



Accent Normalization Using Self-Supervised Discrete Tokens with Non-Parallel Data

Qibing Bai^{1,2,4}, Sho Inoue^{1,2}, Shuai Wang^{*3,1,2}, Zhongjie Jiang⁴, Yannan Wang⁴, Haizhou Li^{1,2,5}

¹School of Data Science, ²SRIBD, The Chinese University of Hong Kong, Shenzhen, China

³School of Intelligence Science and Technology, Nanjing University, Suzhou, China

⁴Tencent Ethereal Audio Lab, Tencent, China

⁵Department of ECE, National University of Singapore, Singapore

Abstract

Accent normalization converts foreign-accented speech into native-like speech while preserving speaker identity. We propose a novel pipeline using self-supervised discrete tokens and non-parallel training data. The system extracts tokens from source speech, converts them through a dedicated model, and synthesizes the output using flow matching. Our method demonstrates superior performance over a frame-to-frame baseline in naturalness, accentedness reduction, and timbre preservation across multiple English accents. Through token-level phonetic analysis, we validate the effectiveness of our token-based approach. We also develop two duration preservation methods, suitable for applications such as dubbing.¹

Index Terms: accent conversion, speech synthesis, voice conversion, flow matching, duration control

1. Introduction

Accent conversion (AC) seeks to alter speech from one accent to another while preserving the speaker characteristics. We study the way to convert the non-native (L2) accented speech into a native (L1) accented one. Such accent normalization (AN)² technology offers a wide range of practical applications, such as improving pronunciation for language learners [1], enhancing the authenticity of movie dubbing [2], and personalized text-to-speech (TTS) systems [3].

Early deep learning approaches focused on *reference-based methods* [4–7], which rely on native accent speech samples to generate accent-neutral representations. Some studies [4, 5, 7] use pre-trained automatic speech recognition (ASR) models to extract phonetic posteriorgram (PPG) or bottleneck features, while [6] propose to generate reference speech with native text-to-speech (TTS). A recent method [8] leverages speech tokens for style transfer. However, the requirement for reference speech during inference limits their practical applications.

The *reference-free method* [9–11] has gained attention, focusing on the mapping between non-native and native accent representations [9, 10] or converting PPGs for unseen speakers [11]. [12] provides a comprehensive evaluation, while [13] proposes to use discrete self-supervised tokens with small-scale paired data. Although these approaches eliminate reference speech requirements, they still rely on paired parallel data during training. The challenges in assembling diverse accent corpora have inspired the development of methods that are both reference-free and independent of parallel corpora [14–18].

*: Corresponding author.

¹Samples: <https://github.com/Plping/TokAN-Demo>. Code available at <https://github.com/Plping/TokAN>.

²Also referred to as foreign accent conversion (FAC).

Recent accent normalization approaches address parallel data limitations. [14] proposes a two-step process: converting non-native speech to text via ASR, and then generating native speech using TTS. [15] introduces a pseudo siamese network with an auxiliary decoder to separate accent features. [16] leverages TTS representations as accent-neutral features. Building on this, [18] incorporates a non-autoregressive (NAR) architecture and enhances robustness through pre-training techniques. The TTS-guided framework was then augmented with flow matching and normalizing flow [19, 20], respectively.

Recent work [16, 18–20] relies on the TTS-synthesized targets, and the quality of synthetic targets could be a limitation, both in terms of voice cloning and duration modeling. When the source speech is noisy, it is hard to generate target speech with the same voice. Moreover, the duration is learned from the synthetic targets instead of native utterances, this injects error accumulation and also applies to PPG-based methods.

We propose an accent normalization method using self-supervised discrete tokens and non-parallel data. Our method first quantizes both input speech and TTS-generated targets into discrete tokens, which reduces quality requirements for synthetic targets. A token-to-token conversion model, trained on deduplicated tokens where consecutive identical tokens are merged, transforms source-accented sequences into native ones. This deduplication removes duration information and focuses the model on the phonetic mapping between accents. A flow matching token-to-Mel synthesizer then recovers duration and generates Mel-spectrograms. Additionally, we explore two methods for preserving the utterance duration: a basic scaling approach and a flow-matching-based technique, which are particularly valuable for applications such as dubbing. Our contributions are three-fold:

- We perform accent normalization using self-supervised discrete tokens and non-parallel data, mitigating the impact of synthetic data quality through token-only training.
- We validate our approach across multiple English accents through comprehensive evaluations, supported by a token-level phonetic analysis.
- We explore duration modeling techniques that preserve the total duration of the original utterance

2. Related Work

2.1. Self-Supervised Discrete Tokens in Speech Generation

Discrete tokens from self-supervised learning (SSL) [21] exhibit a strong correlation with phonetic content [22], making them effective representations for speech synthesis. These tokens serve as a proxy in various generation tasks, including text-to-speech (TTS) [23] and voice conversion (VC) [24].

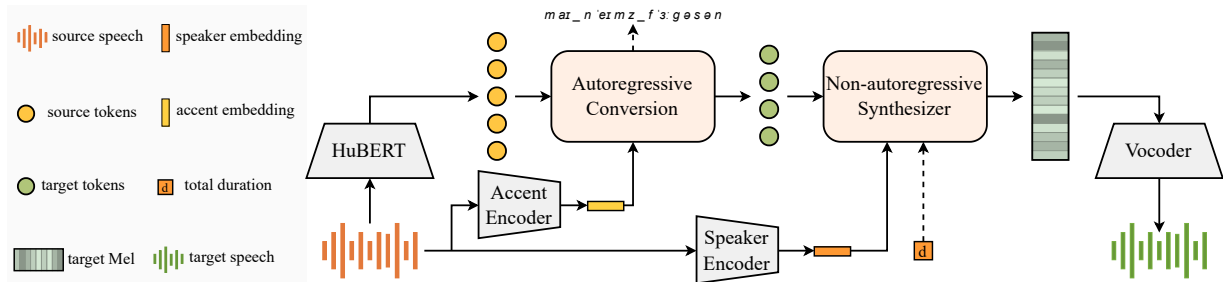


Figure 1: Pipeline of accent normalization. Trainable modules include the conversion model and synthesizer, while gray trapezoids indicate pre-trained modules. The conversion model can generate auxiliary phone outputs. The synthesizer, with a flow matching duration predictor, enables total duration control.

Their discrete nature enables the application of textual techniques to speech processing, such as spoken language modeling [25], speech-to-speech translation [26], and speech generation in large language models [27]. We leverage HuBERT [21] to extract discrete tokens for conversion and synthesis.

2.2. Flow-Matching-Based Speech Synthesis

Flow matching [28] has emerged as a powerful approach for acoustic feature generation in TTS systems [29], achieving competitive naturalness. Recent non-autoregressive TTS research explores flow matching duration predictors [30–32]. While these predictors show higher word error rates (WER) compared to regression, they enable more diverse rhythmic patterns [30]. We apply flow matching to accent normalization. While using regression-based duration prediction by default, we investigate a flow matching duration predictor with awareness of the total duration.

3. Methodology

Figure 1 illustrates our accent normalization pipeline. The system first converts input speech into discrete HuBERT tokens through feature extraction, quantization, and deduplication. An autoregressive conversion model then transforms these source tokens into target tokens with normalized phonetic patterns, removing the source accent. A non-autoregressive synthesizer predicts token-level durations and generates the Mel-spectrogram, conditioned on the source speaker embedding. The synthesizer can employ a flow matching duration predictor for total duration control.

We train the conversion model and synthesizer on separate datasets. The synthesizer is trained on native-accented speech to model normalized duration and acoustic features, while the conversion model requires parallel data to remove source accents. Since token representations largely exclude timbre information, content-parallel training data suffices. This token-based approach prevents timbre or prosodic artifacts in synthetic training speech from degrading target token quality.

3.1. Autoregressive Token Conversion

HuBERT’s discrete tokens primarily encode content-related information, with minimal paralinguistic features when extracted from appropriate layers [21]. These unsupervised tokens capture phonetic rather than semantic patterns [22], preserving source accent characteristics in the token sequence. We employ an autoregressive model to normalize these accented patterns.

While the conversion model typically requires text-parallel data, such pairs are challenging to collect. For non-parallel L2-

accented datasets, we generate synthetic targets using a native TTS. Previous approaches [18–20] using synthetic data rely on direct speech supervision, which can lead to error accumulation during training. Our approach, operating on deduplicated tokens, mitigates this issue by eliminating most paralinguistic and duration information from synthetic data.

To enhance content understanding, we incorporate CTC-based [33] phone prediction as an auxiliary training task, complementing the model’s implicit content recognition from input tokens. The model’s handling of diverse input accents is further improved through accent-conditional processing, where accent embeddings from a pre-trained classifier are integrated into the encoder using AdaLN [34].

3.2. Non-Autoregressive Token-to-Mel Synthesis

The synthesizer maps native-normalized tokens to speech through a non-autoregressive architecture comprising a duration predictor, token encoder, and Mel decoder. Using flow matching [6], the system processes input tokens and speaker embedding to generate speech in multiple steps.

First, the Transformer-based token encoder converts discrete tokens into continuous representations, conditioned on speaker embeddings. The duration predictor then assigns duration values to these token representations for expansion. The expanded representations serve as the condition for the Mel decoder, which transforms noisy Mel-spectrograms into high-likelihood ones via velocity prediction. The decoder employs a UNet-Transformer architecture [29] with AdaLN-Zero [34] speaker conditioning, enhanced by classifier-free guidance (CFG) [35] for improved speech quality.

Preservation of total duration. While the default 1-D convolutional duration predictor, trained with mean square error, generates robust durations [30], it offers limited diversity and control. For applications like dubbing where matching source duration is crucial, we explore two approaches. The first simply scales predicted durations uniformly to match the target total duration (denoted as “ours w/ dur. scaling” in experiments).

We also develop a flow matching duration predictor conditioned on *average token duration*, enabling direct total duration control. This approach generates durations that naturally sum closer to the target length, requiring minimal scaling adjustment. Our experiments compare both methods, with this total-duration-aware system labeled as “ours w/ dur. control”.

3.3. Training Pipeline

Since discrete HuBERT tokens serve as the intermediary between the conversion model and the synthesis model, the two models can be trained independently using distinct datasets.

The synthesizer is directly trained on high-quality target-accented data, typically a TTS dataset. The conversion model is trained on non-parallel data with synthetic targets:

- Train a TTS model on native-accented data
- Generate TTS-synthesized targets for L2-accented data
- Extract discrete SSL tokens for both the source and target
- Train the conversion model on these token pairs

In line with [13, 23], BART-style pre-training [36] is employed to enhance model performance, additionally with the phone supervision. This modular design enables the use of different datasets optimized for each component while maintaining the effectiveness of the overall system.

4. Experimental Setup

4.1. Datasets

We evaluate our method on seven English accents: six from L2-ARCTIC [37] (Arabic, Chinese, Hindi, Korean, Spanish, and Vietnamese) and native American English from ARCTIC [38]. Each accent consists of four speakers. We conduct evaluations on this extended L2-ARCTIC dataset. For training, we utilize multiple data sources:

- LibriTTS-R [39] for the auxiliary TTS model and synthesizer
- The extended L2-ARCTIC described above (~ 20 hours)
- A proprietary Chinese-accented ASR dataset (~ 300 hours)

LibriTTS-R is also used to pre-train the conversion model. The proprietary Chinese-accented dataset is used exclusively for training and validation. Since L2-ARCTIC uses the same prompt sentences across speakers, we construct the data splits: 50 sentences for validation and 80 for testing, holding out one unseen speaker per accent.

4.2. Model Architecture

Proposed. We employ the HuBERT-large model³ (17th layer features) with K-means clustering on LibriTTS-R for 1000 discrete tokens. The conversion model follows an encoder-decoder architecture (each with 5 layers and a hidden dimension of 1024) with accent information from a pretrained classifier⁴. For synthesis, we use flow matching [28] with Resemblyzer⁵ speaker embeddings and BigVGAN [40] vocoding.

Baseline. We implement a frame-to-frame conversion approach following [19]. The model consists of three training stages: TTS training, speech encoder pre-training via distillation, and speech encoder fine-tuning. We utilize pitch and energy prediction during inference for more native prosody.

4.3. Training & Inference Configurations

For synthetic target generation, we employ Matcha-TTS [29] with voice cloning for the extended L2-ARCTIC dataset and a fixed speaker embedding for the extra Chinese-accented data.

The conversion model pre-training uses token noise with probabilities of 0.2 for masking, 0.1 for insertion, and 0.1 for replacement. We set the CTC loss weight to 1.0. The synthesizer training applies a CFG rate of 0.2 for both decoder and duration predictor.

³<https://github.com/facebookresearch/fairseq/blob/main/examples/hubert>

⁴https://huggingface.co/Jzuluaga/accent-id-commonaccent_xlsr-en-english

⁵<https://github.com/resemble-ai/Resemblyzer>

During inference, we use a beam size of 10 for the conversion model. The token-to-Mel synthesizer operates with 32 sampling steps and CFG strengths of 2.0 and 0.5 for the Mel decoder and flow matching duration predictor, respectively.

4.4. Evaluation Metrics

Subjective Metrics. We evaluate speech naturalness (NAT) and accentedness (ACT) using MUSHRA tests, and speaker similarity (SIM) using best-worst scaling (BWS). The BWS results are aggregated via the counting algorithm [41]: $(N_{best} - N_{worst})/N_{occurrence}$. Each evaluation involves 21 raters assessing 20 samples, with the American accent excluded from the accentedness evaluation.

Objective Metrics. We assess three aspects of conversion quality. For *intelligibility*, we compute the word error rate (WER) using a native-only ASR model⁶, simulating listener’s perception of L2-accented speech. *Timbre preservation* is measured by speaker encoding similarity (SECS) using Resemblyzer, which exhibits relatively high accent-independence compared to other speaker models. For *accentedness reduction*, we compute metrics against the synthetic targets. Following the three aspects of accent (pronunciation, rhythm, and intonation) [42], we employ PPG distance (Δ PPG) [43], frame distortion (FD), and F0 correlation (F0 corr.), respectively.

5. Results

5.1. Comparison with Baseline

Naturalness and speaker similarity. Table 1 presents both subjective and objective evaluation results. The proposed system outperforms the frame-to-frame baseline in naturalness and speaker similarity, both subjectively and objectively. This might be attributed to i) the error accumulation during the baseline training process, and ii) the use of CFG in the proposed synthesizer, which influences speaker similarity in particular.

Accentedness and intelligibility. The proposed system achieves more effective accent normalization, particularly through its flexible adjustment of speech rhythm, a critical component of accent. Both systems exhibit higher WERs compared to the source speech, indicating challenges in maintaining intelligibility during accent conversion. The native-only ASR model’s high WERs for the baseline system reflect its lower intelligibility, likely due to its less effective accent reduction.

Table 2 shows the accent-wise results for the subjective evaluation of accentedness. Both the baseline and proposed model can reduce the accentedness on every accent, while the proposed model consistently outperforms the baseline.

The native-only ASR model’s WER reflects the intelligibility of converted speech from a native-only listener’s perspective. As shown in Table 3, our model outperforms the baseline, particularly for Chinese and Vietnamese accents. This improvement may be attributed to our method’s ability to handle syllable-timed rhythm patterns from these L1 languages. For the Hindi accent, however, our method has lower performance, likely due to Hindi’s fine-grained features (e.g., retroflex consonants) that are not well captured in our discrete token inventory.

5.2. Token Distributions of Phonemes

To evaluate pronunciation normalization, we analyze phoneme-specific token distributions using forced alignment on the test

⁶<https://huggingface.co/facebook/s2t-medium-librispeech-asr>

Table 1: Evaluation results of the accent conversion systems.

System	Source-length	Subjective			Objective				
		NAT (↑)	ACT (↓)	SIM (↑)	WER (% ↓)	SECS (↑)	ΔPPG (↓)	FD (↓)	F0 corr. (↑)
Source	✓	60.50 \pm 2.34	46.40 \pm 2.48	-	15.86	-	0.51	67.29	0.74
Source resynthesis	✓	-	-	-	18.75	0.8863	0.47	66.79	0.75
Baseline [19]	✓	48.76 \pm 2.36	37.61 \pm 2.07	-0.031	21.54	0.7846	0.49	60.68	0.72
Ours	×	60.38 \pm 2.17	26.14 \pm 1.79	0.040	16.25	0.8739	0.30	35.24	0.77
Ours w/ dur. scaling	✓	54.24 \pm 2.22	26.80 \pm 1.79	0.007	16.47	0.8771	0.31	62.71	0.77
Ours w/ dur. control	✓	54.72 \pm 2.23	26.61 \pm 1.82	-0.017	16.76	0.8774	0.32	61.57	0.77

Table 2: Accent-wise subjective results of accentedness.

System	Zh	Hi	Vi	Ar	Es	Ko
Source	46.20 \pm 5.25	59.65 \pm 7.02	43.53 \pm 5.66	52.61 \pm 6.14	44.79 \pm 6.04	33.54 \pm 4.80
Baseline	39.09 \pm 4.78	41.82 \pm 5.81	40.56 \pm 5.59	34.94 \pm 4.71	37.99 \pm 5.52	32.76 \pm 4.21
Ours	30.41 \pm 4.98	28.03 \pm 4.70	27.49 \pm 5.94	23.54 \pm 4.46	24.18 \pm 3.69	24.34 \pm 3.30

Table 3: Accent-wise WERs with native-only ASR

System	Zh	Hi	Vi	Ar	Es	Ko	Us
Source	20.82	11.75	31.40	15.09	15.79	13.78	2.53
Baseline	26.76	19.98	34.36	22.96	22.28	18.91	5.62
Ours	17.51	24.93	18.73	17.33	17.05	13.94	5.55

set. Computing KL divergence between these distributions and those from American native speech reveals the effectiveness of our conversion approach. Figure 2 illustrates the analysis for Chinese-accented speech before and after conversion.

The post-conversion results show a more prominent low-valued diagonal line, indicating successful transformation toward native-like pronunciations. Significant improvements appear in common Chinese-English mispronunciation patterns: high-front vowels (orange), substitution of AO for dark L (red), confusion between voiced and unvoiced plosives (purple), and dental-alveolar fricative distinctions (pink).

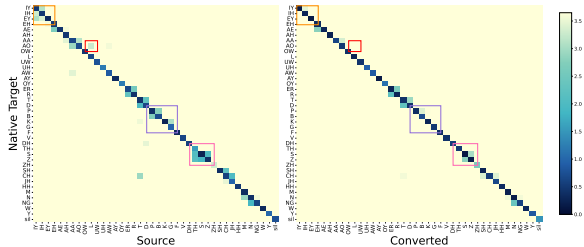


Figure 2: Token-based phoneme divergence from native targets. Left: source Chinese-accented speech. Right: converted speech.

5.3. Total-Duration Preservation

As detailed in Sec. 3.2, we compare two duration control methods: direct scaling of predicted durations (“ours w/ dur. scaling”) and a flow matching duration predictor with total-duration awareness (“ours w/ dur. control”). Without scaling, our base model (“ours”) shows an average absolute difference of 0.513 seconds (around 12.89%) from source durations on the testing set. In contrast, The duration control approach achieves an absolute difference of 0.064 seconds (around 1.64%). Figure 3 demonstrates their typical behavior. The scaling approach produces more uniform durations, shown by the diagonal alignment, while the control approach yields more varied durations, particularly for vowels and fricatives.

While “ours w/ dur. control” achieves less frame disturbance and similar SECS scores compared to direct scaling, sub-

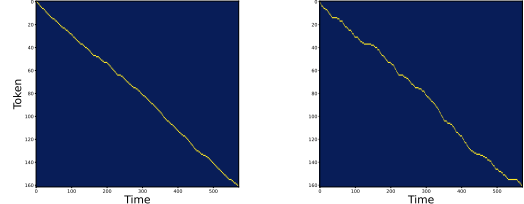


Figure 3: Comparison of two total-duration controlling methods: “ours w/ dur. scaling” (left) and “ours w/ dur. control” (right).

jective SIM ratings favor the scaling approach. This preference likely stems from their different handling of source disfluencies: scaling preserves the slower speaking rate, while the control method produces a more acted speaking style that may affect perceived speaker identity.

5.4. Ablation Study

Auxiliary modules. Removing accent embeddings or phone supervision leads to noticeable performance degradation, as shown by increased WER in Table 4. Phone supervision proves particularly crucial for content preservation.

Training data. Removing the extra Chinese-accented data results in a slight performance decline. Removing the pre-training stage causes significant performance degradation. This suggests that the pre-training step is crucial for the model to learn various linguistic patterns and native token distributions.

Table 4: Ablation results for conversion model

System	WER (% ↓)	ΔPPG (↓)	FD (↓)	F0 corr. (↑)
Source	15.86	0.51	67.29	0.74
Ours	16.25	0.30	35.24	0.77
- accent embeddings	16.82	0.31	35.75	0.76
- phone supervision	19.55	0.33	34.97	0.77
- extra data	17.13	0.31	39.75	0.77
- pre-training	27.59	0.40	30.83	0.76

6. Conclusion, Limitation, & Future Work

In this paper, we introduce an accent normalization method using self-supervised discrete tokens and non-parallel data. Subjective evaluations on multiple English accents show significant improvements in speech naturalness, speaker similarity, and accentedness reduction. We also investigate two methods for controlling total duration, suitable for dubbing scenarios. Subjective evaluations indicate a preference for direct duration-scaling, whereas the flow matching method generates dynamic durations close to the total duration.

Despite improved naturalness and accent normalization, the high WER after conversion limits practical use. Future work will explore more robust tokenizers to address this limitation.

7. Acknowledgements

This work was supported by National Natural Science Foundation of China (Grant No. 62401377), Shenzhen Science and Technology Program (Shenzhen Key Laboratory, Grant No. ZDSYS20230626091302006), Shenzhen Science and Technology Research Fund (Fundamental Research Key Project, Grant No. JCYJ20220818103001002), Program for Guangdong Introducing Innovative and Entrepreneurial Teams (Grant No. 2023ZT10X044).

8. References

- [1] D. Felps, H. Bortfeld, and R. Gutierrez-Osuna, "Foreign accent conversion in computer assisted pronunciation training," *Speech communication*, vol. 51, no. 10, pp. 920–932, 2009.
- [2] O. Türk and L. M. Arslan, "Subband based voice conversion." in *Proc. Interspeech*, 2002, pp. 289–292.
- [3] L. Sun, H. Wang, S. Kang, K. Li, and H. M. Meng, "Personalized, cross-lingual tts using phonetic posteriorgrams." in *Proc. Interspeech*, 2016, pp. 322–326.
- [4] Z. Guanlong, S. Sinem, L. John, C.-H. Evgeny, and G.-O. Ricardo, "Accent conversion using phonetic posteriorgrams," in *Proc. ICASSP*, 2018, pp. 5314–5318.
- [5] G. Zhao, S. Ding, and R. Gutierrez-Osuna, "Foreign accent conversion by synthesizing speech from phonetic posteriorgrams." in *Proc. Interspeech*, 2019, pp. 2843–2847.
- [6] W. Li, B. Tang *et al.*, "Improving accent conversion with reference encoder and end-to-end text-to-speech," *arXiv preprint arXiv:2005.09271*, 2020.
- [7] S. Ding, G. Zhao, and R. Gutierrez-Osuna, "Accentron: Foreign accent conversion to arbitrary non-native speakers using zero-shot learning," *Computer Speech & Language*, vol. 72, p. 101302, 2022.
- [8] X. Zhang, X. Zhang *et al.*, "Vevo: Controllable zero-shot voice imitation with self-supervised disentanglement," in *ICLR*, 2025.
- [9] G. Zhao, S. Ding, and R. Gutierrez-Osuna, "Converting foreign accent speech without a reference," *TASLP*, vol. 29, pp. 2367–2381, 2021.
- [10] T.-N. Nguyen, N.-Q. Pham, and A. Waibel, "Accent conversion using pre-trained model and synthesized data from voice conversion." in *Proc. Interspeech*, 2022, pp. 2583–2587.
- [11] W. Quamer, A. Das, J. Levis, E. Chukharev-Hudilainen, and R. Gutierrez-Osuna, "Zero-shot foreign accent conversion without a native reference," in *Proc. Interspeech*, 2022, pp. 4920–4924.
- [12] W.-C. Huang and T. Toda, "Evaluating methods for ground-truth-free foreign accent conversion," in *APSIPA ASC*, 2023.
- [13] Z. jia, H. Xue, X. Peng, and Y. Lu, "Convert and speak: Zero-shot accent conversion with minimum supervision," in *Multimedia*, 2024.
- [14] S. Liu, D. Wang *et al.*, "End-to-end accent conversion without using native utterances," in *Proc. ICASSP*, 2020, pp. 6289–6293.
- [15] M. Jin, P. Serai, J. Wu, A. Tjandra, V. Manohar, and Q. He, "Voice-preserving zero-shot multiple accent conversion," in *Proc. ICASSP*, 2023, pp. 1–5.
- [16] Y. Zhou, Z. Wu, M. Zhang, X. Tian, and H. Li, "Tts-guided training for accent conversion without parallel data," *Signal Processing Letters*, vol. 30, pp. 533–537, 2023.
- [17] D. Jia, Q. Tian, K. Peng, J. Li, Y. Chen, M. Ma, Y. Wang, and Y. Wang, "Zero-shot accent conversion using pseudo siamese disentanglement network," *arXiv preprint arXiv:2212.05751*, 2022.
- [18] X. Chen, J. Pei, L. Xue, and M. Zhang, "Transfer the linguistic representations from tts to accent conversion with non-parallel data," in *Proc. ICASSP*, 2024, pp. 12 501–12 505.
- [19] Q. Bai, S. Wang, Z. Liu, M. Zhang, W. Rao, Y. Wang, and H. Li, "Diffusion-based method with tts guidance for foreign accent conversion," in *Proc. ISCSLP*, 2024, pp. 284–288.
- [20] T. N. Nguyen, S. Akti, N. Q. Pham, and A. Waibel, "Improving pronunciation and accent conversion through knowledge distillation and synthetic ground-truth from native tts," in *ICASSP*, 2025.
- [21] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *TASLP*, vol. 29, pp. 3451–3460, 2021.
- [22] K. Choi, A. Pasad, T. Nakamura, S. Fukayama, K. Livescu, and S. Watanabe, "Self-supervised speech representations are more phonetic than semantic," in *Proc. Interspeech*, 2024.
- [23] E. Kharitonov, D. Vincent *et al.*, "Speak, read and prompt: High-fidelity text-to-speech with minimal supervision," *Trans. ACL*, vol. 11, pp. 1703–1718, 2023.
- [24] W.-C. Huang, Y.-C. Wu, and T. Hayashi, "Any-to-one sequence-to-sequence voice conversion using self-supervised discrete speech representations," in *Proc. ICASSP*, 2021, pp. 5944–5948.
- [25] K. Lakhotia, E. Kharitonov *et al.*, "On generative spoken language modeling from raw audio," *Trans. ACL*, vol. 9, pp. 1336–1354, 2021.
- [26] A. Lee, P.-J. Chen *et al.*, "Direct speech-to-speech translation with discrete units," in *Proc. ACL*, 2022.
- [27] Q. Fang, S. Guo, Y. Zhou, Z. Ma, S. Zhang, and Y. Feng, "Llama-omni: Seamless speech interaction with large language models," *arXiv preprint arXiv:2409.06666*, 2024.
- [28] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," in *ICLR*, 2023.
- [29] S. Mehta, R. Tu, J. Beskow, É. Székely, and G. E. Henter, "Matcha-tts: A fast tts architecture with conditional flow matching," in *Proc. ICASSP*, 2024, pp. 11 341–11 345.
- [30] M. Le, A. Vyas *et al.*, "Voicebox: Text-guided multilingual universal speech generation at scale," in *Advances in NeurIPS*, 2023, pp. 14 005–14 034.
- [31] Z. Ju, Y. Wang *et al.*, "Naturalspeech 3: Zero-shot speech synthesis with factorized codec and diffusion models," *arXiv preprint arXiv:2403.03100*, 2024.
- [32] S. E. Eskimez, X. Wang *et al.*, "Total-duration-aware duration modeling for text-to-speech systems," in *Proc. Interspeech*, 2024.
- [33] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006.
- [34] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *Proc. ICCV*, 2023, pp. 4195–4205.
- [35] J. Ho and T. Salimans, "Classifier-free diffusion guidance," in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.
- [36] M. Lewis, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.
- [37] G. Zhao, S. Sonsaat, A. Silpachai, I. Lucic, E. Chukharev-Hudilainen, J. Levis, and R. Gutierrez-Osuna, "L2-ARCTIC: A Non-native English Speech Corpus," in *Proc. Interspeech*, 2018.
- [38] J. Kominek and A. W. Black, "The cmu arctic speech databases," in *Fifth ISCA workshop on speech synthesis*, 2004.
- [39] Y. Koizumi, H. Zen *et al.*, "LibriTTS-R: A Restored Multi-Speaker Text-to-Speech Corpus," in *Proc. Interspeech*, 2023.
- [40] S. gil Lee, W. Ping, B. Ginsburg, B. Catanzaro, and S. Yoon, "BigVGAN: A universal neural vocoder with large-scale training," in *ICLR*, 2023.
- [41] A. M. V. Ravillion, "A comparison of best-worst scaling and rating scale for timbre characterisation," 2020.
- [42] X. Zhou, M. Zhang, Y. Zhou, Z. Wu, and H. Li, "Accented text-to-speech synthesis with limited data," *TASLP*, vol. 32, pp. 1699–1711, 2024.
- [43] C. Churchwell, M. Morrison, and B. Pardo, "High-fidelity neural phonetic posteriorgrams," in *ICASSP 2024 Workshop on Explainable Machine Learning for Speech and Audio*, 2024.