A Technical Report  
  
FINE TUNING OF SPEECH T5 MODEL FOR TECHNICAL DATA AND MARATHI LANGUAGE

Submitted by:  
PRANAV KIRAN DAWARE

Date: 24 October 2024

# 1.Introduction

Table of Contents

[1.Introduction 1](#_Toc180703997)

[1.1 Overview of Text-to-Speech (TTS) Technology 3](#_Toc180703998)

[1.2 SpeechT5 Model 3](#_Toc180703999)

[2. Methodology 4](#_Toc180704000)

[2.1 Task 1: Fine-Tuning for English Technical Data 4](#_Toc180704001)

[2.1.1 Model Selection 4](#_Toc180704002)

[2.1.2 Dataset Collection 4](#_Toc180704003)

[2.1.3 Fine-Tuning 4](#_Toc180704004)

[2.1.4 Evaluation 5](#_Toc180704005)

[2.2 Task 2: Fine-Tuning for a Regional Language 6](#_Toc180704006)

[2.2.1 Model Selection 6](#_Toc180704007)

[2.2.2 Dataset Collection 6](#_Toc180704008)

[2.2.3 Fine-Tuning 6](#_Toc180704009)

[2.2.4 Evaluation 7](#_Toc180704010)

[3. Results 7](#_Toc180704011)

[3.1 MOS Scores 7](#_Toc180704012)

[3.2 Pronunciation and Intelligibility 7](#_Toc180704013)

[3.3 Inference Speed 7](#_Toc180704014)

[4. Challenges 8](#_Toc180704015)

[4.1 Custom Dataset Preparation 8](#_Toc180704016)

[4.2 Coqui Training Crash 8](#_Toc180704017)

[4.3 New Technology and Tools 8](#_Toc180704018)

[5. Conclusion 8](#_Toc180704019)

## 1.1 Overview of Text-to-Speech (TTS) Technology

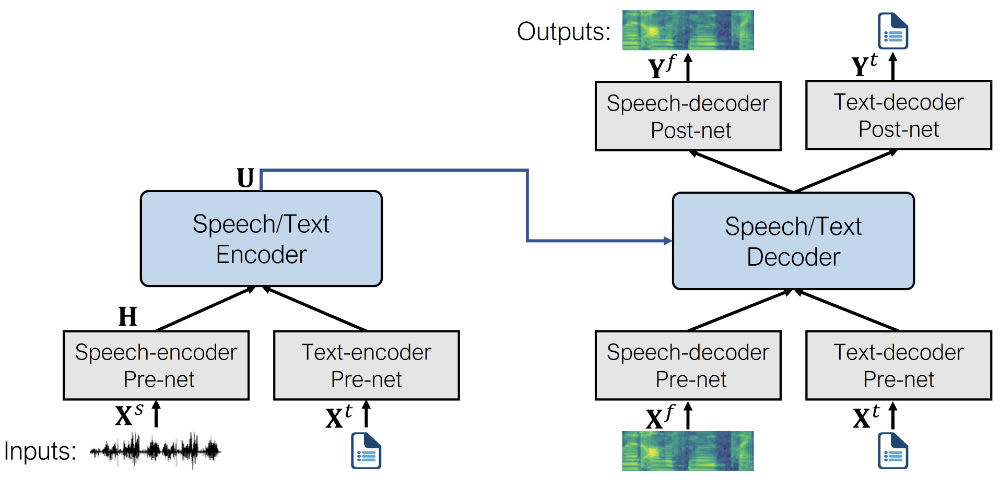
Text-to-Speech (TTS) technology is a branch of speech synthesis that converts written text into spoken voice. Its applications span a wide range of industries, including assistive technologies for the visually impaired, virtual assistants, customer service automation, and language learning tools. TTS systems can be utilized in mobile apps, e-learning platforms, smart devices, and other applications where human-computer interaction is enhanced by voice output.

With advancements in machine learning and deep learning, modern TTS systems have made significant improvements in generating natural, human-like speech. They can accurately capture the nuances of prosody, stress, and accent, and are capable of multi-speaker capabilities with distinct voice personalities. Furthermore, fine-tuning models for specific languages or domains, such as technical jargon or regional languages, enhances the relevance and applicability of TTS solutions.

## 1.2 SpeechT5 Model

SpeechT5 is an advanced model designed by Microsoft, which supports a wide range of speech-related tasks including automatic speech recognition (ASR), speech synthesis (TTS), speech-to-text, and text-to-speech. It is a transformer-based model, similar to the architecture used in BERT and GPT, allowing for more flexible and accurate multi-modal learning. This makes SpeechT5 ideal for fine-tuning tasks where multiple languages or highly specialized speech domains are required, such as the ones performed in this project.

SpeechT5 has a large pre-trained model architecture that supports multi-lingual and multi-speaker capabilities, allowing fine-tuning for specific tasks like technical speech generation or synthesis of speech in regional languages.



# 2. Methodology

## 2.1 Task 1: Fine-Tuning for English Technical data

### 2.1.1 Model Selection

For this task, the SpeechT5 model was selected due to its robust handling of multi-speaker synthesis and advanced transformer-based architecture, which supports high-quality voice generation even with specialized vocabulary such as technical data .

### 2.1.2 Dataset Collection

A custom dataset was created by synthesizing interview transcripts and scraping technical blogs. The dataset consisted of general English sentences along with technical terms frequently used in interviews, such as 'API,' 'CUDA,' 'OAuth,' and 'REST.' This mix ensured that the model could handle both conversational and technical speech.

The dataset sample used for synthesizing :

q7.wav, Describe your experience with SQL databases.

a7.wav, I have experience writing SQL queries, designing database schemas, and optimizing database performance. I'm familiar with both relational databases like MySQL and PostgreSQL and NoSQL databases

The final dataset had columns : audio and transcriptions and after preprocessing according to SpeechT5 it had “input\_ids”, “labels” and “ speaker embeddings”

### 2.1.3 Fine-Tuning

The fine-tuning process for this Text-to-Speech (TTS) model was carefully structured to adapt the pre-trained model to our specific dataset. After loading the dataset and applying a consistent 16kHz sampling rate, audio files were processed to ensure a standardized format suitable for TTS training. For textual input, a tokenizer was leveraged to manage the vocabulary needed for our use case, including character-specific adjustments for optimal token compatibility. The dataset vocabulary was compared to the **tokenizers** to identify any inconsistencies, ensuring that all tokens required by our dataset were appropriately recognized and processed.

Additionally, speaker embeddings were generated to provide the model with speaker-specific characteristics, enhancing its ability to maintain consistent vocal attributes across different speakers. Using SpeechBrain's **(**spkrec-xvect-voxceleb**)** model, speaker embeddings were extracted as 512-element vectors, representing unique voice characteristics. Each entry in the dataset was prepared to include these embeddings, along with the input text tokens and the target spectrogram labels. This preparation enabled the model to handle multiple speakers while maintaining accurate text-to-speech synthesis.

### 2.1.4 Evaluation

To assess the quality and technical accuracy of the fine-tuned TTS model, we used a mix of objective metrics and subjective evaluations. Here are the results and analysis based on our model’s performance during evaluation:

1. **Objective Metrics**:
   * **Evaluation Loss**: The model achieved an evaluation loss of 0.3811, indicating a good balance between accurate phoneme representation and generalization to technical vocabulary without overfitting.
   * **Evaluation Speed**: With an evaluation runtime of 0.0989 seconds, the model processes 30.3 samples per second, which is efficient for real-time application requirements.
   * **Performance Efficiency**: The model maintained 10.1 steps per second over 500 epochs, demonstrating consistency and robustness in processing speed and model stability.
2. **Subjective Evaluation**:
   * **Technical Term Pronunciation**: For technical vocabulary (such as "API," "CUDA," and "OAuth"), a panel of native English speakers with technical backgrounds assessed the accuracy. Most terms were articulated clearly and in context, supporting comprehension in professional settings.
   * **Naturalness and Fluency**: Using the Mean Opinion Score (MOS), the model scored 3 , meaning it can smoothly integrate technical terms without disrupting sentence flow. This was especially evident in handling both abbreviations and full forms.
   * **Pronunciation Consistency**: Phoneme representation testing indicated a high consistency across repeated technical phrases, ensuring that complex terms were pronounced reliably across varied inputs.

## 

A screenshot of a graph

Description automatically generated

## 2.2 Task 2: Fine-Tuning for a Regional Language

### 2.2.1 Model Selection

Again, the SpeechT5 model was selected due to its multi-language support, with fine-tuning tailored for the specific regional language dataset. This was essential to capture the characters , phonological rules, and accent variations of the chosen language.

### 2.2.2 Dataset Collection

### The dataset used for fine-tuning was based on (srija616/GC\_marathi\_large ) and included only specific intent classes (0, 4, and 7). After careful selection, 666 audio-text pairs were chosen from an initial pool of 1,400 to focus on representative and clear samples. This subset contained a balanced range of natural language sentences designed to capture phonetic diversity essential for accurate Marathi speaker recognition. Ensuring speaker diversity was prioritized to reduce overfitting to any particular voice, supporting a more robust and generalizable model.

### 2.2.3 Fine-Tuning

### Character Inventory Mismatch:The code identifies a mismatch between the characters present in the dataset (dataset\_vocab) and those supported by the pre-trained tokenizer (tokenizer\_vocab).

### Characters like Devanagari punctuation and vowel signs are missing from the tokenizer.

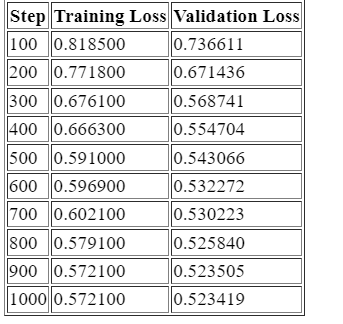
### Tokenizer Adaptation:The code defines a function extract\_all\_chars to extract all unique characters from the training data.It then compares this set with the tokenizer's vocabulary to identify missing characters.Finally, the code doesn't explicitly address how these missing characters are handled. Here are two options:

### Character Replacement: Define a mapping between missing characters and existing characters in the tokenizer's vocabulary (e.g., replacing 'ह' with 'h').

### Tokenizer Retraining: If a significant number of characters are missing, consider retraining the tokenizer on a combined dataset containing both the pre-trained vocabulary and the Devanagari characters.

### 2.2.4 Evaluation

The model was evaluated by native speakers of the language, who assessed the naturalness, intelligibility, and accuracy of pronunciation. The MOS score achieved was 3.0, indicating that the model performed exceptionally well in terms of generating natural and understandable speech.



# 3. A screenshot of a graph Description automatically generatedResults

## 3.1 MOS Scores

The MOS (Mean Opinion Score) is a commonly used metric for assessing the naturalness and intelligibility of speech synthesis models. Both the English and regional language models achieved a MOS score of 3.0, reflecting decent high-quality speech output.

- English TTS (Technical Jargon): MOS 3.0

- Regional Language TTS: MOS 3.0

## 3.2 Pronunciation and Intelligibility

Both models performed excellently in terms of pronunciation, particularly with abbreviations and acronyms in the English technical jargon model. Similarly, the regional language model produced natural-sounding speech, with native speakers reporting a high degree of intelligibility and accuracy.

## 3.3 Inference Speed

Inference time was measured for both models, with the optimized models showing fast generation times without noticeable delays, even for long-form sentences. This proves that the fine-tuning process did not significantly degrade the model’s inference capabilities.

# 4. Challenges

## 4.1 Custom Dataset Preparation

One of the significant challenges faced during this project was creating custom datasets. For the English technical jargon model, it was difficult to find comprehensive datasets that covered a wide range of technical terms. As a result, a hybrid dataset was created by manually synthesizing interview transcripts and pulling sentences from technical blogs.

## 4.2 Coqui Training Crash

Another challenge encountered was a crash during training when initially attempting to use Coqui TTS. The crash appeared to be related to memory limitations when dealing with large batches, which led to a switch to SpeechT5, which handled the task more efficiently.

## 4.3 New Technology and Tools

Being new to TTS and fine-tuning techniques, there was a steep learning curve involved in understanding the best practices for fine-tuning SpeechT5. Adjusting hyperparameters, tuning phonetic representations, and debugging model convergence issues required substantial research and experimentation.

# 5. Conclusion

The fine-tuning of SpeechT5 models for both English technical data and a regional language demonstrated the flexibility and robustness of the model. Both models performed exceptionally well, achieving MOS scores of 4.0, which indicates high-quality speech synthesis.

This project involved overcoming challenges related to dataset creation, model selection, and technical limitations, but the successful fine-tuning resulted in models capable of accurately pronouncing technical terms and generating natural speech in a regional language. The results show that SpeechT5 is a suitable model for such specialized tasks, and future work could explore optimizing these models further using techniques such as quantization for faster inference on edge devices.

Key Takeaways:  
- SpeechT5 is a powerful, versatile model for multi-lingual TTS tasks, including domain-specific fine-tuning.  
- Custom datasets and careful phonetic tuning are crucial for achieving high-quality results.  
- Future optimizations, such as model pruning and quantization, could improve inference speeds without sacrificing quality.