

A Self-Refining Framework for Enhancing ASR Using TTS-Synthesized Data

Cheng-Kang Chou^{*1,2}, Chan-Jan Hsu^{*1}, Ho-Lam Chung², Liang-Hsuan Tseng²,
Hsi-Chun Cheng², Yu-Kuan Fu³, Kuan Po Huang², Hung-Yi Lee²

¹MediaTek Research ²National Taiwan University ³Nvidia

Abstract—We propose a self-refining framework that enhances ASR performance with only unlabeled datasets. The process starts with an existing ASR model generating pseudo-labels on unannotated speech, which are then used to train a high-fidelity text-to-speech (TTS) system. Then, synthesized speech text pairs are bootstrapped into the original ASR system, completing the closed-loop self-improvement cycle. We demonstrated the effectiveness of the framework on Taiwanese Mandarin speech. Leveraging 6,000 hours of unlabeled speech, a moderate amount of text data, and synthetic content from the AI models, we adapt *Whisper-large-v2* into a specialized model, *Twister*. *Twister* reduces error rates by up to 20% on Mandarin and 50% on Mandarin-English code-switching benchmarks compared to *Whisper*. Results highlight the framework as a compelling alternative to pseudo-labeling self-distillation approaches and provides a practical pathway for improving ASR performance in low-resource or domain-specific settings.

Index Terms—Automatic Speech Recognition, Whisper, Pseudo Labeling, Text-to-Speech, Code-Switching, Self-Refining

I. INTRODUCTION

Automatic speech recognition (ASR) has become a cornerstone technology in modern human–computer interaction, powering applications such as voice assistants, real-time transcription, and accessibility tools. As ASR systems continue to evolve, the demand for transcribed speech data has increased substantially. State-of-the-art generalist ASR models leverage large-scale speech datasets, generally involving at least 25,000 hours and scaling up to millions of hours [1]–[4].

Even with these advances, the growing spectrum of multilingual and domain-specific applications calls for additional attention to model development. Replicating the success of supervised pair training across diverse use cases remains challenging, primarily due to the limited availability of transcribed speech data. In contrast, textual data is considerably more abundant [4]. Contemporary textual corpora curated for generative language models [5], [6], when converted into spoken form, can yield speech data volumes over 100 million hours, far exceeding current reserves.¹ This virtually unlimited data resource presents a compelling opportunity to address the

scarcity of speech data, particularly in underrepresented scenarios such as low-resource or tail languages, code-switching contexts, and specialized domains. There is thus a strong incentive to leverage text-to-speech systems (TTS) to generate paired speech for textual data, to leverage these resources in supervised approaches.

While previous research has touched on the use of TTS systems in scenarios such as personalization [7]–[9] and domain adaptation [10], [11], these applications have largely remained limited in scope until [12]. The emergence of high-fidelity controllable TTS systems [13]–[16] has made it feasible to generate large-scale synthetic speech corpora with realistic prosody and acoustic variability. The high similarity of this synthetic speech to real speech, as reflected by MOS scores, presents new opportunities to extend prior work and enhance the domain adaptability of ASR systems.

In this work, we draw inspiration from recent advancements and propose a generalizable, self-refining framework for ASR systems. Contrary to prior work [12], our work focuses on enhancing existing models on the target language without any handcrafted paired data. The process starts with an existing ASR model generating pseudo-labels on unannotated speech, which are then used to train a high-fidelity text-to-speech (TTS) system, a step that has been covered in prior work [16]. Then, we create synthesized speech text pairs from the TTS model and bootstrap them into the original ASR system, completing the closed-loop self-improvement cycle. Throughout the process, only unlabeled single modality data is needed, making the framework highly extensible.

To illustrate the effectiveness of our framework, we focus on adapting *Whisper-large-v2* to Mandarin, a homophonic language that adds an additional layer of modeling difficulty. We also include the setting of Mandarin-English mixing scenarios to account for the frequent inclusions of code-switching content in real-world communications in Asian linguistic environments. In our demonstration, only 6,000 hours of unlabeled audio and less than 1GB of text (to generate 10,000 hours of synthetic content) is used to fuel the training process.

The resulting ASR model, referred to as *Twister*, achieves substantial performance gains compared to its base model *Whisper-large-v2*. *Twister* reduces error rates by up to 20% on Mandarin and 50% on Mandarin-English code-switching benchmarks. Compared to traditional pseudo-labeling self-distillation approaches [2], [17], the inclusion of the TTS

^{*}Equal contribution. This work was conducted by Cheng Kang Chou during his research internship at MediaTek Research, in collaboration with research scientist Chan-Jan Hsu. Correspondence to: b09705011@ntu.edu.tw.

¹A 15TB corpus roughly equates to 2-3 trillion tokens. Converting to spoken form with a spoken speed of 120 tokens per minute, this corresponds to more than 100 million hours of speech data.

model in the self-refinement loop lowers the real speech data required by 10x, while reaching comparable or better performance. Moreover, this process has the added advantage of being further scalable along two dimensions: by incorporating additional textual content to generate more synthetic speech pairs during the TTS generation phase, and by performing the refinement loop iteratively.

Overall, we present a framework that highlights the potential of TTS-generated speech as a practical alternative to real audio for improving ASR models. We envision this approach as a scalable, flexible, and accessible solution, especially for resource-constrained ASR applications. To support further research, we have open-sourced our model and the accompanying synthetic datasets.

II. RELATED WORK

A. Non-English ASR

Despite the strong multilingual performance of recent ASR models, many prior studies have expressed needs to further improve recognition in low-resource languages using either real [18]–[23] or pseudo-labeled data [17], [21]. However, such data for long-tail languages is much more limited compared to English, with fewer domains represented and primarily consisting of evaluation datasets [24]. Beyond the challenge of acquiring large-scale, realistic data, homophonic languages such as Mandarin introduce an added layer of complexity, as pseudo-labeling can easily reinforce incorrect but identically sounding homophones. In contrast, typographically rendered text produced through precise inputs is typically much more accurate. Motivated by these challenges, we adopt Mandarin ASR as a representative case study of the broader lower-resource ASR problem. Through this lens, we investigate the use of synthetic speech as a scalable approach to addressing data scarcity in the development of ASR models.

B. Code-switching ASR

Code-switching refers to the practice of alternating between two or more languages within a conversation or even a single utterance, and is commonly present in real-world communications in Asian linguistic environments. Traditional ASR models trained on monolingual corpora struggle to generalize across language boundaries, leading to degraded performance in this field. Common decoding errors include misinterpreting code-switching as direct translation or producing phonetic approximations based on the original language. Mitigation efforts of prior work [25]–[27] include providing multiple language tokens [28], or using Speech In-Context Learning (SICL) [29]. We explore synthesizing code-switching data on the utterance level to enhance code-switching capabilities of the model.

C. Realistic TTS

Achieving realistic TTS relies on two key elements: precise modeling of the textual input and the production of natural-sounding speech. In recent developments, Large Language Models (LLMs) have been increasingly leveraged for

text modeling, owing to their sophisticated understanding of semantic content and a degree of inherent knowledge of pronunciation. This comprehensive linguistic competence enables LLMs to more effectively address the sequence-to-sequence challenge of mapping text tokens to corresponding speech tokens. The representation of speech tokens have largely transitioned from unsupervised [30], [31] to supervised approaches [32], resulting in increased semantic information density and improved alignment with text. Finally, prior work has highlighted the efficacy of optimal-transport conditional flow matching (OT-CFM) [14]–[16] in generating speech.

D. ASR with Synthetic Data

The dual nature of speech production and perception in human cognition was introduced as early as the Speech Chain concept [33], which was later captured and modeled by deep learning methods [34]–[36]. Recent progress in large-scale supervised ASR and TTS systems has further advanced the enhancement of ASR using synthetic speech in real-world audio scenarios [7]–[12]. We extend prior work by scaling up both the quantity and quality of data, which not only generalizes the approach to model natural speech in the wild, but also enables direct comparisons with another prominent direction of pseudo-labeling self-distillation ASR enhancement [2], [17].

III. METHODS

In our framework, we employ a specialized TTS system to refine an ASR system, where the TTS system is trained on pseudo-labels from the ASR system. This self-refinement process is illustrated in Fig 1, which what we need initially are three objects framed by red dashed lines. The required data consists solely of N unpaired speech $\mathcal{S} = \{\mathcal{S}_i\}_{i=1}^N$ for generating pseudo-labels $\hat{\mathcal{T}} = \{\hat{\mathcal{T}}_i\}_{i=1}^N$ via the ASR and M unpaired text corpus $\mathcal{T} = \{\mathcal{T}_i\}_{i=1}^M$ for generating paired synthetic speech $\hat{\mathcal{S}} = \{\hat{\mathcal{S}}_i\}_{i=1}^M$ from the TTS. Specifically, $(\mathcal{S}, \hat{\mathcal{T}})$ trains the TTS, while $(\hat{\mathcal{S}}, \mathcal{T})$ is then processed to train the ASR.

In light of recent progress in realistic TTS models, we build on existing TTS models and refer to prior work for their training details, focusing our methodology on training the ASR system. Consistent with previous approaches to ASR training, a core part of our modular pipeline is on the curation of data, which includes two phases: **data collection** (Section III-A) and **data filtering** (Section III-B). To address the distributional gap between synthetic and real data, we incorporate augmentation techniques including **alignment** (Section III-C), **concatenation and random perturbation** (Section III-D). Finally, we bootstrap the original ASR model with the processed semi-supervised pairs to enhance its performance.

A. Data Collection

We start our pipeline with collected speech \mathcal{S} and a ASR model \mathcal{F}_θ to generate pseudo-labels $\hat{\mathcal{T}}$, the data pairs are collected to be $\mathcal{D}_{pseudo} := \{(\mathcal{S}_i, \hat{\mathcal{T}}_i)\}_{i=1}^N$, which served as the training data for our TTS model \mathcal{G}_ϕ . The trained TTS model

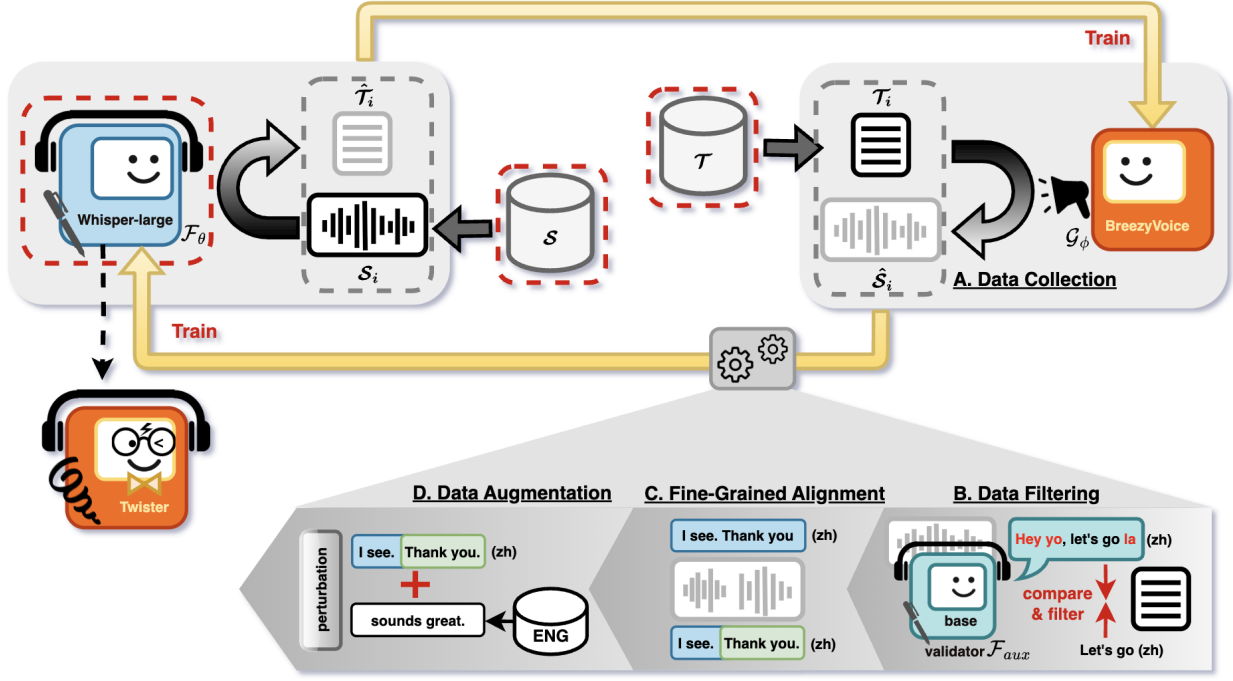


Fig. 1. Overview of the self-refining framework. The process begins by generating pseudo-labeled speech-text pairs from a pre-existing ASR model (framed by red dashed lines, \mathcal{F}_θ) and a collection of unpaired speech (framed by red dashed lines, \mathcal{S}). A TTS model is trained on this pseudo-labeled data and subsequently used to synthesize speech from a large-scale pre-collected text corpus (framed by red dashed line, \mathcal{T}). To ensure data quality, filtering and forced alignment are applied. To support long-form transcription and code-switching, utterance concatenation is performed. Additionally, random audio perturbations are introduced to enhance robustness against acoustic variability. The final curated dataset is used to train the target ASR model.

\mathcal{G}_ϕ is then served as a speech generator based on the collected corpus \mathcal{T} , based on the size of the corpus, we can virtually unlimited speech-text pairs with the dataset described as:

$$\mathcal{D}_{synthesized} := \{(\hat{\mathcal{S}}_i, \mathcal{T}_i)\}_{i=1}^M$$

where $\hat{\mathcal{S}}_i := \mathcal{G}_\phi(\mathcal{T}_i)$.

B. Data filtering:

In generative models such as text-to-speech (TTS), hallucination can occasionally occur, where the synthesized speech includes content not present in the given prompt. Common manifestations of such errors include the insertion of unrelated lexical material and the generation of unintelligible speech. This mismatch between the generated audio and the intended text can propagate to automatic speech recognition (ASR) models trained on such data, leading to systematic omissions of certain patterns of words and phrases during inference.

To address these issues, we incorporate a pre-filtering stage guided by a lightweight auxiliary model, \mathcal{F}_{aux} , or called **validator**, to filter out low-quality data pairs from $\mathcal{D}_{synthesized}$. The quality of each pair is estimated by measuring the similarity between given corpus \mathcal{T}_i and transcriptions generated by \mathcal{F} , i.e., $\mathcal{T}_i^{aux} := \mathcal{F}_{aux}(\hat{\mathcal{S}}_i)$. We define a metric with phoneme error rate (PER) of each sample as $\delta_i := \text{PER}(\mathcal{T}_i, \mathcal{T}_i^{aux})$ with maximum threshold α . Data points that deviate significantly from the original text, indicated by the PER exceeding the threshold (with $\delta_i \geq \alpha$), are removed from the dataset. The remaining dataset is defined as follows:

$$\mathcal{D}_{filtered} := \{(\hat{\mathcal{S}}_i, \mathcal{T}_i) \mid \delta_i \leq \alpha\}$$

The deliberate choice to compare phonemicized versions of the text allows us to discount orthographic variations, such as homophones, which are not indicative of hallucination errors. If the validator’s transcription significantly deviates from the original input text.

C. Fine-grained Alignment

Although the TTS audio is synthesized from a predefined text corpus, fine-grained alignment between the speech and text with respect to prosodic boundaries is not revealed during the generation process. The explicit fine-grained alignment is important for two reasons. First, timecode supervision is indispensable for models that handle long-range audio using a divide-and-conquer approach. For example, *Whisper* has a maximum speech context length of 30 seconds. Audio longer than this must be split into smaller segments, with ASR applied to each segment individually before the transcripts are stitched together (More details in Section III-D). In this process, precise timecodes are essential to enable seamless merging of segments and ensure optimal overall performance. Second, from an application standpoint, ASR systems are widely used to generate subtitles, and accurate phrase-level segmentation is crucial to make subtitles easy for viewers to follow.

To mitigate this issue, we employ Montreal Forced Aligner [37] to conduct forced alignment, resegmenting the transcrip-

tions to fine-grained snippets based on the acoustic characteristics of the synthesized speech, rather than relying solely on the per-instance boundary of the original speech-text pair. This process yields time-aligned speech segments with durations of approximately 3 to 5 seconds. Consequently, the model is exposed to segmentation cues that closely correspond to the underlying speech signal, thereby enhancing its ability to generalize to authentic prosodic variations encountered in real-world scenarios.

D. Data Augmentation

1) *Preserving Long-form ASR Capabilities*: For ASR model \mathcal{F}_θ that accepts audio inputs up to L_{\max} seconds per inference window (e.g., 30 seconds for *Whisper*), clips exceeding this duration are classified as *long-form*. Long-form audio modeling requires segmenting the input into shorter chunks to satisfy the model’s input constraints. Training the model solely on short utterances (typically 5 to 25 seconds) creates a distributional mismatch between the training data and the L_{\max} -second segments encountered during long-form transcription and degrades performance. To address this challenge, we augment L_{\max} -second audios from existing samples to simulate chunked long-form audio.

To obtain long-form audio from short-form ones, we recursively append utterances to an existing clip until the total length slightly exceeds L_{\max} , and cutoff the audio at the L_{\max} second mark. However, pairing this with the transcription corresponding to the first L_{\max} seconds (denoted as $\mathcal{T}_{\leq L_{\max}}$) is suboptimal, as this introduces segmentation discontinuities that destabilize chunk merging. Instead, we backtrack to the nearest prosodic boundary L_{bound} within the L_{\max} time constraint, identified through forced alignment (cf. Section III-C). The transcript is then truncated at L_{bound} , omitting any text corresponding to speech in $(L_{\text{bound}}, L_{\max}]$. A special tag is appended to the end of the text sequence to indicate continuation. During inference, the subsequent chunk begins from the last detected prosodic boundary.

2) *Enhancing Code-switching ASR Capabilities*: Given the limited availability of open-source datasets with code-switching, synthesizing code-switching data becomes crucial for developing robust multilingual models. To this end, we simulate code-switching scenarios by concatenating utterances from different languages up to a maximum duration of L_{\max} seconds. While this approach primarily reflects sentence-level code-switching, it provides a reasonable approximation of word-level code-switching observed in real-world data.

3) *Random Audio Perturbations*: Finally, we apply random audio perturbations such as background noise injection and temporal blurring to improve the model’s robustness to acoustic variability.

E. Data Mixing

While not a distinct processing step, we ensured that all types of data are thoroughly represented in the final corpora. The resulting datasets exhibit multi-faceted diversity, comprising a balanced mix of long and short audio segments across English, Mandarin, and code-switching utterances.

IV. EXPERIMENTS

We demonstrate the effectiveness of the framework by selecting Taiwanese Mandarin as our adaptation target. The refined ASR model is named as *Twister* (TTS-enhanced Whisper), which supports English, Mandarin, and bilingual code-switching settings.

A. Model

We utilize *Whisper-large-v2* as our ASR model, denoted as \mathcal{F}_θ , due to its reasonable performance on the target languages. While this model alone suffices for the backbone of our framework, recent advancements in developing a corresponding \mathcal{G}_ϕ have yielded encouraging outcomes. Namely, *BreezyVoice*, a Taiwanese Mandarin TTS system based on the *CozyVoice* architecture, demonstrates state-of-the-art, highly realistic speech synthesis. Given its strong performance, we adopt *BreezyVoice* directly as \mathcal{G}_ϕ , and refer readers to the original work for details regarding its training procedure of the TTS. We use *Whisper-base* as the validator to prefilter synthesized data pairs.

B. Data

Our raw datasets $\{(T_i, \hat{S}_i)\}_{i=1}^N$ include three types of speech sources: Mandarin, English, and Mandarin-English code-switching. For Mandarin, we curated textual content from ODC-By licensed FineWeb2 [38]. Subsequently, we synthesized a large-scale speech corpus of approximately 10,000 hours using *BreezyVoice* (Section III-A), corresponding to $\mathcal{D}_{\text{synthesized}}$. To ensure acoustic diversity and speaker variation, we synthesized speech using voice samples from over 200 Mandarin speakers. To meet the demands of English and code-switching data, we incorporated open-source ASR datasets **CommonVoice** [39] and **NTUML2021** [26], which are distributed under permissive licenses. The distribution and volume of these raw datasets are summarized in Table I. These datasets serve as the basis for further data augmentation.

TABLE I
DATA DISTRIBUTION OF COLLECTED RAW CORPUS

| Dataset Name | Type | Language | Total Hours |
|------------------|--------|----------------|-------------|
| ODC Synth | Synth. | Mandarin | 10,000 |
| CommonVoice17-EN | Real | English | 1,738 |
| NTUML2021 | Real | Code-switching | 11 |

We set the data filtering threshold α to 0.6, using the validator model *Whisper-base*, resulting in a filtered Mandarin speech dataset with 4,000 hours of audio remaining. The audios are subsequently aligned to obtain fine-grained timestamps. At this stage, the dataset is fully prepared for augmentation.

The augmentation process aims to construct two distinct subsets: a Mandarin set and a mixed (code-switching) set. The Mandarin set is formed exclusively from concatenating Mandarin clips sampled from **ODC-Synth**, and some short-form English clips are preserved for replay purpose. For the mixed set, we expand the limited natural code-switching corpus

NTUML2021 by creating artificial code-switching samples through random mix-and-match combinations of English clips from **CommonVoice17** and Mandarin clips from **ODC-Synth**. Table II summarizes the data quantities of the final dataset, which comprises a mix of long and short audio segments in English, Mandarin, with English included minimally solely to mitigate forgetting. During training, data points are sampled uniformly from all instances across all datasets.

TABLE II
DATA DISTRIBUTION OF FINAL TRAINING CORPUS

| Dataset Name | Type | Length | Language | Total Hours |
|----------------|------------|--------|----------|-------------|
| Mandarin Long | Synth. | Long | Zh | 4,000 |
| Mandarin Short | Synth. | Short | Zh | 70 |
| English | Real | Short | En | 10 |
| Code-switching | Real+Synth | Long | C.S. | 1,715 |

C. Training Details

To enable unified processing of Mandarin, English, and code-switching utterances, we initialized a shared language embedding by taking the element-wise average of the language-token embeddings for $\langle |zh| \rangle$ and $\langle |en| \rangle$. This strategy provides a language-neutral initialization that is universal across intended use cases. This approach is inspired by the zero-shot inference results of the base model, *Whisper-large-v2*. As shown in Table III, the mixed-language embedding achieves comparable performance on monolingual Mandarin and English benchmarks, and even improves accuracy on code-switching tasks, despite not being explicitly trained for this configuration.

We train *Whisper-large-v2* on the described dataset, totaling 10000 steps with a batch size of 256. The learning rate was set to 2×10^{-5} . Training was conducted on 8 NVIDIA H100 GPUs.

TABLE III
MER OF *Whisper-large-v2* ACROSS DATASETS USING DIFFERENT LANGUAGE TAGS. MIXED EMBEDDING PERFORMS BEST AT CODE-SWITCHING SCENARIOS WHILE RETAINING PERFORMANCE ON MONOLINGUAL SETS.

| Dataset \ Setting | Mixed | ZH | EN |
|---------------------|-------|--------|-------|
| ASCEND-EN | 29.63 | 101.54 | 27.20 |
| ASCEND-ZH | 17.65 | 13.75 | 78.50 |
| ASCEND-Mixed* | 21.90 | 22.69 | 80.61 |
| CommonVoice16-zh-TW | 11.94 | 9.02 | 42.08 |
| CSZS-zh-en* | 26.25 | 44.28 | 54.27 |
| ML-2021-long* | 6.82 | 6.13 | 94.76 |

* Code-switching datasets.

D. Evaluation Data

For performance evaluation, we use different datasets to cover Chinese-Mandarin, Taiwanese-Mandarin, English, code-switching, short-form, long-form datasets.

1) *ASCEND*: ASCEND [40] is a spontaneous, multi-turn conversational dialogue corpus featuring Chinese-English code-switching, collected in Hong Kong. To further analyze the impact of language on performance, we partitioned the

dataset into three subsets: EN (English-only), ZH (Mandarin-only), and Mixed (Mandarin-English code-switching).

2) *CommonVoice16*: CommonVoice16-zh-TW [39] is a dataset for Taiwanese-Mandarin, which is a subset of CommonVoice 16-1 version, which is publicly available, serving as an evaluation set for assessing short-form Mandarin ASR performance.

3) *CSZS-zh-en*: CSZS-zh-en [25] is a dataset that contains code-switching data adopted from the Amazon Polly text-to-speech system to synthesize utterances.

4) *ML-lecture-2021-long*: ML-Lecture-2021-Long [26] is a testing dataset comprising approximately 5 hours of recordings derived from the "Machine Learning" course at National Taiwan University, which is publicly available. It features code-switching utterances with a predominance of Taiwan-Mandarin, serving as an in-domain evaluation set for assessing long-form code-switching ASR performance.

5) *FormosaSpeech*: FormosaSpeech [24] is a multi-speaker evaluation benchmark for Taiwanese Mandarin, comprising both news and text reading materials, and featuring monologues as well as dialogues.

6) *Formosa-Suite*: The Formosa Suite represents our own in-domain in-house Taiwanese Mandarin speech corpora designed to evaluate long-form ASR performance. It comprises four subsets covering various subjects: **Formosa-Go** (tourism and location narratives), **Formosa-Show** (talk shows and stand-up comedy), **Formosa-Course** (online-course lectures across academic disciplines), and **Formosa-General** (a broad mix of topics including technology, lifestyle, and food, etc.). Each subset contains 3-minute clips and ranges from 5 to 10 hours in total test duration, collectively covering a diverse range of speaking styles, domains, and speaker conditions.

E. Evaluation Metrics

We adopt Mixed Error Rate (MER) as evaluation metric to assess model performance on code-switching speech recognition tasks. MER computes character error rate (CER) for Mandarin segments and word error rate (WER) for English segments, thereby aligning with the natural granularity of individual languages.

V. RESULTS

To evaluate the effectiveness of our proposed model, *Twister*, we conduct a comprehensive comparison against a suite of baselines built upon the *Whisper* architecture. The primary baseline is *Whisper-large-v2*, the ASR model prior to self-refinement.

In addition, we include two models that also claims improvement on Taiwanese Mandarin. *Whisper-large-v3* and *COOL-Whisper*. Coincidentally, both models leverage pseudo-labeling techniques, followed by distillation approaches using the paired data. *Whisper-large-v3* is an upgraded version of *Whisper-large-v2*, trained on 1 million hours of high-quality speech-text pairs and an additional 4 million hours of pseudo-labeled audio. Using a 4.4% Mandarin composition in the *Whisper-v1* training data as a reference, we estimate

TABLE IV

MER OF DIFFERENT TAIWAN MANDARIN ASR SYSTEMS. BEST PERFORMING RESULTS ARE MARKED IN **BOLD**, AND RELATIVE WORD ERROR RATE REDUCTION COMPARED TO WLV2-AUTO ARE MARKED IN BRACKETS. COLUMNS WITH AN ASTERISK DENOTE CODE-SWITCHING DATASETS.

| Dataset\Model | WLV2-Oracle↓ | WLV2-Auto↓ | WLV3-auto↓ | COOL-Whisper↓ | Twister (Ours)↓ |
|-----------------------------|-------------------|-------------|------------|---------------|------------------------|
| Short Audio Datasets | | | | | |
| ASCEND-OVERALL* [40] | 21.14 (AUTO) | 21.14 | 23.22 | 19.71 | 17.74 (-16.08%) |
| - ASCEND-EN | 27.20 (EN) | 27.36 | 27.21 | 29.39 | 26.64 (-2.63%) |
| - ASCEND-ZH | 13.75 (ZH) | 17.49 | 17.41 | 18.90 | 16.04 (-8.29%) |
| - ASCEND-MIX* | 21.01 (AUTO) | 21.01 | 25.13 | 17.34 | 16.38 (-22.01%) |
| CommonVoice16-zh-TW [39] | 9.02 (ZH) | 9.84 | 8.95 | 11.86 | 7.97 (-19%) |
| CSZS-zh-en* [25] | 29.49 (AUTO) | 29.49 | 26.43 | 20.90 | 13.01 (-55.88%) |
| Long Audio Datasets | | | | | |
| ML-lecture-2021-long* [26] | 6.13 (ZH) | 6.13 | 6.41 | 6.37 | 4.98 (-18.76%) |
| Formosa-Go | 15.03 (ZH) | 15.03 | 14.90 | 16.83 | 13.61 (-9.44%) |
| Formosa-Show | 29.18 (ZH) | 29.18 | 27.80 | 29.78 | 27.58 (-5.48%) |
| Formosa-Course | 9.50 (ZH) | 9.50 | 9.67 | 11.12 | 9.94 (+0.44%) |
| Formosa-General | 11.45 (ZH) | 11.45 | 11.46 | 13.33 | 11.37 (-0.69%) |
| FormosaSpeech [24] | 22.34 (ZH) | 22.34 | 21.22 | 26.71 | 22.09 (-1.12%) |

that *Whisper-large-v3* was exposed to approximately 220,000 hours of Mandarin data throughout training. *COOL-Whisper* is a lightweight model comparable in size to *Whisper-medium*, with approximately half the number of parameters of *Whisper-large-v2*. It was trained using k2d [17] on a dataset comprising 60,000 hours of Taiwanese Mandarin course materials. Given the inherent code-switching present in these recordings, *COOL-Whisper* provides a strong baseline for evaluation on code-switching benchmarks. We summarize evaluation results in Table IV. The column *WLV2-Auto* is the standard inference scheme of *Whisper-large-v2* where the language tag is automatically decided by the model, whereas *WLV2-Oracle* denotes the best-case performance among three different inference configurations (automatic detection, forced Mandarin token, and forced English token).

A. Comparisons with *Whisper-large-v2*

Compared to the original *Whisper-large-v2* model, *Twister* demonstrates significant performance improvements on both Chinese and code-switching datasets. Notably, on the CSZS benchmark, it achieves a substantial WERR of 55.88%, prior methods that rely on more elaborate evaluation schemes [26]. In benchmarks where automatic language detection underperforms compared to settings where the language is provided (i.e., *WLV2-Auto* >> *WLV2-Oracle*), *Twister* demonstrates moderate to substantial improvements. Specifically, it achieves relative WERRs of 8.29% and 19% on ASCEND-ZH and CommonVoice16-zh-tw, respectively. This suggests that our mixed embedding approach effectively mitigates language detection errors, particularly in cases involving noisy or ambiguous audio inputs. Interestingly, a slight performance gain is also observed on ASCEND-EN, which we attribute to the use of synthetic data simulating Asian-accented English. Finally, we observe consistent gains on long-form audio, indicating that our augmentation method successfully extends short-form data into plausible long-form variants that transfer well to real-world long-form speech. Overall, the performance improvements of *Twister* compared to *Whisper-large-v2* demonstrate

that the framework emerges as a positive reinforcement loop for self-refinement. These results encourage future research to explore further scaling in the TTS generation phase or dynamics of iterative refinement.

B. Comparisons with other Models

Additional baselines included in our comparison are *Whisper-large-v3* and *COOL-Whisper*, both of which utilize large-scale audio datasets and *Whisper-large-v2* in conjunction with a distillation methodology. From Table IV, it can be observed that *Twister* performance outperforms other approaches in all but one benchmark. Our findings emphasize the critical role of content in distilling speech-text pairs for ASR, aligning with prior TTS-based approaches [9]. Moreover, our method is notably more data-efficient: whereas previous methods utilize over 60,000 hours of raw speech data, *Twister* achieves superior results with at least 10 times less. These findings demonstrate that integrating TTS into the framework effectively decreases the dependence on real speech data, addressing a longstanding bottleneck in low-resource speech modeling.

VI. CONCLUSION

In this work, we propose a self-refining framework that enhances ASR performance with only unlabeled datasets. We outline a methodology that leverages high-quality TTS-synthesized audio to bootstrap and enhance the performance of the original ASR system. Results show that our framework is effective in Mandarin and code-switching scenarios for both long-form and short-form audio, with performance gains up to 55.88% compared to the original model *Whisper-large-v2*. Compared to pseudo-labeling self-distillation approaches employed by *Whisper-large-v3* and *Cool-whisper*, our method achieves comparable or superior performance with significantly higher data efficiency. Overall, our framework addresses the limitations posed by real data scarcity and offers a generalizable solution for adapting ASR systems to low-resource and underrepresented speech domains.

ACKNOWLEDGMENTS

We thank NVIDIA for providing access to the Taipei-1 supercomputer.

REFERENCES

- [1] NVIDIA, “Canary-1b,” <https://huggingface.co/nvidia/canary-1b>, 2023.
- [2] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever, “Robust speech recognition via large-scale weak supervision,” *OpenAI*, 2022.
- [3] Microsoft Research, “Phi-4-mini technical report: Compact yet powerful multimodal language and multimodal models,” *arXiv preprint arXiv:2503.01743*, 2024.
- [4] Yu Zhang, Wei Chen, Zelin Wang, et al., “Usm: Scaling automatic speech recognition beyond 100 languages,” *arXiv preprint arXiv:2303.01037*, 2023.
- [5] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample, “Llama: Open and efficient foundation language models,” *arXiv preprint arXiv:2302.13971*, 2023.
- [6] Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al., “Olmo: Accelerating the science of language models,” in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024, pp. 15789–15809.
- [7] Xianrui Zheng, Yulan Liu, Deniz Gunceler, and Daniel Willett, “Using synthetic audio to improve the recognition of out-of-vocabulary words in end-to-end asr systems,” in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5674–5678.
- [8] Vladimir Bataev, Roman Korostik, Evgeny Shabalin, Vitaly Lavrukhin, and Boris Ginsburg, “Text-only domain adaptation for end-to-end asr using integrated text-to-mel-spectrogram generator,” in *Proc. Interspeech 2023*, 2023, pp. 2928–2932.
- [9] Karren Yang, Ting-Yao Hu, Jen-Hao Rick Chang, Hema Swetha Koppula, and Oncel Tuzel, “Text is all you need: Personalizing asr models using controllable speech synthesis,” in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.
- [10] Hsuan Su, Ting-Yao Hu, Hema Swetha Koppula, Raviteja Vemulapalli, Jen-Hao Rick Chang, Karren Yang, Gautam Varma Mantena, and Oncel Tuzel, “Corpus synthesis for zero-shot asr domain adaptation using large language models,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 12326–12330.
- [11] Hsuan Su, Hua Farn, Fan-Yun Sun, Shang-Tse Chen, and Hung-Yi Lee, “Task arithmetic can mitigate synthetic-to-real gap in automatic speech recognition,” in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2024, pp. 8905–8915.
- [12] Guanrou Yang, Fan Yu, Ziyang Ma, Zhihao Du, Zhifu Gao, Shiliang Zhang, and Xie Chen, “Enhancing low-resource asr through versatile tts: Bridging the data gap,” in *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2025, pp. 1–5.
- [13] Eren Gölge and The Coqui TTS Team, “Coqui tts,” <https://github.com/coqui-ai/TTS>, 2021. If you want to cite [name], feel free to use this (but only if you loved it [emoji]).
- [14] Shivam Mehta, Ruiho Tu, Jonas Beskow, Éva Székely, and Gustav Eje Henter, “Matcha-tts: A fast tts architecture with conditional flow matching,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 11341–11345.
- [15] Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, et al., “Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens,” *arXiv preprint arXiv:2407.05407*, 2024.
- [16] Chan-Jan Hsu, Yi-Cheng Lin, Chia-Chun Lin, Wei-Chih Chen, Ho Lam Chung, Chen-An Li, Yi-Chang Chen, Chien-Yu Yu, Ming-Ji Lee, Chien-Cheng Chen, et al., “Breezyvoice: Adapting tts for taiwanese mandarin with enhanced polyphone disambiguation—challenges and insights,” *arXiv preprint arXiv:2501.17790*, 2025.
- [17] Liang-Hsuan Tseng, Zih-Ching Chen, Wei-Shun Chang, Cheng-Kuang Lee, Tsung-Ren Huang, and Hung-yi Lee, “Leave no knowledge behind during knowledge distillation: Towards practical and effective knowledge distillation for code-switching asr using realistic data,” in *2024 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2024, pp. 118–125.
- [18] Dawit Ketema Gete, Bedru Yimam Ahmed, Tadesse Destaw Belay, Yohannes Ayana Ejigu, Sukairaj Hafiz Imam, Alemu Belay Tessema, Mohammed Oumer Adem, Tadesse Amare Belay, Robert Geislinger, Umma Aliyu Musa, et al., “Whispering in amharic: Fine-tuning whisper for low-resource language,” *arXiv preprint arXiv:2503.18485*, 2025.
- [19] Per E Kummervold, Javier de la Rosa, Freddy Wetjen, Rolv-Arild Braaten, and Per Erik Solberg, “Whispering in norwegian: Navigating orthographic and dialectic challenges,” in *Proc. Interspeech 2024*, 2024, pp. 3984–3988.
- [20] Golshid Shekoufandeh, Paul Boersma, and Antal van den Bosch, “Improving the inclusivity of dutch speech recognition by fine-tuning whisper on the jasmin-cgn corpus,” *arXiv preprint arXiv:2502.17284*, 2025.
- [21] Sanjay Rijal, Shital Adhikari, Manish Dahal, Manish Awale, and Vaghawan Ojha, “Whisper finetuning on nepali language,” *arXiv preprint arXiv:2411.12587*, 2024.
- [22] Mark Bajo, Haruka Fukukawa, Ryuji Morita, and Yuma Ogasawara, “Efficient adaptation of multilingual models for japanese asr,” *arXiv preprint arXiv:2412.10705*, 2024.
- [23] Vincenzo Timmel, Claudio Paonessa, Reza Kakooee, Manfred Vogel, and Daniel Perruchoud, “Fine-tuning whisper on low-resource languages for real-world applications,” *arXiv preprint arXiv:2412.15726*, 2024.
- [24] Hao-Chien Lu, Chung-Chun Wang, Jhen-Ke Lin, and Tien-Hong Lo, “The NTNU ASR system for Formosa speech recognition challenge 2023,” in *Proceedings of the 35th Conference on Computational Linguistics and Speech Processing (ROCLING 2023)*, Jheng-Long Wu and Ming-Hsiang Su, Eds., Taipei City, Taiwan, Oct. 2023, pp. 397–402, The Association for Computational Linguistics and Chinese Language Processing (ACLCLP).
- [25] Kuan-Po Huang, Chih-Kai Yang, Yu-Kuan Fu, Ewan Dunbar, and Hung-yi Lee, “Zero resource code-switched speech benchmark using speech utterance pairs for multiple spoken languages,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 10006–10010.
- [26] Chih-Kai Yang, Kuan-Po Huang, Ke-Han Lu, Chun-Yi Kuan, Chi-Yuan Hsiao, and Hung-yi Lee, “Investigating zero-shot generalizability on mandarin-english code-switched asr and speech-to-text translation of recent foundation models with self-supervision and weak supervision,” in *2024 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*. IEEE, 2024, pp. 540–544.
- [27] Feng-Ting Liao, Yung-Chieh Chan, Yi-Chang Chen, Chan-Jan Hsu, and Da-shan Shiu, “Zero-shot domain-sensitive speech recognition with prompt-conditioning fine-tuning,” in *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2023, pp. 1–8.
- [28] Puyuan Peng, Brian Yan, Shinji Watanabe, and David Harwath, “Prompting the hidden talent of web-scale speech models for zero-shot task generalization,” in *Proc. Interspeech 2023*, 2023, pp. 396–400.
- [29] Siyin Wang, Chao-Han Yang, Ji Wu, and Chao Zhang, “Can whisper perform speech-based in-context learning?,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 13421–13425.
- [30] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi, “High fidelity neural audio compression,” *Transactions on Machine Learning Research*.
- [31] Zhihao Du, Shiliang Zhang, Kai Hu, and Siqi Zheng, “Funcodec: A fundamental, reproducible and integrable open-source toolkit for neural speech codec,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 591–595.
- [32] Keyu An, Qian Chen, Chong Deng, Zhihao Du, Changfeng Gao, Zhifu Gao, Yue Gu, Ting He, Hangrui Hu, Kai Hu, et al., “Funaudiollm: Voice understanding and generation foundation models for natural interaction between humans and llms,” *CoRR*, 2024.
- [33] P.B. Denes and E. Pinson, *The Speech Chain*, Anchor books. Worth Publishers, 1993.
- [34] Andros Tjandra, Sakriani Sakti, and Satoshi Nakamura, “Listening while speaking: Speech chain by deep learning,” in *2017 IEEE Automatic*

Speech Recognition and Understanding Workshop (ASRU). IEEE, 2017, pp. 301–308.

- [35] Andros Tjandra, Sakriani Sakti, and Satoshi Nakamura, “Machine speech chain with one-shot speaker adaptation,” *Interspeech 2018*, 2018.
- [36] Sahoko Nakayama, Andros Tjandra, Sakriani Sakti, and Satoshi Nakamura, “Speech chain for semi-supervised learning of japanese-english code-switching asr and tts,” in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 182–189.
- [37] Michael McAuliffe, Michaela Socolof, Elias Stengel-Eskin, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger, “Montreal forced aligner [computer program],” <http://montrealcorpus-tools.github.io/Montreal-Forced-Aligner/>, 2017, Retrieved 05 May 2017.
- [38] Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Martin Jaggi, Leandro von Werra, and Thomas Wolf, “Fineweb2: A sparkling update with 1000s of languages,” Dec. 2024.
- [39] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, “Common voice: A massively-multilingual speech corpus,” in *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, 2020, pp. 4211–4215.
- [40] Holy Lovenia, Samuel Cahyawijaya, Genta Winata, Peng Xu, Yan Xu, Zihan Liu, Rita Frieske, Tiezheng Yu, Wenliang Dai, Elham J Barezi, et al., “Ascend: A spontaneous chinese-english dataset for code-switching in multi-turn conversation,” in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 2022, pp. 7259–7268.