R2S: Real-to-Synthetic Representation Learning for Training Speech Recognition Models on Synthetic Data

Minh Tran[1], *Debjyoti Paul*[2], *Yutong Pang*[2], *Laxmi Pandey*[2], *Jinxi Guo*[2], *Ke Li*[2], *Shun Zhang*[2], *Xuedong Zhang*[2], *Xin Lei*[2]

[1]Computer Science Department, University of Southern California, USA
[2]Meta AI, USA

mtran@ict.usc.edu

Abstract

We investigate the use of synthetic speech to enhance the performance of Automatic Speech Recognition (ASR) systems. While pre-trained ASR models have demonstrated impressive capabilities, their performance can still vary across different conditions and speakers. Conversely, text-to-speech technology allows for precise control over factors such as environmental noise and speaker accents, producing clean speech that poses fewer challenges for ASR systems. Building on this insight, we propose a novel method called R2S (Real-to-Synthetic), which aligns the representation spaces of real and synthetic speech. Our approach incorporates a Gradient Reversal Layer to promote invariant representations between real and synthetic speech, and a Residual-Vector Quantization module to generate pseudo-labels from synthetic speech, guiding the representations of real speech. Our experimental results on three datasets demonstrate that the proposed method can boost ASR performance by 4-5% and successfully align the representation space of real and synthetic speech. Our qualitative results further demonstrate that R2S can suppress speaker-dependent features thanks to the alignment with synthetic speech.

Index Terms: speech recognition, synthetic data, representation learning

1. Introduction

Automatic Speech Recognition (ASR) systems have seen remarkable advancements, partly driven by the power of pre-trained models such as Whisper [1]. These systems have demonstrated impressive accuracy across a wide range of situations, making them increasingly reliable in various applications. However, their performance is not without limitations. Factors such as environmental noise, speaker accents, and inherent voice variances can introduce challenges in the speech encoding process that converts raw speech to high-level representations, leading to inaccuracies in speech recognition. To address this issue, speaker adaptation [2–4] and domain adaptation [5–7] methods are proposed.

These adaptation methods have traditionally focused on the role of speaker attributes/conditions in ASR systems, either for personalization or generalization. Personalization approaches [2–4, 8] enable speaker attributes or conditions to influence the predictions while generalization approaches [9–12] encourage speaker-invariant or condition-invariant features for robust ASR.

Inspired by the second line of adaptation methods, we explore speaker-invariant representation learning. Unlike previous methods that require speaker attribute labels (*e.g.,* speaker ID) to learn speaker-invariant representations, we only use synthetic speech for adaptation. Motivated by the fact that synthetic speech (of a particular target speaker) is invariant to speaker, noise, accent, and other undesired properties, leading to a lower Word-error-rate (WER) (see Table 1) compared to real speech of the same content, we investigate whether aligning the representation space of real and synthetic speech can help improve ASR performance. To achieve this, we introduce R2S , a novel method that contains two components: 1) a Gradient Reversal Layer (GRL) placed after the encoder that classifies whether the input is real or synthetic speech to encourage the encoder to produce real-synthetic invariant features, and 2) a linear head trained with Connectionist Temporal Classification loss (CTC-loss) using discrete tokens generated by a Residual Vector Quantizer (Residual-VQ) on features extracted from synthetic speech. Together, the two components enable the encoder of ASR systems to better align with the representations of synthetic speech. Our experimental results with the pre-trained Whisper model on three datasets, namely, LibriSpeech, CommonVoice, and VCTK, demonstrate that the proposed method can boost ASR performance by 4-5% and successfully align the representation space of real and synthetic speech. In summary, the contributions of this paper are as follows: **a)** We present the first study on leveraging synthetic speech to guide the representation learning of real speech; **b)** We propose R2S , a novel method that aligns real and synthetic speech via two components: 1) a GRL that encourages the encoder to produce real-synthetic invariant features, and 2) a mechanism to quantize synthetic features into discrete tokens to guide the real speech representation learning process via a CTC-loss; **c)** We validate the usefulness of the proposed method on three speech recognition datasets.

	LS-test-other	Voxpopuli-en	VCTK
Real	12.39	10.38	4.56
Synthetic	6.91	1.67	2.61

Table 1: *ASR (Word-Error-Rate) performance of Whisper-base on Librispeech test-other, Voxpopuli English test set, and VCTK dataset for real (original test sets) and synthetic (text-to-speech) speech. The performance gap suggests potential performance gain when shifting representations from real to synthetic speech.*

2. Related work

Existing speaker adaptation methods can be categorized into two main approaches. The first approach leverages the identity of speakers to extract speaker representations, which are

[1] This work was done during an internship at Meta AI.

then used to generate personalized predictions in ASR systems. For instance, Senior et al. [2] and Peddinti et al. [3] adapt ASR systems using i-vectors. Pironkov et al. [13] and Adi et al. [8] explore multi-task learning, where speech recognition is trained jointly with auxiliary tasks such as speaker recognition [8] or gender classification [4, 13, 14]. The second approach focuses on adversarial learning-based speech recognition, intending to make acoustic representations independent from speaker characteristics or recording conditions. This line of work aims to improve the robustness and generalization of ASR systems. Tsuchiya et al. [9] and Meng et al. [10] propose using a Gradient Reversal Layer (GRL) [3] in conjunction with speaker classification during ASR training to encourage the model to produce speaker-invariant features, achieving relative WER improvement ranging from 4-7%. Serdyuk et al. [11] implements GRL along with noise condition classification to produce noise-invariant features for speech recognition. Liang et al. [12] use Invariant Representation Learning (IRL) to encourage speech recognition models to generate features similar to clean conditions, given augmented noisy speech. Sun et al. [5] explore adversarial learning with accent speech, where they use a gradient reversal layer followed by an accent classifier to encourage the speech recognition model to generate accent-invariant features.

Tjandra *et al.* [15] propose a novel framework that integrates speech recognition and synthesis into a closed-loop system, mimicking the human speech communication process. They demonstrate that this mutual learning significantly enhances the performance of both systems. Rosenberg *et al.* [16] explore the use of multi-speaker speech synthesis to generate natural-sounding speech capturing prosody, speaker, and style variations, employing the generated data as an augmentation to improve in-domain speech recognition. Chen *et al.* [17] tackles the challenge of limited acoustic diversity in synthesized speech used for augmentation by integrating generative adversarial networks (GANs) with multi-style training. Hu *et al.* [**?**] and Huang *et al.* [18] explore strategies to mitigate discrepancies between synthetic and real data distributions, such as filtering out low-quality samples, to address issues like structured noise and unrealistic speaking styles. Unlike these works, R2S focuses on realigning speech representations to single-speaker and noise-free speech representations, rather than generating diverse data that simulate the target domain.

Most related to our work, Meghanani *et al.* [19] propose using features derived from clean (real) speech to guide the representation learning of the corresponding augmented speech via a Soft-DTW loss [20]. Experimental results show that SCORE can effectively suppress undesired (perturbed) information and enable better performance on content-related downstream tasks such as speech recognition, phoneme recognition, and spoken term discovery. In contrast to SCORE, we explore using clean (synthetic) speech as the target representation for real speech. Furthermore, we explore alternative methods to Soft-DTW, which suffers from prohibitively high memory usage for long sequences.

3. Method

3.1. Data preparation

Given a real speech datasets with transcripts $\mathcal{D} = \{(x_i^{(r)}, y_i)\}_{i=1}^{N}$, we use a a text-to-speech model [21] to generate synthetic speech for each sample in D, resulting in triplets $\mathcal{D}' = \{(x_i^{(r)}, x_i^{(s)} = TTS(y_i), y_i)\}_{i=1}^{N}$.

The system is structured into two main components: a linguistic frontend and an acoustic backend. The linguistic frontend processes plain text and converts it into phonetic form (grapheme to phoneme), and prosodic details. The processed information is then relayed to the acoustic backend, which is equipped with a transformer-based prosody model that estimates phone-level fundamental frequency (F0) values and their durations, a transformer-based spectral model that computes frame-level mel-cepstral coefficients, along with F0 and periodicity features. Then, a sparse WaveRNN-based neural vocoder [21] synthesizes the final audio waveform. We utilize this TTS pipeline to produce audio for each generated text domain, which is then used to fine-tune ASR models. Since our goal is to convert real speech into a clean and condition-invariant representation space, we use a single target speaker for all TTS speech without any noise augmentation.

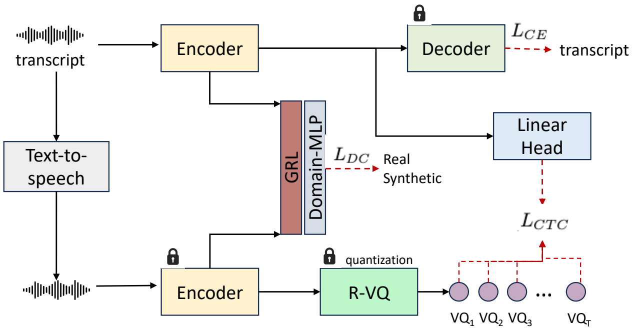

Figure 1: ***An architectural overview of the R2S pipeline.*** *Real speech is passed to a trainable encoder while synthetic speech (with the same transcriptions) is passed to a frozen teacher encoder. Their extracted representations are then forwarded to a domain classifier (with a Gradient Reversal Layer) to classify whether the input is real or synthetic speech. We also use a frozen teacher model to guide the representations of the real speech with a CTC-loss on the pre-computed discrete tokens of corresponding synthetic speech. The decoder is frozen, while the encoder is jointly optimized with the original Cross-entropy loss for ASR, domain confusion loss, and the CTC-loss.*

3.2. Base model

We use Whisper-base [1] as our backbone architecture, with around $78M$ parameters. Whisper is a Seq2Seq Transformer model [22] pre-trained on approximately 700K hours of weakly supervised speech recognition data. The model has shown strong generalizability with competitive performance across a wide-range of public benchmarks in zero-shot settings without the need for any fine-tuning. The model contains an Encoder and a Decoder. Given input speech signal x, the encoder first extracts hidden representations from x to produce H, then the decoder auto-regressively predicts the probability distribution of the next token y_i given previously predicted tokens $y_{<i}$ for the transcription. The model is trained with a cross-entropy loss over the predicted probability distributions

$$L_{CE} = -\Sigma_{i=1}^{N} log P(y_i | y_{<i}, H) \qquad (1)$$

In this work, we use a frozen teacher encoder (Enc_T) to extract synthetic speech representations to train a student encoder (Enc_S) that processes real speech inputs. Both Enc_T and Enc_S share the same initialization from the pre-trained Whisper model.

3.3. Adversarial Learning

The Gradient Reversal Layer (GRL) [23] is a commonly used method for unsupervised domain adaptation to bridge the representation gap between a source domain and a target domain. During the forward pass, all information flowing through a GRL remains unchanged, but in the backward pass, all gradients passing through the GRL are reversed (from positive to negative, and vice versa). In this work, to encourage the encoder to generate invariant features between real and synthetic speech, we add GRL between the encoder and an MLP-based domain classifier (real vs. synthetic speech). Both the encoder and the domain classifier are jointly optimized with a Binary Cross Entropy loss

$$L_{DC} = -\Sigma_{i=1}^{B} logP(d_i|H_i) \tag{2}$$

where B is the batch size, $d_i \in \{0, 1\}$ is the domain labels, and H_i is the utterance-level representations extracted from the encoders for sample i.

3.4. Discrete tokens feature guidance

With adversarial learning, the domain classifier only relies on a single representation vector for each sample, which may not capture the full complexity and variability of real and synthetic speech. To address this issue, we aim to further encourage the encoder to learn more *fine-grained* information within the extracted synthetic speech representations, allowing the model to better capture the nuances and variations.

The temporal misalignment between real and synthetic samples poses a challenge for using traditional regression functions such as Mean Square Error (MSE) to estimate the synthetic speech representation given a real one. Recently, solutions such as Soft Dynamic Time Warping (Soft-DTW) [20] have been proposed for regression problems without temporal alignment, but suffer from prohibitively high memory usage for long sequences. Hence, we propose to first encode the extracted synthetic features into discrete tokens using a Residual Vector Quantization module [24], and use the CTC-loss to guide the representation learning process for real speech.

3.4.1. Learning Residual-VQ.

Vector quantization (VQ) [25] discretizes high-dimensional data by mapping it to a codebook of vectors, using Euclidean distances to determine the closest entries in the codebook. Zhigidour et al. [24] propose residual vector quantization (R-VQ) to use multiple vector quantizers to recursively quantize the residuals of a waveform, resulting in more refined quantized embeddings. Specifically, the encoder feature $z = Enc_S(x) \in \mathcal{R}^{T\times d}$ is quantized w.r.t. a set of codebooks $Z = \{Z_i\}_{[i\in L]}$ where $Z_i \in \mathcal{R}^{K\times d}$ according to

$$z_q^{(l)} = \mathbf{q}(z_q^{(l-1)}) = \{argmin_{z_k\in Z_l}||z_q^{(l-1)}[t] - z_k||)\}_{t\in T} \tag{3}$$

where l denotes the codebook layer, $\mathbf{q}$ denotes the quantization function that map each element of an encoded sequence to the nearest codebook entry, and T denotes the sequence length.

In this paper, we use the final layer quantization of R-VQ as the pseudo-labels to guide the representation learning process of real speech. To generate meaningful pseudo-labels, we first train the R-VQ using only synthetic speech features. In particular, we place the R-VQ between the Enc_T and decoder layers, and optimize the codebook entries (while freezing everything else) with respect to a combination of the speech recognition loss L_{CE} and a commit loss [25].

$$L_{RVQ}(Z) = L_{CE}(Dec(z_q), Y) + ||sg[z] - z_q||_2^2 + ||z - sg[z_q]||_2^2 \tag{4}$$

where z_q is the final-layer quantized output of the encoder extracted feature z, Y is the ground-truth transcription and $sg[.]$ denotes the stopgradient operator. This pre-training stage allows us to learn a codebook that effectively captures the underlying structure of the synthetic speech data, which is crucial for generating high-quality discrete tokens.

3.4.2. Discrete token predictions.

We use the trained R-VQ to generate offline pseudo-labels for the training state of Enc_S. We add a simple linear layer after Enc_S to map the feature dimension to the vocab size of the codebook entries in R-VQ, and use a CTC loss to guide the learning process of Enc_S to predict the sequences of discrete pseudo-label tokens. The discrete token prediction loss L_{DTP} is formulated as Equation 5, where L_X is the pseudo-labels generated by the last layer of R-VQ.

$$L_{DTP} = \Sigma_{i=1}^{B} L_{CTC}(Enc_S(X), L_X) \tag{5}$$

Overall, the student encoder Enc_S is optimized with Equation 6, where $\lambda_1, \lambda_2, \lambda_3$ are hyper-parameters controlling the trade-offs between the three loss terms.

$$L = \lambda_1 L_{CE} + \lambda_2 L_{DC} + \lambda_3 L_{DTP} \tag{6}$$

4. Experiments

4.1. Datasets & Baselines

We evaluate R2S on three datasets: LibriSpeech [26], Voxpopuli-English [27] (containing 15 accents), and VCTK dataset [28] (containing multiple speakers speaking same contents). We follow the official train/val/test splits for LibriSpeech and Voxpopuli-English, and perform a speaker-independent split for VCTK. The train sets contain 960 hours of audio from 2484 speakers (LibriSpeech), 543 hours of audio from 1313 speakers (Voxpopuli), and 44 hours from 109 speakers (VCTK). We fine-tune Whisper-base on the training sets and evaluate on the corresponding test sets.

We compare R2S with four baselines. **FT-R**: vanilla fine-tuning Whisper on only real speech (original training sets), **FT-S**: vanilla fine-tuning Whisper on only synthetic speech, **FT-[R+S]**: vanilla fine-tuning Whisper on real and TTS speech combined, and SCORE [19].

4.2. Implementation Details

We use `Whisper-base` as the pre-trained ASR model in this study. As mentioned in Section 3, we only fine-tune the encoder of Whisper while freezing everything else to experiment usefulness of aligning real and synthetic representations.

During training, we fine-tune the models with an AdamW optimizer with $lr = 1e^{-5}$ for 20 epochs with early stopping and a batch size of 40. We set $\lambda_1 = 1.0, \lambda_2 = 0.5, \lambda_3 = 0.01$ to balance the loss scales. For the Gradient Reversal Layer, we also set the adaptation factor as in [23] to reduce the impact of noisy signals from the domain discriminator in the early training stage, where the discriminator is not well-trained. Our Residual-VQ contains 16 independent codebooks; each codebook contains 1024 entries with a dimension of 512. We use a

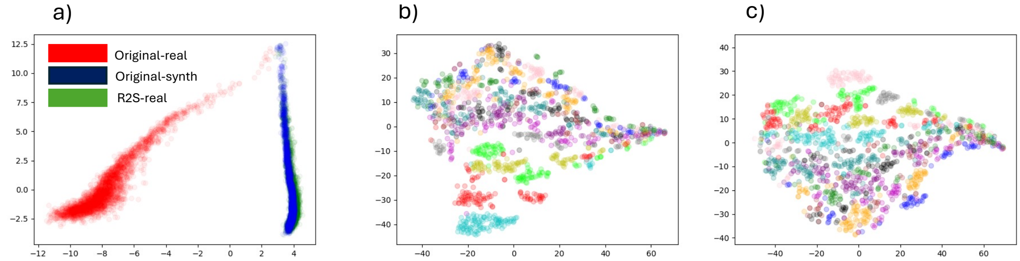

Figure 2: ***Visualization of learned feature representations on the LibriSpeech test-other dataset using t-SNE [29].*** *Panel (a) compares the representation spaces of real and synthetic speech between the original Whisper checkpoint and the adapted R2S model. Panels (b) and (c) provide side-by-side comparisons of representations for 15 randomly sampled speaker identities from LibriSpeech test-other for the original and R2S models, respectively. In panels (b) and (c), each color represents a distinct speaker.*

	LS	Voxpopuli	VCTK
Original	12.39	10.38	4.56
FT-R	10.84	10.03	4.84
FT-S	13.31	11.42	5.46
FT-[R+S]	10.91	10.05	4.98
SCORE [19]	10.78	10.22	4.74
FT[R+S]+GRL	10.52	9.77	4.50
FT[R+S]+CTC	10.39	9.59	4.42
FT[R+S]+GRL+CTC	**10.32**	**9.50**	**4.38**

Table 2: *Performance comparison between R2S and baselines on three speech recognition datasets.*

linear warmup learning rate for 10% of the training process. We use the greedy decoding algorithm with the default parameters provided in the open-source Whisper implementation. For evaluation, we report the Word Error Rate (WER) metric without punctuation. We report results averaged over 5 runs.

Choice of Target Speaker in TTS. We introduce a versatile framework for synthetic data generation that operates independently of the specific target speaker. In our experiments, we fixed the target speaker to simplify the analysis and demonstrate the maximum potential gains achievable when the encoder aligns real speech representations. This design choice reflects the underlying aim of the study—to evaluate the framework's ability to produce speaker-invariant features, regardless of the target speaker.

5. Results

5.1. Quantitative results

We provide the experimental results in Table 2. First, we observe that simply using the (clean) synthetic speech as an augmentation or as the fine-tuning dataset does not result in performance improvement. However, when using synthetic speech as feature guidance, we can observe improvements. In particular, with GRL on a domain classifier between real and synthetic speech, the model improves by 2.9% on the LibriSpeech test-other, 2.6% on the Voxpopuli English test set, and 1.3% on the VCTK dataset. With more fine-grained feature guidance using the Discrete Token Prediction L_{DTP} loss, the improvement becomes more significant, with 4.1%, 4.4% and 3.1% on LibriSpeech, Voxpopuli and VCTK respectively. Since the two losses complement each other on the task of estimating the representation of synthetic speech, fine-tuning the encoder on the losses results in final improvements of 4.7%, 5.3%, and 3.9% on the three evaluation datasets. The improvement provides promising signals of using synthetic speech as guidance for the acoustic modeling process.

5.2. Qualitative results

We visualize the mean-pooled embeddings by the original and adapted Whisper models using t-SNE [29]. Figure 2a) shows the projected embeddings for samples of LibriSpeech test-other. We can observe that the R2S can effectively align the representations of real and synthetic speech. Since the motivation for converting real to synthetic speech features is to remoce undesired information from the encoded representations such as speaker identities or accents, we verify the hypothesis by visualizing the learned embeddings with speaker IDs in Figure 2b) and c) on randomly select 15 identities from LibriSpeech test-other. We can see that the adapted model with R2S better produces speaker-invariant features compared to the original model. We also report a cluster quality metric, the Calinski-Harabaz score [30], to quantify the quality of generated embedding clusters concerning the speaker IDs. A lower DB index suggests more cluster-invariant representations (worse cluster quality). The CH-score is 23.08 for the original model (Figure 2b) and 19.83 for the adapted model (Figure 2c). We also analyzed the learned representations for accented speech from the Voxpopuli dataset (accented English subset), but did not include them in Fig. 2 due to limited space. Nevertheless, our model achieves a lower CH score compared to the original Whisper model (4.66 vs. 5.17), underscoring the effectiveness of R2S in producing accent-invariant features.

6. Conclusion

We present the first study on leveraging synthetic speech to guide the representation learning process of real speech for the task of speech recognition. In particular, we propose R2S with a Gradient Reversal Layer attached to a (real vs. synthetic) speech classifier and a Residual-VQ module to generate pseudo-labels from synthetic speech representations to help the encoder in generating representations more similar to clean and invariant synthetic speech. Experimental results with the Whisper model on three datasets demonstrate the effectiveness of R2S in reducing the WER by $4-5\%$. We qualitatively visualize the learned representations to verify R2S can successfully produce more invariant features with respect to speaker attributes.

7. References

[1] A. Radford *et al.*, “Robust speech recognition via large-scale weak supervision,” in *International conference on machine learning*. PMLR, 2023, pp. 28 492–28 518.

[2] A. Senior *et al.*, “Improving dnn speaker independence with i-vector inputs,” in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2014, pp. 225–229.

[3] V. Peddinti *et al.*, “Reverberation robust acoustic modeling using i-vectors with time delay neural networks.” in *Interspeech*, 2015, pp. 2440–2444.

[4] G. Pironkov *et al.*, “Multi-task learning for speech recognition: an overview.” in *ESANN*, 2016.

[5] S. Sun *et al.*, “An unsupervised deep domain adaptation approach for robust speech recognition,” *Neurocomputing*, vol. 257, pp. 79–87, 2017.

[6] Z. Meng *et al.*, “Domain adaptation via teacher-student learning for end-to-end speech recognition,” in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 268–275.

[7] S. Khurana *et al.*, “Unsupervised domain adaptation for speech recognition via uncertainty driven self-training,” in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6553–6557.

[8] Y. Adi *et al.*, “To reverse the gradient or not: An empirical comparison of adversarial and multi-task learning in speech recognition,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3742–3746.

[9] T. Tsuchiya *et al.*, “Speaker invariant feature extraction for zero-resource languages with adversarial learning,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2381–2385.

[10] Z. Meng *et al.*, “Speaker-invariant training via adversarial learning,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5969–5973.

[11] D. Serdyuk *et al.*, “Invariant representations for noisy speech recognition,” *arXiv preprint arXiv:1612.01928*, 2016.

[12] D. Liang *et al.*, “Learning noise-invariant representations for robust speech recognition,” in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 56–63.

[13] G. Pironkov *et al.*, “Speaker-aware long short-term memory multi-task learning for speech recognition,” in *2016 24th European Signal Processing Conference (EUSIPCO)*. IEEE, 2016, pp. 1911–1915.

[14] Z. Tang *et al.*, “Multi-task recurrent model for speech and speaker recognition,” in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2016, pp. 1–4.

[15] A. Tjandra, S. Sakti, and S. Nakamura, “Listening while speaking: Speech chain by deep learning,” in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 301–308.

[16] A. Rosenberg, Y. Zhang, B. Ramabhadran, Y. Jia, P. Moreno, Y. Wu, and Z. Wu, “Speech recognition with augmented synthesized speech,” in *2019 IEEE automatic speech recognition and understanding workshop (ASRU)*. IEEE, 2019, pp. 996–1002.

[17] Z. Chen, A. Rosenberg, Y. Zhang, G. Wang, B. Ramabhadran, and P. J. Moreno, “Improving speech recognition using gan-based speech synthesis and contrastive unspoken text selection.” in *Interspeech*, 2020, pp. 556–560.

[18] J. Huang, Y. Bai, Y. Cai, and W. Bian, “A study on the adverse impact of synthetic speech on speech recognition,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 10 266–10 270.

[19] A. Meghanani *et al.*, “Score: Self-supervised correspondence fine-tuning for improved content representations,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 12 086–12 090.

[20] M. Cuturi *et al.*, “Soft-dtw: a differentiable loss function for time-series,” in *International conference on machine learning*. PMLR, 2017, pp. 894–903.

[21] C. Wu, Z. Xiu, Y. Shi, O. Kalinli, C. Fuegen, T. Koehler, and Q. He, “Transformer-based acoustic modeling for streaming speech synthesis.” in *Interspeech*, 2021, pp. 146–150.

[22] A. Vaswani, “Attention is all you need,” *Advances in Neural Information Processing Systems*, 2017.

[23] Y. Ganin *et al.*, “Unsupervised domain adaptation by backpropagation,” in *International conference on machine learning*. PMLR, 2015, pp. 1180–1189.

[24] N. Zeghidour *et al.*, “Soundstream: An end-to-end neural audio codec,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 495–507, 2021.

[25] V. D. Oord *et al.*, “Neural discrete representation learning,” *Advances in neural information processing systems*, vol. 30, 2017.

[26] V. Panayotov *et al.*, “Librispeech: an asr corpus based on public domain audio books,” in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[27] C. Wang *et al.*, “Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation,” *arXiv preprint arXiv:2101.00390*, 2021.

[28] V. D. Oord *et al.*, “Wavenet: A generative model for raw audio,” *arXiv preprint arXiv:1609.03499*, vol. 12, 2016.

[29] V. der Maaten *et al.*, “Visualizing data using t-sne.” *Journal of machine learning research*, vol. 9, no. 11, 2008.

[30] T. Caliński *et al.*, “A dendrite method for cluster analysis,” *Communications in Statistics-theory and Methods*, vol. 3, no. 1, pp. 1–27, 1974.