Preserving Speaker Identity in Multilingual Translation: A Neural Machine Translation and Voice Cloning Approach

Srinivasaraghavan Sundar, Preethi Ramesh,
Shruthi Nanditha Ganesh

Department of Statistics, Rutgers University, Piscataway, NJ, USA.
Team No: 8.

Contributing authors: ss4805@scarletmail.rutgers.edu;
pr597@scarletmail.rutgers.edu; sg2057@scarletmail.rutgers.edu;

Abstract

The surge in globalization and the unprecedented rise in virtual collaboration in academia and industries alike has ushered in an era where real-time multilingual communication is no longer optional but essential. Yet current speech translation technologies force an unnatural choice: either communicate through impersonal, synthetic voices or lose the immediacy of real-time interaction. While separate solutions exist for machine translation and voice preservation, maintaining a speaker's identity and voice characteristics across languages remains an unsolved challenge. This project aims to bridge this gap by combining state-of-the-art Neural Machine Translation systems with advanced voice cloning systems, focusing specifically on English-French translation. The proposed system utilizes Marian MT for translation and XTTS for voice synthesis, with OpenAI's Whisper model handling the speech recognition component. Each of the three models in the pipeline were further finetuned with specialized datasets to reduce errors and optimize performance for our specific use case. OpenAI's Whisper-tiny model was finetuned on a section of the Librispeech dataset, achieving an overall WER score of 0.20 and CER score of 0.05, outperforming the base model. Marian MT, utilized for Neural Machine Translation was finetuned on OPUS dataset, resulting in an improved BLEU score of 40.38%. XTTS model was finetuned on Mozilla Common Voice French dataset, achieving improved scores compared to the base model. The enhanced performance metrics across ASR, translation, and voice synthesis components demonstrate the effectiveness of our integrated approach in preserving both linguistic accuracy and speaker identity. Finally, Mlflow is used

to build these experiments, register, evaluate, and serve the most performant versions of these models.

Keywords: Missing Data Imputation, Imputation, Climate Data Analysis, Regression, PMM, MICE

1 Introduction

In today's global landscape, effective multilingual communication tools are critical across diverse fields, from international business and customer service to real-time remote meetings and multimedia entertainment. As technology advances, the demand for cross-language systems that can retain the unique vocal identity of speakers has grown. Traditional translation systems often overlook the preservation of the speaker's voice, leading to an impersonal experience that may lack the intended emotional tone and expressiveness. This chasm is deepened in specialized verticals where technical terms are often lost in translation. Furthermore, ensuring real-time capabilities of these models adds a layer of complexity to the experiments.

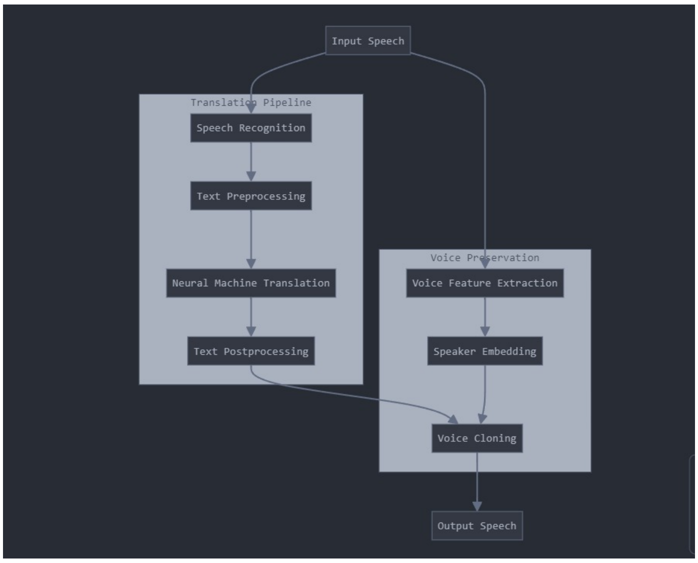

Fig. 1: Overall Architecture

The proposed system employs a dual-pipeline architecture to address the challenges of speech translation while preserving speaker identity. As illustrated in Figure 1, the system processes input speech along two parallel paths: a translation pipeline and a voice preservation pipeline. The translation pipeline handles the linguistic transformation through sequential stages of speech recognition, text preprocessing, neural machine translation, and text postprocessing. Simultaneously, the voice preservation pipeline extracts and processes speaker characteristics through voice feature extraction and speaker embedding generation. These parallel streams converge at the voice cloning stage, where the translated text is synthesized using the preserved voice characteristics of the original speaker. This architecture ensures that both the semantic content and speaker identity are maintained throughout the translation process, ultimately producing output speech that captures both the meaning and personal characteristics of the original utterance.

Despite individual components achieving adequate performance in isolation, the cascading nature of this multi-step pipeline necessitates careful fine-tuning of each model. Minor errors at an earlier point in the pipeline could propagate through the other steps in the channel, resulting in catastrophic errors in the final output. This makes it crucial to finetune each component individually. The challenge is compounded by the resource-intensive nature of fine-tuning large pre-trained models like Whisper, MarianMT, and XTTS, which demands substantial computational resources. To address these constraints, we developed intelligent data preprocessing and feature engineering methods to ensure enhanced performance despite having limitations in compute resources.

2 Literature Review

Recent advancements in machine learning have led to significant progress in speech recognition, machine translation, and speech synthesis individually. However, integrating these components while preserving speaker identity in cross-lingual settings remains an active area of research. This section reviews relevant work in speech translation systems and voice preservation techniques, with a focus on end-to-end approaches and real-time capabilities

Whispy, a sophisticated extension of OpenAI's Whisper model, was designed to overcome the limitations of real-time transcription. While Whisper achieved state-of-the-art performance in offline Automatic Speech Recognition (ASR) tasks, its reliance on pre-processed audio renders it unsuitable for low-latency applications. Whispy addresses this challenge with a buffer-based architecture that processes audio in smaller chunks, enabling real-time transcription while maintaining high accuracy and efficiency. Rigorous testing demonstrated Whispy's robustness across diverse datasets, highlighting its adaptability to practical scenarios such as live conferencing. Today, many architectures build on the strengths of powerful models like Whisper, and Whispy's innovations bridge the gap between offline precision and real-time functionality, marking a significant advancement in ASR technology [1].

Marian, a high-performance neural machine translation (NMT) toolkit, was designed to optimize both training and translation speed. Built entirely in C++ with minimal external dependencies, Marian utilizes dynamic computation graphs for automatic differentiation, enabling efficient training and inference. Its modular encoder-decoder architecture supports advanced translation models, including transformers, and offers features such as multi-GPU training and batched beam search. Evaluations have highlighted Marian's exceptional efficiency, demonstrating up to 30 times faster training speeds compared to Nematus on comparable tasks, all while preserving translation quality. Beyond translation, Marian has been adapted for complex tasks such as grammatical error correction and automatic post-editing, underscoring its adaptability and broad application potential [2].

XTTS: The existing models then were limited to a handful of high- or medium-resource languages. Catering to these issues, XTTS, a multilingual Zero-shot Multi-speaker Text-to-Speech (ZS-TTS) system, was introduced. Built on the Tortoise architecture, XTTS introduced novel modifications to enable multilingual training, enhance voice cloning, and accelerate both training and inference. Unlike existing models such as YourTTS, VALL-E X, and Mega-TTS 2, which supported only a few languages, XTTS extended support to **16 languages**. By enabling cross-lingual TTS without requiring parallel datasets, XTTS demonstrated unparalleled efficiency and adaptability. Additionally, its ability to fine-tune on minimal data for nuanced voice styles established it as a solution for diverse multilingual applications in ZS-TTS technology [3].

Whisper: Prior methods of speech recognition had various limitations, which led to the development of Whisper. Traditional supervised systems such as Deep Speech and SpeechStew relied heavily on limited, high-quality labeled datasets, while unsupervised methods like Wav2Vec 2.0 improved upon this by leveraging vast amounts of unlabeled audio. However, these unsupervised systems often lacked equivalently high-quality pre-trained decoders, requiring dataset-specific fine-tuning protocols that restricted their ability to generalize in diverse environments. Advancements from a study in weakly supervised learning, drawing inspiration from computer vision, showed that using larger, noisier datasets enhanced reliability. Whisper extended this approach extensively by harnessing **680,000 hours of diverse audio data** for both speech recognition and translation, enabling it to perform complex tasks across languages without prior specific training. This development underscored the substantial potential of enlarging weakly supervised datasets, establishing a new benchmark for reliable and adaptable speech recognition technologies [4].

Call Translator with Voice Cloning Using Transformers. Recent advancements in speech technology have addressed the persistent challenges of cross-lingual communication by integrating speech recognition, machine translation, and voice synthesis. Traditional systems often struggled with high word error rates (WER), translation inaccuracies, and a dependency on pre-trained voice models. However, modern approaches have significantly enhanced performance across these domains. Whisper AI achieved a remarkably low WER of 5%, ensuring accurate transcription. NLLB-200 improved translation accuracy by 44% BLEU over comparable models, while Tacotron and transformers enabled advanced voice cloning, replicating unique vocal traits for unseen speakers. This novel system removed the dependency on pre-trained voice models and introduced low-latency speech-to-speech translation. The integration of these models demonstrated the potential of leveraging cutting-edge technologies to enhance real-time multilingual communication [5].

LiveSpeech 2 incorporates the **Mamba architecture**, which enables linear-time decoding for efficient stream processing of infinitely long text sequences. By leveraging **rotary positional embeddings** and semantic guidance during decoding, the model ensures synchronization between text and audio streams while maintaining smooth transitions between audio chunks. Evaluations on datasets such as **LibriTTS** and **LibriLight** demonstrated competitive performance in speaker similarity and audio quality compared to state-of-the-art non-streaming models. These advancements highlight LiveSpeech 2's ability to bridge the gap between offline TTS quality and real-time functionality, marking a significant step forward in streaming TTS technology [6].

3 Dataset Description

The selection and preparation of appropriate datasets posed a significant challenge in developing our speech translation system. Each component required specific data characteristics - clean audio with accurate transcriptions for ASR (LibriSpeech), parallel corpora for translation (OPUS), and diverse speaker samples for voice synthesis (Mozilla Common Voice). This section describes our dataset choices and preprocessing methodology for training and evaluating the complete pipeline.

3.1 LibriSpeech

The LibriSpeech ASR corpus was used for training and fine-tuning Whisper, an (STT) model, to enhance its transcription performance. With approximately 1,000 hours of diverse 16 kHz audio and high-quality transcriptions, it offers robust training data for Whisper to adapt to varied speech patterns, accents, and audio qualities. The clean and noisy subsets allow Whisper to improve both in ideal conditions (train-clean) and challenging environments (train-other). By leveraging the dataset's speaker and linguistic diversity, Whisper can enhance its ability to generalize, providing more accurate and contextually appropriate transcriptions. Details regarding the LibriSpeech dataset is given in table 1

subset	**hours**	**per-spk minutes**	**female spkrs**	**male spkrs**	**total spkrs**
dev-clean	5.4	8	20	20	40
test-clean	5.4	8	20	20	40
dev-other	5.3	10	16	17	33
test-other	5.1	10	17	16	33
train-clean-100	100.6	25	125	126	251
train-clean-360	363.6	25	439	482	921
train-other-500	496.7	30	564	602	1166

Table 1: Description of LibriSpeech dataset

3.2 Opus 100

The Opus 100 dataset was used to develop and evaluate our system. This multilingual parallel corpus was an excellent fit because it covers 100 language pairs, including both high-resource languages like English-French and low-resource combinations like Swahili-Nepali, which allowed the testing of the model's performance across a wide range of linguistic challenges. The dataset is built from diverse sources, such as subtitles, legislative documents, and religious texts, which provided us with rich, varied data for training. The sentence-aligned pairs in the dataset were utilized to train and evaluate a neural machine translation (NMT) model effectively. Despite the dataset's invaluable diversity, challenges such as occasional noise and alignment issues were encountered, necessitating additional preprocessing. Overall, the use of Opus 100 enabled significant advancements in model performance, particularly in handling low-resource languages with improved effectiveness.

3.3 Common Voice French

Common Voice FR, the French subset of Mozilla's Common Voice dataset, is an open-source collection of speech data designed to advance speech recognition and speech synthesis technologies. It includes thousands of hours of validated French audio clips paired with text transcripts, contributed by volunteers worldwide. The dataset features audio clips in WAV format with an average duration of approximately 5 seconds, encompassing over 1,500 unique speakers. Metadata includes speaker demographics such as age, gender, and accent, ensuring coverage of a wide range of regional and linguistic variations. The dataset's diversity in accents, regional variations, ages, and genders makes it a valuable resource for fine-tuning XTTS to generate high-quality, natural-sounding French speech. Details regarding the LibriSpeech dataset is given in table 2

Language	Code	Voices	Total Hours	Validated Hours
Abkhaz	ab	3	<1	<1
Arabic	ar	225	15	9
Basque	eu	508	83	46
Breton	br	118	10	3
Catalan	ca	1,834	120	107
Chinese (China)	zh-ZH	288	12	11
Chinese (Taiwan)	zh-TW	949	43	33
Chuvash	cv	38	2	1
Dhivehi	dv	92	8	5
Dutch	nl	502	23	18
English	en	39,577	1,087	780
Esperanto	eo	129	16	13
Estonian	et	225	12	11
French	fr	3,005	184	173
German	de	5,007	340	325
Hakha Chin	cnh	280	4	2
Indonesian	id	54	5	4
Interlingua	ia	11	2	1
Irish	ga	63	3	2
Italian	it	602	40	36
Japanese	ja	48	2	2
Kabyle	kab	584	192	181
Kinyarwanda	rw	32	1	<1
Kyrgyz	ky	97	20	8
Latvian	lv	82	8	6
Mongolian	mn	230	9	8
Persian	fa	1,240	70	67
Portuguese	pr	316	30	27
Russian	ru	64	31	27
Sakha	sah	35	6	3
Slovenian	sl	42	5	2
Spanish	es	611	31	27
Swedish	sv	44	3	3
Tamil	ta	89	5	3
Tatar	tt	132	26	22
Turkish	tr	344	10	9
Votic	vot	2	<1	<1
Welsh	cy	748	48	42
TOTAL		**58,250**	**2,508**	**2,019**

Table 2: Description of Common Voice Dataset

4 Methodology

This section details the comprehensive methodology involved in developing the speech translation system. The discussion begins with preprocessing steps for spectrogram generation, followed by a detailed look at each components architecture. Finally, the finetuning strategies used for each model are discussed in detail.

4.1 Feature Engineering

Audio preprocessing plays a crucial role in the performance of speech recognition and voice synthesis models. Raw audio waveforms, while complete in information, are often too complex and computationally intensive for direct model processing. Converting these waveforms into mel-spectrograms provides a more efficient and structured representation that captures the essential characteristics of human speech while significantly reducing computational complexity. mel-spectrograms were used for audio processing, providing a visual representation of sound by showing the distribution of energy across different frequencies over time, adjusted to align with human auditory perception. In this process, the audio signal is divided into small chunks using the Short-Time Fourier Transform (STFT), which captures the frequency content of short segments of the signal. These frequency components are then mapped to the mel scale, a perceptually motivated scale that approximates how humans perceive pitch differences. To enhance representation, the mel-spectrogram values are converted to a logarithmic scale, making it easier to highlight quieter sounds relative to louder ones.

Key parameters for generating mel-spectrograms include:

Number of Frequency Bands (Mel Bands): Determines the granularity of the frequency representation.

Hop Length: Defines the time interval between successive chunks, controlling the time resolution.

FFT Size: Specifies the resolution of the frequency components within each chunk.

The resulting mel-spectrograms were saved in .npy format, ensuring faster loading during training and reducing computational overhead. This feature extraction process provides a compact yet rich representation of the audio signal, enabling models to better capture the structural and temporal patterns inherent in speech data. Additionally, the mel-spectrogram can be visualized to interpret the transformed representation of the audio, offering insights into the frequency and temporal characteristics of the signal. An example mel spectrogram that was preprocessed is given in the figure 2.

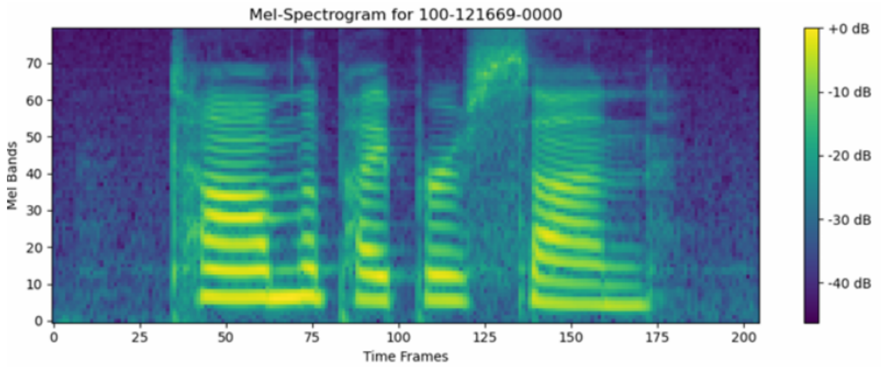

Fig. 2: Mel Spectrogram Sample

4.2 Model Architectures

The speech translation system comprises three distinct models, each handling a specific aspect of the pipeline: OpenAI's Whisper for speech recognition, MarianMT for neural machine translation, and XTTS for voice synthesis. This section details the architecture and key characteristics of each model, focusing on their specific roles within the system and the modifications made to enhance their integration.

4.2.1 OpenAI Whisper Tiny

The **Whisper tiny model**, a lightweight variant of OpenAI's Whisper architecture, was utilized in this project for efficient speech-to-text conversion. Designed with **37 million parameters**, the model offers a balance between computational efficiency and performance, making it suitable for real-time transcription and multilingual applications. It processes audio inputs in the source language and outputs accurate textual transcriptions in the same language.

Key Features

- Multilingual Training:
  - The model was trained on a diverse dataset comprising multilingual audio and labeled text.
  - Enables robust handling of various languages, accents, and noise conditions.
- Transformer-Based Architecture:
  - Whisper Base employs a transformer architecture optimized for automatic speech recognition (ASR).
  - Extracts features from audio signals and decodes them into coherent text representations.

The architecture of Whisper, as shown in the figure 3, follows an **encoder-decoder transformer design** optimized for automatic speech recognition (ASR). The process begins with audio input represented as a **log-mel spectrogram**, which captures time-frequency features of the signal. This spectrogram is passed through 2 convolutional layers (Conv1D) with GELU activation, followed by sinusoidal positional encoding, which helps the model understand the position of features in the sequence.

The resulting features are fed into a series of **encoder blocks**, which process and compress the spectrogram into a rich latent representation. These encoded features are then passed to the **decoder blocks** through cross-attention mechanisms. The **decoder**, which includes learned positional encodings, takes tokenized inputs such as special tokens for task prompts (e.g., SOT for start of transcription) and outputs the **next-token predictions** sequentially. This efficient interaction between the encoder and decoder allows Whisper to transcribe speech accurately while maintaining flexibility for multilingual and multitask training formats.

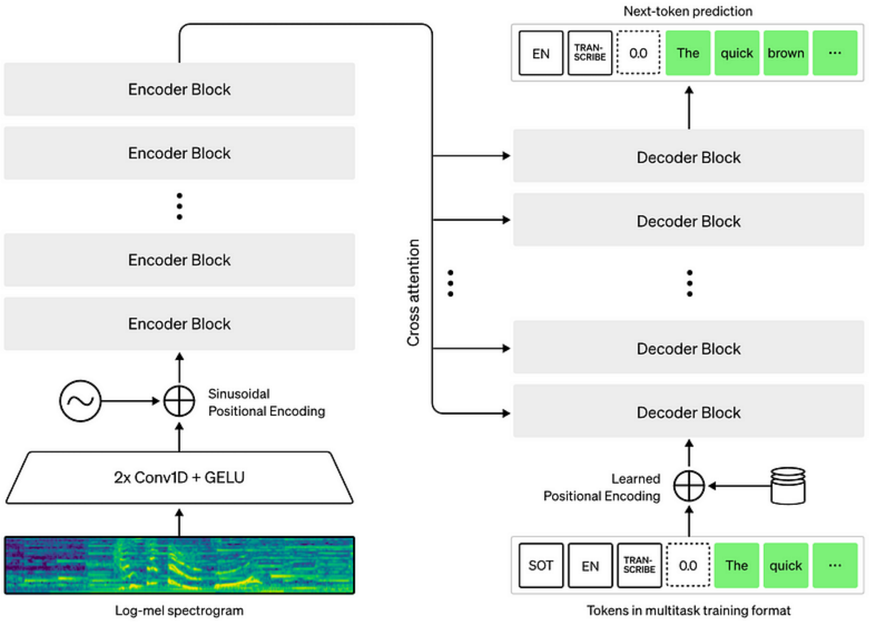

Fig. 3: Whisper Architecture

The **Whisper Base model** was integrated into the pipeline to transcribe multilingual audio inputs accurately. Its ability to handle diverse speech data ensured consistent performance across different linguistic and acoustic contexts. This model served as the foundation for subsequent stages, such as **translation** and **text-to-speech synthesis**, supporting seamless and scalable processing of audio data.

The use of Whisper Base demonstrated the effectiveness of compact, pretrained ASR models in delivering reliable transcription while optimizing resource utilization, aligning with the project's objectives.

4.2.2 MarianMT

Marian NMT, an open-source neural machine translation framework, was developed to offer sophisticated text-to-text translation capabilities. It incorporates advanced **encoder-decoder architectures** with attention mechanisms, enabling it to focus dynamically on specific segments of input to maintain context and meaning, especially in complex sentences. This feature is crucial for delivering high-quality translations.

The framework supports multiple language pairs, making it suitable for a wide range of translation needs, from simple conversational texts to complex technical documents. Marian NMT is designed to leverage **multiple GPUs**, facilitating the parallelization of tasks. This capability allows for rapid processing, which is essential for handling high-volume, time-sensitive translation projects.

The base model of Marian NMT, pre-trained on the **Europarl** and **Common Crawl** datasets, includes **74 million trainable parameters**. These datasets provide a robust linguistic base that enables the model to accurately translate various languages and dialects with exceptional contextual sensitivity. The architecture of the model can be seen in figure 4

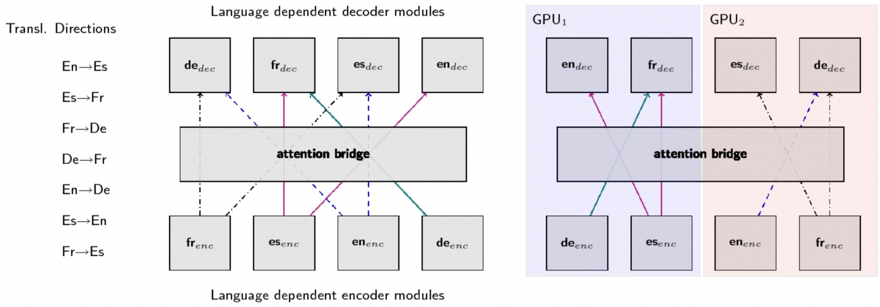

Fig. 4: Marian NMT Architecture

As shown in the attached image, Marian NMT utilizes **language-dependent encoder and decoder modules** interconnected through an **attention bridge**. Each input language is processed by its specific encoder (e.g., en_{enc} for English, fr_{enc} for French), which converts the input text into an intermediate representation. The **attention bridge** dynamically aligns this intermediate representation with the corresponding language-specific decoder (e.g., es_{dec} for Spanish, de_{dec} for German).

The architecture is further optimized for **multi-GPU parallelization**, where encoders and decoders are distributed across GPUs (e.g., *GPU$_1$* and *GPU$_2$* in the diagram). This allows simultaneous handling of multiple translation directions, ensuring efficient resource utilization and high translation throughput. By leveraging these modular components and the attention bridge, Marian NMT can seamlessly process translations across diverse language pairs while maintaining contextual integrity and computational efficiency.

4.2.3 XTTS

The **XTTS V2** model, a multilingual text-to-speech (TTS) system designed for high-quality, natural-sounding speech synthesis across multiple languages and speaker profiles. Trained on diverse multilingual text and audio, it maintains clarity and natural intonation while supporting speaker adaptation for personalized voice synthesis. By leveraging advanced neural TTS techniques, XTTS V2 captures subtle prosodic features like pitch, rhythm, and emphasis. Additionally, it enables voice cloning and customization, allowing the generated speech to mimic the characteristics of a target speaker.

The architecture of XTTS V2, as illustrated in the figure ??, integrates multiple interconnected components to achieve high-quality speech synthesis. Input text

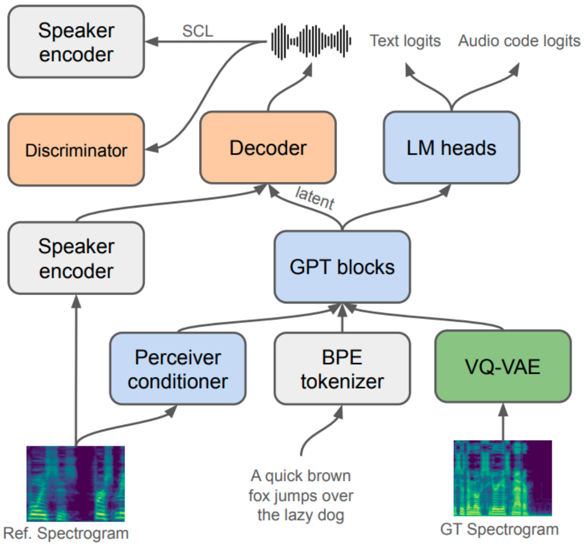

Fig. 5: XTTSv2 Model Architecture

is tokenized using the **BPE tokenizer**, while the **Perceiver conditioner** processes additional contextual information. These inputs, along with speaker embeddings derived from the **Speaker Encoder**, are passed into **GPT blocks**, where latent representations are generated. These latent representations form the foundation for downstream modules: the **Decoder** processes the latent features into audio representations, while **LM heads** generate logits for text and audio codes. A **VQ-VAE** module transforms these intermediate representations into high-quality spectrograms, which align with a ground-truth (GT) spectrogram for fidelity. The **Speaker Encoder** and a reference spectrogram enable voice cloning and adaptation, ensuring that the synthesized speech retains the characteristics of the target speaker. Finally, a **Discriminator** validates the audio quality, ensuring that the generated speech sounds natural and expressive.

XTTS V2 was used to synthesize speech from translated text outputs, providing natural and expressive audio in the target language. The model's ability to handle diverse linguistic inputs and speaker profiles ensured accurate and contextually appropriate voice synthesis. Additionally, XTTS V2's support for speaker adaptation allowed for voice cloning, preserving the vocal identity of the original speaker in the synthesized output. The integration of XTTS V2 enhanced the project's capability to deliver high-quality multilingual speech synthesis, demonstrating its effectiveness in creating accessible and natural-sounding audio.

4.3 Finetuning Strategy

This section details the finetuning strategies utilized for each model.

4.3.1 Finetuning Whisper Model

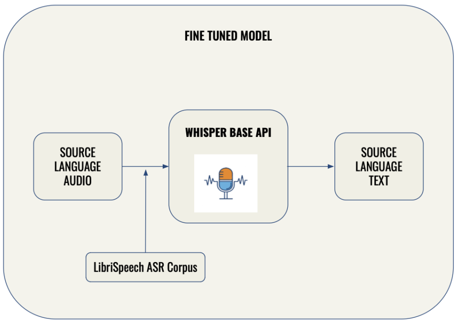

Fig. 6: Whisper fine-tune

The Whisper-tiny model, containing approximately 37 million trainable parameters, was selected for the ASR component due to its balance of performance and computational efficiency. Fine-tuning was performed using a subset of the LibriSpeech dataset, specifically the train-clean-100 portion, comprising 100 hours of English speech. The dataset was split into 22,000 training samples and 2,000 test samples to ensure robust evaluation of the fine-tuning process. Fine-tuning experiments were conducted on an NVIDIA RTX 3060 GPU with 6GB memory. The optimization process explored several hyperparameters and architectural modifications. These included systematic adjustments to the learning rate, experiments with selective freezing of encoder-decoder layers, and variations in batch sizes to optimize training efficiency and model performance. Model performance was evaluated using two standard metrics in ASR: Word Error Rate (WER) and Character Error Rate (CER). WER measures the minimum number of word-level operations needed to transform the predicted transcript into the reference text, while CER performs the same calculation at the character level. Lower scores in both metrics indicate better performance. The fine-tuning objective focused on reducing these error rates compared to the base Whisper-tiny model.

4.3.2 Finetuning MarianMT Model

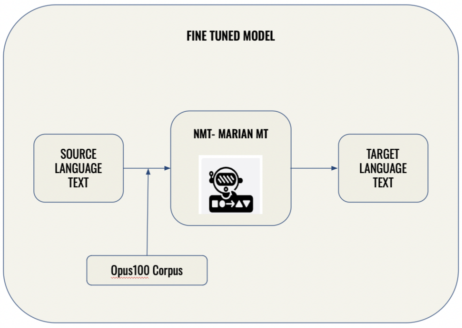

Fig. 7: MarianMT fine-tune

The MarianMT model, with its 74 million trainable parameters, forms the critical translation component of the pipeline. The OPUS dataset, used for fine-tuning, provided high-quality parallel sentences in English and French, carefully curated to cover a diverse range of topics and linguistic structures. From the original dataset, 112,500 sentence pairs were selected, with particular attention to sentence length, complexity, and domain relevance. This was further split into 12,000 training samples and 500 test samples, striking a balance between training robustness and computational efficiency on the NVIDIA A100 40GB GPU. The fine-tuning process explored various hyperparameter combinations, including learning rate adjustments, selective layer freezing, and batch size optimization. These experiments were conducted to enhance the model's ability to handle nuanced translations while maintaining computational efficiency. Translation quality was measured using the BLEU score, which provides a nuanced evaluation of translation accuracy. BLEU works by comparing the model's translations against reference translations created by human experts. It examines how many word sequences (from single words up to four-word phrases) match between the two versions, with longer matching sequences receiving higher weights. This approach helps ensure that the translations maintain both accuracy and natural flow. For instance, a translation that matches individual words but arranges them poorly would receive a lower score than one that preserves proper phrase structure. The fine-tuning process successfully improved upon the base model's BLEU score, indicating enhanced translation quality for English to French conversion.

4.3.3 Finetuning XTTS Model

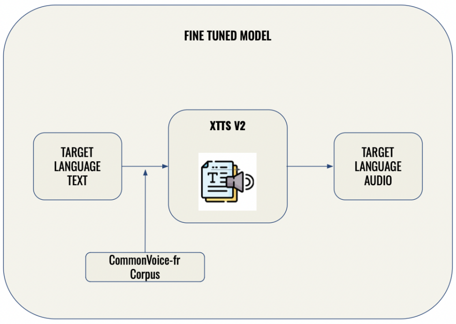

Fig. 8: XTTS fine-tune

The XTTS model, with its substantial 514 million trainable parameters, constitutes the voice synthesis component of the pipeline. The model was fine-tuned using Mozilla Common Voice French dataset, with 20,000 samples split into 18,000 training samples and 2,000 test samples. The fine-tuning process was conducted on an NVIDIA L4 GPU with 22GB memory. The fine-tuning implementation utilized a specialized training configuration designed to optimize voice cloning capabilities for French speech synthesis. The process incorporated several key technical components. The training configuration used a batch size of 3 with gradient accumulation steps of 84 to manage memory constraints, while employing the AdamW optimizer with a learning rate of 5e-06. A MultiStepLR scheduler was implemented for dynamic learning rate adjustment at specific milestones during training. The audio processing pipeline was configured with specific sample rates - 22050Hz for training and 24000Hz for output processing. The model architecture was carefully constrained for optimal input processing, limiting the maximum conditioning length to 132,300 samples (approximately 6 seconds) and the minimum to 66,150 samples (3 seconds). Additional constraints included a maximum waveform length of 255,995 samples (approximately 11.6 seconds) and a maximum text length of 200 tokens. These parameters were crucial for maintaining consistent voice quality while ensuring computational efficiency during the fine-tuning process. MelCE (Mel Cepstral Error) loss quantifies the difference between predicted and reference Mel-frequency cepstral coefficients, providing a measure of how closely the model's generated audio features match the ground truth. Therefore this was selected as the ideal evaluation parameter for evaluating voice cloning.

4.4 Model Lifecycle Management

To ensure systematic tracking and deployment of model versions, MLflow was integrated into the pipeline for model lifecycle management. This approach was implemented across all three components - Whisper, MarianMT, and XTTS - enabling consistent model versioning, performance tracking, and deployment. Each fine-tuned model was logged to MLflow along with its essential metadata, including model parameters, training configurations, and evaluation metrics. For instance, in the Whisper ASR component, both WER and CER metrics were tracked alongside the model

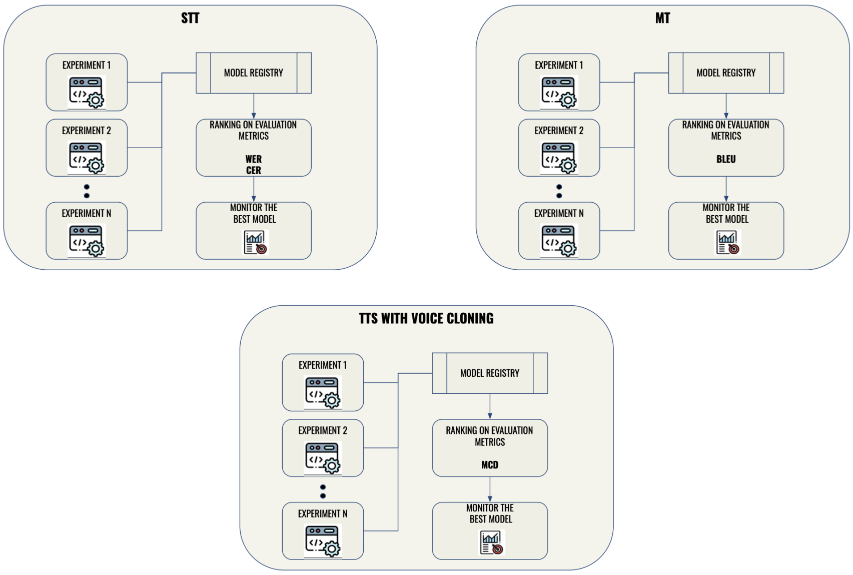

Fig. 9: Model Lifecycle Management

checkpoints. The logging process included saving the model architecture and processor configurations, ensuring reproducibility and maintaining a comprehensive record of each training iteration. The model selection and deployment process was automated through MLflow's registry system. The pipeline includes functionality to retrieve the best-performing model versions based on their evaluation metrics. This systematic approach allows for version control and easy rollback capabilities, with models being tracked through different stages from development to production. The system maintains the ability to load specific model versions as needed, facilitating both experimentation and stable deployment. This infrastructure ensures that model performance can be reliably tracked and compared across different fine-tuning iterations, providing a robust foundation for continuous model improvement and deployment. The figure 9 details the individual pipelines for model deployment.

5 Results

The evaluation of the multi-component speech translation system required distinct metrics for each component to assess performance effectively. The Whisper ASR model's accuracy was measured using Word Error Rate (WER) and Character Error Rate (CER), which quantify transcription accuracy at word and character levels respectively. For the MarianMT component, the BLEU score was employed to evaluate translation quality by comparing the model's output against reference translations. The XTTS voice synthesis model's performance was assessed through standard speech quality metrics focusing on both intelligibility and speaker similarity to ensure effective voice preservation. These metrics collectively provide a comprehensive evaluation of the system's ability to maintain both accuracy and naturalness throughout the translation pipeline.

5.1 Results of ASR:

The fine-tuning experiments for the Whisper model revealed several key insights about model optimization. Initial attempts to reduce computational overhead by freezing encoder layers proved counterproductive - while this approach decreased training time, it led to a significant deterioration in Word Error Rate (WER), indicating the importance of allowing the entire model to adapt to the target domain. The experimental results demonstrate a clear relationship between learning rate and model performance. As shown in the plots and tables, smaller learning rates generally yielded better results, with the model achieving optimal performance at a learning rate of 9e-9. This pattern suggests that gentler parameter updates help the model retain valuable knowledge from its pre-training while making necessary adjustments for the specific task. Fine Tune 8, trained for 8 epochs, achieved the best performance with a WER of 0.2064 and CER of 0.0573, representing significant improvements over the base model's WER of 0.2569 and CER of 0.0698. These results can be seen in figure 10

Model	Learning Rate	WER
Base Model	-	0.2569
Fine Tune 1	5e-5	10.459
Fine Tune 2	5e-9	0.24
Fine Tune 3	1e-9	0.255
Fine Tune 4	5e-10	0.26
Fine Tune 5	7e-9	0.2352
Fine Tune 6	1e-10	0.25591
Fine Tune 7	9e-9	0.2270
Fine Tune 8 (8 epochs)	9e-9	0.2064359

Table 3: Whisper - Fine tuning for WER

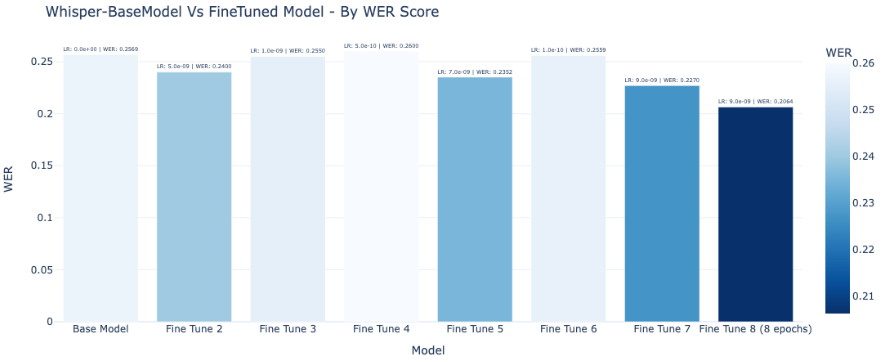

Fig. 10: Whisper-WER

The progression of model performance across different fine-tuning iterations shows a consistent trend of improvement, particularly in the later iterations. The improvement is more pronounced in WER compared to CER, suggesting that the fine-tuning process was particularly effective at correcting word-level errors while maintaining already strong character-level accuracy. This is evident from the visualization where the darker blue bars (later iterations) show progressively better scores. Building upon the previous analysis, the comparison between base and final models demonstrates substantial improvements in transcription accuracy. These results can be viewed in figure 11

The WER improved from 0.2569 in the base model to 0.2064 in the final fine-tuned version (Fine Tune 8), representing a 19.7% reduction in word-level errors.

Similarly, the CER improved from 0.0698 to 0.0573, showing an 18% reduction in character-level errors. These parallel improvements across both metrics indicate that the fine-tuning process successfully enhanced the model's overall transcription capabilities while maintaining a balanced improvement in both word and character-level accuracy.

Model	**Learning Rate**	**CER**
Base Model	-	0.0698
Fine Tune 1	5e-9	0.068
Fine Tune 2	1e-9	0.06943
Fine Tune 3	5e-10	0.0691
Fine Tune 4	7e-9	0.06534
Fine Tune 5	1e-10	0.0695
Fine Tune 6	9e-9	0.06324
Fine Tune 7 (8 epochs)	9e-9	0.0573

Table 4: Whisper - Fine tuning for CER

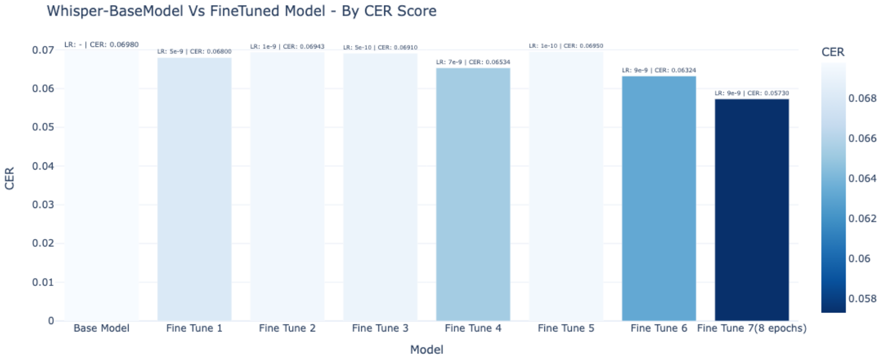

Fig. 11: Whisper-CER

5.2 Results of NMT:

The fine-tuning experiments for the MarianMT model demonstrated interesting patterns in translation quality. The base model began with a BLEU score of 38.54, and through careful tuning of the learning rate, we observed varying levels of performance. Fine Tune 1, using a learning rate of 5e-7, achieved the best performance with a BLEU score of 40.38, representing a 4.8% improvement over the base model. However, the experiments revealed sensitivity to learning rate selection. When the learning rate was decreased to 5e-9 in Fine Tune 2, the performance slightly degraded to a BLEU score of 38.36, slightly below the base model. More notably, increasing the learning rate to 5e-6 in Fine Tune 3 led to a significant performance drop, with the BLEU score falling to 31.23. This pattern suggests that while fine-tuning can enhance translation quality, the model is particularly sensitive to learning rate selection, with moderate learning rates producing optimal results. The learning rate sensitivity pattern in the MarianMT fine-tuning reveals a clear optimal range for model adaptation. At 5e-7 (Fine Tune 1), the model achieved its best performance, suggesting this rate provides the right balance between adapting to new patterns while retaining pre-trained knowledge. When decreased to 5e-9 (Fine Tune 2), the learning became too conservative, resulting in minimal model updates and performance similar to the base model. Conversely, increasing the rate to 5e-6 (Fine Tune 3) proved too aggressive, causing the model to deviate significantly from its well-trained base state, leading to a dramatic drop in BLEU score to 31.23. This pattern demonstrates that translation quality is highly sensitive to the magnitude of parameter updates, with moderate learning rates around 5e-7 providing the optimal trade-off between adaptation and stability. These results can be seen in figure 12

Model	**Learning Rate**	**BLEU Score**
Base Model	-	38.54
Fine Tune 1	5e-7	40.38
Fine Tune 2	5e-9	38.36
Fine Tune 3	5e-6	31.23

Table 5: Marian - Fine tuning for BLEU

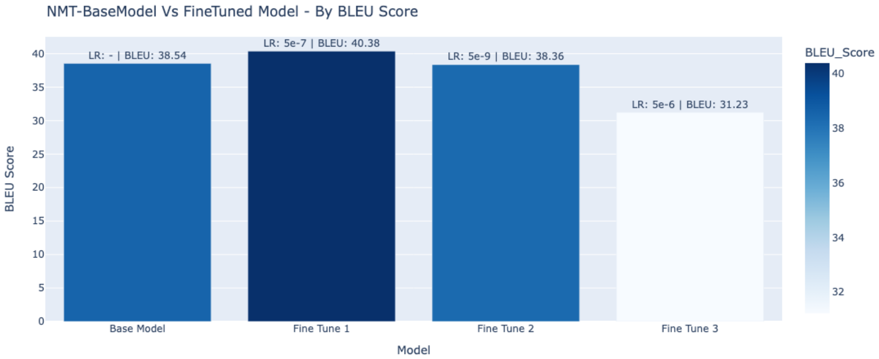

Fig. 12: NMT-BLeU

5.3 Results of Voice Cloning:

The fine-tuning results for the XTTS model demonstrate consistent improvement across multiple loss metrics over eight epochs of training. The model's performance can be analyzed through three key metrics:

The Text Cross-Entropy (Text_CE): loss shows a steady improvement, decreasing from 0.027 to 0.022 over the training period. This reduction indicates enhanced accuracy in the model's text processing capabilities and feature prediction. The gradual nature of this improvement suggests stable learning without overfitting.

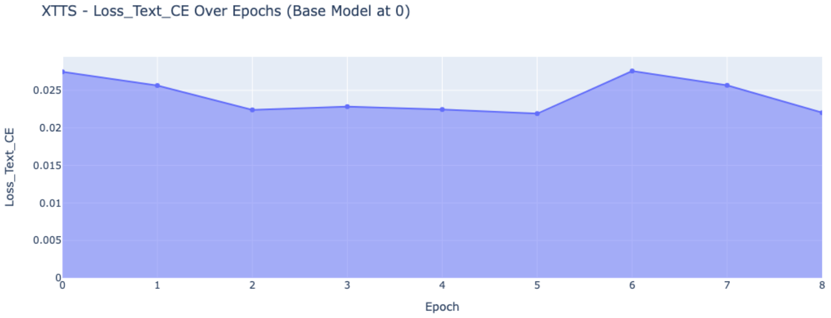

Fig. 13: XTTS - Text Cross-Entropy Loss

The Mel-Spectrogram Cross-Entropy (Mel_CE) loss exhibits the most dramatic improvement, starting at approximately 2.5 and concluding at 1.6 by the final epoch. This significant reduction reflects substantial enhancement in the model's ability to generate accurate acoustic features, particularly in matching the reference mel-frequency cepstral coefficients.

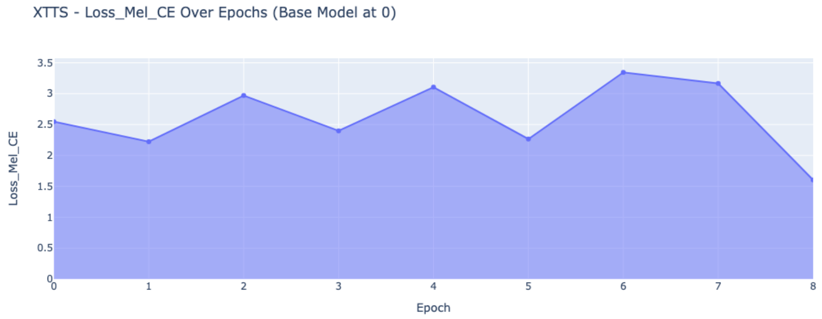

Fig. 14: XTTS - Mel-Spectrogram Cross-Entropy Loss

Total Loss, shows a clear downward trend from 0.03 to 0.01, demonstrating the overall effectiveness of the fine-tuning process. Notably, while there are minor fluctuations across epochs, the general downward trajectory in all three metrics indicates successful model adaptation without significant stability issues.

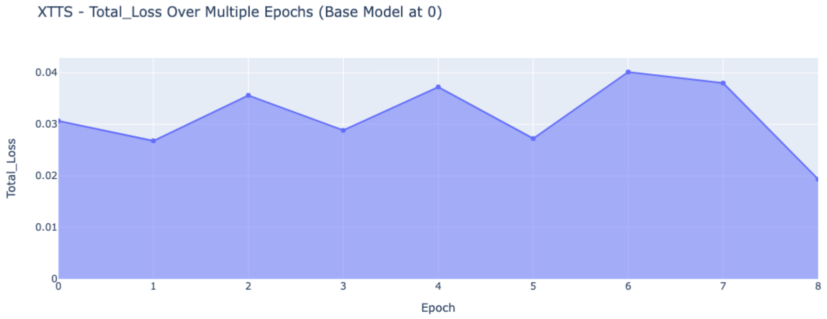

Fig. 15: XTTS - Total Loss

The relationship between the different loss components reveals interesting patterns in the model's learning process. While TextCE loss shows relatively subtle improvements (0.027 to 0.022), the MelCE loss demonstrates more dramatic changes (2.5 to 1.6). This difference in scale and improvement rate suggests that the model made more significant progress in acoustic feature generation compared to text processing. The total loss follows a pattern more closely aligned with the MelCE trajectory, indicating that acoustic feature accuracy plays a dominant role in overall model performance. These improvements have direct implications for voice quality in the final system. The substantial reduction in MelCE loss suggests enhanced accuracy in generating speech characteristics like pitch, timbre, and spectral features, which are crucial for natural-sounding speech. The steady improvement in TextCE loss indicates better text-to-phoneme mapping, potentially leading to more accurate pronunciation and rhythm in the synthesized speech. Combined, these improvements result in more natural-sounding voice synthesis with better preservation of speaker identity characteristics across languages. The results obtained for the base and finetuned XTTS model are shown in figures the above figures.

6 Conclusion

The fine-tuning process yielded significant improvements across all components of the speech translation system. The Whisper ASR model demonstrated a 19.7% improvement in WER (from 0.2569 to 0.2064) and an 18% reduction in CER (from 0.0698 to 0.0573), indicating substantial enhancement in transcription accuracy. The MarianMT component achieved a 4.8% improvement in BLEU score, increasing from 38.54 to 40.38, reflecting better translation quality. The XTTS voice synthesis model showed notable improvements across multiple metrics from the base model (epoch 0) to the final fine-tuned version: Text_CE loss decreased by 18.5% (0.027 to 0.022), Mel_CE loss improved by 36% (2.5 to 1.6), and total loss showed a significant 66.7% reduction (0.03 to 0.01), indicating substantial enhancement in voice synthesis quality. Looking forward, several avenues for enhancement present themselves. A more personalized approach could involve fine-tuning both ASR and voice cloning models using user-specific data, potentially creating more accurate and natural-sounding translations for individual users. Additionally, the translation component could be further enhanced by fine-tuning with domain-specific datasets, addressing specialized vocabulary and conventions in fields like healthcare, legal, or business sectors. This targeted approach would help address the current challenges in translating technical terminology and industry-specific nomenclature.

References

[1] Whispy: Real-Time Speech Recognition with Buffer-Based Architectures. *arXiv preprint arXiv:2405.03484*. Available at: https://arxiv.org/abs/2405.03484

[2] Junczys-Dowmunt, M., Grundkiewicz, R., Heafield, K., & Birch, A. *Marian: Fast Neural Machine Translation in C++*. arXiv preprint arXiv:1804.00344. Available at: https://arxiv.org/abs/1804.00344

[3] XTTS: A Multilingual Zero-Shot Multi-Speaker Text-to-Speech System. *arXiv preprint arXiv:2406.04904*. Available at: https://arxiv.org/pdf/2406.04904

[4] Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. *Robust Speech Recognition via Large-Scale Weak Supervision*. Available at: https://cdn.openai.com/papers/whisper.pdf

[5] Call Translator with Voice Cloning Using Transformers. *IEEE Transactions on Audio, Speech, and Language Processing, 2024*. Available at:

https://ieeexplore.ieee.org/abstract/document/10543304

[6] LiveSpeech 2: Real-Time Streaming TTS with Mamba Architecture. *arXiv preprint arXiv:2410.00767*. Available at: https://arxiv.org/abs/2410.00767