

Improving Code-Switching Speech Recognition with TTS Data Augmentation

Yue Heng Yeo^{*†}, Yuchen Hu[†], Shreyas Gopal[†], Yizhou Peng[†], Hexin Liu[†], and Eng Siong Chng[†]

^{*} Institute for Infocomm Research (I²R), A*STAR, Singapore

[†] College of Computing and Data Science, Nanyang Technological University, Singapore

E-mail: yueheng001@ntu.edu.sg

Abstract—Automatic speech recognition (ASR) for conversational code-switching speech remains challenging due to the scarcity of realistic, high-quality labeled speech data. This paper explores multilingual text-to-speech (TTS) models as an effective data augmentation technique to address this shortage. Specifically, we fine-tune the multilingual CosyVoice2 TTS model on the SEAME dataset to generate synthetic conversational Chinese–English code-switching speech, significantly increasing the quantity and speaker diversity of available training data. Our experiments demonstrate that augmenting real speech with synthetic speech reduces the mixed error rate (MER) from 12.1% to 10.1% on DevMan and from 17.8% to 16.0% on DevSGE, indicating performance gains. These results confirm that multilingual TTS is an effective and practical tool for enhancing ASR robustness in low-resource, conversational code-switching scenarios.

I. INTRODUCTION

Code-switching is an everyday practice where multilingual speakers mix two or more languages into a single conversation, in either intra-sentence or inter-sentence manner, choosing words or grammatical structures that best fit their communicative intent [1]–[3]. In automatic speech recognition, code-switching is particularly challenging because speakers often adjust their intonation, rhythm, and pronunciation when transitioning between languages, demanding ASR systems to track these shifts in real time [4]–[7]. Despite the existing advancements [8]–[12], a significant obstacle for code-switching ASR is the scarcity of realistic, accurately transcribed code-switching datasets, severely limiting model performance.

A prevalent method to address the shortage of code-switching data is audio splicing [13]. This technique synthesizes CS speech by concatenating audio segments from separate monolingual recordings, creating synthetic bilingual utterances without additional data collection. Empirically, ASR systems trained on audio-spliced data have demonstrated improvements in error rates and reduced monolingual bias. However, concatenating audio segments typically results in unnatural prosody and noticeable acoustic discontinuities, introducing co-articulation artifacts that may lead to model overfitting. Consequently, despite being useful for initial experimentation, audio-spliced data exhibits inherent limitations in realism and linguistic coverage compared to advanced TTS-generated synthetic speech, particularly in conversational datasets such as SEAME [10], [14].

Another possible method is TTS augmentation. While early TTS models were considered ineffective for code-switching

ASR due to difficulties in modeling natural prosody, speaker variability, and complex language-switching patterns [10], [15], [16], recent work by Chou et al. [17] demonstrates that synthetic speech generated by advanced TTS models can significantly improve ASR performance. Their self-refining framework, which leverages TTS-synthesized data, achieves significant reduction in error rates ASR task, highlighting the practical effectiveness of TTS augmentation for fine-tuning ASR systems in code-switching scenarios.

Identified key factors for successful augmentation include sufficient text diversity, moderate speaker variation, and appropriate balance between real and synthetic speech [8], [18]. Leveraging such versatile TTS models to generate synthetic data provides an effective and cost-efficient solution because it bypasses the expensive stages of speaker recruitment, studio recording, and manual code-switch transcription by relying solely on readily crawled text and automatically self-labelled speech embeddings to augment real-world code-switching datasets, directly addressing data scarcity [10], [18]. Recent studies have demonstrated substantial performance improvements when ASR systems are trained using synthetic speech generated by advanced multilingual TTS models, significantly reducing the performance gap relative to real-world data [18], [19].

II. RELATED WORKS

Recent advances in ASR increasingly utilize synthetic speech generated by TTS systems to alleviate the data scarcity challenges in low-resource multilingual scenarios [8], [15], [20]. Yang *et al.* [18], for instance, demonstrated significant ASR performance gains by leveraging the multilingual CosyVoice-base TTS model across diverse low-resource domains such as accented speech, minority languages (Korean, Chinese dialects), and specialized vocabulary (e.g., automotive hotwords). Their results emphasized the importance of adequate textual and moderate speaker diversity for effective TTS augmentation.

Nevertheless, while synthetic TTS speech augmentation has proven effective in various linguistic contexts, most prior research has overlooked conversational code-switching, which introduces unique challenges like rapid intra-sentence language switching, informal lexical usage, and complex prosodic patterns [2], [10]. Common prior approaches, such as audio-splicing augmentation by concatenating segments of mono-

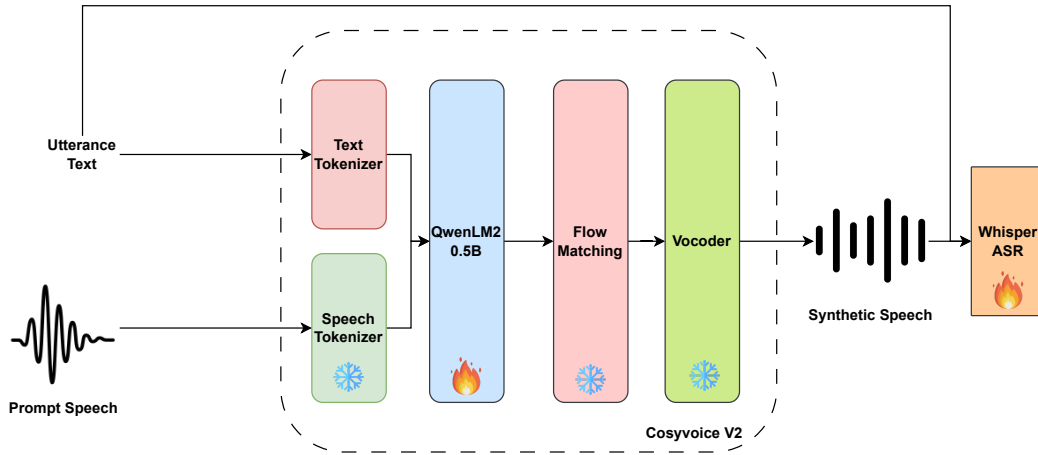


Fig. 1. End-to-end synthetic-data pipeline. Ground-truth text and speech are tokenised, passed through the Qwen-2 language model, a flow-matching decoder and a HiFT vocoder to yield synthetic audio, which is later used to fine-tune Whisper.

lingual audio to form synthetic bilingual utterances, have achieved modest improvements but often produce unnatural prosody and acoustic artifacts, limiting their effectiveness in realistic conversational scenarios [8], [19].

To address these limitations, our study explicitly investigates using the modern multilingual CosyVoice TTS model, fine-tuned specifically on conversational Chinese–English code-switching corpora such as SEAME [9]. By emphasizing realistic conversational structure, spontaneous prosodic variation, and diverse speaker characteristics, we aim to substantially enhance ASR system robustness within complex code-switching conversational contexts, thereby extending existing multilingual augmentation frameworks.

A. Our Contributions

Our contributions in this paper are summarised as follows:

- We demonstrate that multilingual TTS models, specifically CosyVoice fine-tuned on code-switching datasets, effectively capture realistic conversational prosody, informal lexical usage, and rapid intra-sentence language switching. Furthermore, showing that the TTS data is possible for finetuning speech foundation models.
- We identify critical factors for successful TTS augmentation, by adding speaker variation, adding amount of data, and an optimal balance between synthetic and real speech data.
- We confirm the adaptability and effectiveness of our TTS augmentation pipeline by successfully transferring a SEAME-fine-tuned CosyVoice model to a different code-switching corpus (ASCEND), greatly narrowing the gap with real data performance.

III. METHODS

A. CosyVoice TTS

CosyVoice [21] is a scalable, multilingual, zero-shot text-to-speech (TTS) system that relies on supervised semantic tokens

extracted from a multilingual ASR encoder. As summarised in Figure 1, the architecture is organised into four tightly coupled blocks. (1) Text encoder: a language-agnostic BPE front-end converts the input sentence into tokens and aligns them to the speech timeline. (2) Speech tokenizer: built on vector-quantisation over the ASR encoder, this module discretises training audio into low-rate semantic codes. (3) Large language model (LLM) the backbone of the TTS system: treating TTS as an autoregressive sequence generation task, a transformer-based LLM takes in the mixed stream of text and speech tokens and autoregressively predicts the next speech token based on the inputted tokens. (4) Conditional flow-matching decoder: the generated token sequence is up-sampled and passed through a flow-matching network that converts it into mel-spectrograms, which a lightweight vocoder renders as waveform.

CosyVoice is trained in two stages: the speech tokenizer learns from approximately 200k hours of aligned Chinese-English audio, while the full TTS model is trained on an additional 167k hours spanning four languages: Chinese, English, Japanese and Korean. This scale and diversity allow CosyVoice to generate natural, speaker-consistent speech, including fluent intra-sentence code-switching, making it a practical source of synthetic data for strengthening multilingual ASR systems.

B. Whisper ASR

Whisper is a large-scale Transformer-based ASR (ASR) model from OpenAI¹, trained on about 680k hours of multilingual audio. Because its training data spans a broad range of acoustic conditions and spoken languages, Whisper often excels in handling diverse speakers, accents, and noisy recordings.

However, code-switching remains problematic. Although Whisper’s multilingual approach can usually handle multiple languages independently, it can struggle when they appear in

¹Model at <https://github.com/openai/whisper>.

rapid alternation, leading to transcription errors or incorrect language identification.

C. Data Generation Pipeline

Our method has three stages, to data augment reference speech and reproduce more data in terms of speaker variety and volume of data.

1) *TTS tuning*: We begin by adapting CosyVoice 2 to the SEAME domain. Only the QwenLM language-model component is updated; the speech tokeniser, flow-matching decoder and vocoder are kept fixed. During fine-tuning, QwenLM is trained to auto-regressively predict speech-token sequences given SEAME text tokens, allowing it to internalise the corpus’s rapid Mandarin–English alternations, informal phrasing and conversational prosody. This single-module update is computationally lightweight yet sufficient to steer the TTS system toward natural code-switched output while preserving the acoustic fidelity of the original CosyVoice stack.

2) *Synthetic speech generation*: After adaptation, each SEAME transcript is re-synthesised multiple times using different x-vector speaker embeddings sampled from a large pool. The result is a speaker-diverse synthetic corpus that mirrors the original text but enriches timbre, pitch range and speaking-rate variation.

3) *ASR tuning*: The synthetic speech is mixed with the 100 h SEAME ground-truth audio and used to fine-tune Whisper-small model. We compare three conditions: (i) Ground Truth-only, (ii) Ground Truth + TTS (the proposed mix) and (iii) TTS-only. Keeping the Whisper architecture and augmentation recipe unchanged lets us isolate the impact of the additional, speaker-rich synthetic data on code-switching recognition accuracy.

IV. EXPERIMENT SETUP

A. SEAME Dataset

The SEAME (South-East Asia Mandarin-English) corpus is a speech dataset designed specifically to capture spontaneous, conversational code-switching between Mandarin and English among bilingual speakers in Singapore and Malaysia. It contains approximately 192 hours of audio from natural conversations and interviews involving 156 speakers. Conversations cover everyday topics and showcase frequent switches between languages, often even within a single sentence or phrase. Each utterance is carefully transcribed, with clear labels marking language boundaries, making SEAME ideal for training and evaluating automatic speech recognition systems that must handle real-world bilingual interactions. Due to its spontaneous nature, realistic language mixing, and detailed annotations, SEAME is widely used as a standard benchmark for code-switching research and development.

B. Model

a) *CosyVoice fine-tuning*: We adapt the CosyVoice 2’s QWENLM2 (0.5 B parameters) model to the target domain. Optimisation uses Adam with an initial learning rate of 1×10^{-4} . The rate grows linearly during the first 10 000

TABLE I
MIXED-ERROR RATE (MER) OF WHISPER-LARGEV3 ON DEVMAN AND DEVSGE FOR DIFFERENT MIXES OF REAL SPEECH, ORIGINAL-SPEAKER TTS (TTS-O), AND RANDOM-SPEAKER TTS (TTS-R). BOLD MARKS THE LOWEST MER IN EACH TEST SET.

Model	Duration (h)			MER (%)	
	Real	TTS-O	TTS-R	DevMan	DevSGE
Whisper-Largev3	100	-	-	12.1	17.8
	-	100	-	12.5	18.6
	-	-	100	17.7	22.4
	100	100	-	11.1	17.0
	100	-	100	10.1	16.0
	-	-	200	12.2	18.5

updates (warm-up) and then remains constant for the rest of the 200 training epochs.

b) *Whisper ASR fine-tuning*: The ASR back-end starts from the released Whisper-small checkpoint for the ablation studies (~240 M parameters) and is fine-tuned in ESPnet². Input waveforms are converted to 80-bin log-Mel filter-banks (24 kHz, 20 ms window, 12 ms hop); we apply the same SpecAugment two frequency masks (width ≤ 40 bins), five time masks (width $\leq 12\%$ of the utterance) and a five-frame time-warp window. Optimisation uses AdamW ($\beta = 0.9/0.99$, $\epsilon = 1 \times 10^{-6}$, weight decay 0.01). The learning rate follows ESPnet’s `warmup_lr` schedule: it rises linearly from zero to 1×10^{-5} over the first 1 500 updates, then decays with the inverse-square-root rule. Mini-batches are built by counting the total number of spectrogram elements; each update is limited to about 12 M elements, with gradients accumulated over four steps. The language id has been set auto for all experiments.

c) *Evaluation*: ASR quality is reported as mixed-error rate (MER) on DEVMAN and DEVSGE inside ESPNET toolkit for SEAME recipe.

C. Experiment Results

The results in Table I confirm that high-quality TTS is an effective data-augmentation tool for Whisper-Largev3 when synthetic speech is added in addition to the available real recordings. Our baseline of fine-tuning Whisper with only the 100 h of real speech yields 12.1% MER on DevMan and 17.8% on DevSGE. Regenerating those same utterances with CosyVoice2 while keeping the original speaker embeddings (TTS-O) and mixing the two sets one-to-one reduces MER to 11.1% / 17.0%. The most substantial benefit appears when replacing speaker embeddings with randomly sampled ones (TTS-R): combining 100 h of real speech with 100 h of random-speaker synthesis lowers MER further to 10.1% on DevMan and 16.0% on DevSGE, surpassing the ground-truth baseline by roughly two absolute points on each test set. The pattern suggests the crucial ingredient is speaker diversity; synthetic audio that merely repeats original voices contributes less than audio introducing new timbres and prosodies. At the same time, training on 200 h of random-speaker TTS

²<https://github.com/espnet/espnet>

TABLE II

MER (%) OF WHISPER-SMALL ON DEVMAN AND DEVSGE WHEN TRAINED ON SYNTHETIC SPEECH PRODUCED BY A TTS MODEL FINE-TUNED WITH DIFFERENT AMOUNTS OF TARGET-DOMAIN DATA. UTMOS DENOTES THE MEAN OPINION SCORE OF THE CORRESPONDING SYNTHETIC SETS; BOLD MARKS THE BEST VALUE IN EACH COLUMN.

Duration (h)	DevMan	DevSGE	UTMOS
Ground-Truth	13.4	19.2	3.6
10	21.8	26.1	2.9
50	15.2	23.2	3.1
100	13.8	20.1	3.2

alone underperforms the real-only model (12.2% DevMan, 18.5% DevSGE), indicating synthetic data works best as a complement rather than a substitute. Taken together, these findings show that TTS can deliver “almost real” training examples that substantially improve recognition accuracy, provided the synthetic set at least doubles the real-data volume and introduces fresh speaker characteristics rather than duplicating existing ones.

D. Amount of Data to finetune Cosyvoice

To find out how much data we need to finetune Cosyvoice2 to replicate more in-domain data, we ran UTMOS22 [22], an open-source model that predicts a mean-opinion score (MOS) from 1 (poor) to 5 (excellent). The real recordings (ground truth) reach 3.6. When CosyVoice is fine-tuned on reach 10 h of target speech, the MOS reaches to 2.9. Expanding the fine-tune pool to 50 h raises the score to 3.1, and using the full 100 h nudges it to 3.2. The MOS curve climbs only modestly because CosyVoice had already been pre-trained on hundreds of hours of multi-speaker data. The large-scale pre-training taught the model most of the clear pronunciation, smooth pitch, and low noise so even the 10 h version starts from a reasonably high baseline. Extra in-domain hours mainly help the TTS copy the conversational style and rhythm of our corpus, which big MOS models reward only slightly. In contrast, the ASR metrics respond much more: the same jump from 10 h to 100 h cuts MER by roughly nine absolute points. Thus, while MOS gains are little, the larger fine-tune sets remain valuable because they push the synthetic speech closer to the target domain of multi-turn conversation codeswitching speech in ways that matter for Whisper-small’s recognition accuracy.³

E. Amount of Data to Synthesise to finetune Whisper

Table III shows that enlarging the synthetic set beyond the 100-hour ground-truth baseline consistently drives MER down, but the marginal benefit shrinks with each additional block of data. Doubling the training hours to 200 h delivers the most substantial payoff: mixed MER falls by roughly one-quarter compared with the baseline, making this step the single largest improvement in the series. A further rise to 300 h still helps, but the extra gain is only about half of what the previous increment provided. Beyond that point the curve

TABLE III

MER (%) ON DEVMAN AND DEVSGE AS WHISPER-SMALL IS FINE-TUNED WITH INCREASING AMOUNTS OF SYNTHETIC SPEECH. THE LAST COLUMN SHOWS THE RELATIVE MER REDUCTION OBTAINED BY ADDING EACH EXTRA 100-H BLOCK (AVERAGED OVER BOTH TEST SETS).

Synthetic Data (h)	DevMan	DevSGE	Rel. Gain (%)
Ground-Truth	13.4	19.2	–
100	19.0	23.7	–
200	13.3	19.9	23.04
300	11.7	18.2	10.29
400	11.2	17.5	4.06
500	10.9	17.0	2.54

TABLE IV

MER (%) OF WHISPER-SMALL AFTER 300 H OF FINE-TUNING WITH TWO DATA-AUGMENTATION TECHNIQUES (LOWER IS BETTER; BEST SCORES IN BOLD).

Technique	DevMan	DevSGE
GT + Speed perturbation	13.4	19.2
GT + TTS	12.6	18.7

flattens sharply. The 400 h and 500 h conditions shave off just four and three tenths of a percentage point, respectively while incurring the full cost of another one-hundred-hour synthesis run. These results indicate that the ideal amount of data to finetune for Whisper-small lies between two and three times the amount of real data: it captures most of the performance upside without sliding into the zone of rapidly diminishing returns.

F. Data Augmentation Comparison

Another common speech augmentation technique to train ASR systems is speed perturbation. Speed perturbation is a widely used audio-augmentation method that synthetically varies speech tempo by resampling each waveform at a small set of fixed scaling factors typically $0.9 \times$ (slower), $1.0 \times$ (original), and $1.1 \times$ (faster). Because the operation stretches or compresses the time axis while leaving the spectral envelope largely intact, it preserves the speaker’s timbre and linguistic content yet introduces realistic rate-of-speech variability. Applying all three factors effectively triples the amount of training data without requiring additional transcription, providing the acoustic model with broader coverage of temporal dynamics and improving robustness to speaking-rate mismatches at test time.

Table IV contrasts the two similar data augmentation technique fine-tuning for *Whisper-small*. Comparing between the conventional three-speed perturbation ($0.9/1.0/1.1 \times$) with a 300-hour CosyVoice TTS augmentation lowers MER from 13.4% to 12.6% on DEVMAN and from 19.2% to 18.7% on DEVSGE. Both comparison keep the 100 h ground-truth corpus fixed; the performance delta therefore isolates the benefit of speaker and prosody diversity introduced by TTS. Speed-perturbation merely warps temporal dynamics while preserving a single speaker identity, whereas our TTS pipeline injects hundreds of synthetic speaker embeddings, enriching

³<https://github.com/sarulab-speech/UTMOS22>

TABLE V
MER (%) OF WHISPER-SMALL ON THE ASCEND-TEST SET AFTER ~ 9 H OF DOMAIN-SPECIFIC TRAINING, COMPARING GROUND-TRUTH FINETUNING WITH TWO COSYVOICE-GENERATED DATA VARIANTS. BOLD MARKS THE BEST (LOWEST) MER.

Data	ASCEND-Test
Ground-Truth	17.8
CosyVoice zero-shot gen.	25.2
CosyVoice (SEAME-FT) gen.	19.1

the acoustic information to finetune Whisper. Therefore, showing the importance of speaker variety in comparison to purely speed perturbing the data only.

G. Cross-Domain Comparison

ASCEND⁴ is a fully transcribed, 9 h collection of spontaneous Chinese–English code-switching dialogue collected in Hong Kong. The material comprises roughly 12 000 utterances from 38 speakers (21 female, 17 male) with an average duration of 2.4 s per utterance, captured at 16 kHz. Similar to conversational style data in SEAME, ASCEND also reflects conversational turn-taking.

To demonstrate the portability of CosyVoice generated data, we used the SEAME-adapted CosyVoice to synthesise speech for ASCEND, a separate Chinese–English codeswitching dialogue corpus that shares the fast turn-taking and informal lexical mixing typical of everyday conversation. We input the 9 h of ASCEND text transcripts to the SEAME-tuned TTS, and regenerate the exact same utterances using TTS to see the effects of bringing SEAME-adapted CosyVoice to another domain

As summarised in Table V, replacing the collected ASCEND audio with synthetic speech from an unadapted CosyVoice model (“zero-shot”) degrades Whisper-small from 17.8% to 25.2% MER, a 42% relative drop that underscores how strongly ASR performance depends on the prosodic and stylistic match between training and test domains. When the very same TTS engine is first fine-tuned on SEAME, a corpus that shares ASCEND’s conversational, code-switching characteristics and then used to regenerate the ASCEND utterances, MER falls to 19.1%. This SEAME-aligned synthetic data erases three quarters of the error penalty introduced by the zero-shot condition, cutting MER by 24% relative and leaving only a 1.3 % gap to the ground-truth baseline. The result demonstrates that a single round of style adaptation enables CosyVoice to produce training speech that is nearly as effective as real recordings, offering a cost-efficient path for bootstrapping ASR in new conversational code-switching domains without further data collection.

V. CONCLUSIONS AND FUTURE WORK

We investigated multilingual text-to-speech (TTS) as a viable data augmentation technique for addressing the challenge

of limited conversational code-switching data in automatic speech recognition. Our results demonstrate that fine-tuning a modern multilingual TTS model to generate synthetic speech effectively captures the diverse speakers, informal lexical choices, and spontaneous prosody typical of real-world code-switching conversations. The synthetic data produced by our method substantially increases training diversity and realism, providing a practical, cost-efficient way to enhance ASR robustness in low-resource, conversational scenarios.

However, the current augmentation pipeline is constrained by the limited textual diversity inherent in existing transcriptions. As future work, we plan to explore enhancing text variability by employing large language models (LLMs) specifically designed or adapted for multilingual and code-switching text generation. Although reliable and controlled code-switching text generators remain unavailable, their development would enable the synthesis of richer, more varied training examples, further strengthening ASR performance across diverse multilingual conversational domains.

ACKNOWLEDGMENT

The computational resources for this article was performed on resources of the National Supercomputing Centre, Singapore (<https://www.nsc.sg>) in collaboration with Institute for Infocomm Research (I2R A*STAR Singapore)

REFERENCES

- [1] P. Auer, *Code-switching in conversation: Language, interaction and identity*. Routledge, 1999.
- [2] B. E. Bullock and A. J. Toribio, *The Cambridge handbook of linguistic code-switching*. Cambridge University Press, 2009.
- [3] H. Liu, L. P. Garcia, X. Zhang, *et al.*, “End-to-end language diarization for bilingual code-switching speech,” in *Proc. Interspeech*, 2021, pp. 1489–1493.
- [4] H. Li, B. Ma, and K. A. Lee, “Spoken language processing for multilingual environments,” *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 82–91, 2013.
- [5] N. T. Vu, P. K. Gupta, H. Adel, D. Telaar, H. Schütze, and T. Schultz, “First-pass decoding for spoken language understanding systems,” in *Proc. Interspeech*, 2012, pp. 1624–1627.
- [6] H. Liu, X. Zhang, H. Zhang, *et al.*, “Aligning speech to languages to enhance code-switching speech recognition,” *arXiv preprint arXiv:2403.05887*, 2024.
- [7] H. Liu, L. P. Garcia, X. Zhang, A. W. H. Khong, and S. Khudanpur, “Enhancing code-switching speech recognition with interactive language biases,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2024, pp. 10 886–10 890.
- [8] S. Li, X. Chen, and J. Yu, “Data augmentation for code-switching asr: Current challenges and future directions,” in *Proc. IEEE SLT*, 2021, pp. 528–535.

⁴Corpus and license at <https://github.com/HLTCHKUST/ASCEND>.

- [9] D.-C. Lyu, T.-P. Tan, E. S. Chng, and H. Li, “Seame: A mandarin-english code-switching speech corpus in south-east asia,” in *Proc. Interspeech*, 2015, pp. 1982–1986.
- [10] G. I. Winata, Z. Liu, and P. Fung, “Code-switching asr: Advances and challenges,” *IEEE Trans. Audio, Speech, and Language Processing*, vol. 30, pp. 1–17, 2022.
- [11] H. Liu, H. Xu, L. P. Garcia, A. W. H. Khong, Y. He, and S. Khudanpur, “Reducing language confusion for code-switching speech recognition with token-level language diarization,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2023, pp. 1–5.
- [12] X. Zhang, Q. Zhang, H. Liu, *et al.*, “Mamba in speech: Towards an alternative to self-attention,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 1933–1948, 2025.
- [13] A. Hussein, D. Zeinali, O. Klejch, M. Wiesner, B. Yan, *et al.*, “Speech collage: Code-switched audio generation by collaging monolingual corpora,” *arXiv preprint*, vol. 2309.15674, 2023, *Proc. Interspeech 2024*, accepted. [Online]. Available: <https://arxiv.org/abs/2309.15674>.
- [14] T. Nguyen and H.-D. Tran, “Can we train asr systems on code-switch without real code-switch data? case study for singapore’s languages,” 2025, Accepted by *Interspeech 2025*.
- [15] A. Rosenberg, M. Sun, and V. Ramanathan, “Speech recognition challenges for low-resource languages,” *IEEE/ACM Trans. Audio, Speech, and Language Processing*, vol. 27, no. 9, pp. 1531–1542, 2019.
- [16] X. Zhang, D. Liu, H. Liu, *et al.*, “Speaking in wavelet domain: A simple and efficient approach to speed up speech diffusion model,” in *Proc. EMNLP*, 2024, pp. 159–171.
- [17] C.-K. Chou, C.-J. Hsu, H.-L. Chung, *et al.*, “A self-refining framework for enhancing asr using tts-synthesized data,” *arXiv preprint arXiv:2506.11130*, 2025.
- [18] G. Yang, F. Yu, Z. Ma, *et al.*, “Enhancing low-resource asr through versatile tts: Bridging the data gap,” *arXiv preprint arXiv:2410.16726*, 2024. [Online]. Available: <https://arxiv.org/abs/2410.16726>.
- [19] W. Zhang, H. Sun, and L. Li, “Synthetic speech augmentation strategies for multilingual and code-switching asr,” in *Proc. ICASSP*, 2022, pp. 7292–7296.
- [20] D. S. Park, W. Chan, Y. Zhang, *et al.*, “SpecAugment: A simple data augmentation method for automatic speech recognition,” in *Proc. Interspeech*, 2019, pp. 2613–2617.
- [21] Z. Du, Q. Chen, S. Zhang, *et al.*, “Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens,” *arXiv preprint*, vol. 2403.12345, 2024. [Online]. Available: <https://arxiv.org/abs/2403.12345>.
- [22] T. Saeki, D. Xin, W. Nakata, T. Koriyama, S. Takamichi, and H. Saruwatari, “Utmos: Utokyo-sarulab