UniCoM: A Universal Code-Switching Speech Generator

Sangmin Lee[1], Woojin Chung[1], Seyun Um[1], Hong-Goo Kang[1]

[1]Dept. of Electrical & Electronic Engineering, Yonsei University, South Korea, Seoul

Correspondence: {sangmin_lee, woojinchung, syum}@dsp.yonsei.ac.kr, hgkang@yonsei.ac.kr

Abstract

Code-switching (CS), the alternation between two or more languages within a single speaker's utterances, is common in real-world conversations and poses significant challenges for multilingual speech technology. However, systems capable of handling this phenomenon remain underexplored, primarily due to the scarcity of suitable datasets. To resolve this issue, we propose **Uni**versal **Co**de-**M**ixer (UniCoM), a novel pipeline for generating high-quality, natural CS samples without altering sentence semantics. Our approach utilizes an algorithm we call **S**ubstituting **WORD**s with **S**ynonyms (SWORDS), which generates CS speech by replacing selected words with their translations while considering their parts of speech. Using UniCoM, we construct **C**ode-**S**witching **FLEURS** (CS-FLEURS), a multilingual CS corpus designed for automatic speech recognition (ASR) and speech-to-text translation (S2TT). Experimental results show that CS-FLEURS achieves high intelligibility and naturalness, performing comparably to existing datasets on both objective and subjective metrics. We expect our approach to advance CS speech technology and enable more inclusive multilingual systems.

1 Introduction

Multilingual speech technology has advanced rapidly in recent years. Early research primarily focused on similar language families (Toshniwal et al., 2018) or a small range of languages (Zhou et al., 2017), but the latest developments over the recent few years have significantly increased model sophistication. Researchers have scaled models (Radford et al., 2023; Zhang et al., 2023) to support over 100 languages directly, or more than 1,000 languages (Pratap et al., 2024) through language-specific modules (Houlsby et al., 2019). However, most approaches are designed for monolingual utterances and struggle to process speech containing multiple mixed languages effectively.

In particular, code-switching (CS), the practice of alternating between languages within a conversation, is prevalent in multilingual communities. While CS-ASR has been widely studied, most efforts have centered on major languages (e.g., Arabic (Alharbi et al., 2024), Indic (Kumar et al., 2021), Mandarin (Liu et al., 2024), paired with English), leaving research on broader languages relatively underexplored. In comparison, research on CS-S2TT remains at an early stage, with recent efforts (Weller et al., 2022; Shankar et al., 2024).

Given this landscape, we identify two key questions: *(1) Why has CS research lagged behind multilingual speech technologies?* and *(2) Why are existing studies confined to a few major languages?* We attribute these limitations primarily to the scarcity and accessibility of CS datasets. To elaborate, CS occurs at both the inter-sentential (between sentences) and intra-sentential (within a sentence) levels (Gardner-Chloros, 2009), and constructing datasets for these cases, particularly intra-sentential CS with diverse language pairs, requires a substantial number of fluent multilingual speakers. However, their rarity and the high cost of data collection pose significant challenges.

In this context, there are two potential data augmentation methods to address these challenges: CS speech generation using a multilingual text-to-speech (TTS) system (Nakayama et al., 2018; Yu et al., 2023) or combining existing monolingual sources (Seki et al., 2018; Dhawan et al., 2023). The first approach generates natural CS samples by preserving speaker identity across utterances. However, scaling this method is challenging due to the complexities of massively multilingual TTS, such as resource imbalance and prosodic variation. The second approach synthesizes CS samples from abundant monolingual data but may lose linguistic nuances and speaker identity when utterances from different languages are simply concatenated, degrading the quality of intra-sentential CS.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 13273–13288
November 4-9, 2025

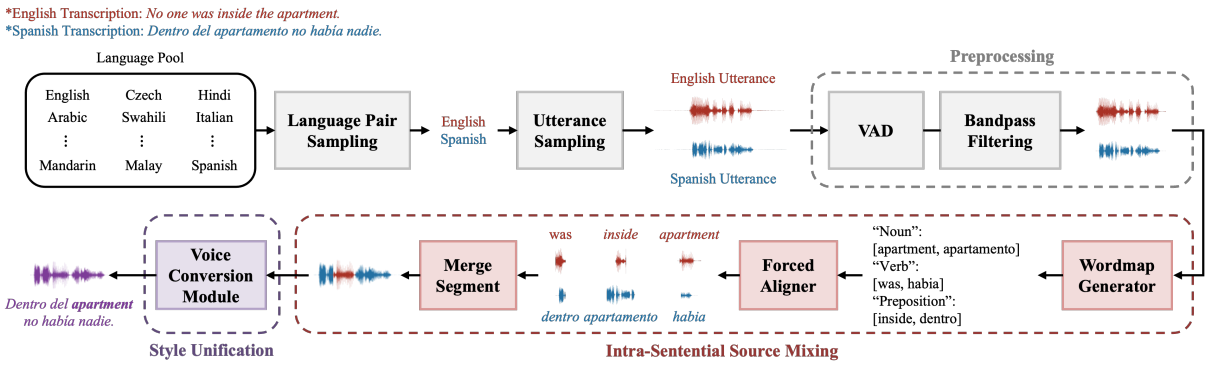


Figure 1: The overall pipeline of UniCoM. The bold region of the final transcription denotes the code-switching.

To address these limitations, we propose **Uni**versal **Co**de-**M**ixer (UniCoM), capable of generating intra-sentential CS samples across diverse languages while maintaining both contextual relevance and speaker consistency. UniCoM consists of three stages: preprocessing, source mixing, and style unification. In the preprocessing stage, potential artifacts are removed from the input sources to ensure the stability of the pipeline. The source mixing stage introduces **S**ubstituting **WORD**s with **S**ynonyms (SWORDS), a novel algorithm to generate speech-text pairs for intra-sentential CS. Finally, the style unification stage standardizes speaker styles across different sources. Consequently, building on UniCoM, we introduce CS-FLEURS: a comprehensive multilingual CS corpus encompassing 253 language pairs—marking a significant advancement over traditional single language pair datasets. Furthermore, the inclusion of over 70 n-way parallel sentences offers potential for future exploration in CS-S2TT tasks. Experimental results demonstrate that CS-FLEURS achieves comparable intelligibility and naturalness to human-generated CS datasets. We anticipate our approach will serve as a foundation for generalizable CS speech technology, paving the way for broader applications. Our contributions are summarized as follows:

- We introduce UniCoM[1], a novel code-switching speech dataset generation pipeline that is applicable to a wide variety of languages.
- We propose SWORDS, an algorithm for intra-sentential CS generation across diverse languages while preserving linguistic and speaker identity.
- We release CS-FLEURS, a first-ever, large-scale, and massively multilingual CS speech corpus.

[1] https://github.com/sanghyang00/unicom

2 Related Work

2.1 Code-Switching Speech Dataset

There are a few speech corpora that can be utilized for CS research. The Fisher corpus (Cieri et al., 2004) and Spoken Wikipedia Corpus (Baumann et al., 2019) were not originally intended for CS but contain some CS data, making them useful for CS tasks. However, its applicability to CS is limited due to data scarcity. The ASCEND corpus (Lovenia et al., 2021) includes both inter- and intra-sentential CS speech, facilitating the modeling of diverse CS patterns. However, its language coverage was limited to Mandarin-English, constraining applicability to broader multilingual settings. While other CS speech corpora (Deuchar, 2008; Lyu et al., 2010; Hamed et al., 2018; Diwan et al., 2021) exist, they shared the challenges related to limited resources and language spans.

2.2 Code-Switching Speech Generation

Nakayama et al. (2018) and Yu et al. (2023) proposed a code-switching speech synthesis approach using multilingual TTS. The method translates text, aligns similar words between languages, and generates speech via TTS. However, scaling to a large number of languages remains challenging for TTS, restricting its applicability to major languages.

Seki et al. (2018) and Dhawan et al. (2023) generate inter-sentential code-switching samples by concatenating independent utterances from different monolingual speech sources. However, this approach struggles to produce intra-sentential code-switching data due to challenges in preserving the unique characteristics of each language, such as phrase structure, making it less representative of real-world code-switching scenarios. Additionally, the lack of speaker consistency in concatenated samples raises concerns about speech naturalness.

Order	Example	Frequency (%)	Languages
SOV	Sam apples ate	45	Abaza, Abkhaz, Amharic, Akkadian, Armenian, Azerbaijani, Basque, etc.
SVO	Sam ate apples	42	Chinese, many European languages, Swahili, Thai, Vietnamese, etc.
VSO	Ate Sam apples	9	modern Arabic, Berber languages, Filipino, Irish, Māori, Welsh, etc.
VOS	Ate apples Sam	3	Austronesian languages, Car, Chumash, Fijian, Malagasy, etc.
OVS	Apples ate Sam	1	Äiwoo, Hixkaryana, Urarina
OSV	Apples Sam ate	1 ↓	Tobati, Warao, Haida

Table 1: A comparison of phrase order across languages (Meyer, 2009; Tomlin, 2014). Mixing sources with "ate apples" as the target may result in the loss of linguistic characteristics due to phrase order variations.

3 UniCoM

In this section, we propose **Uni**versal **Co**de-**M**ixer (UniCoM), a novel code-switching speech data generation pipeline. UniCoM is a universal and generalized framework designed for broad applicability across a wide range of languages. Unlike previous methods that generate inter-sentential code-switching samples by simply concatenating sentences from two different languages, UniCoM enables intra-sentential code-switching speech generation while preserving the unique linguistic features of each language, such as phrase structure and word order. Moreover, UniCoM preserves speaker identity by unifying the speaker's style, making the generated speech more reflective of naturally occurring code-switching in real-life scenarios. In the following sections, we provide a detailed description of each step in the proposed framework.

3.1 Preprocessing

Before applying the generation model, we implemented two preprocessing strategies to refine the baseline data. First, we identified that some samples in the existing monolingual dataset contained buzzing or white-noise-like artifacts, which could disrupt the subsequent generation pipeline. Consequently, we observed that these artifacts significantly degraded sample quality after the voice conversion (VC) stage, resulting in crackling sounds. To address this issue, we applied bandpass filtering to remove potential artifacts while preserving the majority of speech components. Specifically, we set cutoff frequencies below 80 Hz and above 7000 Hz to eliminate unwanted noise while preserving speech integrity. The second challenge we addressed was amplitude inconsistency. In most speech corpora, particularly multilingual ones, variations in recording environments lead to differences in volume, with some samples significantly quieter or louder than others. Moreover, we also observed that these amplitude variations adversely affected VC performance. To mitigate this issue, we applied amplitude normalization to ensure that all input speech samples were scaled uniformly.

3.2 Intra-Sentential Source-Mixing

Compared to inter-sentential CS, intra-sentential CS requires a significantly more complex process. Specifically, in inter-sentential CS, contextual consistency is less critical as language transitions occur at sentence boundaries. In contrast, intra-sentential CS happens within a sentence, making it essential to maintain contextual coherence. Therefore, a more sophisticated algorithm was necessary—one that goes beyond simply concatenating segments from source speech in different languages.

3.2.1 Substitution Strategy Selection

Given these challenges, the first aspect we considered was how to generate an intra-sentential CS utterance while preserving the original sentence's meaning and unique linguistic characteristics. To achieve this, we determined that replacing parts of a sentence with semantically equivalent segments from other languages would be an effective approach. From this perspective, we explored two primary rearrangement methods: (1) phrase-level and (2) word-level substitution.

Phrase-Level Substitution. We initially hypothesized that phrase-level substitution would better preserve the CS ratio. However, this approach introduced structural issues due to cross-linguistic variation in phrase syntax. As shown in Table 1, semantically equivalent substitution of phrase-level segments often disregards language-specific grammar, yielding unnatural outputs.

Word-Level Substitution. In contrast, part-of-speech (POS)—a word-level feature—is cross-linguistically common (Kornfilt, 2020), enabling substitutions that preserve both naturalness and

meaning. We thus adopted word- or POS-level substitutions with a flexible number of substitutions, allowing natural CS sentence generation while preserving semantics and syntactic structure.

3.2.2 SWORDS Algorithm

In this context, we propose the **S**ubstituting **WORD**s with **S**ynonym (SWORDS) algorithm. SWORDS is composed of four steps: (1) Sampling of equivalent sentence pairs, (2) Wordpair mapping generation, (3) Segmentation using forced alignment, (4) Completion through recombination.

This approach introduces two key properties. First, SWORDS is the first method tailored for intra-sentential CS speech generation, whereas existing approaches were designed for inter-sentential CS. Second, SWORDS preserves language-specific structure and sentence semantics through linguistically informed substitution, yielding natural outputs which highly resemble real-world CS scenarios. Details of each step are provided in the following, and the pseudo-code is shown in Alg. 1.

Sampling of Equivalent Sentence Pairs. We first organized n-way parallel sentences with equivalent meanings from a multilingual speech-text dataset. Next, we constructed one-to-one sentence-level mapping pairs for every language combination, promoting the word pair mapping generation process.

Wordpair Mapping Generation. Next, we decomposed the sentence pairs with equivalent meanings into word pairs with corresponding meanings. These word pairs were then classified based on the part of speech (e.g., noun, verb, adverb, adjective, and interjection). This process was conducted using a large language model (LLM), GPT-4o-mini (Achiam et al., 2023) as the word-level mapping generator. By inputting the previously constructed sentence pairs into the LLM with input prompts, we generated a dictionary of equivalent word pairs organized by POS. The details of the prompt and processing steps are illustrated in A.1.

Segmentation Using Forced Alignment. Consequently, to extract speech segments for each word pair generated from the previous step, we leveraged MMS-FA (Pratap et al., 2024) for forced alignment between text and speech, clipping each sample to match the target words. Although MMS-FA requires Romanization, it offers an effective trade-off between alignment speed and accuracy compared to Whisper-based ones (Radford et al., 2023; Bain et al., 2023). Details of the forced alignment methods we considered are provided in A.2.

Algorithm 1 SWORDS algorithm.

Input: Multilingual speech-text paired dataset $\mathcal{D}$, Number of substitutions N, POS substitution categories $\mathcal{P}$
Output: $SWORDS(\mathcal{D}, N, \mathcal{P})$

// Step 1: Sample equivalent sentence pairs
$S_{pairs} \leftarrow$ SampleEquivalentSentences($\mathcal{D}$)
$lang_1, lang_2 \leftarrow$ SelectLanguagePair(S_{pairs})
$utt_1, txt_1 \leftarrow$ SelectSample($lang_1$)
$utt_2, txt_2 \leftarrow$ SelectSample($lang_2$)

// Step 2: Wordpair mapping generation
$W_{pairs} \leftarrow$ LLMWordMapping(txt_1, txt_2)
$W_{pairs} \leftarrow$ ClassifyByPOS(W_{pairs})
$W_{sub} \leftarrow$ RandomSelect($W_{pairs}, N, \mathcal{P}$)

// Step 3: Forced alignment segmentation
$segment_{sub} \leftarrow$ [ ]
for each word w in W_{sub} **do**
    $segment_w \leftarrow$ MMS-FA(w, utt_1, utt_2)
    Append $segment_w$ to $segment_{sub}$
end for

// Step 4: Recombination
$lang_{mat} \leftarrow$ RandomSelect($lang_1, lang_2$)
for each word w in W_{sub} **do**
    $txt_{cs} \leftarrow$ Substitute($word, txt_1, txt_2, lang_{mat}$)
end for
for each seg in $segment_{sub}$ **do**
    $utt_{cs} \leftarrow$ Substitute($seg, utt_1, utt_2, lang_{mat}$)
end for

return utt_{cs}, txt_{cs}

Completion Through Recombination. Finally, we rearranged the clipped segments to produce the final CS speech sample. During the rearrangement process, we randomly select the matrix language, with the other language acting as the embedded language. In the final generation phase, a predefined number of words are randomly selected from the word pairs and substituted into the matrix language utterance, resulting in a source-mixed sample. Specifically, UniCoM enables the selection of various combinations of part-of-speech categories and word pairs for substitution, offering flexibility as a hyperparameter. This allows users to adjust the pool of part-of-speech categories and the number of substitutions based on specific requirements.

3.3 Style Unification

Since intra-sentential CS occurs within a sentence, preserving the speaker's identity throughout the utterance is essential. In the style unification stage, we aimed to align speaker styles across the segments, which were sampled from different utterances to ensure the naturalness of the generated samples. Particularly, our objective was to make the samples as close as possible to real-world instances of intra-sentential CS, with emphasis on enhancing speaker similarity.

Dataset	# language pair	duration (h)	# utt	# tok	CMI	I-index	# parallel
Spoken Wikipedia†	1 (en-de)	6.0	2.5k	10.9k	-	-	*N/A*
Miami-Bangor†	1 (en-es)	5.0	2.3k	5.5k	-	-	*N/A*
ASCEND	1 (en-zh)	10.6	12.3k	11.4k	-	-	*N/A*
MUCS2021	2 (bn/hi-en)	148.2	86.8k	27.9k	-	-	*N/A*
CS-FLEURS	**253**	**1.6k**	**654.7k**	**179.6k**	0.11	0.19	73

Table 2: Overall comparison between human-generated CS speech datasets and CS-FLEURS. CMI and I-index values are scaled between 0 and 1, † indicates that CS samples are partial, and statistics reflect only the CS subset.

Accordingly, we adopt kNN-VC (Baas et al., 2023) as the voice conversion (VC) model for two main reasons. First, kNN-VC retrieves ground-truth self-supervised features of the target speaker using a k-nearest neighbor (kNN) algorithm, uniquely distinguishing it from other VC models. Given that the neural vocoder is relatively language-invariant, this retrieval-based mechanism preserves intelligibility while effectively transferring speaker characteristics across languages. As a result, kNN-VC produces highly intelligible and natural speech, even in cross-lingual settings. Since our VC pipeline targets cross-lingual scenarios, kNN-VC serves as an effective choice for unifying speaker identity across concatenated segments.

Second, kNN-VC achieves fast inference during the voice conversion process, primarily due to its adoption of the lightweight GAN-based vocoder (Kong et al., 2020), in combination with the retrieval-based method described earlier. This design enables substantially faster voice conversion compared to approaches that rely on diffusion-based methods or employ larger vocoders, while still maintaining competitive performance. Although such alternatives may offer higher fidelity in certain scenarios, kNN-VC provides a more favorable trade-off between synthesis quality and inference speed. As a result, kNN-VC is well-suited for both online and offline generation scenarios required by UniCoM. Further details behind the selection of the VC model are provided in Appendix C.

4 CS-FLEURS

This section describes CS-FLEURS, an intra-sentential CS speech dataset spanning 253 language pairs, constructed using UniCoM. Overall metadata of CS-FLEURS is illustrated in Table 2.

4.1 Baseline Dataset

To select the baseline dataset with a sufficient range of languages, we considered three datasets: CommonVoice (Ardila et al., 2019), FLEURS (Conneau et al., 2023), and FLEURS-R (Ma et al., 2024).

CommonVoice is a crowdsourced multilingual dataset covering over 120 languages and 30,000 hours. While its scale is a strength, variations in recording conditions often result in noisy samples.

FLEURS is a 102-language multilingual speech-text dataset with two key advantages over CommonVoice. First, it supports both ASR and S2TT via n-way parallel sentences, enabling CS-ASR/S2TT dataset creation. Second, its longer average sample length benefits the VC module, which relies on sufficient reference duration.

FLEURS-R is a denoised version of FLEURS generated using a speech restoration model (Koizumi et al., 2023), retaining the original structure while improving clarity and reducing noise. Since noise is a major obstacle in ensuring the quality of VC, we opted to use FLEURS-R as the baseline dataset.

4.2 Language Selection

We leveraged 23 languages from FLEURS-R (a subset focusing on European languages) that overlap with the VoxPopuli dataset's (Wang et al., 2021) language coverage. We refer to these overlapping languages as in-domain (ID) languages. The rationale behind this selection is that, although kNN-VC generates converted speech by retrieving ground-truth speaker embeddings, typological differences between source and target languages might lead to unnatural-sounding speech due to mismatched phonetic and prosodic characteristics across languages. To mitigate such artifacts, we restricted the coverage of CS-FLEURS to cover the languages that exhibit similar phonetic and orthographic properties (mostly Indo-European or Latin character languages) as discussed in previous linguistic studies (Bradlow et al., 2010; Bella et al., 2021), to ensure more consistent, intelligible, and natural-sounding voice conversion results. While samples from out-of-domain (OOD) languages often produced plausible outputs, we conservatively limited our evaluation to in-domain (ID) languages to ensure overall quality and stability.

	Transcription
English	Hiking is an outdoor activity which consists of walking in natural environments, often on hiking trails.
Dutch	Wandelen is een buitenactiviteit waarbij je in een natuurlijke omgeving wandelt, meestal op wandelpaden.
Code-Switched	Wandelen *is an outdoor activity which consists of walking in natural environments often on* wandelpaden.

Table 3: Example transcription pair from CS-FLEURS, with the matrix language in italics and code-switched (or embedded language) segments highlighted in red.

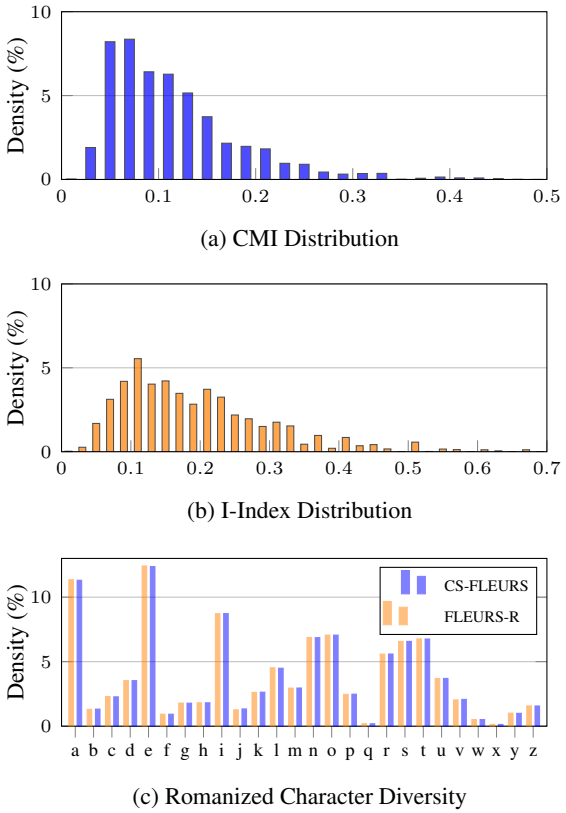
4.3 Source Mixing

In Sec. 3.2.2, we demonstrated that UniCoM can substitute various POS and adjust the number of word pairs as needed. However, excessive word substitution can lead to unnatural results, deviating from real-world CS scenarios. To maintain naturalness and ensure the generated speech reflects authentic CS patterns, we limited sampling to a maximum of three word pairs and restricted POS categories to nouns, verbs, and interjections, following prior linguistic studies on natural CS (Ahn et al., 2020; Yao, 2020; Chi and Bell, 2024).

4.4 Dataset Statistic

Data Size and Language Span. Recent trends in deep learning models underscore the importance of scaling up dataset size. A larger and more balanced dataset contributes to better generalization performance, allowing models to learn richer and more nuanced representations. In this context, CS-FLEURS presents a significant advantage. It encompasses over 250 diverse language pairs and features an extensive amount of speech data. Although individual language pairs contain modest amounts of data, the broad linguistic coverage and overall scale make CS-FLEURS a valuable resource for CS research in massively multilingual settings.

Code-Switching Metrics. The code-mixing index (Das and Gambäck, 2014) (CMI) is a widely used metric for quantifying the intensity and proportion of code-mixing within a given text. It measures the ratio between the matrix (primary) language and the embedded (inserted) language. A higher CMI indicates a greater extent of CS. The I-index (Guzmán et al., 2017), on the other hand, indicates the ratio of the occurrence points of CS within the utterance or text. It aims to measure how evenly CS occurs throughout the speech sample. Unlike previous CS speech corpora that did not provide such information, CS-FLEURS presents the distributions of both CMI and I-index, as shown in Figure 2a and 2b. Moreover, language types



(a) CMI Distribution

(b) I-Index Distribution

(c) Romanized Character Diversity

Figure 2: Data statistics of CS-FLEURS.

(e.g., matrix, embedded) and metrics are incorporated into the metadata of CS-FLEURS, enabling it to be a valuable tool for uncovering the linguistic characteristics of CS and exploring its relationship with speech patterns in future research. To support qualitative analysis, Table 3 shows an annotated transcription pair from CS-FLEURS.

Phoneme Diversity. Quantifying phoneme diversity in a multilingual corpus is inherently challenging, and this task becomes even more complex when applied to CS-FLEURS, a code-switching corpus. To address this, we applied Romanization (Hermjakob et al., 2018) uniformly across all textual data based on prior studies showing that Romanization preserves phonetic characteristics to a reasonable extent (Pratap et al., 2024; Ding et al., 2024). By leveraging Romanized character diversity, we approximate phoneme diversity, even if not perfectly. When compared to the baseline dataset, CS-FLEURS functions as a code-switching speech corpus without compromising phoneme diversity, as shown in Figure 2c. This feature is particularly useful for training speech-to-text models, where phoneme diversity is crucial for performance.

Instructions
1. Assess the overall audio quality of each utterance independent of any background noise levels. (MOS)
2. Assign higher scores when the speaker's voice remains consistent throughout the utterance without perceptible changes. (SIS)
3. Focus on evaluating the absolute quality (MOS) or speaker consistency (SIS) of each sample; comparisons with other samples should be avoided.
4. Select only one score from {1, 2, 3, 4, 5}

(a) General instructions for MOS and SIS evaluation.

Score	Description
5	Exhibits near-human naturalness and prosody; indistinguishable from human speech.
4	Perceptibly synthetic but with high fidelity and natural intonation; pleasant to listen to.
3	Noticeably synthetic with moderate artifacts; acceptable for extended listening.
2	Prominent synthetic artifacts and degraded quality; impacts listener comfort.
1	Severely degraded with overwhelming artifacts or distortions; difficult to comprehend.

(b) Mean Opinion Score (MOS) scoring criteria.

Score	Description
5	Speaker identity remains perfectly consistent; no perceptible change in voice.
4	Speaker identity is mostly consistent; minor variations without affecting overall perception.
3	Some noticeable changes in speaker identity; the utterance remains generally consistent.
2	Pronounced changes in speaker identity; difficult to perceive as a consistent speaker.
1	Speaker identity is completely inconsistent; the utterance appears to be from multiple speakers.

(c) Speaker Identity Score (SIS) scoring criteria.

Table 4: Instructions on subjective evaluations.

5 Experiments

In this section, we evaluate the validity of CS-FLEURS to prove the effectiveness of UniCoM. All experiments were performed on two RTX 3090 GPUs with 24GB VRAM each, and subjective metrics were evaluated through a user study involving a total of 42 participants and over 2,500 utterances.

5.1 Evaluation Metrics

Romanized Character Error Rate (RER). Given the extensive multilingual capabilities and CS nature of CS-FLEURS, evaluating a unified metric for all combinations was challenging due to the lack of trainable open-source models for massively multilingual CS-ASR. To resolve this issue, we first converted all transcriptions into Romanization pairs and measured the Romanized character Error Rate (RER) to assess intelligibility, based on the previous studies that proved Romanization standardizes orthographic diversity while preserving phonetic characteristics (Pratap et al., 2024; Ding et al., 2024). This characteristic makes RER analogous to measuring Phoneme Error Rate (PER), where a lower RER indicates high intelligibility. Specifically, we fine-tuned the XLS-R (Babu et al., 2021) using the CommonVoice 17.0 dataset for our Romanization-based ASR model.

Pair	it-es	pt-es	ro-es	it-ro	it-pt	de-es
RER	20.5	22.8	23.0	23.1	23.9	24.8
MOS	4.50	4.50	4.69	4.62	4.89	4.87
SIS	4.42	4.85	4.71	4.65	4.80	4.94

pl-es	it-pl	cs-it	∼	bg-fi	da-mt	da-fi
24.9	25.1	25.2	∼	38.1	38.1	38.3
4.28	4.75	4.32	∼	4.84	4.55	4.21
4.29	4.76	5.00	∼	4.86	4.73	4.53

bg-el	bg-en	hr-da	da-sl	da-el	bg-da	**Avg.**
38.7	38.7	38.9	39.3	40.2	41.9	**31.6**
4.36	4.05	4.71	4.59	4.00	4.60	**4.44**
4.74	4.18	4.71	4.65	4.65	4.65	**4.74**

Table 5: Evaluation on the 9 best- and worst-performing language pairs of CS-FLEURS. *Avg.* indicates the mean value across 253 pairs; full results are in Appendix B.

Mean Opinion Score (MOS). Since UniCoM generates CS speech through VC, ensuring the naturalness of the generated samples is crucial. To this end, we adopted MOS, a subjective metric to evaluate perceptual speech quality where listeners rate the speech on a scale from 1 to 5, with higher scores indicating more natural and intelligible speech. To be specific, all raters were proficient in English, and many had fluency in Chinese, Spanish, German, French, Dutch, or Italian, primarily through prior residence or extended stays in relevant regions. Some also had a passive understanding of other (mostly European) languages included in the evaluation. Each rater evaluated a set of samples containing five utterances per language pair, and this protocol was applied consistently across all 42 raters. The detailed instructions provided to raters are summarized in Table 4b.

Speaker Identity Score (SIS). Given the intra-sentential CS nature, it is essential to ensure that

speaker identity is preserved within each sample. To evaluate this aspect, we introduce the Speaker Identity Score (SIS), a metric designed to assess the consistency of speaker identity. The SIS is measured similarly to MOS, where raters evaluate the confidence that a given sample contains only a single speaker. The score also ranges from 1 to 5, reflecting the perceived preservation of speaker identity within the sample. Similar to the MOS evaluation, each rater was assigned five utterances per language pair, and this procedure was uniformly applied to all 42 raters. The guidelines given to raters are outlined in Table 4c.

5.2 Dataset Quality Evaluation

Intelligibility Evaluation. In Table 5, results demonstrate that CS-FLEURS maintains sufficient intelligibility, with an average RER of 31%. Given that CS speech is less intelligible than monolingual speech and CS-FLEURS originates from a VC pipeline, results highlight the potential of UniCoM to mitigate CS speech corpus scarcity.

Naturalness Evaluation. Table 5 also indicates that samples from CS-FLEURS show only minor deviations from human-generated datasets in terms of human perception. Specifically, MOS values generally exceed 4, even for the lowest-scoring languages, while samples from CS-FLEURS maintain high speaker consistency, with SIS scores exceeding 4.5 for the majority of language pairs.

5.3 Comparative Evaluation

Moreover, to validate the practicality of CS-FLEURS, we conduct comparisons with existing human-annotated CS datasets across both ID and OOD languages. For ID languages, we used the Spoken Wikipedia Corpus (SWC) and the Miami-Bangor Corpus (MBC), while for OOD languages, we utilized the ASCEND and MUCS2021 corpora. We considered only CS samples from each dataset and evaluated their relative quality using the same metrics described in previous sections. Especially, for OOD languages, the quality of generated datasets might be inevitably degraded due to the large linguistic difference as mentioned in Section 4.2. To distinct it from the original CS-FLEURS and ensure a fairer comparison, we categorize them separately as CS-FLEURS-O.

In-Domain Languages. As illustrated in Table 6a, CS-FLEURS exhibits competitive intelligibility compared to human-generated CS datasets in ID languages. Notably, CS-FLEURS achieved comparable RER and MOS scores to SWC, a professionally produced and well-aligned dataset. Moreover, CS-FLEURS outperformed MBC, which is a noisy dataset, in both objective and subjective evaluations. These findings highlight the quality of CS-FLEURS and prove the strength of UniCoM.

Dataset	Intelligibility RER ↓	Naturalness MOS ↑	Naturalness SIS ↑
SWC (de)	**25.8**	**4.51**	**4.90**
MBC (es)	56.9	4.20	4.40
CS-FLEURS (de)	30.1	4.36	4.83
CS-FLEURS (es)	28.9	4.00	4.89
CS-FLEURS (avg)	31.6	4.44	4.74

(a) In-domain (ID) language comparison.

Dataset	Intelligibility RER ↓	Naturalness MOS ↑	Naturalness SIS ↑
ASCEND (zh)	-	**4.85**	4.66
MUCS2021 (hi)	48.3	4.50	**4.87**
MUCS2021 (bn)	57.6	3.87	4.28
CS-FLEURS-O (zh)	41.5	4.50	4.37
CS-FLEURS-O (hi)	37.7	3.66	5.00
CS-FLEURS-O (bn)	42.7	2.88	4.71
CS-FLEURS-O (avg)	**40.8**	3.68	4.69

(b) Out-of-domain (OOD) language comparison.

Table 6: Quality comparison between CS-FLEURS and human-generated CS datasets.

Out-of-Domain Languages. Table 6b shows that CS-FLEURS achieves performance comparable to existing datasets under OOD conditions, with slightly lower MOS but showing better RER scores. For SIS, it performs on par with human-generated CS corpora. These results underscore the generalizability of UniCoM, despite the typical degradation observed in OOD compared to ID settings.

5.4 Impact of CS-FLEURS in CS-ASR

Finally, to evaluate the actual contribution of CS-FLEURS to CS-ASR performance, we conducted experiments using the aforementioned two ID human-generated CS datasets along with corresponding language pairs in CS-FLEURS (CSF). We ensured an equal-sized training set for each experiment and fine-tuned XLS-R with a concatenated vocabulary for our CS-ASR model.

English-German Pair. As shown in Table 7, training the CS-ASR model with CS-FLEURS improves transcription performance on the English-German pair. In particular, evaluation on the SWC dataset shows that combining our synthetic data yields better results than using SWC alone, highlighting its value as a data augmentation method. More-

Language Pair	Train Data	Eval Data	CER ↓
En-De	SWC	SWC	26.7
	CSF		45.8
	SWC + CSF		**23.0**
	SWC	CSF	48.3
	CSF		**23.7**
	SWC + CSF		26.5
En-Es	MBC	MBC	100
	CSF		71.3
	MBC + CSF		**35.8**
	MBC	CSF	100
	CSF		**20.1**
	MBC + CSF		22.9

Table 7: Impact of CS-FLEURS in CS-ASR training and evaluation for two in-domain language pairs.

over, when evaluated on its own, CS-FLEURS also proves effective as a primary training resource.

English-Spanish Pair. Similarly, when training solely on MBC for the English-Spanish pair, the high noise level results in models producing only blank tokens. In contrast, joint training with CS-FLEURS substantially improves performance on MBC, demonstrating its effectiveness for data augmentation. Notably, training exclusively on the synthetic data yields the best results on the corresponding evaluation set (CSF), reinforcing its capacity both as a core training source and as a means of enhancing existing datasets. However, both tables reveal performance degradation in cross-dataset inference due to domain mismatch across datasets, as previously discussed by Radford et al. (2023). To summarize, results for both language pairs highlight the strength of CS-FLEURS in two key aspects: (1) as a strong standalone training dataset for CS-ASR, and (2) as an effective data augmentation method for existing CS-ASR corpora.

6 Conclusion

In this paper, we introduce UniCoM, a universal pipeline for code-switching speech generation. Specifically, we propose a SWORDS algorithm to facilitate the generation of intra-sentential code-switching samples while maintaining the natural meaning of the original speech. Our pipeline then employs voice conversion to ensure the naturalness of generated samples by unifying speaker styles across utterances without compromising intelligibility. Finally, we release CS-FLEURS, a massively multilingual code-switching speech dataset designed for the CS-ASR task while offering future utility for CS-S2TT scenarios. In experiments, CS-FLEURS exhibits high intelligibility and naturalness compared to human-generated datasets, while contributing to the improvements in CS-ASR training and performance. We believe our work will serve as a cornerstone for a more generalized code-switching speech technology in the future.

7 Limitations

Since the generation pipeline includes a pre-trained voice conversion model, the performance of UniCoM is inevitably limited for out-of-domain languages for the voice conversion model. Consequently, to generate a high-quality and natural CS dataset, the language span of CS-FLEURS is constrained to combinations of languages within the in-domain set of the voice conversion model. Notably, in terms of orthographic features, CS-FLEURS includes languages that predominantly use the Latin script. While Bulgarian leverages the Cyrillic script and Greek utilizes the Greek alphabet, the remaining languages are based on the Latin script.

In future research, we aim to expand the scope of UniCoM, enabling the application of UniCoM to languages with diverse orthographies and wider linguistic coverage, towards a truly universal method.

8 Ethical Statements

This study follows ethical guidelines, prioritizing fairness, transparency, and accountability. The code for UniCoM and demo version of CS-FLEURS is publicly available on anonymous GitHub via the footnote link, and the datasets used for validating UniCoM and CS-FLEURS (e.g., CommonVoice, FLEURS-R, ASCEND, MUCS2021) are fully open-source. CS-FLEURS follows the Creative Commons Attribution 4.0 (cc-by-4.0) license, in accordance with FLEURS-R, and will be publicly released after undergoing post-processing of metadata.

We ensured consistency and fairness in comparisons across all experiments. In cases where fair comparisons were not possible due to specific conditions, we added the † token to indicate this to potential readers. For the user study, all participants voluntarily participated through community outreach. Each participant was thoroughly informed about the study process, procedures, and the intended use of the results. The payment was appropriate given the participants' demographics. We recognize the impact of massively multilingual code-switching speech research and are committed to conducting our work while adhering to ethical research practices.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Emily Ahn, Cecilia Jimenez, Yulia Tsvetkov, and Alan Black. 2020. What code-switching strategies are effective in dialogue systems? *Society for Computation in Linguistics*, 3(1).

Sadeen Alharbi, Reem Binmuqbil, Ahmed Ali, Raghad Aloraini, Saiful Bari, Areeb Alowisheq, and Yaser Alonaizan. 2024. Leveraging llm for augmenting textual data in code-switching asr: Arabic as an example. In *Synthetic Data's Transformative Role in Foundational Speech Models*, pages 26–30.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber. 2019. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*.

Matthew Baas, Benjamin van Niekerk, and Herman Kamper. 2023. Voice conversion with just nearest neighbors. *arXiv preprint arXiv:2305.18975*.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick Von Platen, Yatharth Saraf, Juan Pino, et al. 2021. Xls-r: Self-supervised cross-lingual speech representation learning at scale. *arXiv preprint arXiv:2111.09296*.

Max Bain, Jaesung Huh, Tengda Han, and Andrew Zisserman. 2023. Whisperx: Time-accurate speech transcription of long-form audio. *arXiv preprint arXiv:2303.00747*.

Timo Baumann, Arne Köhn, and Felix Hennig. 2019. The spoken wikipedia corpus collection: Harvesting, alignment and an application to hyperlistening. *Language Resources and Evaluation*, 53:303–329.

Gábor Bella, Khuyagbaatar Batsuren, and Fausto Giunchiglia. 2021. A database and visualization of the similarity of contemporary lexicons. In *International Conference on Text, Speech, and Dialogue*, pages 95–104. Springer.

Ann Bradlow, Cynthia Clopper, Rajka Smiljanic, and Mary Ann Walter. 2010. A perceptual phonetic similarity space for languages: Evidence from five native language listener groups. *Speech communication*, 52(11-12):930–942.

Jie Chi and Peter Bell. 2024. Analyzing the role of part-of-speech in code-switching: A corpus-based study. In *The 18th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1712–1721. ACL Anthology.

Ha-Yeong Choi, Sang-Hoon Lee, and Seong-Whan Lee. 2023. Diff-hiervc: Diffusion-based hierarchical voice conversion with robust pitch generation and masked prior for zero-shot speaker adaptation. *arXiv preprint arXiv:2311.04693*.

Christopher Cieri, David Miller, and Kevin Walker. 2004. The fisher corpus: A resource for the next generations of speech-to-text. In *LREC*, volume 4, pages 69–71.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2023. Fleurs: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805. IEEE.

Amitava Das and Björn Gambäck. 2014. Identifying languages at the word level in code-mixed indian social media text.

Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. 2020. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. *arXiv preprint arXiv:2005.07143*.

Margaret Deuchar. 2008. The miami corpus: Documentation file. *Bangortalk, bangortalk. org. uk/docs/Miami_doc. pdf*.

Kunal Dhawan, KDimating Rekesh, and Boris Ginsburg. 2023. Unified model for code-switching speech recognition and language identification based on concatenated tokenizer. In *Proceedings of the 6th Workshop on Computational Approaches to Linguistic Code-Switching*, pages 74–82.

Wen Ding, Fei Jia, Hainan Xu, Yu Xi, Junjie Lai, and Boris Ginsburg. 2024. Romanization encoding for multilingual asr. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pages 468–475. IEEE.

Anuj Diwan, Rakesh Vaideeswaran, Sanket Shah, Ankita Singh, Srinivasa Raghavan, Shreya Khare, Vinit Unni, Saurabh Vyas, Akash Rajpuria, Chiranjeevi Yarra, et al. 2021. Multilingual and code-switching asr challenges for low resource indian languages. *arXiv preprint arXiv:2104.00235*.

Penelope Gardner-Chloros. 2009. *Code-switching*. Cambridge university press.

Gualberto A Guzmán, Joseph Ricard, Jacqueline Serigos, Barbara E Bullock, and Almeida Jacqueline Toribio. 2017. Metrics for modeling code-switching across corpora. In *Interspeech*, pages 67–71.

Injy Hamed, Mohamed Elmahdy, and Slim Abdennadher. 2018. Collection and analysis of code-switch egyptian arabic-english speech corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Ulf Hermjakob, Jonathan May, and Kevin Knight. 2018. Out-of-the-box universal romanization tool uroman. In *Proceedings of ACL 2018, system demonstrations*, pages 13–18.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR.

Yuma Koizumi, Heiga Zen, Shigeki Karita, Yifan Ding, Kohei Yatabe, Nobuyuki Morioka, Yu Zhang, Wei Han, Ankur Bapna, and Michiel Bacchiani. 2023. Miipher: A robust speech restoration model integrating self-supervised speech and text representations. In *2023 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 1–5. IEEE.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033.

Jaklin Kornfilt. 2020. Parts of speech, lexical categories, and word classes in morphology.

Mari Ganesh Kumar, Jom Kuriakose, Anand Thyagachandran, Arun Kumar A, Ashish Seth, Lodagala V.S.V. Durga Prasad, Saish Jaiswal, Anusha Prakash, and Hema A. Murthy. 2021. Dual script e2e framework for multilingual and code-switching asr. In *Interspeech 2021*, pages 2441–2445.

Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. 2022. Bigvgan: A universal neural vocoder with large-scale training. *arXiv preprint arXiv:2206.04658*.

Hexin Liu, Leibny Paola Garcia, Xiangyu Zhang, Andy WH Khong, and Sanjeev Khudanpur. 2024. Enhancing code-switching speech recognition with interactive language biases. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10886–10890. IEEE.

Songting Liu. 2024. Zero-shot voice conversion with diffusion transformers. *arXiv preprint arXiv:2411.09943*.

Holy Lovenia, Samuel Cahyawijaya, Genta Indra Winata, Peng Xu, Xu Yan, Zihan Liu, Rita Frieske, Tiezheng Yu, Wenliang Dai, Elham J Barezi, et al. 2021. Ascend: A spontaneous chinese-english dataset for code-switching in multi-turn conversation. *arXiv preprint arXiv:2112.06223*.

Dau-Cheng Lyu, Tien Ping Tan, Engsiong Chng, and Haizhou Li. 2010. Seame: a mandarin-english code-switching speech corpus in south-east asia. In *Interspeech*, volume 10, pages 1986–1989.

Min Ma, Yuma Koizumi, Shigeki Karita, Heiga Zen, Jason Riesa, Haruko Ishikawa, and Michiel Bacchiani. 2024. Fleurs-r: A restored multilingual speech corpus for generation tasks. *arXiv preprint arXiv:2408.06227*.

Charles F Meyer. 2009. *Introducing english linguistics*. Cambridge University Press.

Gabriel Mittag, Babak Naderi, Assmaa Chehadi, and Sebastian Möller. 2021. Nisqa: A deep cnn-self-attention model for multidimensional speech quality prediction with crowdsourced datasets. *arXiv preprint arXiv:2104.09494*.

Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. 2017. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*.

Sahoko Nakayama, Takatomo Kano, Quoc Truong Do, Sakriani Sakti, and Satoshi Nakamura. 2018. Japanese-english code-switching speech data construction. In *2018 Oriental COCOSDA-International Conference on Speech Database and Assessments*, pages 67–71. IEEE.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, et al. 2024. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Hiroshi Seki, Shinji Watanabe, Takaaki Hori, Jonathan Le Roux, and John R Hershey. 2018. An end-to-end language-tracking speech recognizer for mixed-language speech. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4919–4923. IEEE.

Bhavani Shankar, Preethi Jyothi, and Pushpak Bhattacharyya. 2024. Costa: Code-switched speech translation using aligned speech-text interleaving. *arXiv preprint arXiv:2406.10993*.

Russell S Tomlin. 2014. *Basic Word Order (RLE Linguistics B: Grammar): Functional Principles*. Routledge.

Shubham Toshniwal, Tara N Sainath, Ron J Weiss, Bo Li, Pedro Moreno, Eugene Weinstein, and Kanishka Rao. 2018. Multilingual speech recognition with a single end-to-end model. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4904–4908. IEEE.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson,

Juan Pino, and Emmanuel Dupoux. 2021. Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. *arXiv preprint arXiv:2101.00390*.

Orion Weller, Matthias Sperber, Telmo Pires, Hendra Setiawan, Christian Gollan, Dominic Telaar, and Matthias Paulik. 2022. End-to-end speech translation for code switched speech. *arXiv preprint arXiv:2204.05076*.

Mingfa Yao. 2020. Structure patterns of code-switching in english classroom discourse. *English Literature and Language Review*, 6(8):133–141.

Haibin Yu, Yuxuan Hu, Yao Qian, Ma Jin, Linquan Liu, Shujie Liu, Yu Shi, Yanmin Qian, Edward Lin, and Michael Zeng. 2023. Code-switching text generation and injection in mandarin-english asr. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Yu Zhang, Wei Han, James Qin, Yongqiang Wang, Ankur Bapna, Zhehuai Chen, Nanxin Chen, Bo Li, Vera Axelrod, Gary Wang, et al. 2023. Google usm: Scaling automatic speech recognition beyond 100 languages. *arXiv preprint arXiv:2303.01037*.

Shiyu Zhou, Yuanyuan Zhao, Shuang Xu, Bo Xu, et al. 2017. Multilingual recurrent neural networks with residual learning for low-resource speech recognition. In *INTERSPEECH*, pages 704–708.

Appendix

A Technical Details of SWORDS

In this section, we present additional details of the SWORDS algorithm, covering the generation of word-level mappings using LLMs, along with the input prompts and post-processing steps, and the selection criteria in the forced alignment module.

A.1 Word-Level Mapping Generation

Hyperparameter of LLM. For hyperparameters of LLM, we set the temperature to 0.0 to produce fully deterministic results and to ensure output validity in word-level mapping. All other hyperparameters were kept at their default values.

Input Prompt. For the input prompt, we utilized two system prompts and one user prompt, as shown in Tab 8. The system prompts included a role-playing prompt to enhance language-specific performance and a formatting prompt to ensure a consistent output structure. The user prompt was designed to generate word-level mappings based on the input language and text. However, due to the complexity of the desired output—a YAML-formatted dictionary with nested lists—occasional inconsistencies in the output format were observed.

Post-Processing. To address this issue, we applied post-processing to enforce a consistent data structure across all LLM outputs. Specifically, we removed unpaired elements, intentionally inserted missing keys, and converted different data structures (e.g., dictionaries, tuples) into lists. This post-processing ensured that all LLM outputs maintained a consistent and valid word-level mapping structure, which was then used in the forced alignment process. The overall word-level mapping generation process is illustrated in Figure 3.

A.2 Selection of Forced Alignment Model

Whisper-Based Models. The first approach we considered was utilizing methods based on Whisper, a multilingual ASR model. To achieve a superior alignment performance, we used the largest Whisper model, which was crucial for maintaining high-quality output in our final dataset. However, as the model size increased, the alignment time grew substantially, making real-world applications difficult. Although we considered using a smaller model, this resulted in a decline in alignment performance, which was critical for us, as the quality of the generated results was our top priority.

MMS-FA. As an alternative, we employed MMS-FA with default hyperparameters (e.g., clipping threshold), which offers forced alignment with a smaller model size and full GPU-based computation. Although it requires orthographic unification, MMS-FA provides faster and effective alignment compared to the Whisper-based methods, making it more suitable for real-world applications.

B Language-Pair-Specific Result

In this section, we present a detailed analysis of the experimental results for each language combination, providing insights into the system's performance across different language pairs. The full results for 253 language pairs are shown in Table 9 and Table 10, while the distributions of metrics across the language pairs are illustrated in Figure 4. These analyses would help evaluate performance variations and identify emerging patterns.

C Selection of Voice Conversion Model

In this section, we elaborate on the criteria used to select the VC module for UniCoM. Given our objective, we considered two key factors: (1) sufficient intelligibility in cross-lingual settings and (2) fast generation speed. We placed greater emphasis on the latter, as slower generation can substantially increase the overall construction time, especially for large-scale datasets. To assess the suitability of each model for UniCoM, we conducted an experiment comparing different voice conversion models. Specifically, we evaluated two additional VC models—Diff-HierVC (Choi et al., 2023) and SeedVC (Liu, 2024)—in cross-lingual voice conversion scenarios. Both adopt decomposition-based (e.g., pitch, content, etc) approaches, in contrast to the retrieval-based architecture of kNN-VC.

Results in Table 11 show that kNN-VC satisfies both criteria effectively. While it exhibited slightly lower intelligibility and naturalness compared to the diffusion-based alternatives, the real-time factor (RTF)—which measures the time required to process one second of audio—demonstrates that diffusion-based models are significantly slower than kNN-VC, regardless of whether a lightweight (Kong et al., 2020) or heavyweight (Lee et al., 2022) vocoder is used. We considered such minor degradations acceptable, as utterances in the embedded language of code-switched speech often exhibit imperfect pronunciation or reduced intelligibility by nature. Based on this trade-off, we chose to adopt kNN-VC.

Type	Detailed Prompt	Format
System (Role)	You are a language expert specializing in *lang1* and *lang2*.	matches: noun: - [[n1, n2]] verb: - [[v1, v2]] adverb: - [[a1, a2]] adjective: - [[a'1, a'2]] interjection: - [[i1, i2]]
System (Formatting)	The final outputs must be returned in YAML format, and each component in part of speech is a list of words with the same meaning. The YAML file structure must strictly adhere to the following format: *format*	
User (Input)	Find pairs of words with the same meaning and sort it with the part of speech information in the given two sentences from different languages. *lang1* sentence: *trans1*, *lang2* sentence: *trans2*	

Table 8: Detailed prompting strategy of word-level mapping generation. Italicized text indicates the hyper-parameter of the pipeline.

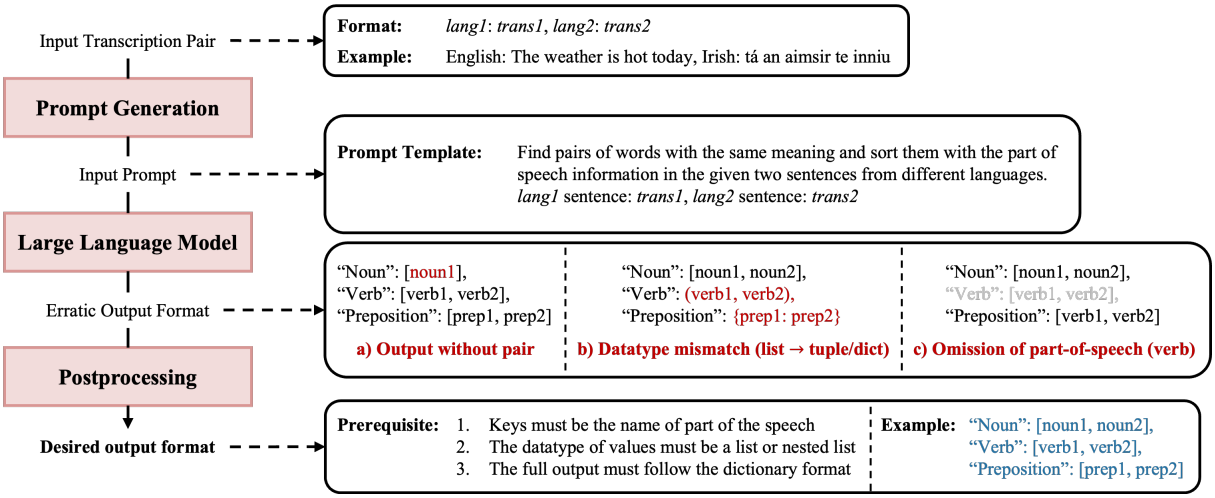


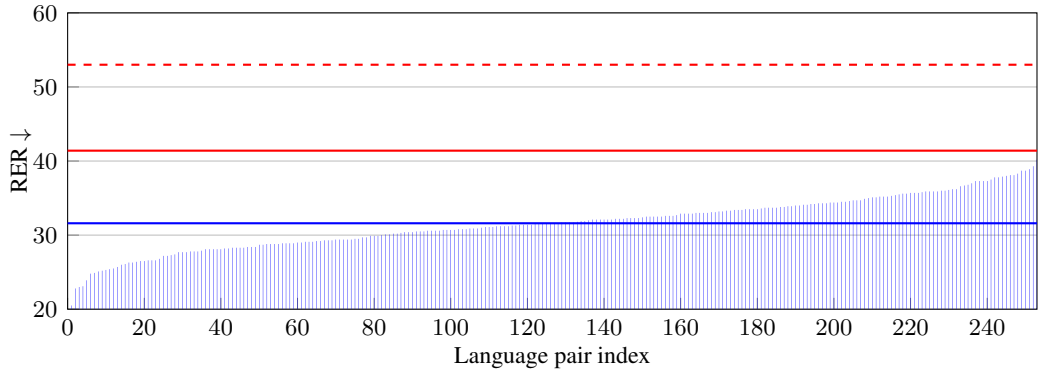
Figure 3: The overall pipeline of word-level mapping generation. Italicized text indicate hyper-parameter of the input prompt.

Lang. pair	RER	MOS	SIS	Lang. pair	RER	MOS	SIS	Lang. pair	RER	MOS	SIS
bg-hr	36.1	3.94	4.44	nl-fr	32.6	4.8	4.9	de-ro	26.6	4.31	4.89
bg-cs	32.9	4.53	4.61	nl-de	27.8	4.53	4.95	de-sk	29.9	4.41	4.94
bg-da	41.9	4.6	4.65	nl-el	36.0	4.63	4.93	de-sl	31.2	4.6	4.89
bg-nl	37.9	4.0	4.76	nl-hu	32.7	4.8	4.85	de-es	24.8	4.87	4.94
bg-en	38.7	4.05	4.18	nl-it	29.1	4.67	4.62	de-sv	28.4	4.54	4.6
bg-et	37.5	4.0	4.56	nl-lv	32.5	4.75	4.62	el-hu	34.0	4.38	4.58
bg-fi	38.1	4.84	4.86	nl-lt	33.4	4.44	4.84	el-it	29.9	4.6	4.94
bg-fr	37.0	3.5	4.58	nl-mt	34.3	4.75	4.83	el-lv	33.4	4.44	4.76
bg-de	34.6	4.0	4.69	nl-pl	31.4	4.44	4.8	el-lt	33.5	4.62	4.94
bg-el	38.7	4.36	4.74	nl-pt	31.7	4.44	4.64	el-mt	35.5	4.5	4.78
bg-hu	35.9	3.67	4.41	nl-ro	30.7	4.79	4.93	el-pl	32.3	4.81	4.88
bg-it	31.4	4.06	4.78	nl-sk	33.3	4.44	4.63	el-pt	32.3	4.0	4.75
bg-lv	34.3	3.94	4.75	nl-sl	35.0	4.55	4.61	el-ro	31.1	4.43	4.89
bg-lt	34.9	4.28	4.94	nl-es	28.7	4.43	4.71	el-sk	34.0	4.69	5.0
bg-mt	37.8	3.44	4.72	nl-sv	32.6	4.53	4.38	el-sl	35.7	4.81	4.88
bg-pl	33.3	4.88	4.89	en-et	35.6	4.78	4.73	el-es	29.5	4.44	4.81
bg-pt	33.8	4.0	4.73	en-fi	35.9	4.57	4.88	el-sv	35.2	4.47	4.5
bg-ro	31.9	3.95	4.84	en-fr	32.9	3.94	4.47	hu-it	27.7	4.35	4.79
bg-sk	34.1	4.53	4.88	en-de	30.1	4.36	4.83	hu-lv	30.1	4.3	4.8
bg-sl	35.9	4.28	4.75	en-el	36.6	4.5	4.88	hu-lt	31.6	4.41	4.95
bg-es	31.8	4.23	4.47	en-hu	33.5	4.67	4.73	hu-mt	32.2	4.88	5.0
bg-sv	37.3	4.07	4.65	en-it	29.3	4.16	4.53	hu-pl	29.7	4.61	4.83

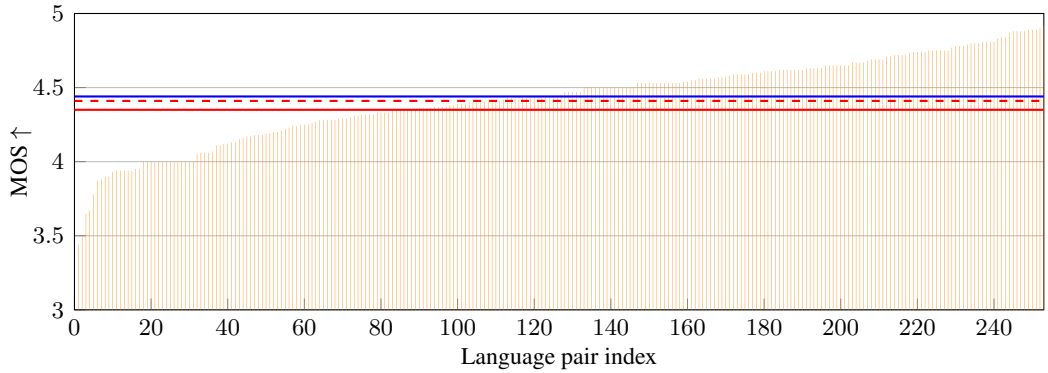
Table 9: Full experimental results for 253 language pairs. All the languages are denoted with ISO-639-1 code.

Lang. pair	RER	MOS	SIS	Lang. pair	RER	MOS	SIS	Lang. pair	RER	MOS	SIS
hr-cs	29.2	4.29	4.75	en-lv	32.5	4.0	4.81	hu-pt	30.2	4.4	4.75
hr-da	38.9	4.71	4.71	en-lt	33.5	4.0	4.82	hu-ro	29.2	4.88	4.8
hr-nl	34.7	4.12	4.79	en-mt	34.5	4.35	4.72	hu-sk	30.9	4.74	4.94
hr-en	35.1	4.75	4.7	en-pl	32.1	4.5	4.63	hu-sl	32.5	4.5	4.94
hr-et	34.2	4.44	4.85	en-pt	30.8	3.93	4.44	hu-es	27.2	4.59	4.94
hr-fi	34.7	4.36	4.78	en-ro	30.9	4.47	4.73	hu-sv	32.1	4.57	4.84
hr-fr	34.4	4.19	4.13	en-sk	33.8	4.5	4.5	it-lv	26.0	4.67	4.67
hr-de	31.1	4.15	4.76	en-sl	35.7	4.65	4.87	it-lt	26.6	4.89	4.61
hr-el	35.6	4.57	4.89	en-es	28.9	4.0	4.89	it-mt	27.4	4.45	4.58
hr-hu	32.2	4.56	4.44	en-sv	33.7	4.24	4.88	it-pl	25.1	4.75	4.76
hr-it	28.3	4.18	4.9	et-fi	31.6	4.61	4.94	it-pt	23.9	4.89	4.8
hr-lv	30.6	4.35	4.83	et-fr	34.3	4.59	4.86	it-ro	23.1	4.62	4.65
hr-lt	31.5	4.65	4.8	et-de	30.7	4.44	4.88	it-sk	26.4	4.65	4.95
hr-mt	33.7	4.37	4.59	et-el	35.9	4.61	4.79	it-sl	28.4	4.78	4.95
hr-pl	29.3	4.36	4.69	et-hu	32.0	4.29	4.93	it-es	20.5	4.5	4.42
hr-pt	30.7	4.53	4.5	et-it	28.8	4.84	4.94	it-sv	27.9	3.78	4.68
hr-ro	28.9	4.33	4.88	et-lv	30.7	4.53	4.88	lv-lt	28.4	4.4	4.62
hr-sk	30.3	4.69	4.69	et-lt	32.4	4.56	4.86	lv-mt	31.2	4.17	4.69
hr-sl	31.7	4.62	4.4	et-mt	33.5	4.39	4.65	lv-pl	28.8	4.33	4.56
hr-es	27.7	4.62	4.61	et-pl	31.3	4.54	4.78	lv-pt	28.3	4.53	4.44
hr-sv	33.4	4.37	4.65	et-pt	31.6	4.31	4.72	lv-ro	28.1	4.11	4.76
cs-da	35.8	4.26	4.36	et-ro	30.4	4.73	4.93	lv-sk	28.8	4.72	4.81
cs-nl	31.3	4.65	4.47	et-sk	32.3	4.11	4.75	lv-sl	31.5	4.5	4.84
cs-en	31.4	4.37	4.7	et-sl	34.5	4.63	4.78	lv-es	26.3	4.06	4.82
cs-et	31.2	3.65	4.5	et-es	28.2	4.65	4.78	lv-sv	30.9	4.25	4.57
cs-fi	31.7	4.77	4.88	et-sv	33.1	3.95	4.8	lt-mt	32.1	4.81	4.67
cs-fr	30.6	4.0	4.93	fi-fr	35.4	4.18	4.88	lt-pl	29.3	4.75	4.63
cs-de	28.1	4.5	4.75	fi-de	31.0	3.94	4.47	lt-pt	29.1	4.56	4.81
cs-el	33.2	4.29	4.63	fi-el	36.2	4.27	4.71	lt-ro	28.3	4.72	4.76
cs-hu	29.4	4.92	4.75	fi-hu	32.5	4.32	4.9	lt-sk	30.4	4.68	4.9
cs-it	25.2	4.32	5.0	fi-it	29.4	4.67	4.94	lt-sl	32.2	4.59	4.86
cs-lv	28.1	4.65	5.0	fi-lv	31.2	4.17	4.72	lt-es	26.3	3.9	4.78
cs-lt	29.9	4.44	5.0	fi-lt	32.5	4.38	4.47	lt-sv	32.3	4.69	4.62
cs-mt	30.6	4.41	4.67	fi-mt	33.9	4.28	4.82	mt-pl	30.5	4.47	4.56
cs-pl	25.7	4.53	4.94	fi-pl	32.1	4.47	4.73	mt-pt	30.6	4.35	4.77
cs-pt	27.8	3.94	4.61	fi-pt	32.6	4.06	4.8	mt-ro	28.9	4.53	4.71
cs-ro	26.5	4.41	4.77	fi-ro	31.5	4.9	4.87	mt-sk	31.7	4.81	4.86
cs-sk	26.1	4.63	4.76	fi-sk	33.0	4.44	4.83	mt-sl	33.9	4.8	4.76
cs-sl	29.4	4.74	4.71	fi-sl	35.3	4.44	5.0	mt-es	27.8	4.5	4.65
cs-es	25.3	4.63	4.82	fi-es	28.9	4.41	4.79	mt-sv	32.9	4.59	4.74
cs-sv	30.8	3.9	4.81	fi-sv	33.7	4.13	4.56	pl-pt	28.2	4.35	4.58
da-nl	36.2	4.58	4.59	fr-de	29.1	4.72	4.45	pl-ro	26.6	4.62	4.94
da-en	37.3	4.12	4.89	fr-el	35.2	4.33	4.75	pl-sk	27.3	4.63	4.62
da-et	38.0	4.88	4.45	fr-hu	33.0	4.33	4.61	pl-sl	29.5	4.62	4.79
da-fi	38.3	4.21	4.53	fr-it	27.2	4.65	4.63	pl-es	24.9	4.28	4.29
da-fr	36.8	4.53	4.88	fr-lv	32.1	3.88	4.75	pl-sv	31.1	4.56	4.71
da-de	31.9	4.42	4.88	fr-lt	33.7	4.18	4.59	pt-ro	25.4	4.13	4.85
da-el	40.2	4.0	4.65	fr-mt	34.1	4.2	4.69	pt-sk	29.4	4.4	5.0
da-hu	37.3	4.47	4.89	fr-pl	30.5	4.72	4.61	pt-sl	31.4	4.19	4.38
da-it	33.0	4.65	4.24	fr-pt	28.8	4.2	4.57	pt-es	22.8	4.5	4.85
da-lv	36.7	4.37	5.0	fr-ro	29.0	4.25	4.8	pt-sv	30.5	4.25	4.81
da-lt	37.8	4.39	4.56	fr-sk	33.2	4.74	4.41	ro-sk	29.0	4.75	4.83
da-mt	38.1	4.55	4.73	fr-sl	34.4	4.22	4.89	ro-sl	29.7	4.74	4.89
da-pl	36.0	4.24	5.0	fr-es	26.8	4.5	4.68	ro-es	23.0	4.69	4.71
da-pt	35.2	4.28	4.87	fr-sv	33.1	3.87	4.74	ro-sv	30.2	4.53	4.68
da-ro	34.4	4.89	4.83	de-el	32.9	4.4	4.71	sk-sl	30.4	4.61	4.65
da-sk	37.3	4.56	4.87	de-hu	29.4	4.5	5.0	sk-es	26.5	4.5	4.71
da-sl	39.3	4.59	4.65	de-it	25.5	4.53	4.86	sk-sv	32.1	4.78	4.74
da-es	33.0	4.83	4.11	de-lv	28.7	4.35	4.63	sl-es	28.3	4.89	4.94
da-sv	35.7	4.35	4.71	de-lt	30.0	4.5	5.0	sl-sv	34.2	4.53	4.72
nl-en	33.6	4.71	4.7	de-mt	30.8	4.39	4.89	es-sv	28.1	4.0	4.83
nl-et	34.7	4.62	4.74	de-pl	28.1	4.44	4.8	**Avg.**	**31.6**	**4.44**	**4.74**
nl-fi	35.1	4.32	4.71	de-pt	27.7	4.32	4.75				

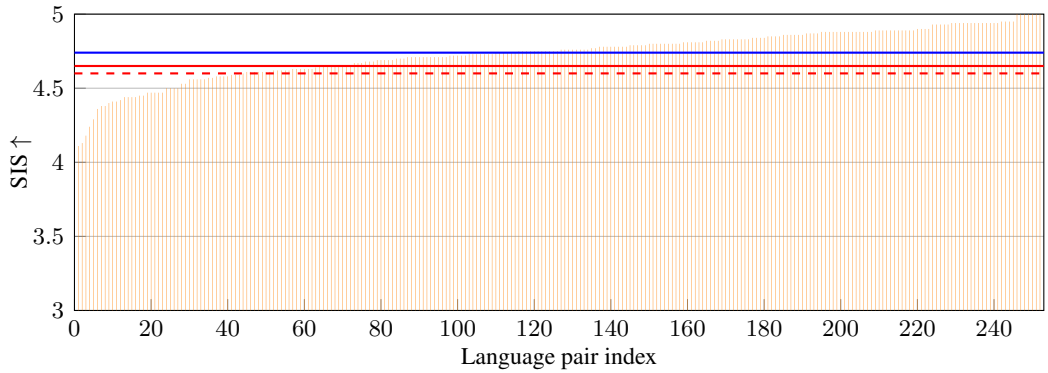
Table 10: Full experimental results for 253 language pairs. All the languages are denoted with ISO-639-1 code.



(a) RER Distribution



(b) MOS Distribution



(c) SIS Distribution

Figure 4: Performance distribution of CS-FLEURS. The blue line indicates the average for CS-FLEURS. Solid and dashed red lines represent the average for human-generated in-domain and out-of-domain datasets, respectively.

Model	Vocoder	RER ↓	NISQA ↑	SECS ↑	RTF ↓	Relative Speed ↑
kNN-VC	HiFi-GAN	25.0	<u>4.35</u>	<u>0.70</u>	**0.01**	×**27.00**
Diff-HierVC	HiFi-GAN	20.4	4.12	0.51	<u>0.18</u>	<u>×1.50</u>
	BigVGAN	<u>19.7</u>	4.23	0.50	0.19	×1.42
SeedVC	BigVGAN	**17.5**	**4.59**	**0.75**	0.27	×1.00

Table 11: Comparison of voice conversion models under cross-lingual settings. NISQA scores are obtained using the NISQA-v2 (Mittag et al., 2021) model, while speaker embedding cosine similarity (SECS) is computed based on embeddings from a pre-trained ECAPA-TDNN (Desplanques et al., 2020) model trained on VoxCeleb (Nagrani et al., 2017). For each metric, the best performance is indicated in bold, and the second-best is underlined.