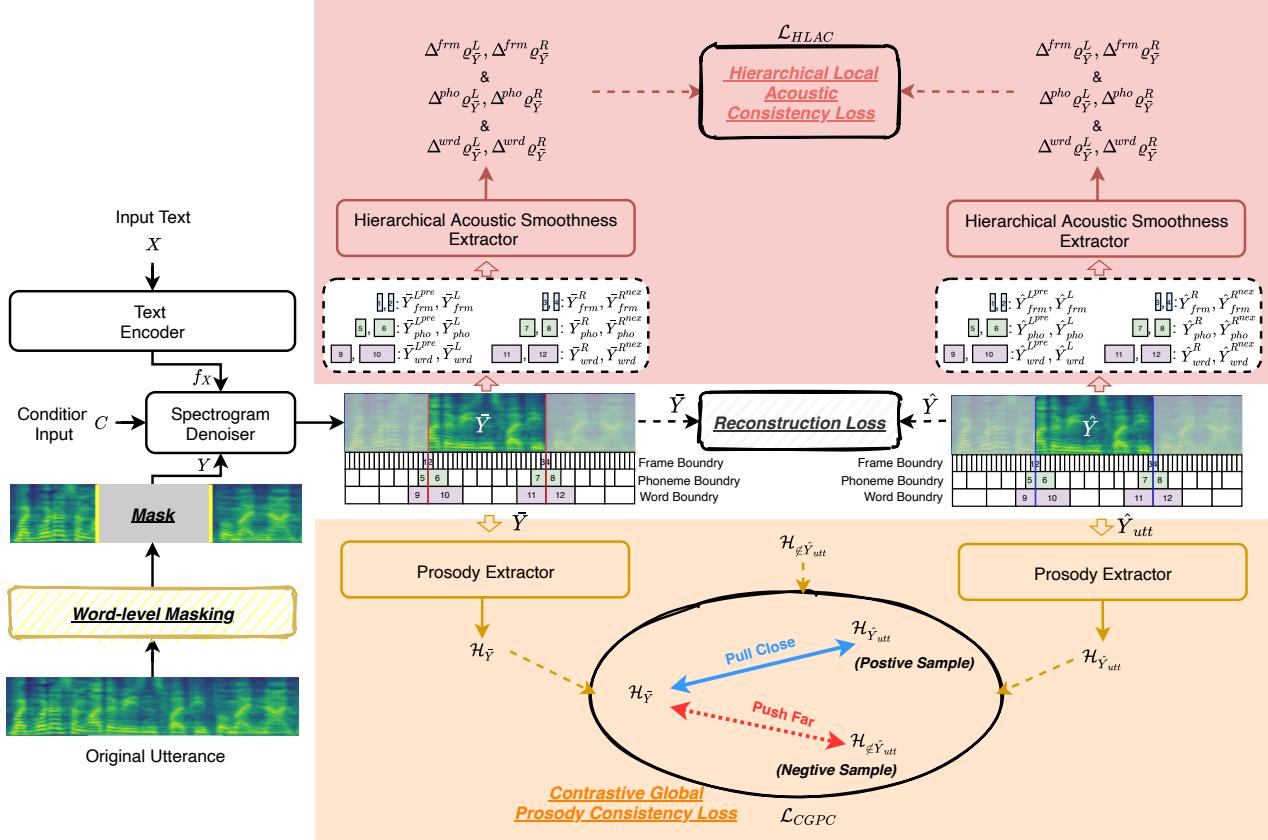


Fig. 1: The overall workflow of FluentEditor2. The total loss function comprises Reconstruction Loss, and Local Hierarchical Acoustic Smoothness and Contrastive Global Prosody Consistency Losses.

III. FLUENTEDITOR2: METHODOLOGY

A. Overview Architecture

As shown in Fig.1, our FluentEditor2 adopts the mask prediction-based diffusion network as the backbone, which consists of a text encoder, and a spectrogram denoiser. The spectrogram denoiser seeks to adopt the Denoising diffusion probabilistic model (DDPM) to learn a data distribution $p(\cdot)$ by gradually denoising a normally distributed variable through the reverse process of a fixed Markov Chain of length T . Assume that the input phoneme sequence of the input phoneme sequence is $X = (X_1, \dots, X_{|X|})$, from which the text encoder extracts the frame-level text embedding $f_X = (f_{X_1}, f_{X_2}, \dots, f_{X_{|X|}})$. The masked acoustic feature sequence $Y = \text{Mask}(f_X, \lambda)$ is generated by replacing entire word-level spans of f_X with random vectors based on a probability λ . The spectrogram denoiser then aggregates several inputs: the text embedding f_X , the masked acoustic feature Y , and the condition input C to guide the reverse process of the diffusion model $\Theta(Y_t|t, C)$, where $t \in T$, and Y_t is a noisy version of the clean input \hat{Y}_{utt} . Similar to [5], the condition input C includes the frame-level text embedding f_X , the acoustic feature sequence \hat{Y}_{utt} , the masked acoustic feature sequence Y , the speaker embedding e_{spk} , and the pitch embedding e_{pitch} . In the generator-based diffusion models, $p_\theta(\hat{Y}_{utt}|Y_t)$ is the implicit distribution imposed by the neural network $f_\theta(Y_t, t)$ that outputs \hat{Y} given Y_t . And then Y_{t-1} is sampled using the posterior distribution $q(Y_{t-1}|Y_t, \hat{Y})$ given Y_t and the

predicted \hat{Y} .

To model speech fluency, we design *Hierarchical Local Acoustic Smoothness Consistency loss* \mathcal{L}_{HLAC} and *Contrastive Global Prosody Consistency Loss* \mathcal{L}_{CGPC} separately on the basis of the original *reconstruction loss*, to ensure that the acoustic and prosody performance of speech generated in the editing area is consistent with the context and the original utterance. For reconstruction loss, we follow [5] and employ Mean Absolute Error (MAE) and the Structural Similarity Index (SSIM) [65] losses to calculate the difference between \hat{Y} and the corresponding ground segment \hat{Y} .

Hierarchical Local Acoustic Smoothness Consistency loss \mathcal{L}_{HLAC} and *Contrastive Global Prosody Consistency Loss* \mathcal{L}_{CGPC} serve the purposes of ensuring the smoothness of connecting points between edited speech regions and neighboring acoustic segments, as well as ensuring that the local prosody of the masked speech aligns with the global prosody of the ground truth.

In the following subsection, we will first introduce the key operations, the word-level masking strategy, for implementing \mathcal{L}_{HLAC} and dive deeper into \mathcal{L}_{HLAC} and \mathcal{L}_{CGPC} in detail, building upon the underlying training principles of acoustic and prosody consistency losses.

B. Word-level Masking Strategy

Traditional speech editing systems often adopt random frame-level masking strategies [5], [9] to mask speech

segments and reconstruct them during training. However, the random frame-level masking often disrupts phoneme or word boundaries, leading to unnatural transitions and misalignment. In addition, to support the calculation of multi-granularity acoustic smoothness at the boundary between edited and non-edited speech segments, we propose a *Word-level Masking (WLM) Strategy*. Specifically, we first obtain the “speech frame-phoneme-word” alignment with the help of the MFA (More details please refer to Section IV-C.), and then mask all the speech frames corresponding to several consecutive words according to the word timestamp.

In this way, the Hierarchical structure integrity of the mask regions area and the non-mask regions can be maintained, and more natural speech editing can be achieved.

C. Fluency-Aware Training Criterion

The FluentEditor2 model introduces novel loss functions aimed at improving the fluency of edited speech by ensuring both Hierarchical Local Acoustic Smoothness and Contrastive Global Prosody Consistency. These two complementary mechanisms ensure fluent transitions between edited and unedited regions, even in cases involving substantial modifications. By addressing acoustic smoothness at multiple hierarchical levels (frame, phoneme, and word), while maintaining global prosodic coherence, the model produces more fluent and coherent speech. This process is key to fluency-aware training, where both aspects are jointly optimized.

1) Hierarchical Local Acoustic Smoothness Consistency Loss: Unit-selection TTS systems traditionally rely on two critical loss functions: *Target Loss* and *Concatenation Loss* [21], [23], [24]. The Target Loss evaluates the proximity of candidate units to the target [24], while the Concatenation Loss [21] measures the smoothness between concatenated speech units.

Building upon the principles of *Concatenation Loss* [21], we propose the *Hierarchical Local Acoustic Smoothness Consistency Loss* (\mathcal{L}_{HLAC}). This loss ensures smooth transitions across multiple hierarchical levels: frame, phoneme, and word, between the edited regions and their surrounding context. Thus, the \mathcal{L}_{HLAC} loss is defined as an integration of frame-level, phoneme-level, and word-level smoothness:

$$\mathcal{L}_{HLAC} = \mathcal{L}_{AC(frm)} + \mathcal{L}_{AC(pho)} + \mathcal{L}_{AC(wrd)} \quad (2)$$

Take frame-level as an example, $\mathcal{L}_{AC(frm)}$ consists of two components, with L and R denoting the left and right boundaries, respectively:

$$\mathcal{L}_{AC(frm)} = \mathcal{L}_{AC(frm)}^L + \mathcal{L}_{AC(frm)}^R \quad (3)$$

Next, we employ the *Mean Squared Error* (MSE) [66] to quantify the proximity between the predicted \bar{Y} and the ground truth \hat{Y} in terms of the Euclidean distance:

$$\begin{aligned} \mathcal{L}_{AC(frm)}^L &= \text{MSE}(\Delta^{frm} \varrho_{\bar{Y}}^L, \Delta^{frm} \varrho_{\hat{Y}}^L) \\ \mathcal{L}_{AC(frm)}^R &= \text{MSE}(\Delta^{frm} \varrho_{\bar{Y}}^R, \Delta^{frm} \varrho_{\hat{Y}}^R) \end{aligned} \quad (4)$$

Note that $\Delta \varrho_{\bar{Y}}^L$ and $\Delta \varrho_{\bar{Y}}^R$ represent the Euclidean distances between adjacent frames, extracted by the Hierarchical Acous-

tic Smoothness Extractor. For example, illustrated in Fig.1, the blue boxes (1-4) represent frame-level units. $\Delta^{frm} \varrho_{\bar{Y}}^L$ represents the difference between the Mel-spectrogram features at the left boundary \hat{Y}_{frm}^L (box 2) and those in the preceding frame $\hat{Y}_{\text{frm}}^{L\text{pre}}$ (box 1). A smaller difference in these Euclidean distance values indicates smoother transitions [67].

$$\Delta^{frm} \varrho_{\bar{Y}}^L = |\theta(\bar{Y}_{\text{frm}}^L) - \theta(\bar{Y}_{\text{frm}}^{L\text{pre}})| \quad (5)$$

Similarly, $\Delta^{frm} \varrho_{\bar{Y}}^R$ represents the difference between the Mel-spectrogram features at the right boundary \hat{Y}_{frm}^R (box 3) and those in the following frame $\hat{Y}_{\text{frm}}^{R\text{nex}}$ (box 4). Smoothness constraints are applied at the right boundary using \bar{Y}_{frm}^R (box 3) from the masked region and $\bar{Y}_{\text{frm}}^{R\text{nex}}$ (box 4) from the non-masked region, which is the frame immediately following the edited region. This ensures coherence on both sides of the edited segment.

$$\Delta^{frm} \varrho_{\bar{Y}}^R = |\theta(\bar{Y}_{\text{frm}}^{R\text{nex}}) - \theta(\bar{Y}_{\text{frm}}^R)| \quad (6)$$

Please note that θ for frame-level means the Euclidean distance is directly calculated using the Mel-spectrogram acoustic features from adjacent frames. For phoneme and word levels, θ represents a function to compute the average value across the multiple frames that make up a phoneme or word. This ensures smoothness is evaluated based on aggregated frame information for these levels.

For phoneme-level and word-level consistency loss functions $\mathcal{L}_{AC(pho)}$ and $\mathcal{L}_{AC(wrd)}$, we similarly apply smoothness constraints between phoneme or word boundaries to ensure smooth transitions across consecutive phonemes or words respectively. In Fig.1, the green boxes (5-8) indicate phoneme-level units, while the purple boxes (9-12) represent word-level units, showing the application of smoothness constraints at each level. The smoothness loss functions at both levels are defined as:

$$\begin{aligned} \mathcal{L}_{AC(pho)} &= \mathcal{L}_{AC(pho)}^L + \mathcal{L}_{AC(pho)}^R \\ \mathcal{L}_{AC(wrd)} &= \mathcal{L}_{AC(wrd)}^L + \mathcal{L}_{AC(wrd)}^R \end{aligned} \quad (7)$$

2) Contrastive Global Prosody Consistency Loss: The Contrastive Global Prosody Consistency Loss (\mathcal{L}_{CGPC}) ensures that the prosody features in the predicted masked region \bar{Y} align with those of the original speech \hat{Y} . We achieve this by comparing high-level prosody representations $\mathcal{H}_{\bar{Y}}$ and $\mathcal{H}_{\hat{Y}}$, which are obtained through a pre-trained prosody extractor based on the *Global Style Token (GST)* model [68].

To enhance the model’s ability to predict masked regions while preserving global prosody coherence, a *contrastive learning mechanism* is integrated into the \mathcal{L}_{CGPC} loss. This mechanism pulls the prosody of the masked region closer to that of the full utterance (positive sample) and pushes it away from other utterances in the batch (negative samples). The pull-close operation ensures that the masked region aligns with the overall prosody of the current utterance, while the push-far operation enforces distinction from unrelated utterances, ensuring both fluency and coherence.

For each batch \mathcal{B} with b samples, where i refers to the index of the current utterance, positive samples $\mathcal{H}_{\bar{Y}_{\text{utt}_i}}$ are defined as

the prosody features of the entire utterance. Negative samples $\mathcal{H}_{\notin \hat{Y}_{utt_i}}$ represent prosodic features from other utterances in the batch. The \mathcal{L}_{CGPC} loss loss is formulated as follows:

$$\begin{aligned} \mathcal{L}_{CGPC} = & \sum_{i=1}^b -\log \frac{\exp(\text{sim}(\mathcal{H}_{\bar{Y}}, \mathcal{H}_{\hat{Y}_{utt_i}})/\tau)}{\exp(\text{sim}(\mathcal{H}_{\bar{Y}}, \mathcal{H}_{\hat{Y}_{utt_i}})/\tau)} \\ & + \sum_{k \neq i} \exp(\text{sim}(\mathcal{H}_{\bar{Y}}, \mathcal{H}_{\notin \hat{Y}_{utt_k}})/\tau) \end{aligned} \quad (8)$$

Here, $\text{sim}(\cdot, \cdot)$ represents the cosine similarity between the prosody features of the masked region $\mathcal{H}_{\hat{Y}_{utt_i}}$ and the prosody features of both the entire utterance (positive sample) and other unrelated utterances (negative samples), while τ is the temperature parameter that controls the sharpness of the contrast between positive and negative samples. The similarity is computed as:

$$\text{sim}(\mathcal{H}_{\bar{Y}}, \mathcal{H}_{\hat{Y}_{utt_i}}) = \frac{\mathcal{H}_{\bar{Y}} \cdot \mathcal{H}_{\hat{Y}_{utt_i}}}{\|\mathcal{H}_{\bar{Y}}\| \|\mathcal{H}_{\hat{Y}_{utt_i}}\|} \quad (9)$$

Here, c_{ij} denotes the similarity between prosody features of the masked region $v_i^{[M]}$ and other samples v_j , while τ is the temperature parameter used in contrastive learning.

In conclusion, by combining *Local Hierarchical Acoustic Smoothness Loss* and *Contrastive Global Prosody Consistency Loss*, FluentEditor2 ensures that edited speech retains natural fluency and consistent prosody across masked regions.

D. Run-time Inference

In run-time, users can edit the speech output by simply modifying the corresponding text. This process allows for intuitive control of the speech through text-based editing. Specifically, users can manually define the desired modification operation, such as *insertion*, *replacement*, or *deletion*. For each operation, the corresponding speech segment in the text is treated as the masked region. Following the approach in [5], our FluentEditor2 model reads both the edited text and the remaining acoustic features, $\bar{Y} - \hat{Y}_{mask}$, from the original speech. The model then predicts the acoustic feature sequence \bar{Y} for the edited word or segment. Once \bar{Y} is predicted, it is smoothly concatenated with the surrounding acoustic features $\bar{Y} - \hat{Y}_{mask}$, ensuring fluent transitions between the edited and unedited portions of the speech. This process results in the final output speech that maintains natural continuity in terms of both timing and prosody, preserving the overall speech quality while reflecting the textual edits.

IV. EXPERIMENTS

A. Experimental Corpora

We evaluate the performance of FluentEditor2 on the VCTK [27] and LibriTTS [12] datasets. The VCTK dataset includes speech data from 110 English speakers, featuring various accents, sampled at 22050 Hz with 16-bit quantization. In contrast, the LibriTTS dataset comprises high-quality English speech recordings at a 24 kHz sampling rate from 2,456

speakers, derived from audiobooks, with a total duration of approximately 585 hours.

To ensure precise alignment between text and speech data, we employ the Montreal Forced Aligner (MFA) [4] for accurate forced alignment of both datasets. Additionally, each dataset is partitioned into training, validation, and testing sets with proportions of 98%, 1%, and 1%, respectively.

B. Participant Information

To ensure the reliability of our subjective evaluations, we conducted listening tests with participants who were native Mandarin speakers with verified English proficiency (CET-4 and CET-6). All evaluators are graduate students specializing in speech processing, who possess solid expertise in assessing speech quality and fluency. This specialized background ensures the authority and robustness of the evaluation results.

C. Experimental Setup

The configurations of text encoder and spectrogram denoiser are referred to [5]. The diffusion steps T of the FluentEditor2 system are set to 8. Following GST [68], the prosody extractor comprises a convolutional stack and an RNN. The dimension of the output prosody feature of the GST-based prosody extractor is 256. To preserve the integrity of both phonemes and words, we adopt a word-level continuous masking strategy. Through extensive experimentation, we found that a masking rate of 80% yields the best results. Section V will report the results for various masking ratios. The pre-trained HiFiGAN [69] vocoder is used to synthesize the speech waveform. We set the batch size is 16. The initial learning rate is set at 2×10^{-4} , and the Adam optimizer [70] is utilized to optimize the network. The FluentEditor2 model is trained with 2 million training steps on one A100 GPU. The masking ratio for WLM in FluentEditor2 is set to 80%.

To obtain frame, phoneme, and word-level alignment, we follow a forced alignment process based on the MFA [4], which generates time-aligned phoneme boundaries from the provided phonetic transcription and speech signals. 1) The alignment process begins with frame-level mapping, where each audio sample is converted into Mel-spectrogram frames based on the sampling rate and hop size. This establishes a temporal representation of the audio at a fine-grained level, serving as the foundation for subsequent phoneme and word-level alignments. 2) Phoneme-level alignment is derived from the time intervals provided by the MFA [4] in the form of TextGrid files. These intervals are mapped to the corresponding Mel-spectrogram frames, ensuring accurate timing for each phoneme. Silent phonemes are accounted for to maintain alignment precision, and phoneme durations are calculated by counting the frames assigned to each phoneme. 3) The word-level alignment is achieved by extending the phoneme-level alignment using a phoneme-to-word mapping. Each frame, already linked to a phoneme, is further mapped to the corresponding word. Word durations are then computed by aggregating the frames associated with the phonemes that constitute each word.

TABLE I: Objective evaluation results of comparative study.

Method	VCTK			LibriTTS		
	MCD (↓)	STOI (↑)	PESQ (↑)	MCD (↓)	STOI (↑)	PESQ (↑)
EditSpeech	7.019	0.690	1.426	5.559	0.663	1.333
CampNet	8.693	0.434	1.340	7.736	0.294	1.249
A ³ T	6.344	0.750	1.560	5.447	0.695	1.379
FluentSpeech	5.902	0.801	1.953	4.573	0.791	1.869
FluentEditor	5.896	0.805	1.969	4.574	0.794	1.876
FluentEditor2	5.836	0.815	1.977	4.537	0.793	1.875

D. Evaluation Metrics

Following FluentSpeech [5], we designed rich subjective and objective metrics, that will be described below.

For subjective evaluation, we conduct a Mean Opinion Score (MOS) listening test [71] to assess speech quality, Fluency-aware MOS (FMOS) [6] to evaluate fluency, and Intelligibility-aware MOS (IMOS) [72] to measure intelligibility. Note that FMOS allows the listener to feel whether the edited segments of the edited speech are fluent compared to the context. We keep the text content and text modifications consistent among different models to exclude other interference factors, only examining speech fluency. Additionally, Comparative FMOS (C-FMOS) [71] is employed for the ablation study to assess the impact of different modifications on fluency. To minimize the vocoder’s impact on subjective metrics, we follow [72] and insert the predicted target regions back into their original positions within the speech.

For objective evaluation, we directly assess the WLM-reconstructed mel-spectrograms, including both the editing regions and the boundaries. The Mel-Cepstral Distortion (MCD) [73], Short-Time Objective Intelligibility (STOI) [74], and Perceptual Evaluation of Speech Quality (PESQ) [75] metrics are computed for these reconstructed spectrograms, providing objective measures of speech quality and intelligibility. This approach ensures a rigorous and consistent comparison of the WLM reconstruction across different models, with the metrics focusing solely on the quality of the reconstructed features. To maintain fairness and accuracy in our comparative experiments, we use an identical test set for all systems, ensuring that the metrics reflect speech quality within the editing regions.

For the test data, we follow [72] and adopt ChatGPT¹ to modify the sentence to generate realistic and diverse <original sentence, modified sentence> paired data, allowing for a comprehensive evaluation of the model’s ability to handle different editing sceneries. Please note that each system’s test set remains identical to ensure that metrics are relevant only to the speech within the editing regions, thereby maintaining accuracy and fairness in the comparative experiments.

¹ChatGPT, based on the GPT-4 model, is accessed through OpenAI’s API (GPT-4-turbo variant) for text modification tasks. For more details on the model and API, visit <https://openai.com>.

E. Comparative Study

To comprehensively evaluate the performance of our proposed FluentEditor2, we conducted a comparative study involving five state-of-the-art neural TSE systems ²:

1) **EditSpeech** [8]: This system utilizes a transformer-based model for text-based speech editing. EditSpeech directly predicts the masked segments of the mel-spectrogram by leveraging the transformer’s powerful attention mechanism to reconstruct speech conditioned on both text and surrounding context.

2) **CampNet** [9]: CampNet introduces a context-aware mask prediction network that simulates the process of speech editing by predicting the content of masked segments based on their surrounding context. The model focuses on ensuring that the predicted segments are consistent with the rest of the utterance both acoustically and contextually.

3) **A³T** [10]: This system proposes an alignment-aware acoustic-text pre-training approach that integrates both phoneme-level information and partially-masked spectrograms as input. By combining these two modalities, A³T enhances the model’s capability to accurately predict missing segments with phoneme-to-speech alignment.

4) **FluentSpeech** [5]: FluentSpeech adopts a diffusion model framework, using a stepwise denoising process to predict the masked speech segments. By leveraging the surrounding speech as context, FluentSpeech achieves smoother and more natural-sounding transitions in edited speech.

5) **FluentEditor** [6]: FluentEditor incorporates Concatenation Loss, which combines both acoustic and prosodic consistency into the loss function, significantly improving the naturalness and coherence of the generated speech at editing boundaries.

6) **FluentEditor2 (Ours)**: Our proposed FluentEditor2 builds upon FluentEditor by introducing two new loss functions: the Hierarchical Local Acoustic Smoothness Consistency Loss (\mathcal{L}_{HLAC}) and the Contrastive Global Prosody Consistency Loss (\mathcal{L}_{CGPC}). These losses are specifically designed to ensure that edited segments integrate smoothly at both the acoustic and prosodic levels, resulting in high-quality, fluent speech.

To further clarify the contribution of our model, we also include the FluentEditor2-generated speech to serve as an

²We note that there was another work [72] focused on TSE, but it was a contemporaneous work and hadn’t been peer-reviewed and open-sourced, so we didn’t add it to the baseline.

TABLE II: Subjective evaluation results of comparative study.

Method	VCTK			LibriTTS		
	Insertion	Replacement	Deletion	Insertion	Replacement	Deletion
Ground Truth	4.428 \pm 0.078	4.422 \pm 0.075	4.424 \pm 0.078	4.441 \pm 0.071	4.412 \pm 0.085	4.417 \pm 0.073
EditSpeech	3.583 \pm 0.160	3.634 \pm 0.133	3.637 \pm 0.177	3.575 \pm 0.150	3.633 \pm 0.118	3.542 \pm 0.145
CampNet	3.678 \pm 0.120	3.716 \pm 0.103	3.765 \pm 0.114	3.601 \pm 0.135	3.738 \pm 0.117	3.713 \pm 0.095
A ³ T	3.833 \pm 0.104	3.828 \pm 0.094	3.772 \pm 0.091	3.597 \pm 0.144	3.779 \pm 0.110	3.732 \pm 0.089
FluentSpeech	3.891 \pm 0.200	3.936 \pm 0.072	3.832 \pm 0.157	3.888 \pm 0.164	4.010 \pm 0.180	3.983 \pm 0.165
FluentEditor	4.073 \pm 0.140	4.114 \pm 0.080	3.991 \pm 0.139	4.053 \pm 0.123	4.105 \pm 0.139	4.044 \pm 0.148
FluentEditor2	4.314 \pm 0.107	4.286 \pm 0.088	4.177 \pm 0.137	4.151 \pm 0.119	4.191 \pm 0.128	4.108 \pm 0.124

upper bound for naturalness and fluency. To this end, five ablation systems are developed: 1) **w/o \mathcal{L}_{HLAC}** , where the Hierarchical Local Acoustic Smoothness Consistency Loss is excluded, 2) **w/o \mathcal{L}_{HLAC} -frame**, where the frame-level loss term of the Hierarchical Local Acoustic Smoothness Consistency Loss is excluded, 3) **w/o \mathcal{L}_{HLAC} -phoneme**, where the phoneme-level loss term of the Hierarchical Local Acoustic Smoothness Consistency Loss is excluded, 4) **w/o \mathcal{L}_{HLAC} -word**, where the word-level loss term of the Hierarchical Local Acoustic Smoothness Consistency Loss is excluded, and 5) **w/o \mathcal{L}_{CGPC}** , where the Contrastive Global Prosody Consistency Loss is removed. These ablation studies help us assess the individual contributions of the proposed loss functions in enhancing the fluency and naturalness of edited speech.

V. RESULTS AND DISCUSSION

A. Evaluation of Reconstructed Speech

We randomly selected 400 test samples from both the VCTK and LibriTTS datasets and reported the objective results in Table I. Following the methodology of [5], we evaluated objective metrics for the masked regions using reconstructed speech. FluentEditor2 consistently outperformed all baselines in terms of both speech quality and fluency, achieving superior results across all three metrics: MCD, STOI, and PESQ.

For instance, FluentEditor2 achieves the lowest MCD values, indicating superior speech quality and reduced distortion, while also attaining the highest STOI and PESQ scores, reflecting better speech intelligibility and overall perceptual quality. These results highlight FluentEditor2’s effectiveness in producing fluent and high-quality speech. Note that objective metrics do not fully reflect the human perception [76], we further conduct subjective listening experiments.

B. Evaluation of Edited Speech

For the FMOS evaluation, we selected 50 audio samples for each operation (insertion, replacement, and deletion) from the test set of each dataset and invited 20 listeners to assess speech fluency. Following the methodology outlined in [8], we evaluated insertion, replacement, and deletion operations, and present the FMOS results in Table II. FluentEditor2 demonstrates notable improvements in fluency-related perceptual scores across both the VCTK and LibriTTS datasets. While the scores of FluentEditor2 are slightly below the ground truth values, it consistently achieves competitive

results with FMOS scores of 4.314 for insertion and 4.286 for replacement, as seen in the VCTK dataset. These scores reflect the effectiveness of the fluency-aware training criteria in improving acoustic quality and prosody.

It is worth noting that the lower scores may be due to the inability to compare the edited speech with the pre-edit version directly. The perceived fluency might be even higher if comparisons could be made with the actual edited speech. Despite this, the significant advancements in FluentEditor2’s fluency scores highlight the effectiveness of our approach and suggest substantial potential for further improvement.

C. Ablation Study

For the ablation study, we combined subjective and objective evaluations to assess the impact of each module. Specifically, 400 test samples were selected from the VCTK and LibriTTS datasets, with the objective metric MCD used to measure spectral differences and assess overall speech quality. For subjective evaluation, 50 edited audio inserts were randomly chosen, and C-FMOS was used to evaluate fluency and naturalness from a human perspective.

Table III presents results after individually removing each level of \mathcal{L}_{LHAC} , include frame, phoneme, and word, as well as the complete removal of \mathcal{L}_{LHAC} . The results show that removing the phoneme-level constraint leads to the largest increase in MCD, reaching 6.004 for VCTK and 4.564 for LibriTTS, which underscores its crucial role in maintaining phonetic coherence and ensuring natural transitions. This highlights the importance of phoneme-level constraints in defining precise editing boundaries. Complete removal of \mathcal{L}_{LHAC} led in a further increase in MCD and a decrease in C-FMOS (-0.281 for VCTK and -0.231 for LibriTTS), indicating that the combined contributions of frame, phoneme, and word-level constraints are necessary for producing fluent, coherent speech edits. Each hierarchical level provides a unique function, with frame-level maintaining temporal alignment, phoneme-level ensuring phonetic accuracy, and word-level preserving semantic consistency.

Furthermore, the removal of \mathcal{L}_{CGPC} led to a decrease in C-FMOS (-0.201 for VCTK and -0.226 for LibriTTS) and an increase in MCD, confirming its critical role in preserving prosodic quality. This constraint is essential for maintaining rhythm, intonation, and continuity, which are crucial for generating natural-sounding edits with consistent speaking style.

TABLE III: Objective and subjective results of ablation study.

Method	VCTK		LibriTTS	
	C-FMOS	MCD(\downarrow)	C-FMOS	MCD(\downarrow)
FluentEditor2	0.00	5.836	0.00	4.537
w/o \mathcal{L}_{HLAC}	-0.281	5.917	-0.231	4.544
w/o \mathcal{L}_{HLAC} -frame	-0.139	5.900	-0.177	4.594
w/o \mathcal{L}_{HLAC} -phoneme	-0.125	6.004	-0.203	4.564
w/o \mathcal{L}_{HLAC} -word	-0.172	5.910	-0.139	4.635
w/o \mathcal{L}_{CGPC}	-0.201	5.896	-0.226	4.587

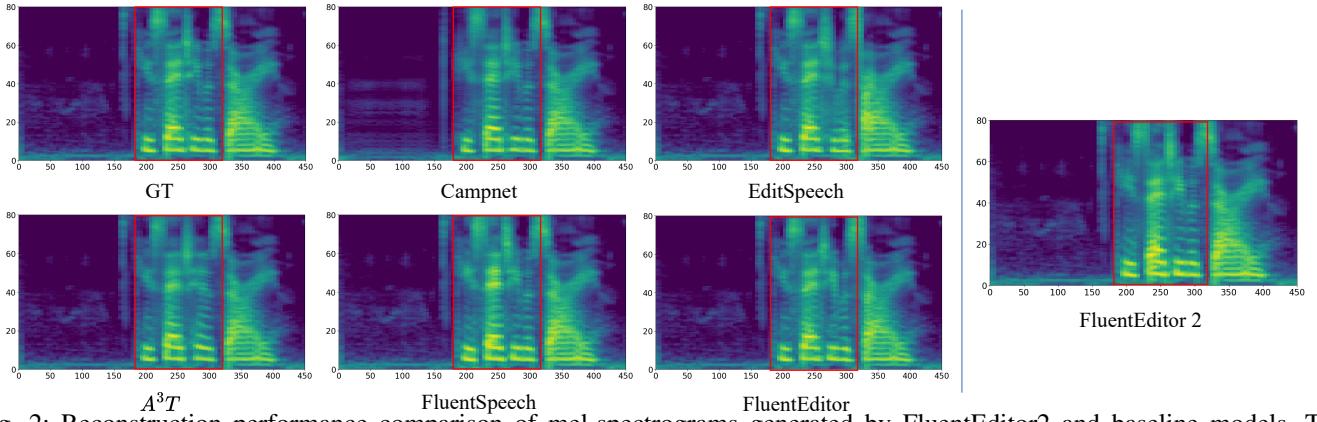


Fig. 2: Reconstruction performance comparison of mel-spectrograms generated by FluentEditor2 and baseline models. The example shows the sentence “he said he was sorry,” with the red box indicating the masked and reconstructed segment “he said he was.”

TABLE IV: Evaluation of the Word-Level Masking Strategy on Baseline Models Using MCD and FMOS Metrics.

Method	VCTK		LibriTTS	
	FMOS	MCD(\downarrow)	FMOS	MCD(\downarrow)
EditSpeech	0.00	7.019	0.00	5.559
EditSpeech + WLM	+0.029	7.004	+0.042	5.547
CampNet	0.00	8.693	0.00	7.736
CampNet + WLM	+0.050	8.588	+0.005	7.164
A ³ T	0.00	6.344	0.00	5.447
A ³ T + WLM	+0.065	6.192	+0.071	5.306
FluentSpeech	0.00	5.902	0.00	4.573
FluentSpeech + WLM	+0.060	5.892	+0.080	5.405
FluentEditor	0.00	5.896	0.00	4.574
FluentEditor + WLM	+0.080	5.869	+0.075	4.572

In summary, these findings demonstrate that both \mathcal{L}_{HLAC} and \mathcal{L}_{CGPC} are indispensable for ensuring the high-quality, fluent, and natural-sounding edits produced by FluentEditor2. Their combined influence allows FluentEditor2 to achieve performance that approaches the Ground Truth, significantly outperforming existing baselines.

D. Evaluation on the Word-Level Masking Strategy.

In this section, we attempt to demonstrate the effectiveness of Word-Level Masking strategy by also arming the WLM policy into all baselines, including A3T, EditSpeech, CampNet, FluentSpeech, and FluentEditor. We use MFA-derived duration alignment information [4], described in section IV-C, to replace the random frame level masking strategies in the baseline with our WLM strategy. Following the methodology

used in the in the section V-C, the evaluation is conducted using both the objective metric MCD and the subjective metric FMOS for a comprehensive assessment. All results are reported at Table IV.

In FluentEditor2, WLM enhances fluency at word boundaries by enabling more accurate alignment and splicing of word, phoneme, and frame boundaries in the edited regions. This improvement results in seamless synthesis across editing points, achieving the lowest MCD and the highest FMOS scores among all models, thereby underscoring the fluency of the generated speech. As indicated by the final row in Table I (objective evaluation) and Table II (subjective evaluation), FluentEditor2 exhibits superior performance, with notable reductions in distortion and significant improvements in fluency compared to other models.

TABLE V: Performance Comparison at Different Mask Ratios Based on MCD, STOI, and PESQ Metrics.

Mask Ratio	MCD (↓)	STOI (↑)	PESQ (↑)
60%	5.968	0.805	1.913
65%	5.934	0.807	1.937
70%	5.887	0.810	1.971
75%	5.970	0.808	1.931
80%	5.836	0.815	1.977
85%	5.947	0.808	1.953
90%	5.966	0.812	1.956
95%	5.987	0.806	1.919

While FluentEditor2 benefits most from WLM, the strategy also improves other models. In FluentEditor, WLM reduces MCD from 5.896 to 5.869 on the VCTK dataset and from 4.574 to 4.572 on the LibriTTS dataset, facilitating more fluid word transitions and higher FMOS scores. This reduction in MCD suggests that WLM decreases spectral distortion and improves speech fluency. Notably, FluentEditor transitions from phoneme-level to word-level masking, highlighting the importance of masking entire words for more coherent and fluent speech synthesis. When applied to other models like A3T, EditSpeech, CampNet, and FluentSpeech, WLM consistently improves both MCD and FMOS. These models, which originally employed non-continuous frame-level masking strategies, often disrupted phoneme and word boundaries. The introduction of WLM significantly amplifies the benefits by preserving the integrity of these boundaries, thus enhancing speech synthesis performance.

E. Effectiveness of Different Mask Ratios in Word-Level Masking

Although [10] reported that 80% was the optimal masking rate for frame-level random selection, we extended our experiments to test similar rates for word-level masking. We varied the masking rate upwards to 60% and downwards to 95% to evaluate the impact of different masking ratios on speech quality and intelligibility. For the effectiveness of different mask ratios in word-level masking, the test data was prepared in the same way as in the section V-A.

As shown in Table V, our results confirm that an 80% mask ratio yields the best overall performance, achieving the lowest MCD (5.836) and the highest STOI (0.815) and PESQ (1.977) scores. This suggests that similar to frame-level masking, word-level masking at 80% offers a strong balance between maintaining speech intelligibility and fluency. When masking ratios deviate from 80%, the performance declines slightly. Lower mask ratios (e.g., 60%–75%) show a higher MCD and lower PESQ, indicating less effective masking, which fails to maintain fluency adequately. Similarly, higher mask ratios (e.g., 90%–95%) also lead to increased MCD and reduced fluency, likely due to excessive masking disrupting the speech’s natural rhythm. These results highlight the importance of selecting an optimal mask ratio, as both under- and over-masking can negatively impact the overall speech quality. In conclusion, the experiments confirm that an 80% mask ratio is the most effective for word-level masking, reinforcing findings from frame-level masking studies and

validating the utility of this ratio across different linguistic levels.

F. Visualization Analysis of Mel-Spectrogram Reconstruction and Editing

For the objective evaluation of mel-spectrograms, we randomly selected one audio sample from the 400 test samples we previously selected from both the VCTK and LibriTTS datasets (as reported in Table I). This sample was then visualized to illustrate the reconstruction performance.

As illustrated in Fig. 2, comparing mel-spectrograms generated by FluentEditor2 and other baseline models demonstrates the superior reconstruction performance of FluentEditor2. Specifically, FluentEditor2 and consistently captures more intricate details in the frequency domain. This results in mel-spectrograms that exhibit richer spectral characteristics, leading to more natural-sounding speech outputs. For example, baseline models like EditSpeech and A³T show relatively coarse spectral reconstructions, which may result in artifacts or unnatural prosody during playback. In contrast, FluentEditor2 maintains both acoustic and prosodic consistency by using acoustic and prosody loss functions, effectively capturing boundary information during the speech reconstruction process. For instance, Fig. 2 shows the sentence “he said he was sorry,” with red boxes highlighting the masked and reconstructed “he said he was” region, clearly showcasing the enhanced smoothness and fidelity of FluentEditor2 and compared to the FluentSpeech baseline.

To further analyze FluentEditor2’s editing capabilities and provide a clearer view of the smooth transitions in the edited regions, we randomly selected one audio sample from each operation (insertion, deletion, and replacement) from the test set of the VCTK dataset for visualization. Fig. 3 provides a detailed visualization of the editing capabilities of FluentEditor2 on mel-spectrograms for common operations like deletion, insertion, and replacement of words or phrases³. The red boxes indicate the edited segments in the speech. When comparing the FluentSpeech baseline with FluentEditor2, it becomes evident that the latter can handle these modifications much more gracefully. In particular:

- For **deletion**, FluentEditor2 and smoothly removes the masked segments, ensuring that the surrounding speech remains coherent and maintains the natural prosody, unlike the baseline, where distortions can sometimes occur at the boundary.
- During **insertion**, the transitions between the existing and newly inserted speech segments in FluentEditor2 are nearly seamless, preserving both prosodic rhythm and intonation. In contrast, the baseline struggles with smooth integration, leading to noticeable disruptions in speech flow.

³Deletion operations result in a reduction in spectrogram length, as segments are removed, whereas insertion leads to a lengthened spectrogram due to added content. For replacement operations, the spectrogram length varies based on the relative duration of the new versus replaced content. Our model, Fluenteditor2, shows smoother transitions at the editing boundaries compared to baselines, with more continuous spectral patterns and fewer abrupt changes.

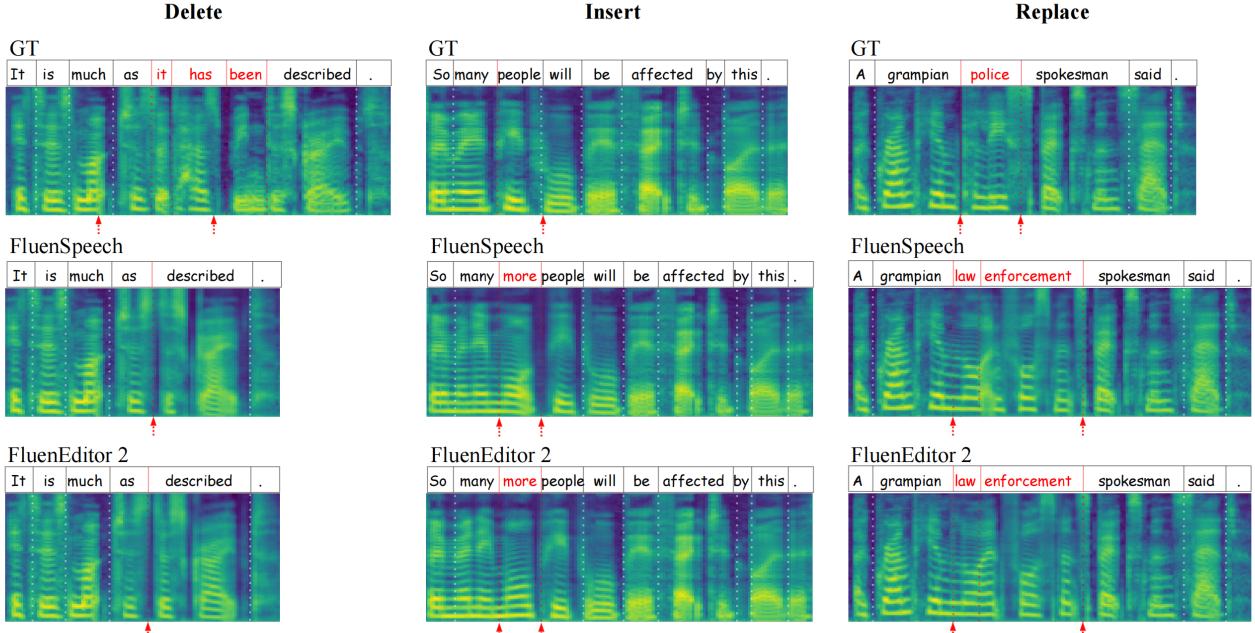


Fig. 3: Visualization of editing effects on mel-spectrograms for speech insertion, replacement, and deletion. Dotted lines represent time step divisions aligned with each word in the sentence. Red highlights indicate the boundaries of the edited regions where the operations were applied.

TABLE VI: Generalization Performance of FluentSpeech on Source-Domain and Cross-Domain Datasets: Comparison of MOS, FMOS, and IMOS across baseline models and FluentSpeech in both Source and Cross Domain Settings.

Method	Source-Domain Dataset			Cross-Domain Dataset		
	MOS	FMOS	IMOS	MOS	FMOS	IMOS
EditSpeech	3.718 ± 0.039	3.739 ± 0.039	3.729 ± 0.046	3.618 ± 0.058	3.630 ± 0.057	3.630 ± 0.060
CampNet	3.582 ± 0.047	3.589 ± 0.077	3.574 ± 0.051	3.420 ± 0.139	3.315 ± 0.201	3.444 ± 0.140
A ³ T	3.835 ± 0.042	3.852 ± 0.045	3.862 ± 0.036	3.738 ± 0.065	3.742 ± 0.064	3.767 ± 0.064
FluentSpeech	3.876 ± 0.08	3.930 ± 0.084	3.886 ± 0.084	3.806 ± 0.116	3.800 ± 0.122	3.797 ± 0.123
FluenEditor	4.052 ± 0.066	4.089 ± 0.075	4.059 ± 0.068	3.945 ± 0.116	3.974 ± 0.118	3.976 ± 0.124
FluenEditor2	4.247 ± 0.057	4.286 ± 0.066	4.259 ± 0.063	4.123 ± 0.067	4.135 ± 0.064	4.148 ± 0.068

- For **replacement**, FluentEditor2 and delivers a well-integrated substitution of words or phrases, resulting in consistent intonation and pacing, while the baseline may exhibit prosody mismatches or abrupt transitions.

These visualizations not only confirm FluentEditor2's effectiveness in reconstructing high-quality mel-spectrograms but also underscore its superior ability to handle intricate speech edits. The detailed prosodic and acoustic features retained by the model are crucial for producing natural, expressive speech even after complex modifications. More visualization results and speech samples are referred to our website: <https://github.com/Ai-S2-Lab/FluentEditor2>. to appreciate the advantages fully.

G. Generalization Study

For the generalization evaluation, we utilize LJSpeech as the cross-domain dataset [77]. Note that LJSpeech primarily contains speech related to Newgate Prison in London, which focuses differently on VCTK dataset. For LJSpeech, we randomly select 50 samples from the test set, consisting of

20 insertions, 20 replacements, and 10 deletions. For VCTK, we also selected 50 samples from the test set with the same operations as the source-domain dataset. Both source-domain and cross-domain datasets are evaluated using MOS, IMOS, and FMOS. Each audio sample was assessed by at least 20 listeners to evaluate speech quality and fluency. The results are presented in Table VI.

The results show that FluentEditor2 outperforms all baseline models in both settings. It achieves the highest MOS (speech quality), FMOS (fluency), and IMOS (intelligibility), demonstrating significant improvements in both perceptual and acoustic consistency. However, Cross-Domain performance shows a noticeable drop across all models, particularly in MOS and FMOS, highlighting the limited ability of existing methods to generalize effectively to Cross-Domain scenarios. This performance gap can be attributed to the fact that most models are primarily trained and evaluated within the same domain, limiting their robustness when faced with unseen data from a different domain.

VI. CONCLUSION

In this paper, we introduced FluentEditor2, a novel text-based speech editing (TSE) model designed to improve the acoustic and prosody consistency of edited speech through two innovative fluency-aware training criteria. The proposed Hierarchical Local Acoustic Smoothness Consistency Loss (\mathcal{L}_{HLAC}) evaluates the consistency of acoustic features at editing boundaries both at the frame and phoneme levels, ensuring that the transitions at concatenation points are smooth and natural. Additionally, the Contrastive Global Prosody Consistency Loss (\mathcal{L}_{CGPC}) uses contrastive learning to align the high-level prosodic features within the edited regions with the surrounding context, while distinctly separating them from irrelevant segments. We validated the effectiveness of FluentEditor2 through extensive objective and subjective experiments on two datasets, VCTK and LibriTTS, covering the three core editing operations: insertion, replacement, and deletion. The results consistently demonstrated that incorporating \mathcal{L}_{HLAC} and \mathcal{L}_{CGPC} significantly enhances the fluency and prosody of edited speech across all datasets and operations. This comprehensive evaluation shows that FluentEditor2 not only produces more natural and seamless transitions but also maintains consistent prosody, providing a superior solution for text-based speech editing.

REFERENCES

- [1] Z. Jin, G. J. Mysore, S. Diverdi, J. Lu, and A. Finkelstein, “Voco: Text-based insertion and replacement in audio narration,” *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–13, 2017.
- [2] R. Liu, Y. Hu, H. Zuo, Z. Luo, L. Wang, and G. Gao, “Text-to-speech for low-resource agglutinative language with morphology-aware language model pre-training,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 1075–1087, 2024.
- [3] R. Liu, B. Sisman, G. Gao, and H. Li, “Controllable accented text-to-speech synthesis with fine and coarse-grained intensity rendering,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 2188–2201, 2024.
- [4] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, “Montreal forced aligner: Trainable text-speech alignment using kaldii,” *Proc. Interspeech 2017*, pp. 498–502, 2017.
- [5] Z. Jiang, Q. Yang, J. Zuo, Z. Ye, R. Huang, Y. Ren, and Z. Zhao, “FluentSpeech: Stutter-oriented automatic speech editing with context-aware diffusion models,” in *Findings of the Association for Computational Linguistics: ACL 2023*. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 11655–11671. [Online]. Available: <https://aclanthology.org/2023.findings-acl.741>
- [6] R. Liu, J. Xi, Z. Jiang, and H. Li, “Fluenteditor: Text-based speech editing by considering acoustic and prosody consistency,” in *Interspeech 2024*, 2024, pp. 3435–3439.
- [7] M. Morrison, L. Rencker, Z. Jin, N. J. Bryan, J.-P. Caceres, and B. Pardo, “Context-aware prosody correction for text-based speech editing,” in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 7038–7042.
- [8] D. Tan, L. Deng, Y. T. Yeung, X. Jiang, X. Chen, and T. Lee, “Editspeech: A text based speech editing system using partial inference and bidirectional fusion,” in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2021, pp. 626–633.
- [9] T. Wang, J. Yi, R. Fu, J. Tao, and Z. Wen, “Campnet: Context-aware mask prediction for end-to-end text-based speech editing,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 2241–2254, 2022.
- [10] H. Bai, R. Zheng, J. Chen, M. Ma, X. Li, and L. Huang, “A³T: Alignment-aware acoustic and text pretraining for speech synthesis and editing,” in *International Conference on Machine Learning*. PMLR, 2022, pp. 1399–1411.
- [11] J. Tae, H. Kim, and T. Kim, “Editts: Score-based editing for controllable text-to-speech,” in *Interspeech 2022*, 2022, pp. 421–425.
- [12] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, “Libritts: A corpus derived from librispeech for text-to-speech,” in *Interspeech 2019*, 2019, pp. 1526–1530.
- [13] H. Reyes, S. Subramaniam, N. Kaabouch, and W. C. Hu, “A spectrum sensing technique based on autocorrelation and euclidean distance and its comparison with energy detection for cognitive radio networks,” *Computers & Electrical Engineering*, vol. 52, pp. 319–327, 2016.
- [14] C.-y. Tseng, S.-h. Lin, Y. Lee, H.-m. Wang, and Y.-c. Chen, “Fluent speech prosody: Framework and modeling,” *Speech communication*, vol. 46, no. 3–4, pp. 284–309, 2005.
- [15] R. Liu, B. Sisman, G. Gao, and H. Li, “Expressive tts training with frame and style reconstruction loss,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1806–1818, 2021.
- [16] J. R. Glass and V. W. Zue, “Multi-level acoustic segmentation of continuous speech,” in *ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing*. IEEE Computer Society, 1988, pp. 429–430.
- [17] L. R. Rabiner, R. W. Schafer *et al.*, “Introduction to digital speech processing,” *Foundations and Trends® in Signal Processing*, vol. 1, no. 1–2, pp. 1–194, 2007.
- [18] K. Honda, “Physiological factors causing tonal characteristics of speech: from global to local prosody,” in *Speech Prosody 2004, International Conference*, 2004.
- [19] K. Qian, Y. Zhang, S. Chang, J. Xiong, C. Gan, D. Cox, and M. Hasegawa-Johnson, “Global prosody style transfer without text transcriptions,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 8650–8660.
- [20] C. Blouin, O. Rosec, P. C. Bagshaw, and C. d’Alessandro, “Concatenation cost calculation and optimisation for unit selection in tts,” in *IEEE workshop on speech synthesis*, 2002, pp. 0–3.
- [21] A. J. Hunt and A. W. Black, “Unit selection in a concatenative speech synthesis system using a large speech database,” in *1996 IEEE international conference on acoustics, speech, and signal processing conference proceedings*, vol. 1. IEEE, 1996, pp. 373–376.
- [22] J. Wouters and M. W. Macon, “Unit fusion for concatenative speech synthesis.” in *INTERSPEECH*. Citeseer, 2000, pp. 302–305.
- [23] M. Dong, K.-T. Lua, and H. Li, “A unit selection-based speech synthesis approach for mandarin chinese.” *J. Chin. Lang. Comput.*, vol. 16, no. 3, pp. 135–144, 2006.
- [24] R. Fu, J. Tao, Y. Zheng, and Z. Wen, “Deep metric learning for the target cost in unit-selection speech synthesizer.” in *INTERSPEECH*, 2018, pp. 2514–2518.
- [25] D. T. Chappell and J. H. Hansen, “A comparison of spectral smoothing methods for segment concatenation based speech synthesis,” *Speech Communication*, vol. 36, no. 3–4, pp. 343–373, 2002.
- [26] P. H. Le-Khac, G. Healy, and A. F. Smeaton, “Contrastive representation learning: A framework and review,” *Ieee Access*, vol. 8, pp. 193907–193934, 2020.
- [27] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, “Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit,” *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, vol. 6, p. 15, 2017.
- [28] B. P. Chapman, A. Weiss, and P. R. Duberstein, “Statistical learning theory for high dimensional prediction: Application to criterion-keyed scale development.” *Psychological methods*, vol. 21, no. 4, p. 603, 2016.
- [29] S. Whittaker and B. Amento, “Semantic speech editing,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2004, pp. 527–534.
- [30] S. Rubin, F. Berthouzoz, G. J. Mysore, W. Li, and M. Agrawala, “Content-based tools for editing audio stories,” in *Proceedings of the 26th annual ACM symposium on User interface software and technology*, 2013, pp. 113–122.
- [31] C. Baume, M. D. Plumley, J. Ćalić, and D. Frohlich, “A contextual study of semantic speech editing in radio production.” *International Journal of Human-Computer Studies*, vol. 115, pp. 67–80, 2018.
- [32] “Descript,” <https://www.descript.com/>, 2020.
- [33] E. Moulines and F. Charpentier, “Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones,” *Speech communication*, vol. 9, no. 5–6, pp. 453–467, 1990.
- [34] B. Sisman, J. Yamagishi, S. King, and H. Li, “An overview of voice conversion and its challenges: From statistical modeling to deep learning,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 132–157, 2020.
- [35] S. H. Mohammadi and A. Kain, “An overview of voice conversion systems,” *Speech Communication*, vol. 88, pp. 65–82, 2017.

- [36] L. Sun, K. Li, H. Wang, S. Kang, and H. Meng, "Phonetic posteriograms for many-to-one voice conversion without parallel data training," in *2016 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2016, pp. 1–6.
- [37] Z. Jin *et al.*, "Speech synthesis for text-based editing of audio narration," Ph.D. dissertation, Doctoral dissertation, Ph. D. dissertation, Comput. Sci. Dept., Princeton ..., 2018.
- [38] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Interspeech 2020*, 2020, pp. 5036–5040.
- [39] P. Guo, F. Boyer, X. Chang, T. Hayashi, Y. Higuchi, H. Inaguma, N. Kamo, C. Li, D. Garcia-Romero, J. Shi *et al.*, "Recent developments on espnet toolkit boosted by conformer," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5874–5878.
- [40] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 10850–10869, 2023.
- [41] M. Jeong, H. Kim, S. J. Cheon, B. J. Choi, and N. S. Kim, "Diff-tts: A denoising diffusion model for text-to-speech," in *Interspeech 2021*, 2021, pp. 3605–3609.
- [42] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, "Grad-tts: A diffusion probabilistic model for text-to-speech," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8599–8608.
- [43] A. Alexos and P. Baldi, "Attentionstitch: How attention solves the speech editing problem," *CoRR*, vol. abs/2403.04804, 2024. [Online]. Available: <https://doi.org/10.48550/arXiv.2403.04804>
- [44] G. B. Simpson, R. R. Peterson, M. A. Casteel, and C. Burgess, "Lexical and sentence context effects in word recognition." *Journal of Experimental Psychology: Learning, Memory, and Cognition*, vol. 15, no. 1, p. 88, 1989.
- [45] R. W. Schafer and L. R. Rabiner, "Digital representations of speech signals," *Proceedings of the IEEE*, vol. 63, no. 4, pp. 662–677, 1975.
- [46] B. Lin, L. Wang, X. Feng, and J. Zhang, "Automatic scoring at multi-granularity for l2 pronunciation," in *Interspeech*, 2020, pp. 3022–3026.
- [47] W. Chen, X. Xing, X. Xu, J. Pang, and L. Du, "Speechformer: A hierarchical efficient framework incorporating the characteristics of speech," in *Interspeech 2022*, 2022, pp. 346–350.
- [48] Y. Lei, S. Yang, X. Wang, and L. Xie, "Mssemotts: Multi-scale emotion transfer, prediction, and control for emotional speech synthesis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 853–864, 2022.
- [49] Y. Hono, K. Tsuboi, K. Sawada, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Hierarchical multi-grained generative model for expressive speech synthesis," in *Interspeech 2020*, 2020, pp. 3441–3445.
- [50] X. Li, C. Song, J. Li, Z. Wu, J. Jia, and H. Meng, "Towards multi-scale style control for expressive speech synthesis," in *Interspeech 2021*, 2021, pp. 4673–4677.
- [51] Y. Ren, J. Liu, and Z. Zhao, "Portaspeech: Portable and high-quality generative text-to-speech," *Advances in Neural Information Processing Systems*, vol. 34, pp. 13963–13974, 2021.
- [52] S. Lei, Y. Zhou, L. Chen, Z. Wu, X. Wu, S. Kang, and H. Meng, "Msstylets: Multi-scale style modeling with hierarchical context information for expressive speech synthesis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.
- [53] Y. Jiang, T. Li, F. Yang, L. Xie, M. Meng, and Y. Wang, "Towards expressive zero-shot speech synthesis with hierarchical prosody modeling," in *Interspeech 2024*, 2024, pp. 2300–2304.
- [54] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1735–1742.
- [55] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.
- [56] T. B. Viana, V. L. Souza, A. L. Oliveira, R. M. Cruz, and R. Sabourin, "A multi-task approach for contrastive learning of handwritten signature feature representations," *Expert Systems with Applications*, vol. 217, p. 119589, 2023.
- [57] X. Chen and K. He, "Exploring simple siamese representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 15750–15758.
- [58] Z. Yang, Y. Cheng, Y. Liu, and M. Sun, "Reducing word omission errors in neural machine translation: A contrastive learning approach," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 6191–6196.
- [59] A. J. Bose, H. Ling, and Y. Cao, "Adversarial contrastive estimation," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, I. Gurevych and Y. Miyao, Eds. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 1021–1032. [Online]. Available: <https://aclanthology.org/P18-1094>
- [60] E. Kharitonov, M. Rivière, G. Synnaeve, L. Wolf, P.-E. Mazaré, M. Douze, and E. Dupoux, "Data augmenting contrastive learning of speech representations in the time domain," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 215–222.
- [61] A. S. Koepke, A.-M. Oncescu, J. F. Henriques, Z. Akata, and S. Albanie, "Audio retrieval with natural language queries: A benchmark study," *IEEE Transactions on Multimedia*, vol. 25, pp. 2675–2685, 2022.
- [62] Y. Wu, X. Wang, S. Zhang, L. He, R. Song, and J.-Y. Nie, "Self-supervised context-aware style representation for expressive speech synthesis," in *Interspeech 2022*, 2022, pp. 5503–5507.
- [63] Y. Meng, X. Li, Z. Wu, T. Li, Z. Sun, X. Xiao, C. Sun, H. Zhan, and H. Meng, "Calm: Contrastive cross-modal speaking style modeling for expressive text-to-speech synthesis," in *Interspeech 2022*, 2022, pp. 5533–5537.
- [64] Z. Wu, Q. Li, S. Liu, and Q. Yang, "Dctts: Discrete diffusion model with contrastive learning for text-to-speech generation," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 11336–11340.
- [65] Y. Ren, X. Tan, T. Qin, Z. Zhao, and T.-Y. Liu, "Revisiting over-smoothness in text to speech," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022, pp. 8197–8213.
- [66] P. S. Mandhare, V. R. Borkar, and M. R. Kumbhakarna, "The generalized approximation of an arbitrary function using its mean and quadratic variance," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 66, no. 12, pp. 2096–2100, 2019.
- [67] N. N. W. N. Hashim, M. A.-E. A. Ezzi, and M. D. Wilkes, "Mobile microphone robust acoustic feature identification using coefficient of variance," *International Journal of Speech Technology*, vol. 24, no. 4, pp. 1089–1100, 2021.
- [68] 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 *International conference on machine learning*. PMLR, 2018, pp. 5180–5189.
- [69] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17022–17033, 2020.
- [70] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [71] P. C. Loizou, "Speech quality assessment," in *Multimedia analysis, processing and communications*. Springer, 2011, pp. 623–654.
- [72] Y. Chen, Y. Jia, S. Zhao, Z. Jiang, H. Li, J. Kang, and Y. Qin, "Diffeditor: Enhancing speech editing with semantic enrichment and acoustic consistency," *arXiv preprint arXiv:2409.12992*, 2024.
- [73] R. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment," in *Proceedings of IEEE pacific rim conference on communications computers and signal processing*, vol. 1. IEEE, 1993, pp. 125–128.
- [74] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *2010 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2010, pp. 4214–4217.
- [75] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *2001 IEEE international conference on acoustics, speech, and signal processing. Proceedings (Cat. No. 01CH37221)*, vol. 2. IEEE, 2001, pp. 749–752.
- [76] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," in *International Conference on Learning Representations*, 2020.
- [77] K. Ito and L. Johnson, "The lj speech dataset," <https://keithito.com/LJ-Speech-Dataset/>, 2017.