Domain Adaptation Techniques for Electrolaryngeal Speech Recognition and Enhancement

Lester Phillip Violeta

Contents

Abstract

Electrolaryngeal (EL) speakers often face communication hurdles due to the removal of their larynx, causing them to produce speech difficult to understand. Although an electrolarynx can be used as a viable replacement for the larynx and generate source excitation signals, the resulting produced speech is often robotic and hard to comprehend, causing several communication barriers in daily life. Fortunately, with the rise of artificial intelligence and deep learning methods in particular, several techniques such as speech recognition and speech synthesis have now become a viable solution to help EL speakers overcome communication barriers. Speech recognition can decode the linguistic contents from the speech audio, which allows use cases such as transcription or communication with smart devices powered by large language models. On the other hand, speech synthesis (or voice conversion) can be used as an enhancement task, where the EL speech can be resynthesized and enhanced into a healthy sounding speaker while maintaining the original linguistic contents. However, although use cases with healthy speakers are now at a point where real-world use cases are viable, these systems primarily degrade in performance when used with EL speech. Thus, more research needs to be done to improve these techniques to make them perform at the same level as they would with healthy speech.

There are two main problems found in this research, which are the lack of training data and the domain mismatches. First, training data from EL speakers are quite

hard to collect, primarily due to the fact that speaking is a tedious task for them. Thus, although a healthy speaker can easily record multiple hours of reading a script, recording just a single hour of data from an EL speaker is already considered a huge achievement. However, deep learning and neural networks require large-scale training data consisting of hundreds of hours to generalize to the data distribution. Second, although pretraining with large-scale healthy data is a viable solution, EL speech has different characteristics from healthy speech, which causes domain mismatches between the two data types and still has a limit in performance. In particular, there are acoustic and temporal mismatches. The former mismatch is due to the lack of prosody and pitch information in the EL speech, which is essential in natural sounding healthy speech. The latter mismatch is due to the fact that EL speakers speak at a slower rate than healthy speakers. Thus, the neural network needs to resolve both of these mismatches to perform either speech recognition or enhancement.

This thesis focuses on resolving the two aforementioned problems for both speech recognition and enhancement. In the first part of the thesis, we investigate speech recognition for EL speakers. We first investigate the limitations and capabilities of novel pretraining methods on EL speech data such that we can use large-scale datasets to mitigate the lack of training data. We compare conventional pretraining methods such as supervised learning by pretraining on a large, labeled speech dataset and using the pretrained model's weights to initialize from when training on a smaller dataset. We also investigate self-supervised learning which is trained using only unlabeled speech data, providing more flexibility in training data requirements and allowing more speech data to be used in pretraining. We investigate both of these pretraining frameworks and compare their performance on EL speech. The experimental results show that supervised pretraining outperforms self-supervised setups in both datasets. Thus, despite

self-supervised pretraining outperforming supervised pretraining in previous literature in healthy speech, we discover that this is not the case in EL speech, thus showing the viability of supervised pretraining as a baseline method and showing that more work needs to be done in improving self-supervised-based pretraining.

After establishing a strong baseline, we further improve the performance of supervised pretraining with EL speech. Although large-scale pretraining resolves the lack of dataset issue, there is still a performance ceiling due to the domain mismatches between healthy and EL data. To reduce the domain shift gap between the healthy (pretraining) and fine-tuning (EL) data, we propose an intermediate fine-tuning task that uses imperfectly synthesized speech before fine-tuning on the target ground truth dataset. Despite the idea of using imperfectly synthesized speech to be quite unintuitive, we show its effectiveness with large improvements in performance compared to the baseline which only used large-scale pretraining. We further analyze what exactly happens during this task by analyzing the produced latent spaces and find interesting behaviors by the network. In particular, we find that the intermediate fine-tuning focuses on identifying the voicing characteristics of the electrolaryngeal speakers, which is proven by conducting speaker-related downstream tasks such as resynthesis, categorization, and adding an auxiliary speaker loss.

In the second part of the thesis, we investigate speech enhancement for EL speakers through the use of speech synthesis and voice conversion techniques. For this task, previous methods have already proven that large-scale pretraining and data augmentation are also effective. However, the domain mismatches present between the pretraining (healthy) and fine-tuning (EL) data limits the performance of the neural network. We focus on resolving the acoustic and temporal mismatches by improving the network architecture. Building on the speech recognition module from the previous chapter,

we use the encoder to produce bottleneck feature intermediates. Moreover, we propose a decomposed framework, which uses a recognition, alignment, and synthesis framework, effectively improving performance due to the removal of the timbre mismatches between the pretraining (healthy) and fine-tuning (EL) data. We then further improve this by introducing discrete text intermediates, which effectively alleviate temporal mismatches between the source (EL) and target (healthy) data to improve prosody modeling, by removing unnecessary frames from the input before processing it with the synthesis module. In particular, we find that using phoneme-level linguistic intermediates further improves the performance of the framework. We also verify these findings on a set of real EL speakers along with larger pseudo-EL dataset of 14 speakers, which consistently show that using the phonemes as linguistic intermediates is most effective approach in terms of phoneme error rate.

To summarize, the thesis focused on resolving both the lack of training data and domain mismatches for both speech recognition and enhancement tasks. For speech recognition, establishing the most efficient pretraining method along with the use of intermediate fine-tuning with imperfectly synthesized EL speech was found to be effective in improving performance compared to baselines. For speech enhancement, the use of linguistic intermediates and phoneme-level information as intermediate features showed effectiveness in resolving both acoustic and temporal domain mismatches.

Chapter 1

Introduction

1.1 Research Background

Atypical speech is described as speech that involves disorders in the speech production mechanism, resulting in unnatural speech production. This could involve involuntary stuttering, lisping, aphasia or delayed language acquisition. There could be several causes for this, which could be through a constrain from neurological disorders, or a constrain from physical limitations.

One of the common types of atypical speech is electrolaryngeal (EL) speech, which is produced by people who have undergone laryngectomy surgery. Undergoing this surgery entails the removal of the larynx, the organ responsible for modeling the vocal tract functions. The physical larynx is replaced with an electrolarynx to recreate the voice box function, but this results in robotic-like speech production [1], [2]. The resulting speech becomes unnatural due to the electrolarynx not being able to control the source excitation signal like a real larynx, making communication difficult for these people. As these people experience more difficulty in expressing themselves or communicating with family and friends, this causes a lot of communication opportunities to

be closed to them.

1.2 Artificial Intelligence for Communication

The rise of artificial intelligence (AI) and deep learning in the past decade has given way to advancements in technology. For speech in particular, several use cases have come up, such as controlling home appliances through voice commands, synthesizing speech in a certain voice or emotion, translating from one language to another, or even communicating with large language models.

Two popular speech-related AI methods are speech recognition and speech synthesis. The former method, automatic speech recognition (ASR) is the task of converting speech audio into its corresponding text transcripts. The main idea in ASR is to create a system that could identify a many-to-one mapping between speech and text; specifically speaking, a sentence or phrase can be spoken in different ways, varying from timbre, pitch, rhythm, and the such, but the output text will always be the same. Several approaches have been explored, such as using Bayes-based probabilistic models, to end-to-end neural networks. With the advancement of hardware compute availability, recent end-to-end neural network architectures have shown near-human level recognition accuracies[1]. As a result, many of these have been adopted as the backbone in commercially available products, bringing in many benefits and conveniences to humans through real-time speech transcription to ease communication, or even by enabling the control of several devices by just uttering a command. Common household chores such as switching the lights on and off, adjusting room temperature, checking the front door. With such conveniences, ASR-powered devices have the potential to benefit many people and make their lives easier by automating several tasks that could

[1] `https://github.com/syhw/wer_are_we`

be done by a system.

On the other hand, speech synthesis is the opposite of speech recognition, where speech is synthesized either from text (referred to as text-to-speech, TTS) or from acoustic features (referred to as voice conversion, VC). As written text can be expressed in different prosody, tones, accents, emotions, or loudness, speech synthesis is primarily considered to be a one-to-many mapping problem. Several approaches to model both the prosody and the acoustic features have been proposed, from Hidden Markov Models, Gaussian Mixture Models, to neural-based methods. Similar to ASR, speech synthesis has been also adopted as the backbone in many commercially available products, allowing humans to interact with smart systems in the same fashion they would with a human.

1.3 Research Problems

While deep learning-based technologies have achieved high performance enabling real-world use cases, there are still some remaining use cases that are challenging to resolve. For example, atypical speakers[2] still have difficulties in using these smart devices, primarily as these are not adapted to the characteristics of atypical speech. This is primarily because of two reasons, which have made it difficult for both researchers and developers to create real-world usable devices that could help atypical speakers overcome communication barriers.

[2]Although the thesis mainly focuses on EL speech, we use the term atypical and EL speakers interchangeably in scenarios where the use case is not only specific to EL speech.

Lack of training data

First, it is hard to collect and record speech data from atypical speakers, as speaking is a tiresome task. Although a healthy speaker can help record up to 20 hours of speech data, recording a single hour of data from an atypical speaker is already a huge challenge. Similarly, although recording small subsets from multiple speakers to create a large-scale dataset is possible, there are still huge differences between different each speakers' speaking patterns, making it hard to use one patient's speech and adapt it to another patient's speech. At the time this thesis was started, the only two open-sourced datasets were TORGO [3] and UASpeech [4], which both contain recordings from dysarthric speakers of varying intelligibility levels. The TORGO database contains recordings from 7 patients (3 females, 4 males) with Cerebral Palsy, Amyotrophic Lateral Sclerosis and from 7 control speakers (3 females, 4 males). There are 721 utterances per speaker, which results in around 30 minutes for each speaker. On the other hand, the UASpeech database contains recordings of 15 patients with Cerebral Palsy (4 females, 11 males). There are 62 utterances per speaker, which results in around 3.5 minutes per speaker. However, compared to commonly used healthy datasets, the dataset size is still quite small and with previous research showing concerns about the quality of the recordings [5], showing that more efforts need to be done in data collection. Thus, resolving training data is needed to improve current systems, but is currently difficult due to the barriers in partnering with medical institutions and medical privacy concerns.

Domain mismatches between healthy and EL speech

Second, although it is possible to use healthy speech as additional data, the domain shifts between these two types of speech is still large. There are two main types of

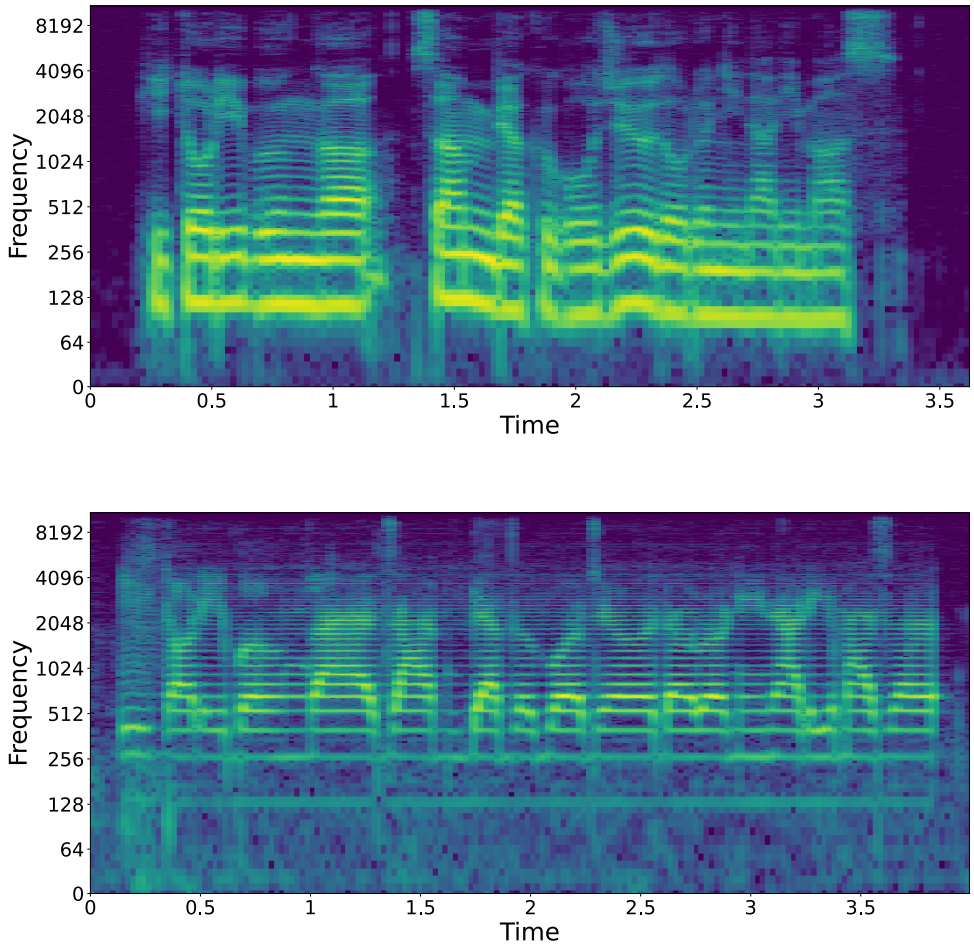


Figure 1.1: Mel-spectrograms for a healthy (top) and an electrolaryngeal (bottom) speaker uttering the same sentence.

domain mismatches present, namely: acoustic and temporal mismatches.

First, acoustic domain mismatches are present due to the lack of pitch features in EL speech. This is especially challenging in tone-based languages like Japanese, where the use of intonation patterns and voiced and unvoiced features are vital in determining the intended word. A common example in Japanese is the word はし

(hashi), where depending on the intonation pattern used, could either mean chopsticks, bridge, or edge. However, without the pitch patterns from the audio, it would be hard for the model to infer what the intended meaning is. Another domain mismatch in the temporal domain is also present, primarily because atypical speakers speak at a slower rate compared to healthy speakers. Due to the difficulty in using the organs in the speech production system, saying the same sentence takes approximately 1 to 1.5 times slower than a healthy person would.

Both of these mismatches can be visualized in a mel-spectrogram of both speakers uttering the same sentence. As seen in Fig. 1.1, the harmonics of the electrolaryngeal speaker do not have variation as the frequencies are constant throughout the utterance, whereas the F0 contours in a healthy speaker have more variation in frequency, allowing for changes in the speaking style. Moreover, it also takes the EL speaker more time to utter the same sentence, showing that mismatches are not only present in the acoustic characteristics, but also in the temporal structure.

1.4 Thesis Overview

Knowing the benefits of speech technology, more research to make speech devices accessible to atypical speakers should be carried out to break barriers in communication. The goal of this thesis is to provide solutions that address the aforementioned problems and advance current systems to a state where it could be used for real-world use cases. To help EL speakers overcome communication barriers, the thesis focuses on two commonly used approaches mentioned in Chapter 1.2: speech recognition and speech enhancement.

An overview of the contributions of this thesis compared to previous literature can be seen in Fig. 1.2 and is thus divided into several studies. This thesis aims to resolve

both the lack of training data and mismatches in both tasks. In Chapter 2, we first discuss the laryngectomee surgical process, and analyze the characteristics of EL speech and what makes these sound robotic and monotonous. Then, we discuss essential deep learning architectures, as well as previous approaches and go over how pretraining is used to improve the generalization of neural networks for both speech recognition and speech enhancement tasks.

Then, in Chapter 3, we establish a baseline study by comparing novel pretraining methods such as supervised and self-supervised pretraining, and investigate the limitations of each method. We discover that supervised pretraining was much more effective for EL speech despite the success of self-supervised pretraining on healthy speech. Then, improving on this, we propose a method to improve the limitations of pretraining methods through the intermediate fine-tuning method by using imperfect synthetic speech as described in Fig. 1.3. Despite its unintuitive approach, the proposed method shows huge improvements over the baseline method. We also analyze what exactly makes the method successful and discover that the network was able to learn the EL speaker features despite being trained on a recognition loss objective.

In Chapter 4, we focus on the speech enhancement task and improve on previous works by utilizing the findings from Chapter 3. In particular, we use the speech recognition encoder to extract strong linguistic features from EL speech audio and use these to replace traditional acoustic features. An overview of the conventional enhancement framework, and how we integrate the previous proposed speech recognition model into this proposed speech enhancement model is shown in Fig. 1.4. We propose a decomposed framework, which uses a recognition, alignment, and synthesis framework, effectively improving performance due to the removal of the acoustic mismatches between the pretraining (typical) and fine-tuning (EL) data. Then, we then further improve

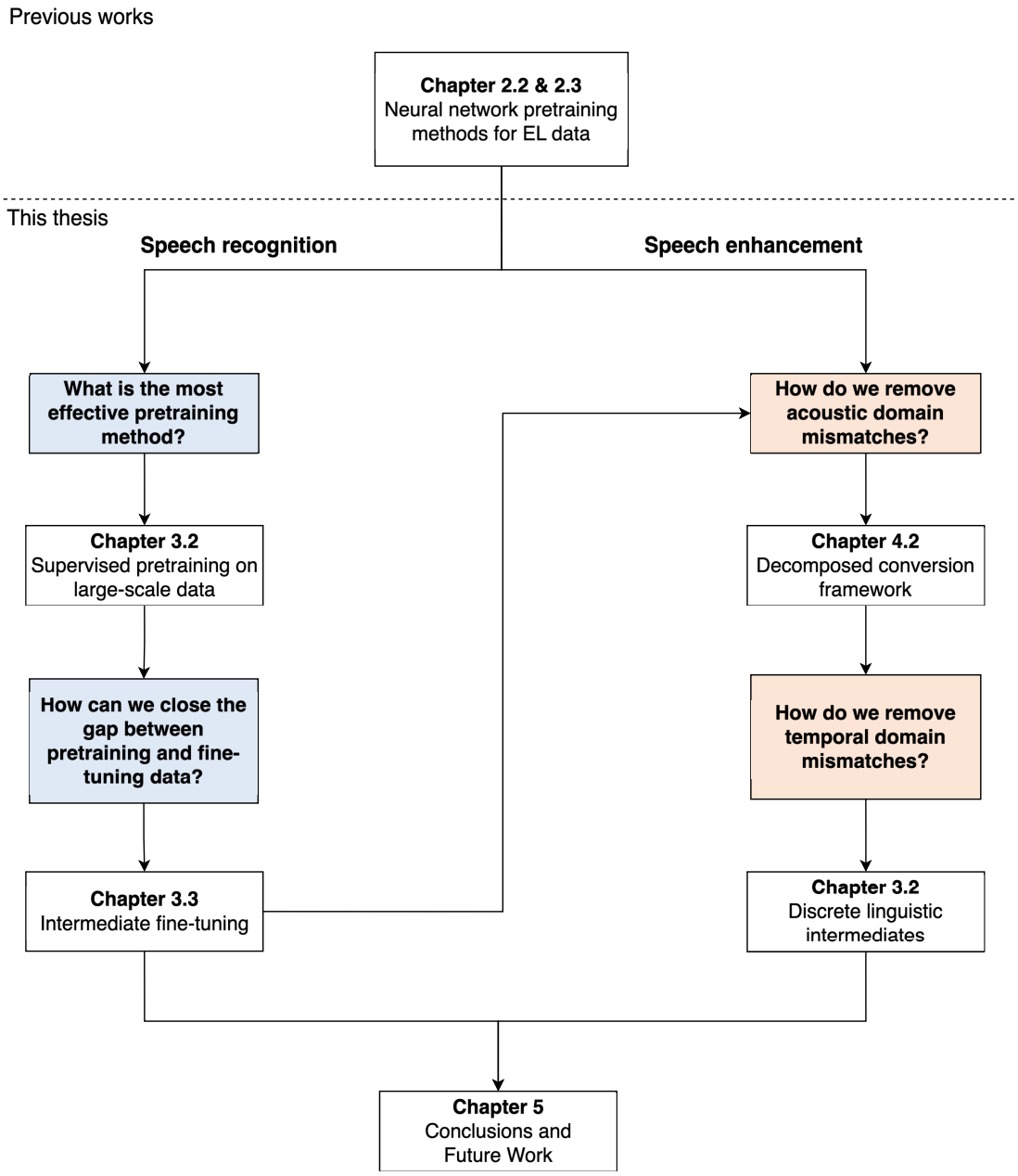


Figure 1.2: A visualization of the contributions of this thesis. We start by analyzing the problems in the speech recognition task. The thesis then tackles the speech enhancement task while also using the learnings from the previous speech recognition task to resolve the discovered issues.

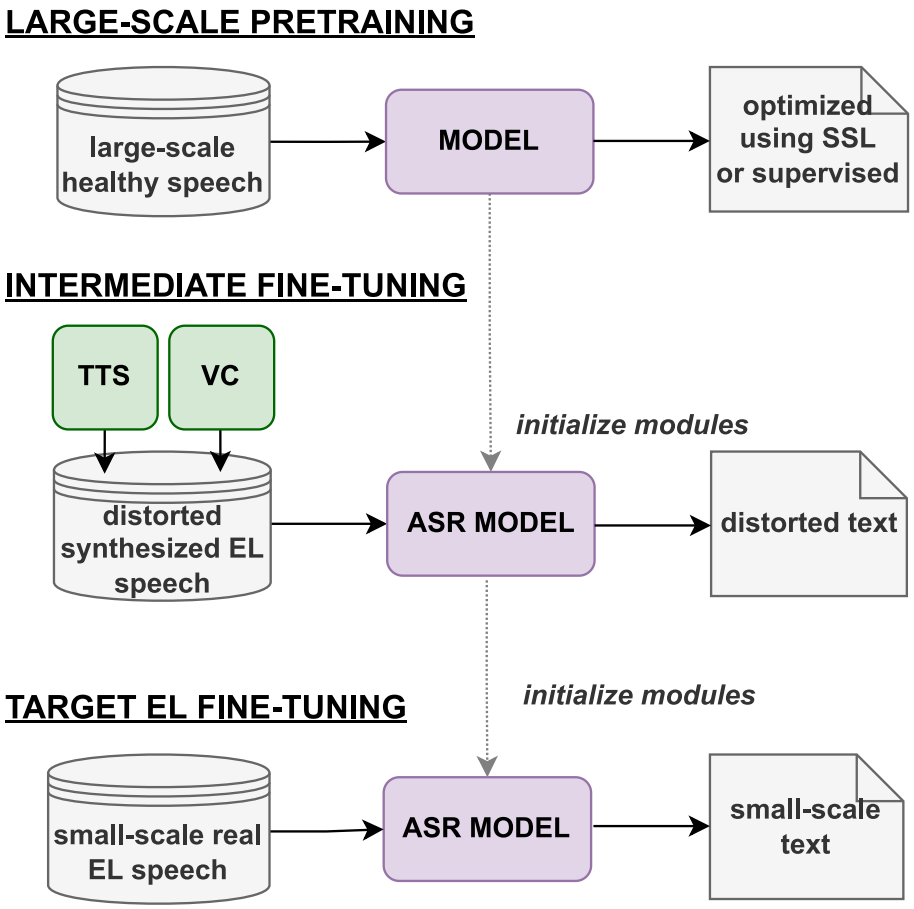


Figure 1.3: A visualization of how the model weights are initialized and trained in different stages. The stages are composed of pretraining, intermediate fine-tuning, and target fine-tuning; where each stage conducts the same speech recognition task but with different input and target data.

this by resolving the temporal mismatches between the source (EL) and target (typical) data, by introducing discrete text intermediates, which effectively alleviate to improve prosody modeling. Our findings show that by simply replacing acoustic features with linguistic features from the speech recognition system, more intelligible and naturally sounding speech can already be synthesized compared to the baseline, and using discrete text intermediates to remove unnecessary frames further improves the results. Furthermore, we also validate these results on another set of pseudo and real-world EL speakers, showing its robustness to different types of EL speakers.

We conclude the thesis in Chapter 5 by summarizing the results and lay the foundations of potential future work.

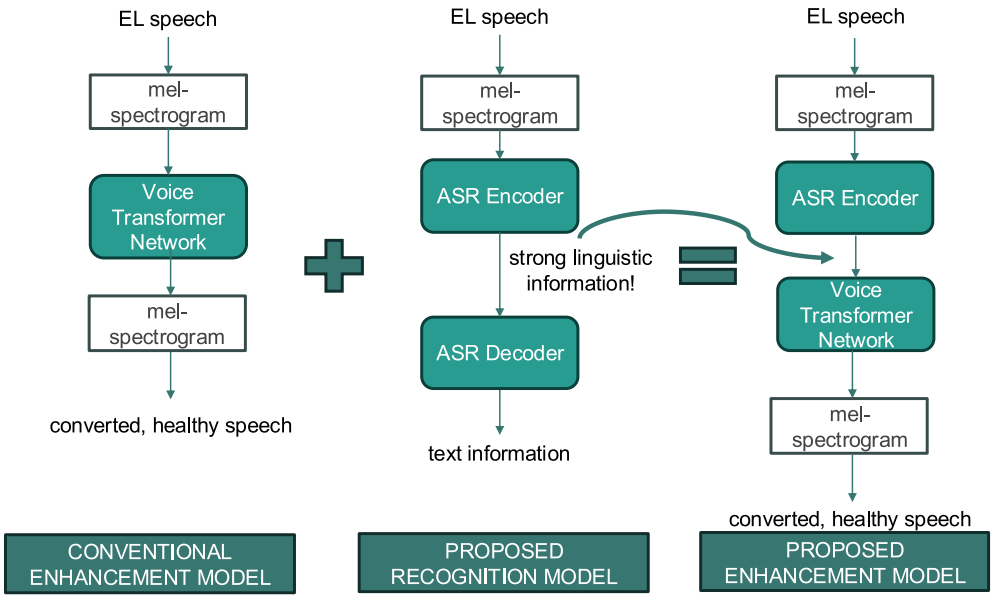


Figure 1.4: A visualization of conventional speech enhancement models and the proposed speech recognition model in Chapter 3. The proposed speech enhancement model is achieved by integrating the speech recognition model into conventional works, subsequently improving the model by introducing strong linguistic information into the architecture.

Chapter 2

Related Work

In this chapter, we provide a background to understand the context of the thesis' goals. In Chapter 2.1, we first describe laryngectomy and how it affects the human speech production process, and the different solutions to allow laryngectomees to regain speaking ability. Then, we give a brief background on essential deep learning concepts and then discuss previous works and approaches for both electrolaryngeal speech recognition in Chapter 2.3 and enhancement in Chapter 2.4.

2.1 Background on Larygneal Surgery

2.1.1 Human speech production process

The larynx is the organ responsible for generating the source excitation signals of human speech. In particular, the larynx is mainly responsible for the voiced and unvoiced features of speech (or sometimes referred to as phonetics in the linguistics field). The voiced and voiced refers to the oscillatory state of any part of the larynx that modifies the airstream [6]. By expelling air from the lungs to the glottis, a sufficient pressure drop created can cause the vocal folds to oscillate. This oscillation made by

the vocal folds is responsible for modulating the air pressure and flow throughout the larynx. In effect, this airflow by the larynx is responsible for creating most of the voiced phonemes. On the other hand, without a sufficient pressure drop, the vocal folds do not oscillate, in which case the produced phoneme is referred to as an unvoiced phoneme [7].

Referring to the source filter theory [8] and visualized in Fig. 2.1, this signal that the larynx produces is a harmonic series, which consists of the F0 accompanied by harmonic overtones, and are multiples of the F0. This produced signal is then amplified and attenuated via a filter with a continuous impulse response, peaking at specific resonances. This filter response represents the transfer function of the vocal tract resonant properties. Combining these two stages results in the individual speech sounds that are used in everyday conversation.

Similar to any other muscle in the body, humans can train this muscle and effectively control the harmonic series to produce and freely alter speech. By contracting, loosening, and using the tongue, variation in F0 can be done and is used linguistically to produce intonation and tone. Thus, without the larynx producing source excitation signals, the resulting speech becomes monotonous and does not have any variation in pitch or prosody. Unfortunately, some patients undergo laryngectomy, the total removal of the larynx, which is commonly caused by laryngeal cancer. Due to the removal of the larynx, laryngectomees have to resort to different speaking methods.

2.1.2 Alternative speaking methods

With the total removal of the larynx, laryngectomees mainly resort to three different kinds of alternative speaking methods, namely: esophageal, tracheoesophageal, or through an external electrolarynx. In the esophageal method, the laryngectomee in-

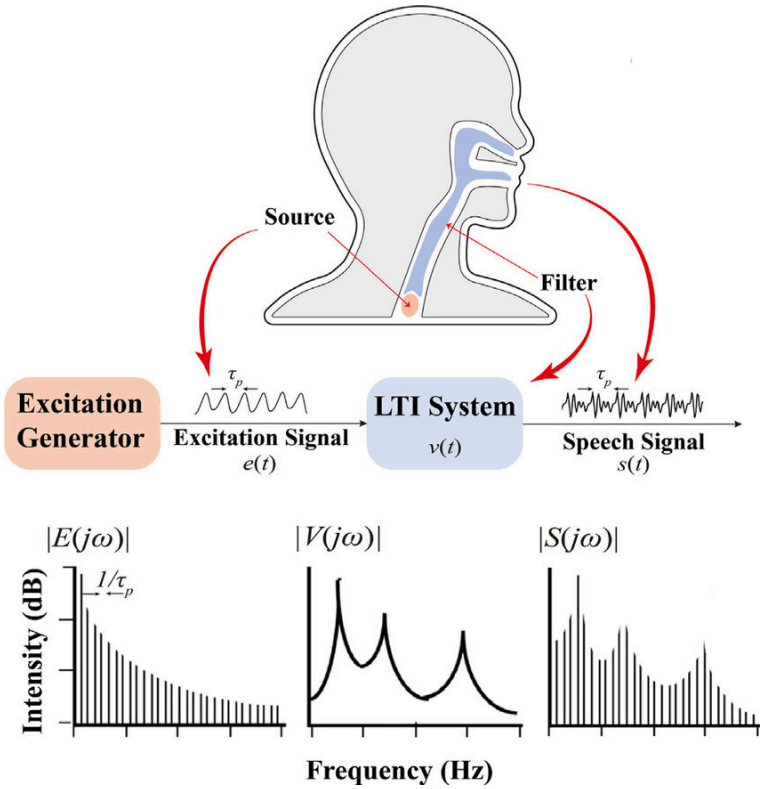


Figure 2.1: A visualization of the source filter theory [8], [9] for human speech production. An excitation signal $e(t)$ is passed to a linear time-invariant (LTI) system $v(t)$ to form the speech signal $s(t)$. Figure from [9].

hales air and uses this to oscillate the esophagus as a replacement to the oscillation of the vocal folds. Although a natural alternative, this speaking method requires skill and strength to sound intelligible. The tracheoesophageal method is similar to esophageal, but uses air from the lungs using a prosthetic valve inserted from the trachea into the

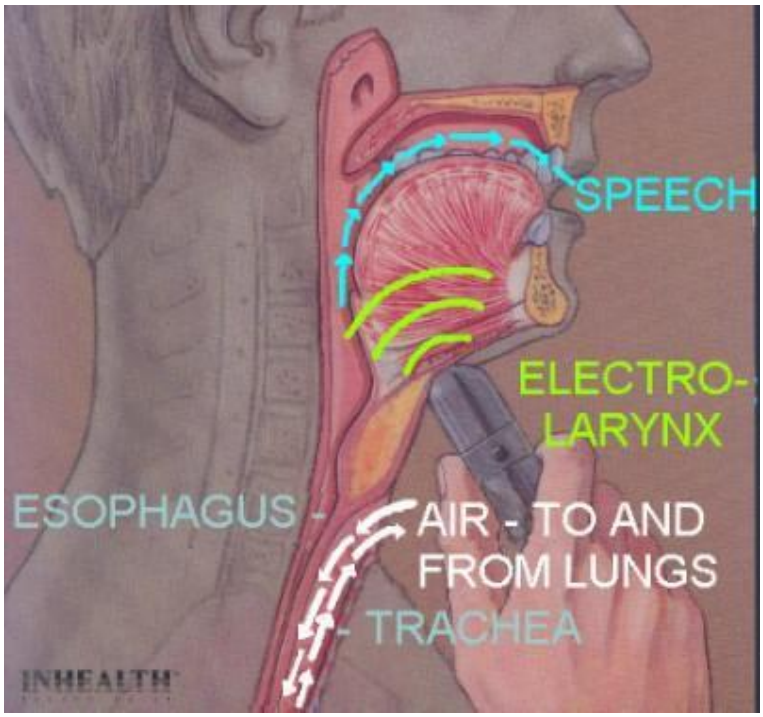


Figure 2.2: A visualization of how an electrolarynx is used. Figure from [10].

esophagus. However, although it becomes easier to generate the air flow, the prosthetic valve needs to be maintained regularly and is harder to be used by elderly people.

Different from these two methods, the electrolarynx is an external device that is pushed onto the lower jaw to create the source excitation signals with a switch. Compared to the previous methods, using this device produces the most intelligible speech, resulting in its popularity. A visualization in Fig. 2.2 shows an example of how to use this device. Although the most viable alternative for speech communication out of the three, there is still a huge gap in intelligibility between the electrolarynx method and natural speech, requiring the need to further provide laryngectomees with alternatives that allow smooth communication.

2.2 Deep Learning Concepts

To understand how to resolve the problems in this thesis, we first discuss some essential deep learning architectures that are commonly used in the field of data-driven deep learning.

2.2.1 Attention-based modeling

The Transformer [11] is one of the most powerful architecture in the deep learning space, being versatile with multiple types of data and success in various tasks. The primary feature of this network is self-attention, which is done through the multi-head attention (MHA) sublayer [11]. To compute MHA, input matrices *query*, *key*, and *value* are processed to learn h different linear projections and perform the attention operation in parallel. This results in different learnable linear projections, which are concatenated together and projected with another linear projection.

Although the effectiveness of the Transformer in the original paper was demonstrated on a text translation task, the architecture has been versatile and can be adapted to process continuous features instead of the original discrete text. This makes it possible to use for both the speech recognition and speech enhancement tasks. In the case of the speech recognition task, the input embeddings are simply removed to be able to directly handle continuous acoustic features such as mel-spectrograms. Specifically, the Conformer network [12], a variant of the Transformer [11] was proposed, and has proven its ability to exploit local features and model long-range global context, resulting in ASR performance with high recognition accuracies [13]. What makes the Conformer network different from the vanilla Transformer network is that the outputs of the MHA block are processed by a convolution module that consists of a pointwise convolution

and a gated linear unit, followed by a 1D convolution layer and batch normalization. The MHA and convolution modules are then sandwiched by two feed-forward modules. This sandwich structure allows the network to capture the long-range context through the MHA and feed-forward modules originally proposed in the Transformer network [11] and capture local contexts in the speech through the proposed convolution module.

To optimize this network for the speech recognition task, a hybrid CTC-Attention [14], [15] multi-task loss is used to exploit the strengths of self-attention while retaining the monotonic alignment of the speech and text through the connectionist temporal classification (CTC) [16] loss. The CTC-Attention loss is composed of the logarithmic linear combination of the attention and CTC loss. The attention module [17] predicts each character using the previous predictions as conditioning features and finds the essential parts of the sequence by assigning weights to the hidden vectors. On the other hand, the CTC loss [18], which has been used in early ASR methods, models the alignments between the encoder outputs and the text label sequence by using an additional blank symbol. Although attention loss has been generally better in many tasks due to its data-driven approach, CTC helps the network in forcing a monotonic alignment between the input and outputs, which covers the weaknesses of attention, especially in longer sequences. Unlike other tasks that use the attention module, the alignment in ASR is always monotonic, thus further improving the performance of attention in ASR.

On the other hand, to optimize this network for speech enhancement, the softmax function in the decoder can also be simply removed to predict the probabilities of the log mel-spectrogram [19]. As this process is typically divided into a two-stage task, the predicted log mel-spectrogram is consequently passed into a vocoder to generate the corresponding waveform. To optimize the acoustic network, a simple L1 or L2 loss is

used by comparing the predicted log mel-spectrogram with the ground truth log mel-spectrogram. On the other hand, the conventional neural vocoder is optimized using a GAN-based training framework [20], which uses a discriminator on the waveform dimension.

The use of self-attention in Transformers also gave way to a new method gaining popularity called self-supervised learning (SSL). The general idea of this method is to make the model learn a universal representation of the data using joint embedding objective functions [21], [22] to compare embedding representations of the data. Inspired by methods in natural language processing, SSL-based pretraining does not require any target text labels to be trained unlike conventional tasks, and are implemented either through a contrastive loss or a masked-region prediction loss [23]. Thanks to this training objective, larger datasets of up to 60k hours were now available to be utilized in large-scale pretraining, allowing a neural network to learn stronger speech representations with a larger number of data available. For speech in particular, several frameworks have been proposed [23] such as generative modeling [24], discriminative/-contrastive modeling [25]–[27], and multitask learning [28]. SSLs are quite versatile as it is an upstream task, and can thus be further used in different downstream tasks such as speech recognition or speech enhancement.

There are two self-supervised pretraining frameworks: contrastive loss [25] and masked region prediction [27]. The main differences between the two frameworks can be visualized in Figure 2.3. The first framework is wav2vec 2.0 [25], one of the most widely used SSL frameworks in speech data based on contrastive loss. This framework uses a multilayer convolutional neural network (CNN) to encode raw speech audio X into a latent space $z_{1:j}$ for j timesteps (Eq. 2.1). The latent space representations $z_{1:j}$ are randomly masked before being passed onto the MHA layers of the Transformer

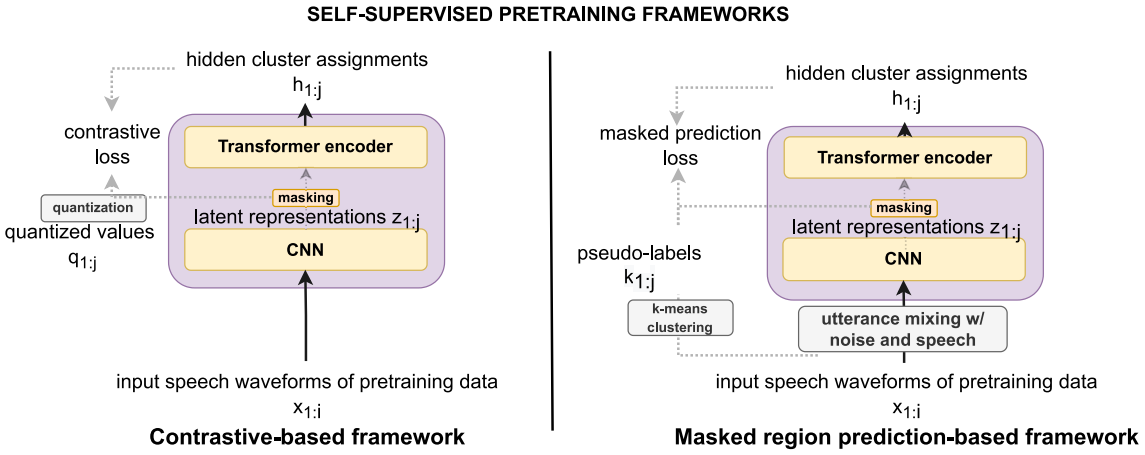


Figure 2.3: Overview of SSL-based frameworks. The SSL models wav2vec 2.0 [25] (left) and WavLM [27] (right) are trained by a contrastive or masked region prediction framework.

encoder backbone (Eq. 2.2) to produce the context representation vector $h_{1:j}$. Also, to represent the targets of the self-supervised task, the latent space representations $z_{1:j}$ are quantized as $q_{1:j}$ (Eq. 2.3) using product quantization. The model is optimized using a contrastive loss function (Eq. 2.4) by identifying a q' from $q_{1:j}$ among a set Q_t of κ-number distractors given the masked context representation vector $h_{1:j}$ from the Transformer encoder through a cosine similarity function (sim), allowing the codebook to learn representations of positive and negative samples. Moreover, the loss is calculated along with a diversity loss function (Eq. 2.5) to increase the use of the quantized representations. Here, V entries in each of the codebooks G are forced to be used equally by maximizing entropy I for each codebook $\bar{p}_{g,v}$ across utterances of a certain batch size. The final loss can be computed as in Eq. 2.5, with α being a tunable hyperparameter.

$$z_{1:j} = \mathrm{CNN}(x_{1:i}) \tag{2.1}$$

$$h_{1:j} = \text{Transformer}(\text{masking}(z_{1:j})) \tag{2.2}$$

$$q_{1:j} = \text{quantization}(z_{1:j}) \tag{2.3}$$

$$\mathcal{L}_c = -\log \frac{\exp(\text{sim}(c_j, q_j)/\kappa)}{\sum_{q' \sim \mathbf{Q}_j} \exp(\text{sim}(c_j, q')/\kappa)} \tag{2.4}$$

$$\mathcal{L}_d = \frac{1}{GV} \sum_{g=1}^{G} \sum_{v=1}^{V} \bar{p}_{g,v} \log \bar{p}_{g,v} \tag{2.5}$$

$$\mathcal{L}_{contrastive} = \mathcal{L}_c + \alpha \mathcal{L}_d \tag{2.6}$$

Aside from wav2vec2.0, there is another SSL framework called WavLM [27] that instead uses masked region prediction to learn speech representations. WavLM does this through a denoising and prediction framework, mixing utterances by adding interfering speech and noise to the input speech, and making the model predict the original clean speech. Its architecture is similar to that of wav2vec 2.0, where the noisy input speech is passed onto a set of CNN layers to make $z_{1:j}$ and randomly masked before passing it onto the Transformer network, which then predicts the pseudo-labels of the masked regions $h_{1:j}$. The target pseudo-labels $k_{1:j}$ are generated by performing two iterations of k-means clustering: one on the MFCC of the original clean speech $x_{1:i}$ (Eq. 2.7) and another on the latent representations $z_{1:j}$ (Eq. 2.8). Moreover, a gated relative position bias is added to the Transformer backbone to better capture the sequence ordering of the noisy input speech. Training the model through this denoising framework proves to make the model robust to acoustic variations in the speech. The masked region loss

objective is defined as in Eq. 2.9, where K is the number of clusters, M is the set of masked indices in the time-domain, and L is the l-th layer output of the Transformer.

$$k_{1:j} = \text{k-means}(\text{MFCC}(x_{1:i})) \tag{2.7}$$

$$k_{1:j} = \text{k-means}(z_{1:j}) \tag{2.8}$$

$$\mathcal{L}_{maskedregion} = -\sum_{l \in K} \sum_{t \in M} \log p(z_{1:j} \mid h^{L}_{1:j}) \tag{2.9}$$

2.2.2 Diffusion modeling

Another quite popular generative modeling method is through diffusion modeling [29] which has also gained popularity with speech generation tasks. Although adaptable to predict discrete text and language modeling tasks, research involving diffusion modeling is more developed when predicting continuous features such as log mel-spectrograms, so we only focus on using this for the speech enhancement task. The diffusion process is divided into two steps: the forward noising which is used as training data, and the backward denoising which is used to generate new samples with conditioning features. The forward process gradually adds Gaussian noise over time t in the data sample x (in the case of this thesis, the mel-spectrogram). The forward process can be parameterized as $z_t = \alpha_t x + \sigma_t \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, allowing the generation of any noised data sample at time t and making the training process straightforward to implement. A noise scheduler is also defined by the parameters α_t and σ_t. To use this framework, a denoiser model is trained $v(z_t; t)$ is used to predict the the denoised sample $\mathbf{z}_{t+1}$ at time step $t+1$ from a noisy sample $\mathbf{z}_t$ at time step t. Then, during inference, the

forward process can be reversed by using the trained denoiser network to iteratively remove noise from a Gaussian noise sample initialized from $\mathbf{z}_1$.

On the other hand, conditional flow-matching with optimal transport (OT-CFM) objective proposed in [30]. Although the formulation is different, diffusion and flow matching are two sides of the same coin [31]. The transformation process, which leverages optimal transport to construct a time-dependent vector field $u_t(x|x_1)$, for $t \in [0, 1]$ using a neural network $v_t(x; \theta)$ with parameters θ, goes from a simple initial Gaussian distribution x_0 into a complex target distribution x_1 (in this case, the mel-spectrograms). Once trained, the network defines a flow ϕ_t that transforms the initial distribution x_0 into x_1. Using the OT-CFM objective [30] of $\phi_t^{\text{OT}}(x_0) = (1-(1-\sigma_{\min})t)x_0+tx_1$, we can use the time derivative of this path as a target, $u_t^{\text{OT}}(\phi_t^{\text{OT}}(x_0)|x_1) = x_1 - (1 - \sigma_{\min})x_0$, to optimize the neural network parameters, where $\sigma_{\min}$ is a small positive constant for stability. Several architectures have also adopted the use of Transformers to predict the denoised features to take advantage of the use of self-attention architectures. Moreover, the network can be easily adapted to use either text or audio (through the use of audio SSLs) as conditioning features. Compared to the L1 or L2 losses used in other works, the use of the diffusion losses to predict the log mel-spectrogram have proven reduced oversmoothing and enhanced synthesis quality. These advantages make it perfect for the speech enhancement task.

2.3 Electrolaryngeal Speech Recognition

As mentioned in Chapter 1.2, speech recognition is a popular method with the rise of artificial intelligence devices. In this chapter, we describe the previous work that has been done to improve speech recognition for different atypical speakers.

2.3.1 Comparison of recognition performance between human and ASR

Although it is straightforward to understand that other humans have difficulty in understanding EL speech, it would also be interesting to see an analysis of how different a human and a neural network understands EL speech. In particular, the research in [32] focused on this problem and found that the recognition of isolated words was more difficult due to the fact that all phonemes become voiced sounds. Three experiments were conducted. First, blind recognition, which means recognizing an isolated word from a dictionary of a million words. Then, a second experiment which provides more prior information by limiting the vocabulary to those of the training set was conducted. Finally, a third experiment where the listener was asked to distinguish two similar words with only a difference in a voiced phoneme. Results showed that recognition of EL speech is hard even for humans, especially for the isolated word case. However, surprisingly, neural network-based ASR systems were no better than humans as well. This was especially shown in the first experiment where there is no prior context given, and both humans and machines had difficulty in recognizing isolated words. However, with enough prior context like in the second experiment, both humans and machines gain dramatic improvements in performance. The most interesting finding is that although it was initially assumed that EL speakers can only produce voiced phonemes, the findings show that this is not always the case, as the context of the phoneme plays an important role in voicing. In many cases, the human and the machine distinguished voiced and unvoiced phonemes in the EL speech, showing that the context of the surrounding words can greatly help recognition.

2.3.2 Pretraining techniques for low-resourced pathological ASR

Research in resolving low-resourced ASR for pathological speech has primarily revolved around large-scale pretraining on healthy speech datasets in a supervised manner. For example, [33] focused on the problem of limited data and used two types of non-standard speech: speech from people with amyotrophic lateral sclerosis (ALS) and accented speech. The results showed that personalized models achieved relative WER reductions of 62% and 35% in the two groups, reducing the absolute WER for ALS speakers to 10% for mild dysarthria and 20% for more severe dysarthria. These works use a non-attention encoder and decoder network such as the Recurrent Neural Network Transducer (RNN-T) [34], and an attention-based, sequence-to-sequence model such as Listen, Attend, and Spell (LAS) [17] network. By pretraining RNN-T and LAS networks on the 960-hour Librispeech dataset, majority of the improvements comes despite only having 5 minutes of training data. Interestingly, the study also finds that finetuning just a particular subset of layers often gives better results than finetuning the entire model. This work was one of the first to show the power of pretraining in resolving the limited training data problem.

Extending this work in [35] saw further improvements by pretraining on a larger dataset with 162k hours. A dataset was collected, which composed of 432 individuals with self-reported disordered speech. As a main result, the performance of the trained models were equalling human-level performance in recognition accuracies at 15% character error rate (CER), showing the effectiveness of large-scale pretraining on healthy speech. As the metadata was also collected from the different participants, the study also further analyzed where the model was strong and what its limitations were. In particular, although the results for each severity group was analyzed, and WERs were

greater than 10% for approximately 26% of the participants. For the typical to mild severity group, although some speakers had high WERs, recognition accuracies were only degraded primarily by limitations in the dataset quality and size rather than having atypical speech features. On the other hand, for speakers with moderate to severe severity in this group, performance degradation was due to both the atypical speech features and low dataset quality and size. Thus, this research also shows the difficulty of collecting data, as careful preprocessing steps still need to be conducted to make these usable for machine learning training.

However, in both of these aforementioned works, one difficulty in reproducing this issue is gaining access to a similarly sized large pretraining dataset with corresponding text labels. To train a speech recognition model, both the speech and text pair need to be prepared for training. Although it is quite easy to collect a large-scale speech dataset, accurately labeling the transcriptions becomes difficult and expensive. Moreover, without any quality assurance, wrong transcriptions can negatively influence the training of the model. Although it has been proven and widely accepted that large-scale pretraining is the key to robust ASR [36], the labelled datasets used in these works have not been open-sourced for public use. As of the time this thesis was conducted, the largest labelled speech dataset is Librispeech at only 960 hours, which is significantly less than the 162k hours used in the aforementioned experiments.

On the other hand, improvements in large-scale pretraining for ASR were proposed in the past few years through SSL. As previously mentioned, SSL-based pretraining did not require any target text labels to be trained unlike conventional tasks [23]. This means that a strong linguistic representation can be learned by simply learning how to reconstruct the input speech and learning how small segmented frames correlate to the other frames of the speech. Thanks to this training objective, larger datasets of up to

60k hours were now available to be utilized in large-scale pretraining, allowing a neural network to learn stronger speech representations with a larger number of data available. Several works have also shown its effectiveness by gaining state-of-the-art performance by fine-tuning on as small as 10 minutes of data [25], [26], [37]. Thus, using these SSL features over traditional acoustic features such as mel-spectrograms have the potential to resolve limited training data. Initial studies in different minimally resourced speech recognition tasks [38], [39] have proven to be successfully, making these a potential investigation point for electrolaryngeal speech recognition.

2.3.3 Data augmentation for low-resourced pathological ASR

Aside from large-scale pretraining, data augmentation is also a popular approach in limited data cases. The idea is to use a speech synthesis system to synthetically generate a larger ASR training dataset to improve pathological ASR performance. Several early methods have been developed, where research such as [40] use voice conversion powered by generative adversarial networks to convert healthy speech into atypical speech. Another approach [41] also uses parallel VC to generate new atypical speech utterances, and focus on expanding the vocabulary of the speech utterances in the training set. Other methods such as [42] use a text-to-speech method for a more controllable synthesis method.

Several other works have also focused on improving generating atypical speech not only for data augmentation, but with the goal of accurately modeling the atypical speech features for medical purposes. By accurately modeling the atypical speech features, this can also consequently be used for data augmentation purposes. For example, [43] first focused on this by adopting a two-stage framework, which consists of a sequence-to-sequence model and a nonparallel frame-wise model. In particular,

the healthy speech is first converted into atypical speech of another speaker in the desired severity using the sequence-to-sequence model, and is consequently transformed into the target speaker using the nonparallel frame-wise model. By doing this, the framework can model both the atypical speech features and the target speaker using individual models. Results show that this framework was able to mimic the severity characteristics in a linear way, which was evaluated by according to three speech language pathologists.

The study in [44] further improves on this two-stage framework, particularly in the second stage by using phonetic posteriorgrams and global style tokens, along with a new parallel dataset dedicated for high-quality evaluation of this task. The research ended with three main conclusions. First, the proposed method shows better preservation of the atypical source features compared to the baseline owing to the use of phonetic posteriorgrams and global style tokens. However, there is still a huge gap of the synthesized samples compared to ground truth recordings. Second, the speaker identity during conversion is impacted by severity but not all speaker information is lost, meaning that features from the pre-operative voice can still be used as a reference utterance. Finally, choosing the appropriate source speaker based on just the severity labels was not sufficient, once again showing the need for data collection and improving the metadata of these datasets.

In all these aforementioned works, the focus has mainly been on developing a high-quality speech synthesis system to generate speech that can accurately represent both the pathological speech features and its linguistic information. However, a weakness of this method is that large datasets are still required to train a high-quality speech synthesis model that represents both the acoustic and linguistic information. Thus, there is a chicken-and-egg problem present as data augmentation cannot be presented

as a solution for limited training data, without also having a large-scale dataset to train the speech synthesis system, resulting in speech that may not truly represent the target atypical speech features of the original speaker.

2.4 Electrolaryngeal Speech Enhancement

Similar to speech recognition, speech synthesis and voice conversion are also a popular method. In the context of atypical speakers, we refer to using voice conversion as speech enhancement as it transforms and enhances atypical speech into speech that is much easier to understand. In this chapter, we describe the previous work that has been done to improve speech enhancement for different atypical speakers.

2.4.1 Sequence modeling techniques in voice conversion

A traditional approach in voice conversion is called parallel voice conversion (Parallel VC), where a dataset of two speakers uttering the same sentences is used to convert the speaker information from the source to the target and also altering the temporal structure. This is different from the now commonly used recognition-synthesis approach [45], where the source and target features have the same length, and only timbre-related information is converted during the process. As mentioned, EL speech requires the conversion of the temporal structure, which requires the focus on sequence modeling with parallel VC.

Statistical-based methods such as the Gaussian Mixture Model [46] and neural-based methods [19], [47]–[49] have been used to transform both the temporal and timbre features of the source speaker into the target speaker. These statistical methods typically use dynamic time warping to model the temporal information. Neural-based methods

have also the option of modeling the target speech either in an autoregressive [19], [47] or non-autoregressive fashion [48], [49]. Due to the parallel dataset, the model effectively learns how to model the target speaker's prosodic features by referencing the source speaker's features. In particular, the autoregressive network uses attention [11] to reference which frames to remove or use when modeling the current frame. On the other hand, the non-autoregressive network uses a duration predictor [48] to determine which input frames should be deleted or elongated when modeling the entire target speech.

On the other hand, non-parallel approaches have also been proposed to alleviate the difficulty in obtaining parallel datasets. For example, phoneme-level representations could be extracted through a cascaded automatic speech recognition and text-to-speech (ASR+TTS) [50], [51] framework to generate the target waveform (or intermediate acoustic features like mel-spectrograms). This removes the necessity of a parallel dataset, as the two systems can be trained individually by using text as a target in the case of ASR, or input in the case of TTS. Although this removes the difficulty of collecting parallel datasets, such an approach has been considered less flexible in modeling the temporal information, as it completely removes the source prosody, which in many cases is still useful prior information. For example, if the speaker likes to elongate a certain phoneme, this prosody can be transferred over in the parallel VC setup. In contrast, in ASR+TTS, there would be no way to transfer prosody information due to the cascaded framework. Moreover, in languages such as Japanese and Chinese where there are multiple pronunciation for a single character, the task can become more unstable if the reading or pronunciation of the character is solely inferred from just the text information. Thus, using these approaches while convenient and effective, removes several advantages that were present in using Parallel VC.

2.4.2 Conventional EL speech enhancement approaches

EL speech enhancement has traditionally been conducted using rule-based algorithms or statistical voice conversion. For example, subtractive-type algorithms [52] were used to reduce the radiated noise by taking into account the frequency-domain masking properties of the human auditory system. The research proposes taking into account the auditory masking properties in the enhancement process. Here, the perceptual weighting technique is used such that the subtraction parameters are adapted. Compared to the baseline using power spectral subtraction, the proposed algorithm efficiently reduces the background noise.

Statistical voice conversion using Gaussian mixture models [53]–[56] were also explored to map the spectral features of an EL speaker into the spectral and source excitation features of a typical speaker. In this line of work, the Gaussian mixture model in [46] is used as the conversion framework while improving it for the EL speech enhancement use case. In particular, [53] improves this by using spectral information to estimate both the spectra and the F0 counters. On the other hand, [54] further improves this by estimating spectral parameters, F0 and aperiodic components independently. Such approaches were validated to improve the quality of noisy EL speech into easily understandable speech for humans.

More recently, with the rise of neural networks, conventional work on neural network-based atypical speech enhancement builds on the aforementioned Parallel VC framework [57], [58]. In particular, works such as [59] by using the Transformer [11], [60] to transform the mel-spectrogram of an EL speech utterance into a mel-spectrogram of a typical speech utterance. Due to the data-hungry nature of Transformers, a two-stage pretraining technique using text-to-speech (TTS) and autoencoder (AE) was used to efficiently learn linguistic information from a large-scale typical speech data. The pro-

cess is done by first training a TTS model with a large-scale dataset. Then, an AE-style pretraining is conducted by using the decoder of the TTS model as initialization parameters, and reconstructing the target speaker by also using it as inputs. Here, the decoder parameters are frozen, such that the encoder is efficiently pretrained. Moreover, since the TTS pretraining technique uses text information as inputs and models strong linguistic information [60], such a speaker-independent pretraining style was beneficial in reducing the speech type and speaker mismatches when fine-tuning. By using such a pretraining technique, the model can be fine-tuned on a parallel dataset despite only having limited data.

The work in [59] further improves this by using synthetic data of both EL and healthy speakers. Due to the huge domain mismatch between typical and EL speech, the network is first fine-tuned on a parallel synthetic EL and typical speech before fine-tuning on the ground truth recordings. Interestingly, it was found that fine-tuning first on synthetic EL speech (even with lots of mispronunciations in synthesis) softens the speech type and speaker mismatches when fine-tuning the network. Further improvements on this work were conducted in [61], where the encoder module was also adapted, effectively training it to extract strong representations from the input EL speech before fine-tuning on the ground truth dataset. Thus, the use of synthetic speech data augmentation and model pretraining was found to be an effective solution in the atypical speech enhancement task.

Other works such as [62] focused on converting atypical speech into healthy speech by using discrete speech units. Inspired by SSL techniques, the research uses HuBERT [63] to extract strong linguistic features and quantize these into linguistic units through k-means clustering. Similar valued tokens that are consecutive to each other are normalized into a single unit, effectively removing unnecessary frames from the input

speech. This results in a speech synthesis-like task, where instead of input text tokens, the discrete speech units are used to generate the mel-spectrogram and waveform of the healthy speaker. The research indicates the proposed method outperforming data augmentation-based baselines, reaching a 28.2% relative average word error rate reduction when compared to original dysarthric speech, and shows robustness against speed perturbation and noise. Thus, this research shows the importance of using discrete speech information and removing unnecessary frames to be vital in conducting the EL speech enhancement task.

2.5 Conclusions

In this chapter, we discussed the related work of the thesis, covering topics such as the human speech production process, preliminary deep learning concepts, and previous approaches for both electrolaryngeal speech recognition and enhancement. We first elaborated on how humans produce speech, speaking alternatives for laryngectomees, and why electrolaryngeal speakers have difficulties in producing understandable speech due to the lack of the source excitation features. Next, we discussed preliminary deep learning concepts such as the attention-based Transformer and diffusion modeling, both of which are essential architectures to be used in this thesis. Then, we discussed previous work on electrolaryngeal speech recognition, which either resolves issues using either pretraining or data augmentation to handle small-scale training datasets. Finally, we discussed previous work on electrolaryngeal speech enhancement, which typically relies on the use of data augmentation methods to handle domain mismatches.

Chapter 3

Electrolaryngeal Speech Recognition

In this chapter, we investigate the approaches to improve electrolaryngeal speech recognition. With the lack of training data available, this study aims to combine the success of pretraining methods with data augmentation techniques and improve performance of speech recognition systems.

3.1 Introduction

Although ASR has been gaining a lot of popularity, research has been only progressed on speakers with healthy speech due to the lack of variety in open-sourced training sets, and minority speakers have not found success in using ASR-powered devices. For example, disabled patients, who would possibly benefit the most from these devices, are hindered from using these devices owing to their loss of control of their speech organs and thus suffer from atypical speech [64]. Thus, aside from daily communication, several benefits become unavailable to atypical speakers, as their produced speech can also be misunderstood by most commercial ASR devices [65].

Although several ASR devices are based on a neural network, the size and architecture of the neural network can vary depending on the target speakers and/or target recognition accuracy. Thus, as one may expect, using a simple network vs. a sophisticated network contains several trade-offs. Using simple, previously proposed architectures such as RNN or LSTM-based models [17], [66] is relatively easy to train and can fit small datasets (on the order of one hour of speech); however, the maximum achievable recognition performance is limited by their simplicity and limited capacity to model fine-grained details in the input. On the other hand, using more sophisticated and recent networks, such as a Transformer [11] or any of its variants that utilize the self-attention module [12], have the potential to achieve high recognition accuracies due to their ability to capture the fine and detailed features along with the long-range contexts of the input speech; however, these need to be trained on a large-scale dataset (around a hundred hours at minimum) in order to generalize well and are harder to train due to the larger number of hyperparameters needed to be tuned.

There are several approaches to resolving the low-resourced problem when training ASR systems; however, we focus on the two most common techniques: large-scale pretraining and data augmentation. ASR systems try to elucidate the posterior distribution $P(Y \mid X)$ given the input speech X and the target text label Y. Without a large dataset, the aforementioned posterior distribution cannot be approximated when using large and data-hungry networks. Thus, large-scale pretraining is done by making use of a large-scale healthy speech dataset to first pretrain the neural network, then subsequently fine-tune on the small-scale atypical speech dataset by initializing the weights from the pretrained network. This weight initialization can be analogous to using the posterior distribution of the large-scale healthy dataset as a prior during training [67], as it already has a clear idea of how to map speech into linguistic content, making

the model only need to adjust slightly to adapt to the speech features of the target dataset. Thus, compared to initializing weights from random values, the network can generalize to the small-scale dataset faster, making large and data-hungry networks viable for use with small-scale datasets. However, although easy to implement, large-scale pretraining is still too naive as a method, as the maximum recognition accuracy will still be limited due to a large domain shift gap in speech features existing between the large-scale healthy dataset and the small-scale atypical dataset. Thus, further investigations in resolving the domain shift gap between the pretraining and target data are still needed to match recognition performance with healthy speakers.

Another approach previously explored is using data augmentations to create a pseudo-large-scale atypical dataset through speech synthesis techniques such as text-to-speech and voice conversion. Like pretraining, the idea is to clearly approximate the posterior distribution $P(Y \mid X)$ by artificially generating speech that matches the probability distribution of the small-scale dataset. Similar to ASR, speech synthesis has also progressed at an incredible rate thanks to the advances in neural networks and has synthesized human-like quality speech, but unfortunately also faces the same problem of requiring large-scale datasets to train these networks. If the data augmentations do not accurately represent the data distribution of the small-scale dataset, the ASR model would fail to generalize no matter how large the dataset is. Thus, data augmentation techniques using speech synthesis to improve ASR typically face a chicken-and-egg problem, as developing a high-quality speech synthesis system that captures both the acoustic and linguistic information for simulating atypical speech would also be very difficult to do without a large-scale dataset.

With all these said, we aim to investigate the limitations of existing neural network techniques in neural ASR to match the performance of EL ASR with that of healthy

speakers. Since using simple networks limits the maximum recognition accuracies no matter what adjustments are made, we focus on resolving the problem of how to use large networks on small-scale datasets. We first discuss investigations on the limits of novel pretraining methods like self-supervised learning systems against conventional large-scale pretraining. Then, we start investigating methods for removing the domain shift gap between the healthy pretraining and EL fine-tuning datasets through an intermediate fine-tuning task. Finally, we thoroughly analyze the results of the intermediate fine-tuning approach and explain what exactly makes this task successful. Our results show that conventional large-scale pretraining was more advantageous than self-supervised-based pretraining methods for EL speech data. Moreover, through the use of intermediate fine-tuning with imperfect synthetic EL data, we find that although unintuitive, the network was able to optimize learning the voicing characteristics of the target EL speakers efficiently and lead to improved results.

3.2 Domain-Mismatch Adaptation with Intermediate Fine-tuning

3.2.1 Pretraining task formulation

We first formulate the large-scale pretraining process similar to [67] by separating it into two stages: pretraining and fine-tuning. We first define a target text label Y and a conditional probability function $P(Y \mid X)$ of Y given input speech X. Thus, the ASR tasks using the large-scale healthy speech X_{PT} and small-scale EL speech X_{FT} can be represented as in Eq. (3.1) and Eq. (3.2) respectively.

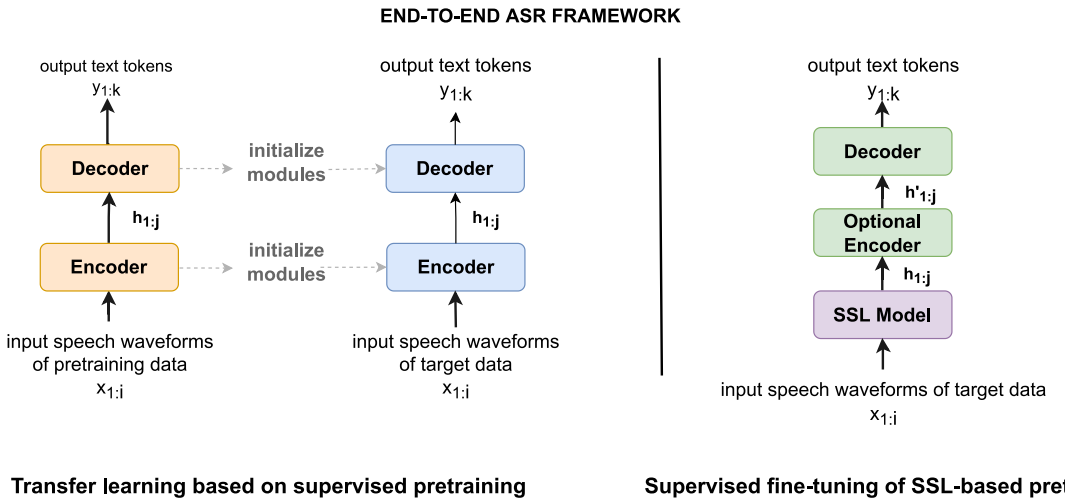


Figure 3.1: Visualization of different types of large-scale pretraining. Supervised pretraining is done by initializing the model's weights from a pretrained model. On the other hand, SSL-based pretraining (right) is similar to supervised training, but instead uses a pretrained SSL model as a feature extractor.

$$\mathcal{T}_{\mathcal{PT}} = \{Y, P_{PT}(Y \mid X_{PT})\} \tag{3.1}$$

$$\mathcal{T}_{\mathcal{FT}} = \{Y, P_{FT}(Y \mid X_{FT})\} \tag{3.2}$$

Equations (3.1) and (3.2) share the same transcript space Y but differ in the input domain X, which induces a domain shift in $P(Y \mid X)$. By using these tasks as a cascaded pipeline, the performance of task $\mathcal{T}_{\mathcal{FT}}$ improves the generalization ability of P_{FT} by using P_{PT} as a prior[1] during training. Since the amount of data X_{PT} available is extremely large, conducting large-scale pretraining would allow task $\mathcal{T}_{\mathcal{FT}}$ to only focus

[1]From hereon, the term "prior" is defined to be the information from the pretrained model by using it as initialization weights.

on learning the voicing characteristics of EL speech, as using the prior P_{PT} should supply sufficient linguistic knowledge, given that both tasks have the same target Y.

As mentioned in Chapter 2.2.1, this large-scale pretraining process can be viewed in two different ways: supervised or self-supervised. Supervised pretraining can be used by first pretraining a network with a large-scale labeled dataset before fine-tuning this on a smaller electrolaryngeal dataset. On the other hand, self-supervised pretraining is more flexible, and does not required labeled data during due to its objective function, enabling the use of more training data. This upstream task can then be used to produce audio features and fine-tuned in a speech recognition downstream task. An overview of each pretraining method can also be found in Figure 3.1.

3.2.2 Limitations in large-scale pretraining

As there are no prior works that discuss the performance of these two pretraining methods on atypical speech, we first need to investigate this aspect and compare both setups to decide which pretraining method to use as a baseline. Nevertheless, regardless of the pretraining method that would be used, there will still be limitations in this approach. Despite large-scale pretraining showing improvements in previous works, this approach is still too naive as the gap in learning from P_{PT} to P_{FT} is too large to overcome due to the huge difference in the input speech features X_{PT} and X_{FT}, limiting the maximum recognition performance. Thus, we aim to soften this learning gap by introducing an intermediate fine-tuning task $\mathcal{T}_{\mathcal{IF}}$, as represented in Eq. (3.3), that uses P_{PT} as a prior. Task $\mathcal{T}_{\mathcal{IF}}$ thereby produces a conditional distribution P_{IF} that P_{FT} can instead use as a prior. Thus, if we can successfully implement a new task $\mathcal{T}_{\mathcal{IF}}$ that can provide a better prior P_{IF} than P_{PT} for task $\mathcal{T}_{\mathcal{FT}}$, we can soften the gap

that the model has to learn and improve the maximum recognition performance.

$$\mathcal{T}_{\mathcal{IF}} = \{Y, P_{IF}(Y \mid X_{IF})\} \tag{3.3}$$

One way to approach this limitation is by going back to previously researched methods. Several methods have been developed to synthetically generate a larger ASR training dataset to improve atypical ASR performance. Research such as [40] uses generative adversarial networks, while [42] uses a text-to-speech method for a more controllable synthesis method. Other approaches such as [41] focus on expanding the vocabulary of the speech utterances in the training set using speech synthesis. In all these aforementioned works, the focus has mainly been on developing a high-quality speech synthesis system to generate speech that can accurately represent both the atypical speech features and its linguistic information. However, as previously mentioned, a weakness of this method is that large datasets are still required to train a high-quality speech synthesis model that represents both the acoustic and linguistic information. The majority of the aforementioned research works also conducted experiments on the UASpeech dataset [4], a atypical speech dataset containing multiple speakers with multiple utterances, which makes their systems hard to reproduce on smaller datasets. Aside from the aforementioned problems, previous literature has also used the data augmentations in one single training run instead of splitting these into multiple runs, whereas we propose to use the data augmentations in a separate training task.

Although not thoroughly explored in speech processing, other fine-tuning methods for neural networks such as intermediate fine-tuning have been commonly studied in other fields such as natural language processing (NLP). Research such as [68], [69] explore the use of using rich-labelled datasets in an intermediate task between a BERT-based [70] pretraining and fine-tuning tasks to increase performance in different proxy tasks and improve robustness to noise. In particular, [68] emphasizes the effectiveness of

using a larger dataset during intermediate fine-tuning for fine-tuning a small target dataset. Although the rich-labelled datasets used in the intermediate fine-tuning are not exactly the same as the target dataset, the goal of the intermediate task is to soften the gap that the model needs to learn when adapting to the target dataset from the large-scale pretraining dataset.

Several works in NLP have also explored the use of artificially-generated data as a pretraining dataset. For example, [71]–[73] explore the use of text datasets without any explicit semantic information such as music, programming code, or an artificially generated language and have found success in making using long short-term memory or an attention-based model like Transformer learn language representations in a BERT-based pretraining [70] task and subsequently perform well in downstream tasks that use natural human language. On the other hand, text datasets generated by randomly sampling from n-gram distributions were not seen to be effective. Thus, a conclusion that can be inferred from the results is that as long as there is some sort of inherent structure available in the dataset, a model trained on this dataset should perform similarly to a model trained on a natural language dataset on the same downstream task.

However, there is yet to be research to be done on what intermediate fine-tuning task would be the most effective in closing this domain shift gap. Thus, we investigate different setups that generate EL speech using different speech synthesis techniques such as text-to-speech and voice conversion. Even when using imperfectly synthesized speech as the dataset during intermediate fine-tuning, speech synthesis techniques use natural inputs like text or speech, making the resulting synthesized speech also inherit the natural and inherent structures of the inputs. Thus, while these inherent and high-level structures in the resulting speech may not be recognizable to the human

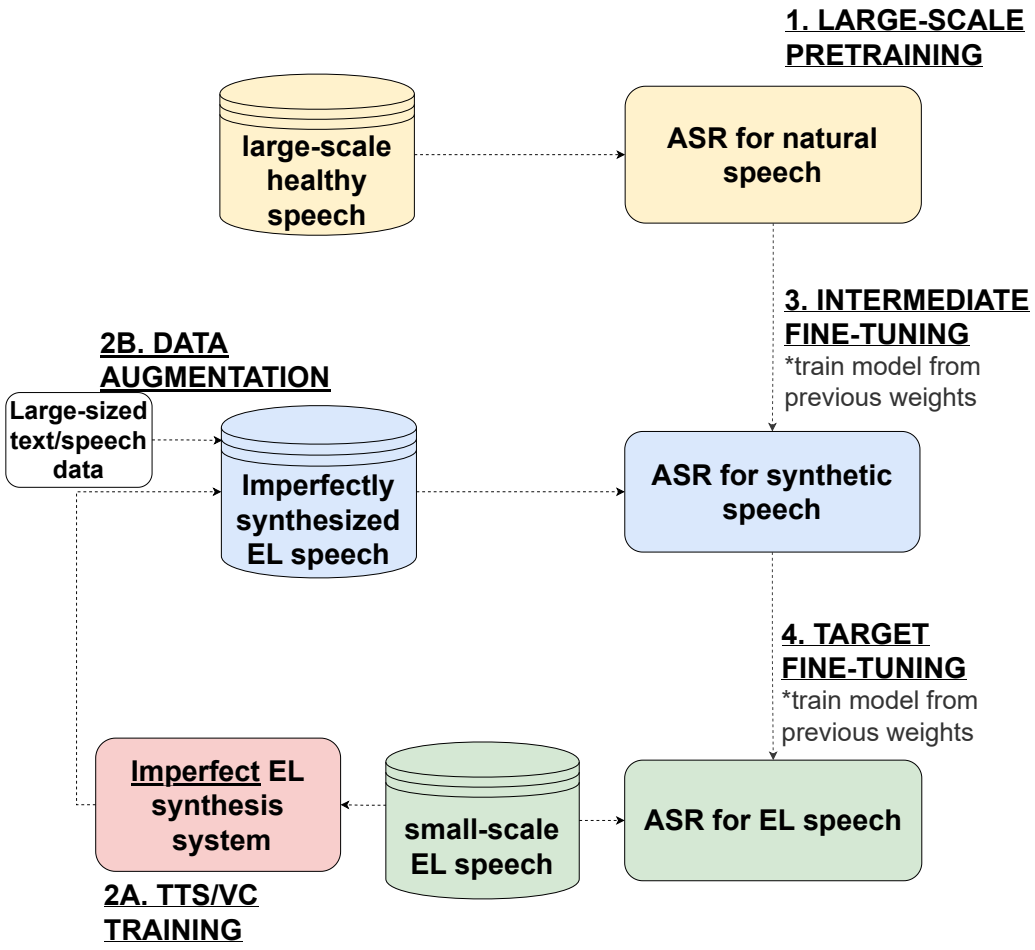


Figure 3.2: Visualization of the training tasks done in the experiment. We propose that adding in the intermediate fine-tuning stage using imperfectly synthesized speech could help the network improve.

ear and may just seem to be distortions, we hypothesize that a neural network could still find these inherent structures and learn speech representations from the imperfect and distorted synthesized speech, similar to how it would learn when using real and undistorted speech.

3.2.3 Intermediate fine-tuning with imperfect EL data

To improve naive large-scale pretraining, we propose an intermediate fine-tuning task that utilizes imperfectly synthesized EL speech in order to close the domain shift gap between the pretraining and fine-tuning data. This intermediate fine-tuning task is conducted between the large-scale pretraining and small-scale target fine-tuning, with the goal of softening the learning gap that the model has to do from the healthy speech to the EL speech. The imperfectly synthesized speech during this intermediate fine-tuning task can be generated using the following speech synthesis methods. These could be considered as noisy inputs.

- **Text-to-speech (TTS):** We input the text information and transform these into speech audio.
- **Voice conversion (VC):** We input the speech audio of a healthy speaker and transform these into the speech audio of the EL speaker using a parallel training strategy.

During intermediate fine-tuning, the model is not expected to improve CER performance, but instead, it should be able to learn the inherent high-level features of the EL speech data that could help it to learn better after fine-tuning with the target EL. Details on the synthesis process can be found in Chapter 3.3.2.

To further show that the model only needs to learn the inherent high-level features and that it does not need to learn linguistic information during the intermediate fine-tuning stage, we distort the target text data labels through two methods. These could be considered as noisy targets.

- **Text-randomization:** We create new text labels of the same lengths by randomly sampling characters from the training text data offline using a uniform

distribution.

- **Text-swapping:** We simply interchange the text information between speech utterances. Both the input speech and target text make sense individually but are not directly correlated.

Similarly, we do not expect the model to immediately output a competitive performance after this stage. However, if our hypothesis of learning the inherent features from the distorted synthetic speech is correct, using these models as initialization weights should have similar performance to the setups with noisy inputs after target fine-tuning.

3.3 Experimental Setups

3.3.1 Large-scale model pretraining architecture

We describe how we implement the pretraining frameworks for this investigation before conducting intermediate fine-tuning. We first conduct the following supervised pretraining setups. We used the Transformer [11] and Conformer [12] as the encoders, both of which have produced the lowest word error rate when using the Librispeech [74] dataset in supervised training[2]. Both encoders had eight attention heads, although the Transformer had 18 layer blocks, while the Conformer only had 12 layer blocks. The decoder for both encoders was a Transformer composed of eight attention heads and eight layer blocks, and trained by CTC-A. The pretraining stage outputs and predicts byte-pair encoding (bpe) tokens [75]; however, in the fine-tuning stage, we initialized a new output layer to decode English and Japanese character tokens instead. For UASpeech, we also compared our results with those obtained using end-to-end models

[2] `https://github.com/syhw/wer_are_we`

found in [76] that predict the character tokens by using a CONV+BLSTM encoder trained using CTC and ATTN with mel-scale filter banks and pitch as inputs.

After pretraining the SSL model as described in Chapter 2.2.1, we fine-tuned it by attaching a decoder to its last layer. Two decoder setups for fine-tuning were compared. The setup first used the original ASR setup in wav2vec 2.0 and WavLM, where a linear projection layer is attached and trained by CTC. Next, following the convention in [77], we trained a model by CTC-A with a vanilla 2-layer 1024-unit BLSTM decoder, as recommended in the ASR setup in [23]. As atypical speech is extremely acoustically different from healthy speech, we also investigated the effectiveness of adding an extra encoder to further pre-process the SSL features before passing it onto the decoder. Two additional encoders were compared with preprocessing the SSL representations, the Conformer [12] and Transformer [11], owing to the strong ability of self-attention to learn representations. We used two attention heads and two layer blocks for both encoders. We then referred to using the SSL representations directly as using an Identity encoder. We found better performance when only the CNN layers of the SSL models were frozen during fine-tuning. The rest of the SSL model components, the encoder, and the decoder were fine-tuned.

3.3.2 Generating imperfectly synthesized speech

For the TTS and VC speech synthesis models, both primarily used the Transformer [11] network to create synthetic EL speech using TTS [78] or VC [19]. The TTS model simply fine-tuned a pretrained model on the EL data. For the VC setup, we followed the same many-to-one setup as described in [43]. One slight difference in this experiment's setup though is that since our dataset only contains one source healthy speaker, we generated multiple source healthy speakers using a pretrained TTS model.

We evaluated the TTS and VC models by synthesizing a version of the EL data and calculated the CER using an ASR model trained on all utterances of EL data. The synthesized speech resulted in imperfect generations with high CER, with TTS-generated at 34.1% and VC-generated at 71.9% despite the ASR model being able to recognize the ground truth speech at only 4.4% CER. However, even with high CER scores and failing to model the linguistic information, one could still immediately figure out that the speaker speaks with an electrolarynx when listening to the samples, as the model could still somehow model the EL voicing characteristics.

3.3.3 Implementation details

We used ESPnet [79], [80], an open-sourced speech processing toolkit, to implement the speech models. Each stage used a Conformer encoder with eight attention heads and 12 layer blocks, and a Transformer decoder composed of eight attention heads and eight layer blocks. The model was optimized by CTC-Attention [14], [15] with $\lambda = 0.8$. We fine-tuned the networks' learning rate in steps of 10^{-n} where $n \in \{2, 3, 4\}$ and used the best results from each run. The language models used are based on the Transformer [11] architecture and trained on the respective large-scale datasets' text. We used the CER to evaluate the results. Results shown when using intermediate fine-tuning are the mean of three runs initialized from different random seeds. In Tables 3.3 and 3.4, we will refer to the Conformer architecture as "Conf." and the Transformer architecture as "Transf.".

3.3.4 Datasets

Large-scale pretraining

We first compare the effects of different pretraining methods, along with the amount of data use during pretraining. To pretrain the supervised models, we used the 960 hours of Librispeech [74]. On the other hand, SSL pretraining varies per model, with wav2vec 2.0 pretrained on 60k hours of speech and WavLM pretrained on 94k hours of speech. In the fine-tuning stage, we used two atypical speech datasets, as summarized in Table 3.1. We first investigated an in-house recorded dataset of Japanese electrolaryngeal speech, which we refer to as ELSpeech in this thesis. The dataset was recorded by a laryngeal speaker by speaking 1000 different sentence utterances both with an electrolarynx and with their normal voice. We use 85% of the data for the training set, and the rest for the test set. Both the healthy and electrolaryngeal speeches were used in the train and dev sets but only the electrolarygneal speech was used in the test set. As the dataset was in Japanese, we used the CER as the evaluation metric.

To further verify the results, we also investigate another atypical dataset of dysarthric speech. The next dataset we used was UASpeech [4], a dataset containing parallel English word utterances of 15 dysarthric speakers and 13 healthy speakers. We followed the recommendations in [76] and used both the dysarthric and healthy speakers to train the ASR system but only used the dysarthric speakers for the test set, and removed excessive silences at the start and end of each utterance using a GMM-HMM system[3]. The train and dev sets were from the B1 and B3 blocks, whereas the test set was from the B2 block. We followed the convention of the said paper and used the word error rate (WER) as our evaluation metric.

[3] `https://github.com/ffxiong/uaspeech`

Table 3.1: Duration (in number of hours) of each split used in the experiment. UASpeech results are obtained after silence removal.

	Healthy		Atypical		
Dataset	**Train**	**Dev**	**Train**	**Dev**	**Test**
ELSpeech	1.21	0.37	1.40	0.46	0.42
UASpeech [4]	2.38	0.45	3.60	0.68	2.09

Intermediate fine-tuning

To verify our proposed intermediate fine-tuning method, we used several EL speech datasets spoken in Japanese. To pretrain the models with large-scale healthy datasets, we used the Japanese LaboroTV Speech dataset (2k hours) [81] for the supervised pretraining setup. For SSL pretraining, we used the Japanese CSJ corpus [82] (661 hours) and an imperfectly synthesized EL version of the entire CSJ corpus. Since the CSJ corpus is relatively small for SSL pretraining, we also used the large-scale English dataset Librilight [83] (60k hours) for comparison.

Thus, for our target EL speech during the target fine-tuning stage, we used two parallel datasets, one recorded from an actual EL patient and one recorded by a healthy speaker using an electrolarynx, which we will refer to as ELREAL and ELSIMU1 respectively. Next, to train the TTS and VC models, we used a slightly larger dataset recorded by a different healthy speaker using an electrolarynx, which we will refer to as ELSIMU2. The utterances here have no overlap with ELSIMU1 or ELREAL. In all cases, the datasets were smaller in number compared to common ASR tasks, containing just around 2 hours of data in total. For the intermediate fine-tuning, we used 20k utterances from the CSJ corpus and converted these to EL using the TTS and VC

Table 3.2: Total duration (in mins) and number of utterances for each split used in our experiment.

	Duration			No. of Utterances		
Dataset	**Train**	**Dev**	**Test**	**Train**	**Dev**	**Test**
HEALTHY	4.38	2.14	2.17	116	40	40
ELREAL	5.77	2.96	2.99	116	40	40
ELSIMU1	6.26	2.71	2.75	116	40	40
ELSIMU2	125.91	-	-	1000	-	-

models. The resulting synthetic speech will be referred to as ELIMPERFECT. The details regarding the number of minutes and utterances of each dataset can be found in Table 3.2. Finally, to analyze the behavior of the neural network with a normal voice, we used a healthy speaker dataset which we refer to as HEALTHY, which is parallel with ELREAL and ELSIMU1.

3.4 Experimental Results: Comparison of Pretraining Methods

We first establish the baseline pretraining method discussed in Chapter 2.2.1 to be used as the base for our proposed method. As there has been no prior work that establishes the most efficient pretraining method, we aim to clarify the advantages of each pretraining method by comparing different setups, and show up to what extent the best setup can achieve with just simple large-scale pretraining on a privately collected electrolaryngeal speech dataset, shown in Table 3.3. To further show the generalizability

of the results on other atypical speech, we also perform the experiments for dysarthric speech in UASpeech, shown in Table 3.4.

Table 3.3: CER comparison between using different SSL features and raw features in ELSpeech. ** indicates end-to-end models pretrained with 960h Librispeech.

Sys.	Features	Encoder	Decoder	Loss	CER%
1	Mel-scale filterbank	**Conf.****	**Transf.****	**CTC-A**	**27.9**
2		Transf.**	Transf.**	CTC-A	61.5
3	wav2vec 2.0	Identity	Linear	CTC	93.3
4		Identity	LSTM	CTC-A	89.2
5		Conf.	LSTM	CTC-A	54.1
6		Transf.	LSTM	CTC-A	50.2
7	wav2vec 2.0 (XLSR)	Identity	Linear	CTC	99.6
8		Identity	LSTM	CTC-A	89.9
9		Conf.	LSTM	CTC-A	90.2
10		Transf.	LSTM	CTC-A	71.5
11	WavLM	Identity	Linear	CTC	54.9
12		**Identity**	**LSTM**	**CTC-A**	**41.8**
13		Conf.	LSTM	CTC-A	42.5
14		Transf.	LSTM	CTC-A	46.0

3.4.1 Comparison of SSL frameworks

First, we analyze the two SSL model frameworks described in Chapter 3.2.1, based on contrastive loss (wav2vec 2.0 [25]) and another based on masked region prediction (WavLM [27]) on both datasets. As seen in Sys. 12 in Table 3.3 (EL Speech) and Sys. 10 in Table 3.4 (Dysarthric Speech), using the WavLM model with an LSTM decoder outperforms all setups using wav2vec 2.0 in both datasets, as it produces the lowest CER (41.8%) and WER (51.8%), making it the best SSL setup. This performance can be attributed to the speech denoising framework used in masked

Table 3.4: WER comparison between using different SSL features, mel-scale filter bank, and raw waveforms in UASpeech. ** indicates models pretrained with 960h Librispeech.

					WER% of each intelligibility rating				
Sys.	**Features**	**Encoder**	**Decoder**	**Loss**	**Overall**	**Very Low**	**Low**	**Mid**	**High**
1	Mel-scale filter bank	Conv+BLSTM [76]	Linear	CTC	58.0	87.1	62.4	55.3	37.8
2		**Conv+BLSTM**** [76]	**LSTM****	**ATTN**	**35.0**	**68.7**	**39.0**	**32.5**	**12.2**
3	Mel-scale filterbank	Conf.**	Transf.**	CTC-A	67.2	90.6	73.6	64.1	46.6
4		Transf.**	Transf.**	CTC-A	68.2	89.6	73.9	66.2	48.7
5	wav2vec 2.0	Identity	Linear	CTC	75.7	93.1	79.4	74.1	60.4
6		Identity	LSTM	CTC-A	96.4	97.9	97.5	95.8	95.0
7		Conf.	LSTM	CTC-A	71.7	92.3	74.6	68.7	55.3
8		Transf.	LSTM	CTC-A	71.8	93.8	83.7	69.7	48.4
9	WavLM	Identity	Linear	CTC	91.5	97.3	92.3	90.5	87.1
10		**Identity**	**LSTM**	**CTC-A**	**51.8**	**71.5**	**50.0**	**46.0**	**40.6**
11		Conf.	LSTM	CTC-A	70.1	94.1	81.3	67.1	48.5
12		Transf.	LSTM	CTC-A	77.2	94.5	83.9	75.3	60.3

region prediction, which makes the model more robust to acoustic variations in speech features found in atypical speech and the attention-based LSTM decoder, thereby improving the representation learning. A similar trend in both datasets is also seen when using an additional Conformer or Transformer encoder (Sys. 5, 6, 13, and 14 in Table 3.3 and Sys. 7, 8, 11, and 12 in Table 3.4) to preprocess the SSL features, where it degrades the performance with WavLM, but improves wav2vec 2.0. This shows that the wav2vec 2.0 features are not as strong in projecting the atypical speech into a latent space and needs further processing for it to properly work. Moreover, as the SSL models are trained in English and the ELSpeech dataset is in Japanese, we also test the performance of the multilingually pretrained wav2vec XLSR model [26], which has been successfully used in decoding languages not included in the pretraining data. However, we do not find any success to this approach, as seen in Table 3.3, where we see a degradation in performance when using XLSR for all setups compared with its

wav2vec 2.0 counterparts.

3.4.2 Comparison of SSL pretraining and supervised pretraining

Next, we compare the SSL pretraining frameworks with conventional pretraining methods. We present two findings when comparing these two pretraining methods. First, we see a trend similar to that in [77], where the best SSL setups in Sys. 12 in Table 3.3 and Sys. 10 in Table 3.4 outperform the pretrained Transformer setups in Sys. 2 in Table 3.3 and Sys. 4 in Table 3.4, proving that SSL pretraining has the potential to outperform supervised pretraining.

However, our second finding is that other supervised setups not used in [77] may still outperform SSL pretraining. As seen Sys. 1 in Table 3.3, using a pretrained Conformer-Transformer model produces the lowest CER (27.9%) for ELSpeech, outperforming the best SSL setup by 13.9%. On the other hand, although the Conformer-Transformer setup (Sys. 3 in Table 3.4) did not perform very well in UASpeech, [76] showed that using a pretrained Conv+LSTM model with 40-dimensional mel-scale filter banks as inputs can still outperform our best SSL setup by 16.8%, as seen in Sys. 2 in Table 3.4.

3.5 Experimental Results: Proposed Intermediate Fine-tuning Method

As we have established large-scale pretraining to be the superior pretraining method, we use this as our baseline method and improve on it. Although supervised pretraining

was shown to be effective in this chapter, there is still an obvious gap between as seen in the high CER and WER scores. Compared to performance on healthy speech datasets [4], the performance of these ASR systems is still quite high as shown in Tables 3.3 and 3.4. As discussed in Chapter 3.2.2, we attribute this low performance due to the domain mismatches found between healthy and atypical speech. In the next chapters, we discuss how the proposed method can alleviate these domain mismatches found in conventional large-scale pretraining methods. Thus, we investigate the use of imperfect synthetic EL data as a solution as proposed and described in Chapter 3.2.

3.5.1 Effectiveness of intermediate fine-tuning

We start the experiments by investigating the SSL pretraining method by instead using imperfect EL data as the SSL pretraining data. Previous research has shown the robustness of SSL models with large improvements in WER when pretrained with in-domain data [84]. As expected and in line with the findings in Chapter 3.4.2, we see in Table 3.5 that SSL is still not as effective for EL speech even when pretrained with large-scale datasets like CSJ or Librilight, as it completely fails to recognize any of the sentences. We also use the CSJ dataset and convert it into EL speech using the VC model, which we refer to as CSJ-ELVC, and use the resulting generations as the SSL training data. However, as seen in Table 3.5, all SSL models from Sys. 1 to 3 fail to adapt to both EL datasets.

Next, we investigate the effectiveness of the intermediate fine-tuning task. As seen in Table 3.5, adding in an intermediate fine-tuning step improves performance of conventional ASR pretraining methods, proving that we can close the gap in the domain shift between the pretraining and target data using imperfect speech during supervised pre-

[4] `https://github.com/syhw/wer_are_we`

Table 3.5: CER of different pretraining and fine-tuning setups on ELREAL and ELSIMU1. Results with intermediate fine-tuning are mean CER of three runs initialized from different random seeds.

Sys.	Pretraining Method (Dataset)	Intermediate Fine-tuning Dataset	ELREAL CER%	ELSIMU1 CER%
1	**SSL (Librilight)**	None	109.7	291.9
2	**SSL (CSJ)**	None	98.4	209.0
3	**SSL (CSJ-ELVC)**	None	94.7	98.7
4	**Supervised**	None	22.2	17.2
5	**(LaboroTV)**	ELSIMU2	21.1	15.6
6		**TTS, text-randomized**	**20.2**	**24.4**
7	**Supervised**	**TTS**	**18.7**	**18.1**
8	**(LaboroTV)**	**VC**	**18.3**	**18.0**
9		**TTS, text-swapped**	**16.1**	**15.7**

training. We first observe that although the VC-generated data discussed in Chapter 3.3.2 had a significantly higher CER (71.9%) than the TTS-generated (34.1%), there is no CER degradation seen between Sys. 7 and Sys. 8. Thus, we show that CER is not an effective proxy in filtering out EL synthetic data for the training dataset, contrary to the technique in [85], primarily because atypical speech can be both unintelligible while also natural at the same time.

In addition to this, we observe that distorting text labels can be effective. As seen in Sys. 9, using text-swapped labels proves to be the most effective in the intermediate fine-tuning step in both datasets. However, when using randomly sampled text labels as seen in Sys. 6, we see a lesser effectiveness in the method. This is similar to the findings found in the NLP experiments in Chapter 3.2.2, where using pretraining text data randomly sampled from n-grams was also not effective in improving the NLP

model performance. Thus, there must be other high-level features that the model was using to improve the target fine-tuning step rather than using the low-level features such as intelligibility.

Another observation is the difference in the results with ELSIMU1 and ELREAL. We see that the intermediate fine-tuning was not as effective and that using the conventional pretraining and fine-tuning method in Sys. 4 results in a lower CER (17.2%) than the other setups. This is most likely due to ELSIMU1 having a smaller domain shift gap with the healthy pretraining dataset, compared to ELREAL. As ELSIMU1 was only recorded by healthy speakers using an electrolarynx, ELSIMU1 most likely mimicked the robotic quality of EL speech well, but not the other aspects such as the speech rate, pronunciation tendencies, and the such that are caused by the removal of the larynx.

3.5.2 Analysis of latent spaces produced in each task

Intuitively, the results do not exactly follow any expectations based on statistics as neural networks are designed to elucidate the posterior distribution $P(Y \mid X)$. Fine-tuning through weight initialization uses the previous task's $P(Y \mid X)$ as its prior, which therefore means that the distributions in each stage need to be as close as possible. Using imperfectly synthesized speech should have made the posterior distribution of the intermediate fine-tuning task distant to that of the pretraining and fine-tuning tasks due to the distorted linguistic information in the speech audio and the distorted text labels; however, results show a contrast to these expectations. Although we have hypothesized in Chapter 3.5.1 that the network learns some inherent features from the structure of the data, we have still yet to find what exactly this inherent feature is. In the following subsections, we dig deeper into the network and analyze

what exactly is happening at each task that makes the method successful.

To implement the following experiments, we take the latent spaces of the three parallel datasets (ELREAL, ELSIMU1, and HEALTHY) produced by each task's encoder and analyze which tasks each latent space succeeds the most in. Moreover, more focus is placed on the results with ELREAL as the goal of the research is to improve ASR for real EL speakers.

Performance of each task in linguistic content proxy tasks

We first conduct a simple proxy task by checking the CER of the test set during each task. An interesting finding in Fig. 3.3 that we find is that using the imperfect synthetic speech first results in CER scores even higher than what we get after the large-scale pretraining task at 77.1% CER. Despite this initial degradation by the intermediate fine-tuning, the network performance still improves after the target EL fine-tuning task, which proves our initial hypothesis in Chapter 3.2 that the network only focuses on learning the inherent structure of the distorted speech data, similar to how it would with real speech. In particular, we see that the text-swapping setup, which had the worse CER after the intermediate fine-tuning stage, performed the best after the target fine-tuning, leading us to think that the network does not focus on using low-level intelligibility features during the intermediate fine-tuning stage.

We further show that even when these are used as conditioning features to synthesize speech, the latent spaces in the intermediate fine-tuning stage contain less linguistic information than that of the pretraining stage. To implement this experiment, we use the extracted latent spaces as the conditioning features to train a neural vocoder to synthesize speech. We use Parallel WaveGAN [86] due to its simple architecture and ability to synthesize speech at a fair quality.

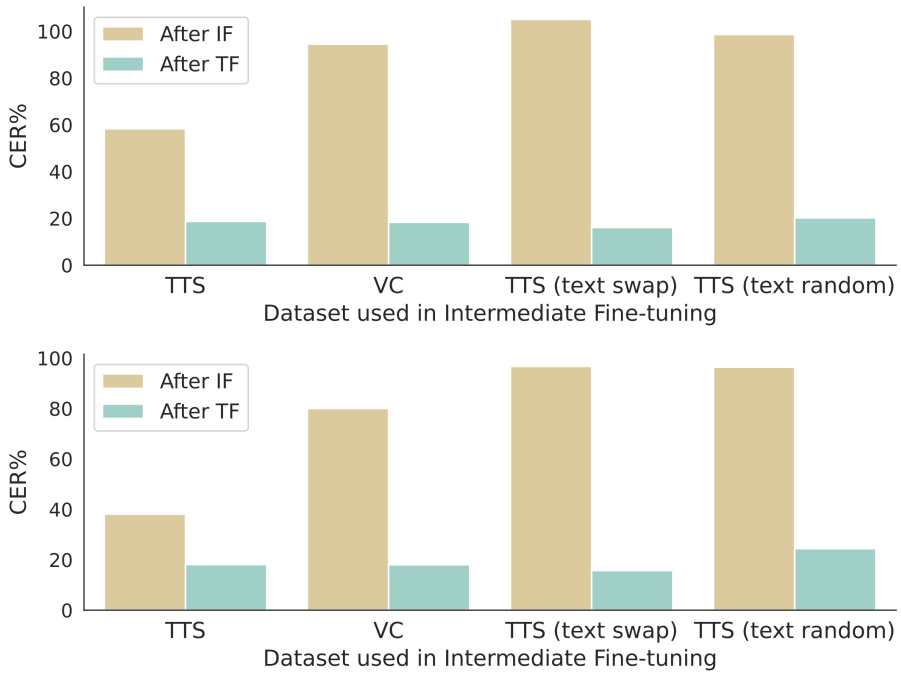


Figure 3.3: Visualization of the ELREAL CER (1) immediately after the intermediate fine-tuning task (IF), and (2) after the target fine-tuning (TF) task for ELREAL (top) and ELSIMU1 (bottom).

As seen in Table 3.6, the speech synthesized when using the intermediate fine-tuning stage (TTS, text-swapped) as conditioning features even produces a higher CER than that of the pretraining stage in both datasets, despite the pretraining task having no contact with any sort of EL speech before, showing that the latent space produced during the intermediate fine-tuning task barely contained any linguistic information. Moreover, as we expect, the target fine-tuning stage produces synthesized speech with the lowest CER in both datasets.

With all these results, we prove that the intermediate fine-tuning does not focus on learning linguistic content as it has consistently proven to be even worse than the pretraining task in linguistic content-based proxies, despite it not having any contact

Table 3.6: Evaluation of the quality of synthetic speech generated by Parallel WaveGAN when using the latent spaces as conditioning features using CER.

Task	ELREAL	ELSIMU1
Pretraining	34.1	54.1
Intermediate Fine-tuning	36.1	56.3
Target Fine-tuning	24.0	50.9
Ground Truth	15.3	14.0

with EL speech before. Although a possible initial hypothesis was that some high-level inherent structure may be hidden in the linguistic content, no evidence shows this to be the case.

As we observe that linguistic content is not learned during the intermediate fine-tuning task, we turn to find other proxies that could verify where the intermediate fine-tuning task performs well compared to the other tasks. Given that the intermediate fine-tuning task is unable to contain linguistic information, we start to hypothesize that the high-level inherent features here may be the voicing characteristics of the EL speakers. In the following experiments, we use the test set utterances of ELREAL, ELSIMU1, and HEALTHY, which are all parallel. Similar to the experiments in the previous chapter, each utterance is passed into the encoder to produce the latent space.

Performance of each task in speaker identification proxy tasks

To start this study, we visualize the latent representations from each task using t-SNE [87]. In our previous work, we verified the differences in the projection between each task for the text-swapping setup, showing that the intermediate fine-tuning task

optimizes finding the high-level inherent structures by learning to separate EL speech representations from healthy speech representations. Here, we analyze the latent spaces further by comparing all setups and each task. Compared to the previous work which visualized each frame as a dot, we instead use a whole utterance to represent a dot, which is calculated by taking the mean of all frame-wise features in that utterance. As shown in Fig. 3.4, we understand the behaviors better through this method. We see that in the large-scale pretraining task, the model projects all three speakers in roughly the same area as expected. During intermediate fine-tuning, we see the distance of the separation between the EL and healthy features differ for each intermediate fine-tuning setup. Looking into the figures deeper and comparing with the resulting CERs in Table 3.5, we further observe that the amount of separation from the EL and healthy speech features are correlated to the resulting CER, where a lower CER also means a greater separation between the healthy and EL speech. On the other hand, no significant correlations to the resulting CER were found when visualizing the target fine-tuning task, showing that the resulting CER is heavily dependent on the results from the intermediate fine-tuning task. This further proves the initial hypothesis that the intermediate fine-tuning task does not learn linguistic-related features and instead focuses on the voicing characteristics of the EL features.

To further prove that the intermediate fine-tuning succeeds in identifying speaker identity, we conduct a categorization task by taking the latent spaces and check whether a model could classify if the input speech was spoken by ELREAL, ELSIMU1, or HEALTHY. As these three datasets are parallel and equal in number of utterances, we are assured that the model will only focus on the pronunciation tendencies and voicing characteristics rather than the linguistic content. To train the model, the latent spaces are passed through three convolutional layers with a ReLU activation

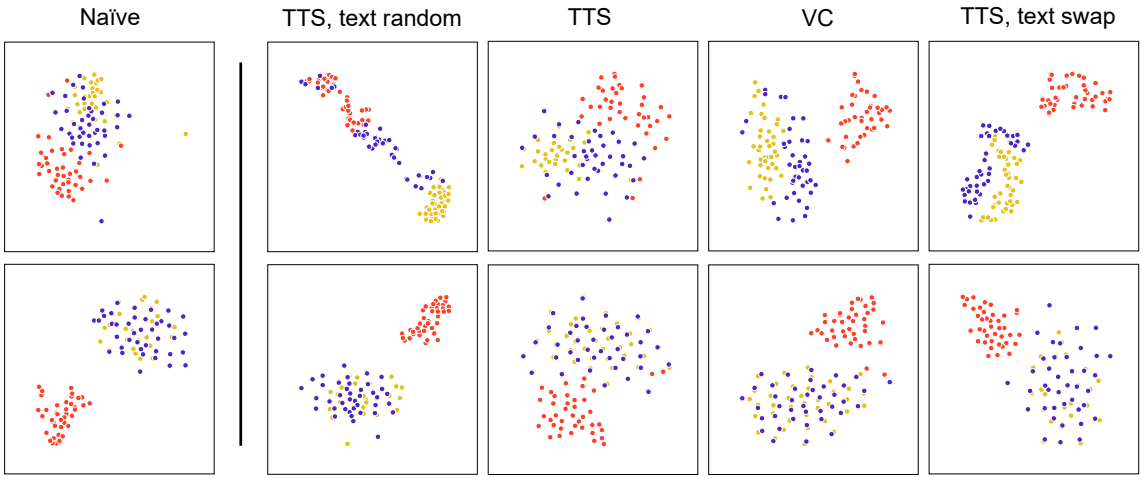


Figure 3.4: Comparison of how the ELREAL, ELSIMU1, and HEALTHY test sets are projected into the latent space, visualized by t-SNE [87]. Each utterance frame-wise mean from the latent space is represented as a single dot, with orange dots representing healthy speech, violet dots representing ELSIMU1, and yellow dots representing ELREAL. The top-left plot shows when using large-scale pretraining model. The bottom-left plot shows target fine-tuning without intermediate fine-tuning. On the right-hand side, the TTS text-randomized, TTS, VC, and TTS text-swapped plots are shown. The top plots are during intermediate fine-tuning, while the bottom plots are after the target fine-tuning.

function in between. The output is average pooled into a 1-dimensional output, which predicts the category (or in this case, a one-hot vector) of the input speech. We train the model for 2000 steps and test on the test set which contains 40 utterances from each speaker.

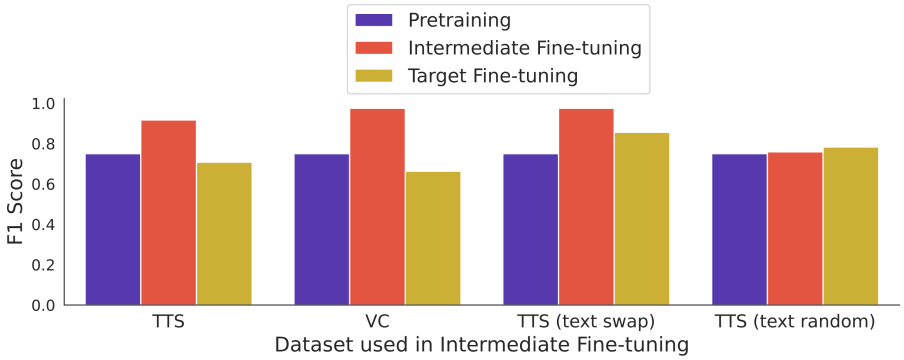


Figure 3.5: Resulting F1 scores when latent spaces from each task are used to categorize the speech category (ELREAL, ELSIMU1, HEALTHY) on a parallel test set.

As seen in Fig 3.5, we find that the intermediate fine-tuning task (TTS, text-swapped) highly succeeds in the categorization task by almost always perfectly classifying the input speech into their respective categories. On the other hand, the latent spaces of the pretraining and target fine-tuning tasks did not perform as well and always performed worse than their intermediate fine-tuning counterparts in categorization tasks. We recall in Chapter 3.5.2 that the setups with the highest CER immediately after the intermediate fine-tuning were also the ones that produced the lowest CER after the target fine-tuning task. We also see a similar tendency wherein the F1-score of categorization follows a similar trend to that of Table 3.5, where the setups with the highest F1 scores (TTS, text swapping) also produced the lowest CER.

Based on these results, we see that the more imperfect and distorted the data used in the intermediate fine-tuning was, the better it performs at identifying speakers and

the better it becomes to use as a prior. A good reason why the network changed its objective during the intermediate fine-tuning, despite using a CTC-Attention loss, is because the data was too distorted to learn any linguistic information from, the network instead found a different structure to learn from, which in this case was the voicing characteristics of each speaker. However, this behavior cannot be learned from any randomly distorted dataset. As shown by the difference in performance in the text swapping and text randomization, there still needs to be some sort of structure available in the data. Thus, from the results, it is conclusive to say that optimizing learning the voicing characteristics during the intermediate fine-tuning can be an efficient task when pretraining ASR models.

3.5.3 Analysis of ASR model behavior during intermediate fine-tuning

One hypothesis why the network works despite having distorted target linguistic labels is that the network instead optimizes to learn the inherent structures, which we conclude as the voicing characteristics, and does not prioritize learning the speech-to-text alignment as much as it usually would. However, this is not to say that the network completely disregards the speech-to-text alignment. For example, since the setup with text random target labels was randomly sampled and each character did not correlate with each other, the intermediate fine-tuning task was less effective because there was no pattern for the model to learn. On the other hand, the setup with text-swapped labels had a pattern that grammatically correlated the characters with each other, which the network could use to find and learn to improve the performance in the target EL fine-tuning task. We can relate this to the success of using imperfect EL speech, since the imperfect speech data were generated by a sequence-to-sequence

model (as described in Chapter 3.3.2), making each frame have an inherent structure that correlated them with the other frames. In this case, the high-level features may have represented the voicing characteristics well enough, which the network found and helped it learn better during the target fine-tuning task.

Moreover, in Fig. 3.6, we find that if we remove the CTC loss constraint during the intermediate fine-tuning task, the final CER degrades in all setups. Thus, the network still needs to learn a monotonic alignment using the CTC loss, even with a weak one. These show that even though there is not a full correlation between speech and text, learning a weak alignment could still benefit the network. These findings also validate our previous hypothesis in Chapter 3.2.1, as we hypothesized that transfer learning would allow the model to focus on only learning the acoustic features of the EL speech. Since the acoustic features EL speech in the temporal structure differ from that of healthy speech, if we prohibit the model from learning any sort of temporal structure by removing the CTC loss, using the prior generated during the intermediate fine-tuning task becomes significantly less effective.

3.5.4 Use of speaker identification loss during intermediate fine-tuning

Since all experiments lead to the conclusion that the intermediate fine-tuning task focuses on learning the voicing characteristics, we prove the effectiveness of learning the voicing characteristics. This auxiliary loss predicts the probability of the utterance being spoken by an EL or healthy speaker and is optimized using a cross-entropy loss. We repeat the text-swapping setup and use the imperfect synthetic EL data and additionally use a healthy single-speaker dataset (JSUT [88]) to compare the effects of the auxiliary loss.

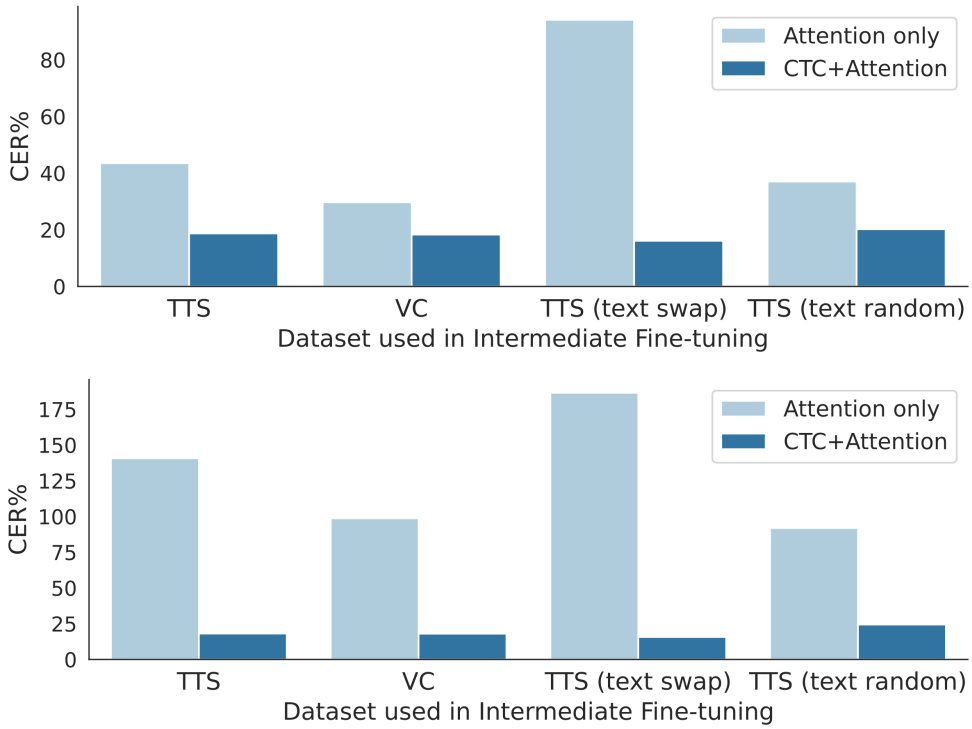


Figure 3.6: Visualization of CER difference on ELREAL after target fine-tuning when (1) using only an attention loss during intermediate fine-tuning, and (2) using the multi-objective CTC-Attention loss during intermediate fine-tuning for ELREAL (top) and ELSIMU1 (bottom).

Let $X = \{X_{\text{TYP}}, X_{\text{EL}}\}$ be the training data, which is composed of a typical and an EL dataset X_{TYP} and X_{EL}. The speech type ID loss L_{SID} identifies whether the speaker is a typical or EL speaker and is optimized using a binary cross-entropy loss. Since we use both EL and typical data, we mask the outputs from the typical speech inputs during the calculation of the CTC/Attention losses L_{ctc} and L_{attn} [14]. The masking avoids making the model learn two highly variant types of speech, improving decoding performance. In these experiments, we mask the outputs from the healthy speakers when computing the CTC-Attention losses; thus, the healthy speech is only used to

Table 3.7: Resulting CER when adding the speaker identification task as an auxiliary loss during intermediate fine-tuning.

Task	ELREAL	ELSIMU1
TTS, text-swapping	16.1	15.7
TTS, text-swapping with speaker ID loss	15.6	40.2

optimize the divergence between EL and healthy speech. Note that the auxiliary loss is only used during intermediate fine-tuning.

After running three trials initialized from different random seeds and averaging the results, we find that using the auxiliary loss results in a better CER for ELREAL, as seen in Table 3.7. On the other hand, using the same method to improve ELSIMU1 was not effective at all. Although we prove that intermediate fine-tuning focuses on learning the voicing characteristics of the data with ELREAL, the results from Table 3.7 lead us to think that there may be intrinsic differences between ELREAL and ELSIMU1 despite both sounding like EL voices to the human ear.

Interestingly, we also notice that the model's generalization ability improves when using this loss. For example, when using the pretrained model, the HEALTHY dataset can be decoded with a low 4.3% CER. Using the TTS text-swapping method expectedly severely increases to 82.6%, as the network has been solely optimized for EL speech. However, when using a speaker identity loss, the degradation becomes minimal, being able to decode at a 6.0% CER. This shows that forcing the encoder to learn different speaker identities not only improves performance in EL speech but also in HEALTHY speech, allowing a more generalizable model.

3.5.5 Differences in pronunciation between ELREAL and ELSIMU1

In the previous experiments, we see that there are differences in results when using these two datasets. As seen in Table 3.5, the intermediate fine-tuning task was less effective when using the ELSIMU1 dataset compared to ELREAL. Moreover, we saw that adding an auxiliary loss was not effective when using it with ELSIMU1 despite being effective with ELREAL. These results lead us to think that although both sound like EL speakers for humans due to the voicing characteristics, they may still be perceived differently by the model. To gain a better understanding of the gap between simulated EL and real EL data and conduct better experiment setups in the future, we analyze the differences in ELREAL and ELSIMU1 as perceived by the model.

We dig deeper into the categorization task by looking at the confusion matrix. As mentioned, a reasonable expectation would be that the model would confuse categorizing ELREAL and ELSIMU1 together due to the use of an EL device and due to the fact that the t-SNE visualization in Fig. 3.4 overlapped both EL speakers and are projected separately from the healthy speech. However, as seen in Fig. 3.8, this hypothesis is wrong, as the model more frequently misrecognizes ELSIMU1 as HEALTHY than ELREAL. Thus, we see that aspects like the imitation of the pronunciation of phonemes caused by the removal of the larynx were not accurately simulated. Moreover, since ELSIMU1 and HEALTHY were from the same speaker, the model may have perceived the pronunciation tendencies to be the same.

This confirms our previous hypothesis in Chapter 3.5.1, where we mention that the effectiveness of intermediate fine-tuning was less, primarily because the domain gap between the large-scale healthy speech and ELSIMU1 was smaller than that of the large-scale healthy speech and ELREAL. Another possibility that this shows is the

Table 3.8: An overview of the DHH speaker dataset collected.

Speaker ID	**Gender**	**No. of utterances**	**Total length (Hours)**
S1	Male	5000	11.41
S2	Male	4982	10.69
S3	Female	902	1.64
S4	Female	2493	4.31
Total	-	13377	28.05

speaker-dependent characteristics of the model, showing that the model can only decode the target EL data. As a conclusion, this also gives an idea of why EL ASR is a difficult task, as simply conducting experiments by simulating the voicing characteristics would not always translate to results with actual EL speakers, thus calling the need to invest more time in collecting data from atypical speakers to further advance the research progress in this problem.

3.5.6 Generalization of proposed techniques to other atypical speech datasets

Although the proposed techniques in this thesis have been applied to EL speech, the pretraining and fine-tuning techniques presented in this thesis can also be applied to other types of atypical speech as demonstrated. We demonstrate the same methods with four deaf and hard-of-hearing (DHH) speakers (S1, S2, S3, and S4) and analyze the network behaviors with different training set sizes and setups. An overview of the DHH speaker dataset collected is shown in Table 3.8.

Then, we describe the different setups used in our experiment. We follow the intermediate fine-tuning (IF) setup detailed in Chapter 3.5.1 and extend upon their findings.

- **Direct**: Directly using an out-of-the-box pretrained model and evaluating the performance without performing the intermediate fine-tuning.
- **Direct - speaker-independent**: Fine-tuning the dataset with the other DHH speakers' audio data. This means that we do not use the target DHH speaker's audio data to fine-tune the model.
- **Naive fine-tuning**: Simply fine-tuning the pretrained model on the target speaker without any extra procedures.
- **IF - TTS**: Based on the intermediate fine-tuning setup described in Chapter 3.5.1. We generate a larger dataset of the target DHH speaker using TTS, and use these during intermediate fine-tuning.
- **IF - TTS with text-swapping**: Similar to the TTS setup, but randomly swapping the text labels within the dataset. Note that the text labels are only swapped during the intermediate fine-tuning stage as described in Chapter 3.5.1.
- **IF - speaker-independent**: Instead of using synthetic speech, we use the real speech recordings from other DHH speakers during intermediate fine-tuning. This allows us to assess if the improvements in intermediate fine-tuning come from the speaker-dependent or speaker-independent characteristics of the data.

Figure 3.7 summarizes our results. They show that applying data augmentation and intermediate fine-tuning significantly improves performance in this low-resource setting, indicating that the method proposed in Chapter 3.5.1 transfers to DHH speech.

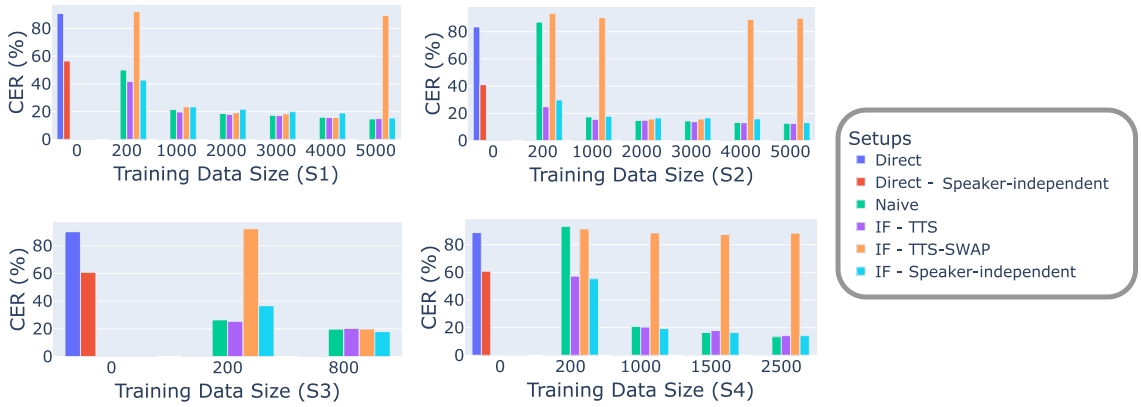


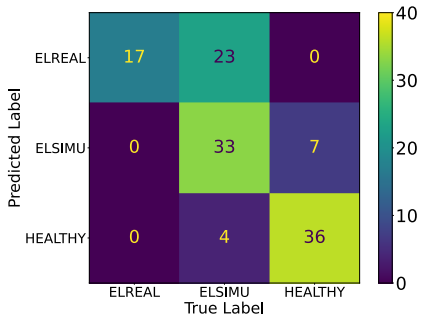
Figure 3.7: Character error rate (CER) of the different developed ASR systems for all speakers (S1, S2, S3, and S4). The x-axis indicates the amount of data used in the training and development sets.

However, we observed that using intermediate fine-tuning became less effective as the training data size increased. Amongst the different setups, we find that the IF-TTS setup was the most stable and showed the most improvements in the low-resourced scenarios. Although the IF-TTS text-swapping setup was seen to be successful in Chapter 3.5.1, we do not see the same trend in our experiments as we observe its instability. Specifically, in cases where the training data is too low or too high, we observed a collapse and the ASR model fails. One hypothesis on why this is the case is because of the difference in speech features between electrolaryngeal speech and DHH speech. For electrolaryngeal speech, the speaker identity is disrupted, but the intelligibility is still there, as several phonemes can still be pronounced almost the same as a typical speaker. On the other hand, DHH speech contains the speaker identity but has unintelligible speech. Thus, since the intelligibility information between the pretraining data (LaboroTV) and the target data (electrolaryngeal speech) could be transferred and shared, the intermediate fine-tuning worked better on the electrolaryngeal dataset.

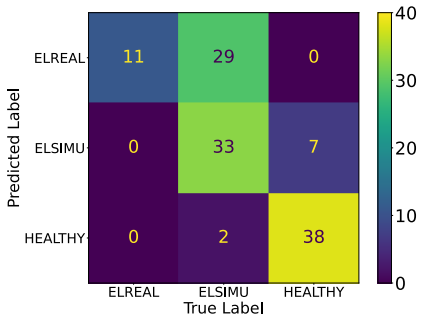
On the other hand, in the case of our experiments, since the intelligibility information between the pretraining data (LaboroTV) and the target data (DHH speech) vastly differed, the information was not effectively shared. ASR models typically focus on the intelligibility information and disregard speaker identity information, which also explains why the text-swapping was not as effective in the case of DHH speakers. Finally, we observe that using data augmentations of the target speaker, even when the quality is low, was still always better than using real recordings of other DHH speakers as shown by the IF - speaker-independent setup results.

In particular, we found that under the same training set size conditions, using imperfect synthetic speech and intermediate fine-tuning can still be successful in improving ASR performance compared to naively fine-tuning the network. However, we note that researchers still need to carefully consider the particularities of their dataset and carefully apply the different proposed methods. For example, while both types of speech are relatively unintelligible, EL speech does not contain any sort of speaker identity, while deaf and hard-of-hearing speech still retains the original speaker identity. Thus, such considerations need to be properly examined before applying the proposed techniques to other datasets.

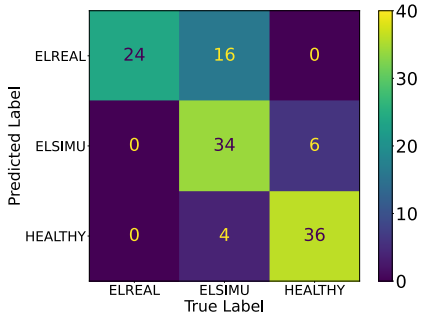
As a final observation, although we verified the validity of the intermediate fine-tuning in the low-resource scenario, we still found that it cannot beat the performance when using real recordings of the target speaker as the training data. By comparing the Naive and IF-TTS setups, results show that using imperfect synthetic speech only became successful in the 200 utterance setup as there was a larger CER difference between using the two setups; however, as more real data was collected and used during training, the effects of using imperfect synthetic speech became close to none. Thus, although we acknowledge the difficulty of obtaining training data from the target



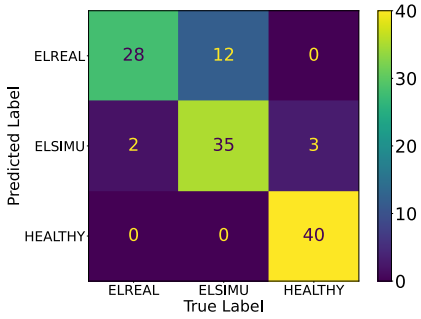
(a) Text-to-speech setup



(b) Voice conversion setup



(c) Text-swapping setup



(d) Text-randomization setup

Figure 3.8: Confusion matrices of categorizing from ELREAL, ELSIMU1, and HEALTHY when using the latent spaces of the target fine-tuning task, shown with different intermediate fine-tuning setups.

speaker, we urge researchers to collect at least 1000 utterances to easily develop efficient personalized ASR systems, and such that they would not need to do any additional procedures such as intermediate fine-tuning and speech synthesis to improve the ASR models. On the other hand, in cases where collecting data is difficult, researchers can still opt to use the intermediate fine-tuning procedure as an alternative, as it has been verified to work even on other types of atypical speech.

This confirms our previous hypothesis in Chapter 3.5.1, where we mention that the

effectiveness of intermediate fine-tuning was less, primarily because the domain gap between the large-scale healthy speech and ELSIMU1 was smaller than that of the large-scale healthy speech and ELREAL. As a conclusion, this also gives an idea on why EL ASR is a difficult task, as simply conducting experiments by simulating the voicing characteristics would not always translate to results with actual EL speakers, thus calling the need to invest more time in collecting data from atypical speakers to further advance the research progress in this problem.

3.6 Conclusions

With ASR providing several conveniences to atypical speech patients, more work should be done to make commercial ASR devices useful for them. We investigated two types of SSL-based pretraining frameworks, contrastive and masked region prediction, on two different atypical speech datasets and compared them with a supervised pretraining framework. Although SSL pretraining frameworks have shown success with minimally resourced healthy speech, this does not seem to be the case with atypical speech. Thus, further investigations in improving the training strategy can be conducted to improve performance in SSL pretraining.

Aside from this, we established that supervised pretraining can still be a strong baseline in atypical ASR. One behavior we notice is that the large-scale networks have difficulties in transcribing the UASpeech dataset compared to the ELSpeech dataset. This is primarily due to the fact that UASpeech only contains word utterances, which makes it difficult for large networks to converge on. The setups used in this study primarily relied on pretraining data composed of sentence utterances to train both the acoustic and language models. Thus, during fine-tuning, the network had a larger difficulty due to the mismatch in both acoustic features (such as the difference in

temporal structure in dysarthric speech) as well as the grammar differences (sentence vs word utterances).

In addition, we presented an elaborate study on the use of imperfectly synthesized speech data as training data for an intermediate fine-tuning task. As recognition accuracies of ASR models have been limited due to the domain shift between the pretraining and fine-tuning data, we aimed to minimize this problem and narrow the gap between the two datasets by introducing the intermediate fine-tuning task. Our results showed a huge performance boost when using the intermediate fine-tuning method over methods that did not use this. Furthermore, a detailed analysis showed that the intermediate fine-tuning does not focus on learning the linguistic content as an ASR task would, but rather focuses more on learning the voicing characteristics of the EL speaker. This was proven in experiments that involve speaker-related downstream tasks, such as resynthesizing waveforms, categorizing speakers, and in a setup where using a speaker ID loss. Thus, we show the effectiveness of the intermediate fine-tuning method for low-resourced EL ASR tasks. Finally, we also showed the differences between using a simulated EL dataset compared to that of a real EL speaker, showing the need for collecting real EL speaker datasets in this research field. Although it is quite possible to ask healthy speakers to record their voices speaking in an electrolarynx, there are still many pronunciation differences that were not recognizable by the human ear, but seemed to be quite obvious to a neural network. Moreover, it also takes skill and practice to effectively use an electrolarynx, which means that if healthy speakers were to collect data using this method, some practice would be required to properly produce intelligible speech.

Chapter 4

Electrolaryngeal Speech Enhancement

In this chapter, we investigate the approaches to improve electrolaryngeal speech enhancement. In particular, we aim to resolve two domain mismatches between healthy and EL speech, particularly in the acoustic and temporal domains. In particular, we study the use of a decomposed network to remove acoustic domain mismatches and improve synthesis performance. Then, we investigate the use of linguistic intermediates to remove temporal domain mismatches.

4.1 Introduction

Voice conversion (VC) [89], the task known as changing the speaker information while keeping linguistic information unchanged, has had rapid improvements in the age of deep learning. One of its sub-applications, atypical speech enhancement[1] [85], [90], [91], has made way for atypical speakers to regain the ability to speak like typical

[1]Note that speech enhancement in this work refers not only to enhancing the recording quality, but enhancing the atypical speech itself.

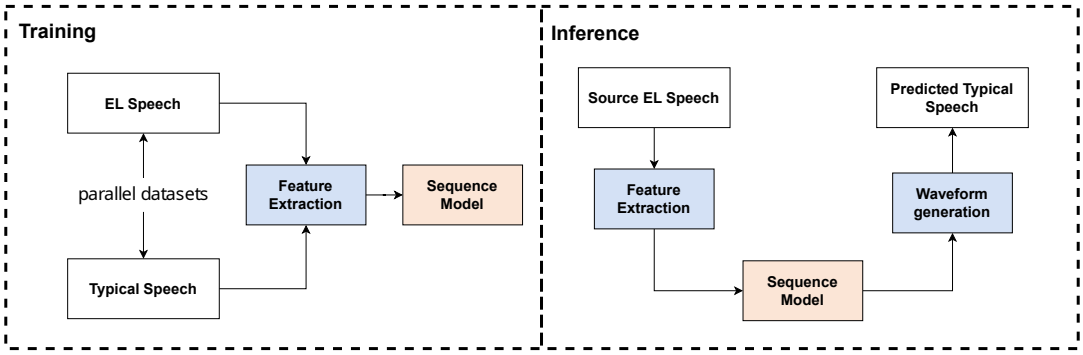


Figure 4.1: An overview of the conventional approach in the EL speech enhancement task. A parallel dataset of the source EL and target typical speakers are used to train a sequence-based network. During inference, the source EL acoustic features are passed into the trained sequence-based network to convert it into acoustic features that of intelligible speech.

speakers.

A conventional approach to the EL speech enhancement task is to use a parallel dataset and learn the temporal alignments between typical and EL speech. A visualization of this approach is shown in Fig. 4.1. Sequence models have generally been used in this task due to its ability to jointly convert multiple speech characteristics, such as the timbre and temporal structure. Previous works such as [57] used a strong sequence-based model such as a Transformer [11], [60] to model the temporal structure of the typical speech. With the data-hungry nature of sequence models, other works such as [59], [91], [92] focused on the use of large-scale synthetic speech and found that it greatly improves speech enhancement performance.

In all of these previous works, the general theme is to resolve the domain mismatch between typical and EL speech, which is what causes the most problems and makes this task difficult, through pretraining and data augmentation. There are two main

types of domain mismatches found in EL and typical speech. First, there are speech and speaker mismatches between the two. Since EL speech lacks source excitation and cannot produce some unvoiced sounds that are present in typical speech, transforming EL speech into typical speech requires the network to also generate the pitch-related features such as prosody and intonation, making the task difficult. Second, there is also temporal structure mismatch between the EL and typical speech. Since EL speakers speak at a slower rate and cannot pronounce some phonemes [32], the network also needs to learn how to map the EL speech speaking rate into that of typical speech and improve the pronunciation. These domain mismatches, combined with only low-resourced dataset available, are what make this task very difficult to resolve.

This study first addresses the acoustic domain mismatch which is commonly encountered in pretraining and fine-tuning approaches. Inspired by the recognition-synthesis [93] framework, which used strong recognition modules to extract dense linguistic information, (in this work, bottleneck features (BNF) from an output of a speech recognition encoder) [94] and HuBERT [63], [95] layer embedding intermediates. These two features are then used as input and output features of an alignment module, which resolves the temporal structure differences between the source EL speaker and target typical speaker. As Chapter 3 focuses on building a speech recognition system, we build on this and use this system to extract strong linguistic features from EL speech audio. Compared to traditional acoustic features such as mel-spectrograms, the recognition module extracts pure linguistic-related information and removes speaker-related information. Thus, we effectively allow the alignment module to focus solely on modeling the temporal structure of typical speech, resulting in better performance.

Then, we address the second domain mismatch of temporal information mismatches between the EL and typical speech. Specifically, we investigate the use of discrete

linguistic intermediates, by using discrete linguistic information (such as text) to improve BNF intermediates. Compared to audio information, discrete text information does not contain any temporal structure. Similar to the aforementioned findings in [62], normalizing the temporal length by using discrete linguistic information is vital in the speech enhancement task. Thus, the motivation of using discrete linguistic intermediates is to find the most effective way of removing as much unnecessary temporal information from the BNFs as possible by only using the frames which correspond to linguistic information. This reduces the burden on the alignment module, as it would no longer need to detect which frames to ignore during the conversion process. In particular, we investigate the following types of discrete linguistic intermediates: CTC-based intermediates, phoneme-level intermediates, and learnable discrete intermediates.

4.2 Bottleneck Feature Intermediates Framework

We first discuss the proposed method to resolve the acoustic domain mismatches. Our proposed framework to remove speech type and speaker information uses bottleneck feature intermediates, which builds on the Transformer-based Parallel VC system [59] described in Chapter 2.4.2. The proposed method is conducted through a three-stage framework, consisting of recognition, alignment, and synthesis modules. An overview of the entire framework can be seen in Fig. 4.2. We detail the task of each module of the proposed framework below.

4.2.1 Problem formulation

Given a parallel VC dataset be $\text{VC}_{\text{parallel}} = \{\mathbf{S}_{\text{src}}, \mathbf{S}_{\text{trg}}\}$, where $\mathbf{S}_{\text{src}}$ and $\mathbf{S}_{\text{trg}}$ denote the source, target speech, respectively, a sequence model aims to learn the relationship

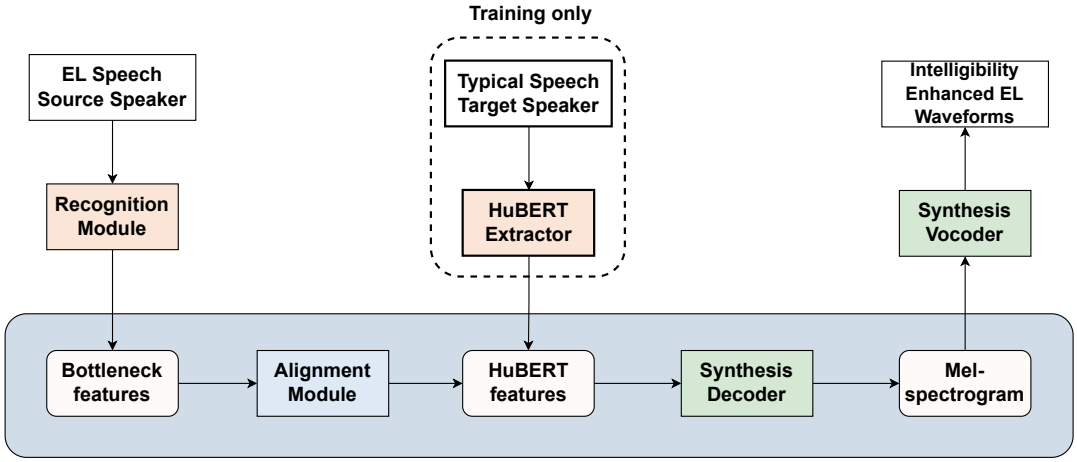


Figure 4.2: An overview of the proposed framework using bottleneck feature intermediates to remove speech type and speaker normalization. The framework contains three main modules to convert from EL to typical speech: the recognition, alignment, and synthesis modules. Note that each module is trained separately.

between these two datasets. In general, a sequence model aims to learn the mapping between a source feature sequence $\mathbf{X} = \mathbf{x}_{1:n} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ and a target feature sequence $\mathbf{Y} = \mathbf{y}_{1:m} = (\mathbf{y}_1, \ldots, \mathbf{y}_m)$. These features are are often of different length, i.e, $n \neq m$ [19]. This length mismatch is especially seen in the EL speech enhancement problem, as EL speakers often have trouble speaking and result in a slower speaking rate compared to healthy speakers, as mentioned in Chapter 2.1.

In traditional VC, a one-dimensional speech signal is commonly represented instead as a sequence of acoustic feature vectors. This makes it easier for neural networks to learn patterns, as learning the patterns inside a one-dimensional signal is a very difficult task (i.e., in a 16 kHz sampling rate, the network needs to correctly learn to predict 16,000 samples per second). Here, each acoustic feature vector is calculated from a speech frame given by a pre-defined frame size and frame shift. In traditional VC and

previous works on EL enhancememt, a commonly adopted acoustic feature is the log mel-spectrogram [19], [59]. However, the main problem with log mel-spectrograms is that it contains both acoustic and linguistic features. Although it is not usually a problem in a healthy VC setup, this becomes a problem in EL speech enhancement, due to the mismatches discussed in Chapter 1.3. Thus, it can be easy to see why performance degradation is present when using log mel-spectrograms as source and target features, as the network needs to both generate the acoustic features and enhance the intelligibility of the linguistic information. The goal of the chapter of this thesis is to discover how to further decompose the framework into different networks focusing on dedicated tasks, and discover which feature can be used to replace log mel-spectrograms.

4.2.2 Recognition module

The recognition module uses a linguistic encoder, which uses the bottleneck features (BNFs) [2] from an ASR encoder to extract the linguistic information. In Chapter 3.4, we showed that an effective approach to improving ASR model performance for EL speech is through a three-stage training framework. First, the model was pretrained on a large-scale typical speech dataset. Next, we fine-tuned the network on synthesized EL speech in an intermediate fine-tuning stage. Due to the limited data in training an EL speech synthesis model, the synthesized speech also contained lots of mispronunciations. Conducting this three-stage framework resulted in better performance for the model. However, results in Chapter 3.5.2 found that since the model used this stage to learn the EL speech characteristics instead of the linguistic information, it was sufficient enough for the imperfectly synthesized EL speech to only represent the EL speech

[2] Although the use of self-supervised features have been successful in typical VC tasks, we discussed these in Chapter 3 to be ineffective in decoding EL speech.

features. Finally, we fine-tuned the network on the ground truth EL speech to learn the linguistic features and decode at a high accuracy. We adopt this framework as the backbone of the recognition module.

The goal of the linguistic encoder now is to be robust enough to remove the speech type features from both typical and EL speech, while also accurately extracting linguistic information. With a unified representation, the performance of a pretraining and fine-tuning framework becomes robust to the speech type mismatches. Although this has been an easy task in typical VC, several ASR works have found developing speaker-independent models [33], [35], [96], [97] for atypical speakers a difficult task due to the high variance in their speech. A naive approach to resolve this would be to simply fine-tune the ASR model on both the EL and typical speech such that the model is not only optimized for EL speech. However, similar to previous works, we found that fine-tuning the ASR model on both types of speech at the same time causes performance degradations.

We further improve the ASR model by using the speech type loss described in Chapter 3.5.4. As findings discovered that the intermediate fine-tuning focuses on learning speech type identity features, we make the network learn both speech types during this stage. Through this approach, we effectively optimize the ASR model for EL speech, while also ensuring that it does not forget how to decode typical speech. We show in Eq. 4.1 the detailed loss calculation during intermediate fine-tuning. After the intermediate fine-tuning stage, we fine-tune on X_{EL} with the CTC/Attention losses L_{ctc} and L_{attn} [14] as usual.

$$L_{\text{ASR}} = L_{\text{SID}}(X) + L_{\text{ctc}}(X_{EL}) + L_{\text{attn}}(X_{EL}) \tag{4.1}$$

4.2.3 Alignment module

The alignment module models the temporal structure from atypical to typical speech. To improve intelligibility, the alignment module needs to fulfill two tasks. First, due to the temporal structure of EL speech, the model needs to convert the speaking rate similar to a typical speaker. Next, since EL speakers cannot produce certain phonemes, the alignment module also needs to correct the phoneme pronunciation. Similar to the baseline described in Chapter 2.4.2, we adopt the use of a Transformer [11], [60] sequence-to-sequence model to resolve these issues. Due to its success, we also adopt the same fine-tuning procedure with synthetic data and then the target data.

We improve this framework by using the BNFs produced by the recognition module as the inputs. These BNFs would further reduce the mismatches in speech type and speakers during pretraining and fine-tuning, as the linguistic encoder allows the alignment module to solely focus on modeling linguistic information. To further reduce the burden on the alignment network, we also use HuBERT layer embeddings as the output features. Aside from HuBERT providing dense linguistic information, using a variant of HuBERT with soft features [95] has also been found successful in removing speaker features and in cross-lingual settings, which would further improve the performance of the synthesis module described later. Moreover, the TTS/AE pretraining described in Chapter 2.4.2 was initially used in our baseline due to the unavailability of a large-scale parallel dataset; however, with the release of [98], we first verify whether parallel VC is indeed better. Although using parallel VC pretraining would directly model the fine-tuning task, this would not contain the speaker-independent properties of the TTS/AE pretraining, which might cause more degradations in the multiple fine-tuning stages due to the speech type and speaker mismatches. However, owing to the proposed framework focusing solely on linguistic features, we hypothetically remove

this possibility.

4.2.4 Synthesis module

As our goal is to force the alignment module to focus only on modeling linguistic information, the task of synthesizing into the target speaker is placed on a synthesis module. Since the typical dataset used as a target speaker is also limited in size, we use a Diffusion model [29] for this module, as this framework has been proven effective in synthesizing speech in a target speaker even in few-shot settings [99]. To improve the few-shot performance of the Diffusion model, similar to [99], we first pretrain on a large-scale multi-speaker dataset with classifier-free guidance [100] and use fixed speaker embeddings and HuBERT layer embedding intermediates as conditioning features. Then, we adapt the model to the few-shot data for another set of iterations. To train the model, we iteratively add noise for N timesteps to the mel-spectrogram and predict the noise at timestep n during training by using the noisy mel-spectrogram at $n-1$ as input along with the conditioning features. During inference, we pass in Gaussian noise and predict the mel-spectrogram after N iterations. Finally, to synthesize the audio waveforms from the predicted mel-spectrograms, we use HiFiGAN (V1) [20] as the vocoder.

4.3 Discrete Linguistic Intermediates Approaches

Next, we discuss the limitations of this framework and address how to resolve the temporal domain mismatches. Since the previous approaches rely on the Parallel VC framework, these directly model the temporal structure of the typical speech from the atypical speech. Unfortunately, aside from the speaker mismatches, there is also a large

Table 4.1: An overview of the main differences of each proposed method.

Chapter	Linguistic intermediate	Parallel data	Alignment modeling	Jointly learned recognition
§4.2	Bottleneck features	✓	Autoregressive	✗
§4.3.1	Bottleneck features	✓	Non-autoregressive	✗
§4.3.2	Bottleneck + CTC-decoded	✓	Autoregressive	✗
§4.3.3	Discrete text	✗	Non-autoregressive	✗
§4.3.4	Discrete text + bottleneck	✓	Non-autoregressive	✗
§4.3.5	Discrete text + bottleneck	✓	Non-autoregressive	✓

mismatch in temporal structure between the source EL and target typical speech, which places a huge burden on the alignment module. A solution to resolve this mismatch is through discrete linguistic intermediates, which normalize the BNF intermediates' temporal structure based on certain prior information[3]. Compared to audio, discrete linguistic information such as text do not have any temporal structure. Thus, by using discrete linguistic intermediates such as text information, the lengths of the source EL speech can be normalized, and consequently reduce the temporal structure mismatches, reducing the burden on the alignment module. This can be done in multiple ways as overviewed in Table 4.1, where each method is described in detail in the following chapters.

4.3.1 Duration predictor-based alignment modeling

We start the investigations with a simple experiment on the type of sequence modeling technique used in the alignment module, specifically between autoregressive (AR)

[3]Note that discrete linguistic information here does not refer to the discrete representations with a finite codebook size, but instead using discrete prior information such as text to normalize the temporal structure.

and non-autoregressive (NAR). In the baseline work [59], an AR modeling technique using the Transformer module [11] was adopted. However, compared to AR modeling, NAR modeling is more robust to mispronunciations as it reduces the error propagation made from the earlier steps through a duration predictor, by simply specifying by how many times each frame will be expanded or which would be deleted. We investigate whether there are significant differences when replacing the alignment module with a NAR network. In particular, we adopt the monotonic alignment search proposed in [101]–[103]. This uses an unsupervised learning method that rapidly aligns source and target features without the need for external aligners. The approach combines the Viterbi and forward-backward algorithms from Hidden Markov Models to extract hard and soft alignments, respectively. The soft alignment matrix A_{soft}, normalized across the source feature dimension, represents the probability distribution of aligning source frames to target frames. To promote monotonic alignment, a Beta-Binomial prior encourages a near-diagonal path in the alignment matrix. Hard alignments A_{hard} are extracted using the Viterbi algorithm, which enforces a binarized, monotonic alignment, ensuring each frame is assigned to a single text token. To reduce the train-test domain gap, the model is conditioned on A_{hard}, and a KL-divergence loss minimizes the difference between A_{soft} and A_{hard}.

4.3.2 CTC-based linguistic intermediates

Another approach in normalizing BNFs is by using the CTC module from the recognition module as visualized in Fig. 4.3. Since the recognition module is trained with a CTC-Attention loss, the resulting CTC module can be used to remove frames in the BNF that correspond to `<blank>` tokens. By removing frames corresponding to `<blank>` tokens, which are not always relevant to linguistic-related information, the

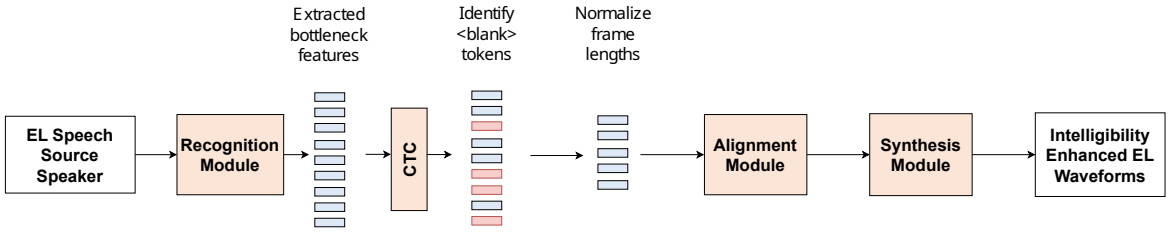


Figure 4.3: CTC-based intermediates. A pretrained CTC decoder produces discrete outputs which are used to remove blank frames from the bottleneck features before being passed as inputs to the alignment module.

temporal dimension of EL speech can be normalized. To vary the normalization strategy, we further explore the impact of varying the context window size on our model's performance. Specifically, for each non-`<blank>` frame in the sequence, we retain all frames within a window size of n frames on either side, such that the $2n+1$ frames surrounding a non-blank frame are preserved. Eq. 4.2 further explains this process, where x_j represents a binary array where each value corresponds to the frame indices classified as non-`<blank>` tokens by the CTC. The output y_i is the modified binary array that factors in the context window to retain more frames surrounding any non-`<blank>` frame. However, a possible drawback of this method is that the normalization process is entirely rule-based (as the CTC module is not trained together with the alignment module), which may lead to a rigid normalization process.

$$y_i = \begin{cases} 1 & \text{if } \exists\ j \in [i-n, i+n] \text{ such that } x_j = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{4.2}$$

4.3.3 Phoneme-level intermediates

Another approach is to adopt the ASR+TTS approach discussed in Chapter 2.4.1 to extract phoneme-level information, and subsequently directly synthesize the typical speech from these. As seen in the top half of Fig. 4.4, BNF normalization is then done by using the trained ASR decoder from the recognition module to decode discrete text, converting the decoded text to phonemes using a rule-based dictionary[4], and subsequently using a TTS module (which substitutes as the alignment and synthesis modules in the framework proposed in Chapter 4.2) to synthesize the decoded text. The continuous BNFs are now completely disentangled from the EL temporal information from the source speech and model the typical speaker's temporal information from the phoneme-level.

4.3.4 Phoneme-level intermediates w/ source BNFs

Although the use of phoneme-level intermediates could possibly help reduce prosody errors, similar to CTC-based linguistic intermediates, a possible drawback of this method is that if the ASR performance for EL speakers is not at a high level, the mispronunciation errors will be propagated to the synthesized speech. It is also possible that some prosodic features that are relevant from the source speaker will contain necessary information to model the typical speech prosody. Moreover, as discussed in Chapter 2.4.2, a downside of a cascaded ASR+TTS framework is that it would be harder to predict the pronunciation of characters directly from text, which could possibly degrade synthesis performance in cases where the pronunciation is changed.

Due to these issues, we further explore experiments where we investigate the effec-

[4] `http://open-jtalk.sp.nitech.ac.jp/`

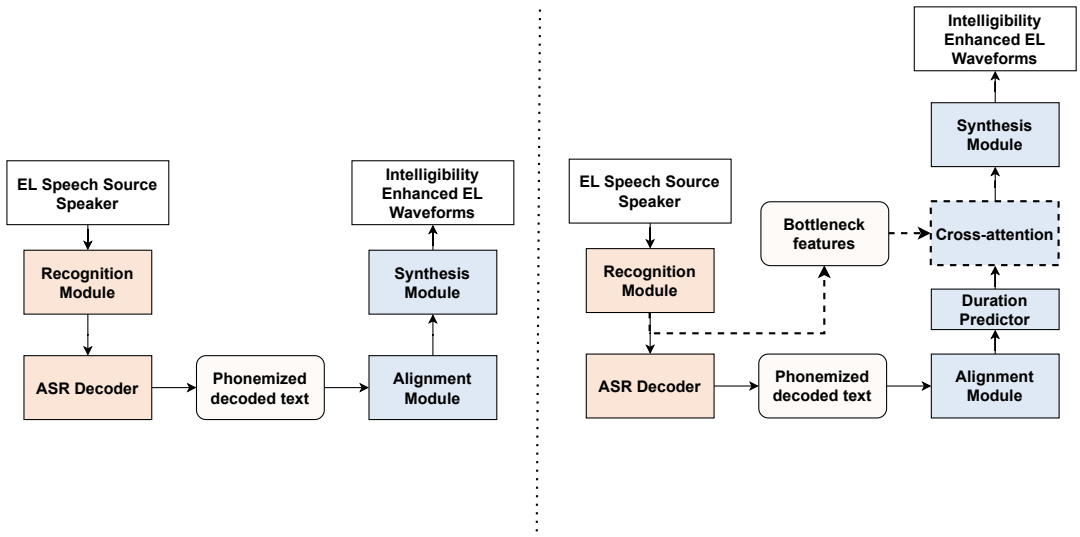


Figure 4.4: Phoneme-level intermediates. The top half of the figure shows a pretrained ASR decoder transcribing text from the input speech, which is converted to phonemes, and subsequently passed to the alignment module. The bottom half shows another variation of the aforementioned setup where the outputs of the alignment module is conditioned with the source BNFs using cross-attention.

tiveness of using both BNFs and the decoded phoneme-level intermediates to condition the TTS module when reconstructing the target typical speaker, and verify whether these could possibly alleviate the insufficient performance in EL ASR systems. To do this, we condition the duration expanded text features with the source BNFs using cross-attention as seen in the bottom half of Fig. 4.4. By using the attention module, this allows the model to jointly learn which frames from the source BNF are relevant in synthesizing the target typical speech in a data-driven manner.

4.3.5 Learnable discretization

Finally, we investigate the use of a learnable discretization on the BNFs. Although similar to the architecture described in Chapter 4.3.3, the main difference of this is that the recognition decoder (through a linear layer) and the synthesis modules are jointly trained in a single network, whereas the architecture in Chapter 4.3.3 uses a fixed recognition decoder. The main advantage of the learnable recognition decoder is that it allows the outputs to be adapted to the synthesis module in a data-driven manner, especially since the recognition module still has a high CER when decoding EL speech. An example of this is the work in [62], which has also shown that dysarthric speech can be normalized by using extracted discretized HuBERT units from typical speech by K-means clustering. We conduct further investigations on EL speech enhancement based on this work.

As seen in Fig. 4.5, we adopt the framework proposed in [62] but make some changes to fit our dataset. We use the BNFs extracted from the ASR encoder, as findings in Chapter 3.4 of HuBERT features not representing linguistic information in EL speech accurately. In the original setup in [62], K-means clustering with 1000 clusters is run on the BNFs of the entire typical speech dataset. In our experiments, we simply use the ground truth text phonemes instead of discretizing K-means clustering. During training, we discretize the BNFs from the typical speech. We explain this choice later in Chapter 4.5.5. The BNFs from the EL speech are also passed through a linear layer and through CTC loss. The discretized BNFs from the typical speech are then passed on to a TTS model to model the typical speech. This framework allows the model to jointly learn how to the discretize units of the EL speech into that of typical speech, while also synthesizing the target speech. We use the same TTS model as used in Chapter 4.3.3 to synthesize the decoded text into waveforms. Similar to an important

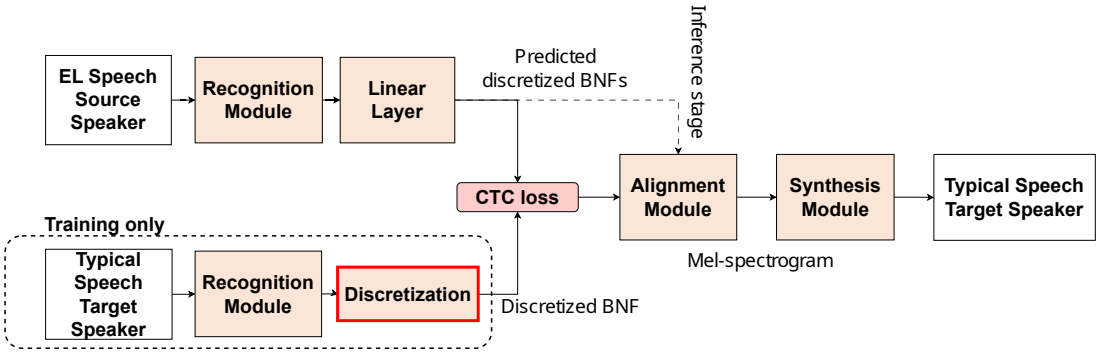


Figure 4.5: Learnable discretized intermediates through a CTC loss based on [62]. The discretization process is through a phoneme-discretization (acquired from the ground truth text). During training, the typical speech BNFs are discretized and the network learns to discretize EL BNFs using a CTC loss. During inference, the EL speech is discretized using the learned model to synthesize the typical speech.

point noted in the original implementation in [62], we also apply pretraining through the large-scale parallel dataset and synthetic data.

4.4 Experimental Settings

4.4.1 Evaluation metrics

For objective evaluations on the initial experiments investigating BNF intermediates, we focus on speaker-related assessment by measuring the synthesis quality through metrics such as CER, mel-cepstral distortion (MCD), log F0 root mean square error (F0 RMSE), and log F0 correlation (F0 CORR). On the other hand, for objective evaluations for experiments using discrete linguistic intermediates, we focus on the temporal structure-related assessment, by measuring the duration difference (DDUR), CER, and phoneme error rate (PER), and a neural-based mean opinion score (UTMOS)

[104]. For both CER and PER, we used the same Conformer model in Table 4.3 trained on the large-scale typical speech, and converted the decoded text into phonemes to calculate PER.

For subjective evaluations of experiments using BNF intermediates, we recruited 15 native Japanese speakers to measure the naturalness of the synthesized speech using a 5-scale mean opinion score (MOS) test. For the subjective evaluations of discrete linguistic intermediates, we recruited 21 native Japanese speakers to conduct the same MOS test [5].

4.4.2 Datasets

An overview of all the EL datasets, which are all recorded in Japanese, is outlined in Table 4.2. The overview contains the speaker's biological sex, duration in minutes and number of utterances for each split. The main EL dataset used in this experiment (denoted in the top segment of Table 4.2), is recorded by a real EL patient (ELREAL), along with a parallel counterpart recorded by a typical speaker (TYPICAL).

In order to validate and analyze the behavior of the proposed methods on a larger scale EL corpus, we also conducted ablation studies on the performance on 14 pseudo EL speakers (recorded by typical speakers using an external electrolarynx), which are denoted on the middle segment of Table 4.2 as EL PS. To make evaluation fair, we convert all EL PS data to a single parallel counterpart spoken by a typical speaker (TYPICAL SET 2), which is the same speaker as TYPICAL. We follow the same train/dev/test split released [6], which used a 116/40/40 split for the train, dev, and test data. It is worthwhile to note that these simulated speakers have little to no prior

[5] Demo page: `https://lesterphillip.github.io/elsie_extended`

[6] `https://github.com/k2kobayashi/PES-corpus/blob/main/README_en.md`

Table 4.2: Total duration (in minutes) and number of utterances of each EL speaker used in the experiments.

	Minutes			**No. of Utterances**		
Dataset	**Train**	**Dev**	**Test**	**Train**	**Dev**	**Test**
EL REAL	5.77	2.96	2.99	116	40	40
TYPICAL	4.38	2.14	2.17	116	40	40
EL PS FEMALE001	10.54	0.76	3.63	140	10	50
EL PS MALE001	9.82	0.69	3.39	140	10	50
EL PS MALE002	14.38	1.03	5.05	140	10	50
EL PS MALE003	9.02	0.63	3.03	140	10	50
EL PS MALE004	9.78	0.71	3.16	140	10	50
EL PS MALE005	11.34	0.82	4.16	140	10	50
EL PS MALE006	10.69	0.80	3.66	140	10	50
EL PS MALE007	10.76	0.80	3.82	140	10	50
EL PS MALE008	11.71	0.87	4.06	140	10	50
EL PS MALE009	11.12	0.82	4.15	140	10	50
EL PS MALE010	11.31	0.86	3.84	140	10	50
EL PS MALE011	11.93	0.91	4.26	140	10	50
EL PS MALE012	9.89	0.67	3.29	140	10	50
EL PS MALE013	9.99	0.73	3.48	140	10	50
TYPICAL SET 2	8.35	0.61	2.74	140	10	50
EL REAL FEMALE001	2.94	1.15	0.71	30	10	10
EL REAL MALE001	1.96	0.80	0.45	30	10	10
EL REAL MALE002	2.54	0.98	0.66	30	10	10
TYPICAL SET 3	1.69	0.64	0.41	30	10	10

experience in using an external electrolarynx, thus compared to a real EL patient, the intelligibility of the recordings may vary based on their experience. Moreover, as found in the results of Chapter 3.5.5, the differences in pronunciation by simulated EL speakers with real EL speakers also need to be considered when checking the results.

Finally, we further validate and analyze the behavior of the proposed methods on another set of real-world EL speakers. As it was difficult to ask the patients to record a large-scale dataset, the total number of utterances was only limited to 50. To make evaluation fair, we convert all EL PS data to a single parallel counterpart spoken by a typical speaker (TYPICAL SET 3), which is the same speaker as TYPICAL and TYPICAL SET 2.

We now outline the pretraining and finetuning datasets used in each module. For the recognition module, we followed the same training framework as in Chapter 3.3 to train a linguistic encoder. We first pretrained on LaboroTV, a large-scale typical speech dataset containing around 2k hours of speech data [81]. Next, we fine-tuned the network on a total of 27k utterances of synthetic EL data (generated using a text-to-speech system) and typical speech. Finally, we fine-tuned the network on our self-acquired EL REAL speech data.

For the alignment module, we first pretrained our network on HiFiCaptain [98], a large-scale parallel dataset of typical speakers, containing around 18k utterances in total. We used the female speaker as the source and the male speaker as the target. Then, we used the same setup as in [59] where we first fine-tuned on synthetic EL (generated using a text-to-speech system), synthesized from text from the JSUT [88] corpus. We then fine-tuned the model on the parallel EL and TYPICAL speech data in Table 4.2. Note that compared to the baseline system in [59], this current split is different, as we composed the evaluation data with longer utterances to show the

Table 4.3: Resulting CER on both EL REAL and TYPICAL speech with different training data setups.

Training data and method	**EL**	**TYPICAL**
Speaker-dependent on typical speech	77.1	4.3
Speaker-dependent on target EL speech (Chap. 3.3)	15.9	61.5
Fine-tuned on multiple EL and typical	28.7	3.8
Fine-tuned on EL and typical speech	18.1	16.6
Proposed fine-tuned on EL and typical with speech type ID loss only	16.2	6.8
Proposed fine-tuned on EL and typical with speech type ID loss and masking	**13.8**	**6.0**

effectiveness of the proposed method. We used the same 116/40/40 split used in the recognition module, so the test data is unseen by both all the modules.

For the synthesis decoder, we used the JVS dataset [105], a dataset containing 30 hours of speech from 100 speakers, to pretrain the model before fine-tuning it on the TYPICAL speech. For the synthesis vocoder, we used a pretrained model on VCTK [106], an English dataset with 44 speakers of around 40 hours in total.

4.4.3 Model architecture

The recognition model used a Conformer architecture [12], which has 12 layers for both the encoder and decoder, the same as in Chapter 3.3. On the other hand, the Transformer model in the alignment module has six layers for both the encoder and decoder, the same as in [59]. The diffusion model of the synthesis decoder was adopted from [107] and uses 512-dimension channels to predict the noise between each timestep.

To handle speech inputs, we replaced the text encoder with the BNF encoder as the conditioning features. To integrate speaker information, we used a pretrained WavLM model[7] (which was fine-tuned for speaker verification) as speaker embeddings and fused it to each residual block using conditional layer normalization [108]. We set the number of diffusion steps N to 100. No changes were made in HiFiGAN (V1). For experiments using discrete text inputs, we used Matcha-TTS [109] as the alignment and synthesis module due to its strong generative modeling.

Table 4.4: Objective and subjective evaluation results on the synthesized speech from different systems, along with the ground truth recorded speech. MOS is calculated with a 95% confidence interval. We detail the input and output features, along with the pretraining method used in the alignment module.

(Sys.) Description	**Inputs**	**Outputs**	**Pretraining**	**MCD (↓)**	**CER (↓)**	**F0 RMSE (↓)**	**F0 CORR (↑)**	**MOS (↑)**
(1) Baseline [59]	Mel	Mel	TTS/AE	7.78	35.0	51.19	0.30	2.42 ± 0.17
(2) Baseline ablation	Mel	Mel	Parallel VC	7.70	33.5	**49.95**	0.35	2.38 ± 0.18
(3) Proposed BNF inter.	**BNF**	**HuBERT**	**Parallel VC**	**7.14**	**19.0**	52.41	0.29	**3.25 ± 0.15**
(4) Pretraining ablation	BNF	HuBERT	TTS/AE	7.45	32.2	50.39	0.28	2.90 ± 0.17
(5) Feature ablation	BNF	Mel	Parallel VC	7.54	29.1	51.16	**0.37**	2.78 ± 0.18
Ground truth	-	-	-	-	4.3	-	-	4.85 ± 0.07

4.5 Results and Discussion

4.5.1 Validation of the recognition module

We first present how to develop a robust linguistic encoder using the speech recognition model discussed in Chapter 3. We investigate different training setups as shown in Table 4.3. First, we see the difficulty in using a speaker-independent model, as

[7] `https://huggingface.co/microsoft/wavlm-base-plus-sv`

Table 4.5: Objective and subjective evaluation results on the synthesized speech to compare the use of BNF intermediates with mel-spectrograms, along with the ground truth recorded speech. MOS is calculated with a 95% confidence interval. We detail the type of linguistic intermediate used in the experiment. Note that Sys. 3 is the same system from Table 4.4. Both Sys. 7 and Sys. 11 were omitted from the MOS test due to low objective scores.

(Sys.) Description	**CER (↓)**	**PER (↓)**	**DDUR (↓)**	**UTMOS (↑)**	**MOS (↑)**
(3) BNF intermediates	19.0	8.4	0.67	3.34	2.77 ± 0.15
(6) NAR-based alignment	20.0	9.1	0.60	2.89	1.95 ± 0.13
(7) CTC-based inter. (n=0)	47.1	40.9	1.11	3.43	-
(8) CTC-based inter. (n=2)	21.5	9.1	**0.32**	3.44	2.58 ± 0.16
(9) Phoneme-level inter.	**17.6**	6.9	0.89	3.43	2.97 ± 0.15
(10) Phoneme-level inter. w/ source BNF	18.8	**6.1**	0.85	3.43	2.86 ± 0.16
(11) Learnable discretized inter.	45.3	21.8	0.52	**3.47**	-
Ground truth	4.3	0.0	-	3.76	4.72 ± 0.08

optimizing on either EL or typical speech results in degradations in the other. Thus, using a model optimized on just EL speech degrades the large-scale pretraining stage on typical speech. Next, fine-tuning on multiple EL and typical speakers yields no measurable performance gains, which is caused by the high variance between these speakers. Finally, we show that simply fine-tuning the model on both EL and typical speech can be effective but not fully optimized, as there is still a performance gap from the speaker-dependent setups.

We show that our proposed method of using a speech type ID loss and masking the typical speech during CTC-Attention loss calculation makes the model learn to decode both EL and typical speech. This is because the model learns how to decode EL speech, while also not forgetting typical speech features learned during pretraining through the speech type ID loss. To verify this, removing the masking of typical speech results in

slightly worse scores. Through this, we can decode both EL and typical speech at an accuracy similar to the speaker-dependent setups.

4.5.2 Comparison of input/output features

We show the effectiveness of the proposed linguistic encoder in this task. We first compare the proposed system with the Transformer-based Parallel VC system [59] described in Chapter 2.4.2. As seen in Table 4.4 our proposed method of using the BNF/HuBERT features (Sys. 3) can significantly improve the synthesized speech with a 16% improvement in CER and 0.83 in naturalness score over Sys. 1, the baseline that uses mel-spectrograms as inputs. This proves our initial hypothesis that the proposed linguistic encoder can effectively remove speech type information while also extracting accurate linguistic information. We also conducted a study by using mel-spectrogram outputs. As shown in Sys. 5, using HuBERT instead of mel-spectrograms as outputs helps further stabilize the model, as it also contains dense linguistic information similar to the BNFs. Aside from this, Sys. 3 and 4 that used HuBERT features and the synthesis decoder had the top naturalness scores, further showing the effectiveness and necessity of a synthesis decoder over directly predicting the mel-spectrogram.

4.5.3 Comparison of pretraining techniques

In Chapter 4.2.3, we discussed that the TTS/AE pretraining also helps in resolving the speech type and speaker mismatches during pretraining and fine-tuning through its speaker-independent pretraining style [60]. However, upon comparing the baseline techniques, we observe Sys. 2 to have slightly better scores than Sys. 1 except in MOS. Thus, we prove that TTS/AE is not sufficient to create a speaker-independent

property. Through the proposed approach in Sys. 3, we can directly model the fine-tuning task by using parallel VC pretraining, while also being able to implement a speaker-independent property by using BNF/HuBERT as input and output features, which reduces the mismatches during each fine-tuning stage. It is important to note that although Sys. 4 used both speaker-independent training styles, since the input (BNF) and output (HuBERT) features were different, the proposed method was not able to fully utilize the effectiveness of AE pretraining. Finally, we find that compared to the other systems, Sys. 3 has the highest F0 RMSE score and the second lowest F0 CORR score, showing that the proposed method truly allowed the alignment module to focus on modeling linguistic information, but caused a small tradeoff in modeling pitch.

4.5.4 Comparison of discrete linguistic intermediates

We extend the use of the BNF intermediates to resolve the domain mismatch in temporal structure. To alleviate the mismatches in temporal structure, we investigate the most effective linguistic intermediate to improve the use of BNFs. The main results are found in Table 4.5 and we use the initially proposed Sys. 3 as the baseline. First, we investigate a simple experiment where we replace the use of autoregressive alignment modeling in Sys. 3 with a non-autoregressive duration predictor, which is described in Sys. 6. We see the non-autoregressive method in Sys. 6 degrades the results in CER, PER, and MOS compared to the initially proposed Sys. 3. One of the main reasons of this degradation is the duration modeling always goes from longer EL utterances to shorter typical speech utterances, whereas the duration predictor used was designed to expand shorter text inputs into longer acoustic features in TTS. Thus, the direct use of non-autoregressive alignment modeling with the vanilla BNFs is not sufficient enough

to improve performance.

We now start investigating the use of discrete linguistic intermediates. We investigate the use of CTC-based linguistic intermediates to remove frames from the BNF intermediates, as described in Sys. 7 and Sys. 8. Unfortunately, compared to Sys. 3, both of these systems are also not effective. We observe that increasing the window size n results in lesser deletions and a much higher PER score, as seen by comparing the results in Sys. 7 and Sys. 8. Thus, this shows that each frame in the BNFs may still be correlated to each other despite corresponding to a `<blank>` token, and removing each frame by solely using the CTC decoder is not an efficient approach since the CTC decoder weights were optimized only during the ASR task.

Next, we investigate the use of phoneme-level intermediates which convert the BNFs to discrete text as described in Sys. 9. Completely removing all prosodic information from the source speech and normalizing it into phoneme-level works better than Sys 3. as seen in the CER, PER, UTMOS, and MOS results. This shows that although the ASR system is imperfect and errors from mistranscriptions are propagated to the synthesis, the resulting speech becomes more natural and unnatural pauses are removed compared to the initially proposed vanilla BNF setup. We note that aside from the mispronunciations in the transcript, the quality improvement is mainly due to the use of a strong generative model in the TTS model to synthesize the speech from the transcribed text.

4.5.5 Investigation of strategies to reduce error propagation from recognition module

Although the phoneme-level intermediates in Sys. 9 results in a much natural prosody, the linguistic information is not entirely accurate as the errors from the ASR

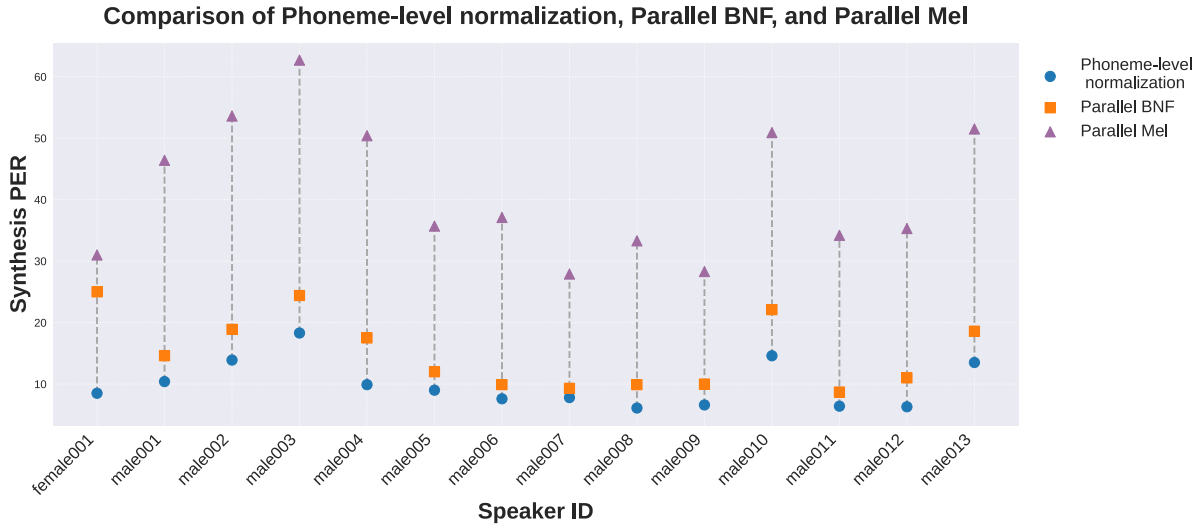


Figure 4.6: Comparison of PER using phoneme-level intermediates, parallel BNF, and parallel mel on a larger pseudo EL corpus. Note that a lower score represents better performance.

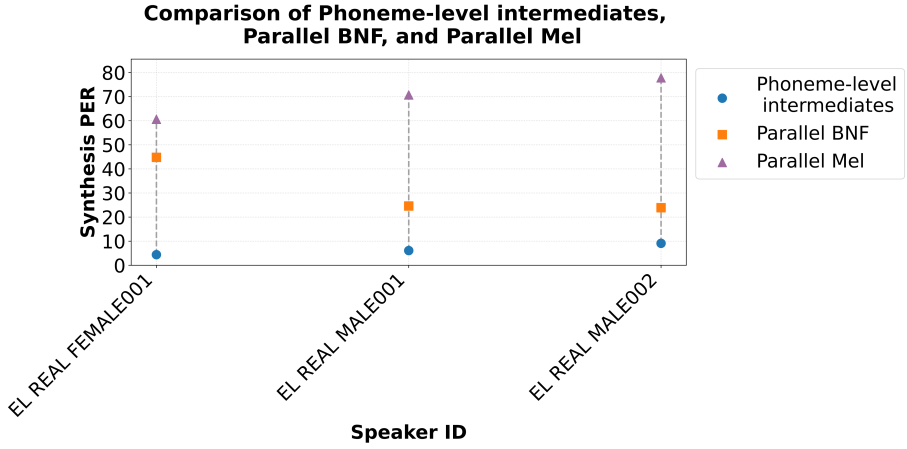


Figure 4.7: Comparison of PER using phoneme-level intermediates, parallel BNF, and parallel mel on another set of real EL speaker corpus. Note that a lower score represents better performance.

module propagate into the synthesis without any way to alleviate these errors. We investigate two strategies such as the use of source BNFs discussed in Chapter 4.3.4 and use of learnable discretization discussed in Chapter 4.3.5.

We first observe that the using BNFs alongside the discrete text is ineffective in reducing the errors made by the ASR module as seen in Sys. 10. Although the evaluation scores are somewhat similar with each other, we investigated further into the synthesized samples. We find that there are instances where mistranscriptions are "self-corrected" by the model, there are unfortunately also many cases where correct transcriptions are "overcorrected" by the model, which cancels out the benefits of the self-correction. Thus, although we indeed observe the effectiveness of cross-attention selecting relevant frames, the use of BNFs as a whole has no assured benefit, as there is no way for the network to know whether the input transcriptions were correct or not.

We also further investigate the learnable discretization normalization inspired by [62] as described in Sys. 11. Our first observation is that using the same BNF discretization method in [62] with K-means completely fails. Although the work in [62] used a Unit HiFiGAN (similarly composed of a duration predictor and upsampling alignment mechanism in Matcha-TTS [109] we used), this previous work also only investigated on the UASpeech [4] dataset, which only consists of word-level utterances, which meant that the model only had to model sub-phonetic units instead of phonetic units in sentence-level datasets like our setup. Thus, we instead opted to discretize the BNFs using the ground truth text, similar to an ASR system. Nevertheless, we still found that the learnable discretized intermediates were not beneficial, as it is still inferior compared to the phoneme-level intermediates method, as seen in the high CER and PER scores. In particular, we found that there were more words mispronounced, which may be due to the simple decoding method used in CTC, which became more sensitive

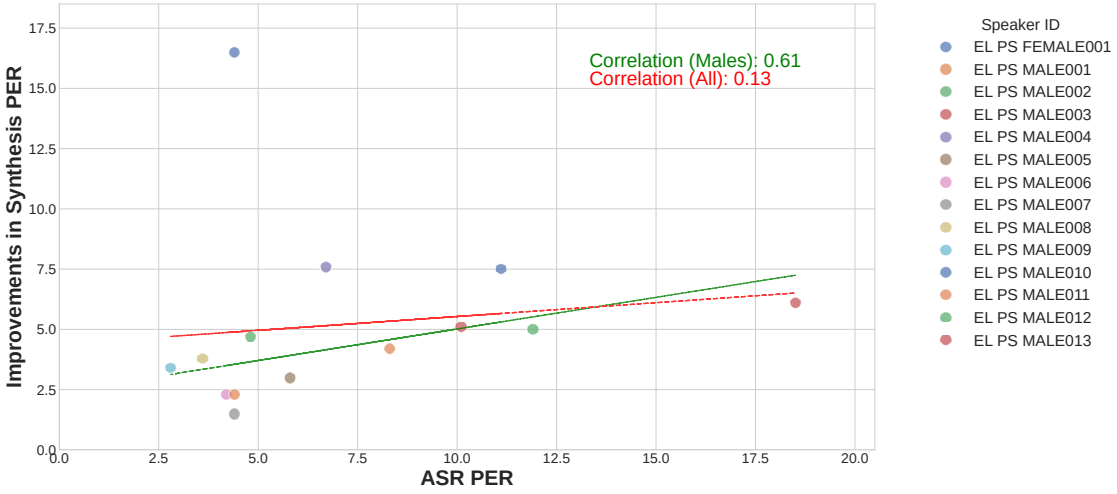


Figure 4.8: Analysis of correlation of the recognition module's performance in an ASR task, with respect to PER.

to mispronunciations due to using the ground truth text to discretize the BNFs instead of a K-means discretization approach. As a final overall observation, we notice that the DDUR and UTMOS metrics did not directly correlate with the subjective MOS results, whereas the CER and PER metrics were more representative of the resulting MOS results, showing that CER and PER may be the most viable option for objective tests in this task.

4.5.6 Generalization of proposed techniques with a larger EL dataset

Based on the previous experiments and Table 4.5, the phoneme-level intermediates method in Sys. 9 seems to be the most promising. As the previous experiments only used a single EL speaker, there may be doubts on the generalizability of the results. To further verify the generalizability of the phoneme-level intermediates method on

other EL speakers detailed in Table 4.2, we conduct experiments with a larger pseudo EL dataset with the best method in Sys. 9 and compare this to Sys. 3 (referred to as Parallel BNF) and Sys. 1 (referred to as Parallel Mel) as baselines. Pretraining on the HiFiCaptain pair and synthetic JSUT EL / JSUT pair datasets were also conducted. We use each EL speaker as described in the bottom half of Table 4.2 and convert all of them to a single typical male speaker (TYPICAL MALE007) to make the evaluations fair. To develop the ASR systems, we simply fine-tune the pretrained model on each speaker and do not conduct the intermediate fine-tuning method discussed in Chapter 4.2.2, as the goal of these experiments is to instead observe the behavior of the network when using BNFs with different ASR PER scores in the recognition module. As seen in Fig. 4.6, we see that in all 14 speakers, the use of phoneme-level intermediates is the most effective and produces the speech with the lowest PER. This pattern is further proven on another set of real world EL speakers, as seen in Fig. 4.7. Despite the real world EL speakers being limited to just 30 training utterances, using phoneme-level intermediates still performed with only a maximum of 9.1% PER, which further proves that the phoneme-level intermediates is the best method to use in terms of maximizing resources. This validates that phoneme-level intermediates is an effective method even for different EL speakers and would be a suitable choice as a framework for the atypical speech enhancement task.

As we notice that the level of improvement over the baselines vary for each speaker, we further investigate the correlation of the ASR PER and the synthesis PER in the 14 pseudo EL speakers. We further find a correlation in the absolute PER improvements when using phoneme-level intermediates over Parallel BNFs. As shown in Fig. 4.8, we find that ASR systems with a poor performance (higher PER score) actually get a higher performance improvements when using phoneme-level intermediates than the

traditional Parallel BNF method in Sys. 3. On the other hand, smaller improvements are found in cases where the ASR PER performs better (lower PER score). This can be further proven with the strong correlation coefficient of 0.61 for male speakers. Although an outlier to this finding is from the FEMALE001 speaker where there is more improvements despite a well performing ASR network, this might be attributed to the fact that the target speaker was a male speaker, which caused the baseline Parallel BNF setup to not perform effectively, which can be seen in Fig. 4.6. Thus, the outlier is mostly due to the baseline not performing well rather than the phoneme-level intermediates method having unstable behaviors.

These findings further show that researchers should opt to use phoneme-level intermediates instead of Parallel BNF or Parallel Mel as their baseline method when working on this task, due to its better performance and ease of accessibility in datasets. The phoneme-normalization strategy has other practical advantages, as there is no need for a parallel dataset, and control over the target speaker timbre would be easily handled by modern TTS systems, making it applicable to a lot of real-world scenarios even when the EL speaker has limited recorded data of their old voice.

However, these findings do not necessarily mean that working on a parallel setup is futile. For example, prosodic information from the source EL containing emphasis on certain words is disregarded with the phoneme-normalization setup but would be well handled by a parallel setup. Moreover, as explained previously, the pronunciation of some characters in Japanese and Chinese may also become more difficult if it is directly inferred from just the input text, whereas the pronunciation in a parallel setup can be retained.

4.6 Conclusions

We investigated methods to reduce domain mismatches between the EL and typical speech data through the use of discrete linguistic intermediates. We first resolved the first mismatch of different speakers during pretraining (typical speech) and fine-tuning (EL speech) by the use of bottleneck feature intermediates, which created a speaker-independent representation for both EL and typical speech, reducing the speech type mismatches between each dataset. Using this proposed method allowed the alignment module to focus on modeling intelligibility, where it scored a 16% absolute improvement in CER and a 0.83 higher naturalness score, compared to the baseline. This was primarily because compared to the baseline which required the alignment module to model both linguistic and speaker features at the same time, the method decomposed these tasks into different networks.

We next resolved the second mismatch of the difference in the temporal structure between the typical and EL speech, by introducing discrete linguistic intermediates to improve modeling the temporal structure of typical speech, and effectively removing the burden from the alignment module. Our findings showed that using phoneme-level intermediates through a cascaded ASR+TTS framework was the most efficient and synthesized speech with another absolute improvement of 1.4% in CER and 0.2 in naturalness, compared to the initialy proposed system. This was primarily because compared to audio information, text information does not contain any sort of temporal structure, and thus using these as prior information to normalize and select relevant frames allowed the alignment module to become more efficient.

Nevertheless, despite the phoneme-level intermediates having the best performance, there are still some fundamental limitations to using this approach as it deletes the source speaker's prosodic information entirely. Although this research investigated

alleviating inadequacies of the phoneme-level intermediates strategy such as using the source BNF and learnable discretization, we find tradeoffs in synthesis quality. Thus, future work still need to investigate how to improve the error propagation and setups discussed in this paper to consider how to reference the source speaker's temporal structure when modeling the target speaker through a parallel VC setup.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, we explored the possible improvements in deep learning approaches to help EL speakers overcome communication barriers in daily life. We focused on resolving two main problems: lack of training data and domain mismatches in both speech recognition and enhancement tasks.

In Chapter 2, we detailed the literature present in this field. We first give a background on the human speech production process, and the main responsibility of the larynx in this process. In cases where a larynx requires surgical removal, several other alternatives for speaking are available, but an electrolarynx is the least tasking of all. Next, we discussed preliminary deep learning concepts such as the attention-based Transformer and diffusion modeling, which were essential architectures used to resolve the problems in this thesis. Then, we discussed previous work on electrolaryngeal speech recognition, which either resolves issues using either pretraining or data augmentation to handle small-scale training datasets. Finally, we discussed previous work on electrolaryngeal speech enhancement, which typically relies on the use of data augmentation methods to handle domain mismatches.

In Chapter 3, we started the thesis by focusing on the speech recognition task. We established a strong baseline in our research by first providing a thorough comparison of novel pretraining techniques: supervised and self-supervised pretraining and compare the performance on two pathological datasets. We investigated two types of SSL-based pretraining frameworks, contrastive and masked region prediction, on two different pathological speech datasets and compared them with a supervised pretraining framework based on Transformer and Conformer architectures. The main takeaway from this study is that although SSL pretraining frameworks have shown success with minimally resourced healthy speech, this does not seem to be the case with EL speech.

Then, we provided improvements in the supervised pretraining framework by proposing an intermediate fine-tuning task that alleviated the learning gap between the pretraining and target fine-tuning tasks. To prove the effectiveness of this method, we presented a thorough study on the effects of using artificially generated EL data in the intermediate fine-tuning task by using different speech synthesis models and the

distortion of text labels in improving CER for EL speech recognition. By doing so, we observe improvements by up to 6.1% compared to the baselines. A further analysis on the latent spaces produced by each task showed that instead of focusing on learning the linguistic content, the network instead tried to first learn the voicing characteristics of the electrolaryngeal speech. These showed that optimizing learning the voicing characteristics of the target speakers can be an efficient task during the intermediate fine-tuning and could lead to improved results.

In Chapter 4, we focused on the speech enhancement task. Building on the findings from Chapter 3 and using this speech recognition model, we used this to also create a speech enhancement model. We analyzed two domain mismatches, which were found in the acoustic and temporal structures. We first resolved the first mismatch of different speakers during pretraining (typical speech) and fine-tuning (EL speech) by the use of bottleneck feature intermediates, which created a speaker-independent representation for both EL and typical speech, reducing the speech type mismatches between each dataset. Using this proposed method allowed the alignment module to focus on modeling intelligibility, where it scored a 16% absolute improvement in CER and a 0.83 higher naturalness score, compared to the baseline. This was primarily because compared to the baseline which required the alignment module to model both linguistic and speaker features at the same time, the method decomposed these tasks into different networks.

We next resolved the second mismatch of the difference in the temporal structure between the typical and EL speech, by introducing discrete linguistic intermediates to improve modeling the temporal structure of typical speech, and effectively removing the burden from the alignment module. Our findings showed that using phoneme-level intermediates through a cascaded ASR+TTS framework was the most efficient and

synthesized speech with another absolute improvement of 1.4% in character error rate and 0.2 in naturalness, compared to the initialy proposed system. This was primarily because compared to audio information, text information does not contain any sort of temporal structure, and thus using these as prior information to normalize and select relevant frames allowed the alignment module to become more efficient.

We conclude this thesis by proposing solutions to the two main problems and improving baseline systems. With the ASR+TTS framework, a simple cascaded, low-latency setup can already be done. Moreover, with a bit of effort on the engineering side, the latency can even be reduced by introducing voice activity detection such that the conversion process can occur in chunks of phrases rather than sentences. However, there is still more work to be done to improve these systems for smooth and uninterrupted real-world use cases. For example, in the speech recognition task, the proposed method performs at a 15% CER, which is still far behind from the recognition rate of the pretrained model at 4.4% on the healthy test set. On the other hand, speech enhancement also still has a large gap in naturalness tests, as the ground truth was rated at 4.72 MOS, while the best performing method was only rated at 2.97 MOS. Moreover, the use of phoneme-level linguistic intermediates have several limitations compared to a parallel setup as discussed in Chapter 4.5.6.

5.2 Future Work

The findings in the current experiments can bring several opportunities to improve the method. For speech recognition, the main problem has been the difficulty of inferring the correct Kanji characters without the intonation patterns from the EL speech. Thus, more work can be done on the decoder langauage model side, which can infer the correct sentence based on the context of the sentence. These can be done by using

post-processing techniques of the decoded text by using large language models, which are also currently done in decoding rare words. By the use of these post-processing techniques, the sentence can be corrected and use the context of the previous sentences instead of just inferring from the phonetic sounds of the input speech. Another approach can also be simply just directly predicting at a phoneme-level text instead of the character-level text.

On the other hand, for speech enhancement, more work can be done by improving the phoneme-level intermediates, in particular the cascaded setup using ASR and TTS. As both networks are separately optimized, this introduces the problem of error propagation. In a scenario where both networks can be jointly optimized using parallel data, this error propagation problem can be mitigated. Moreover, this helps improve the network by being able to use both the phonetic sounds from the speech input and the decoded text when generating the enhanced speech, and can minimize errors in the pronunciation of different Kanji characters. For example, Codec-based quantization and compressing audio into semantic and acoustic tokens have become a popular approach in speech synthesis. Thus, there are several opportunities by creating an encoder that can compress EL speech into effective semantic tokens using the speech recognizer above, or by jointly training the quantization module of the semantic tokens and the synthesizer jointly.

Another opportunity for improvement in the speech enhancement task is making the network work in real-time. Currently, EL speakers have to say the entire sentence, pass it through the speech enhancement framework, and wait for the converted healthy speech to be played in a cascaded fashion. Even with a low-latency network, the cascaded fashion is what makes this framework difficult to adapt to real-world settings. An ideal setup could be one where while the EL speaker speaks, the model already

listens and processes the EL speech, and outputs the healthy voice even when the speaker is not yet finished talking. Although it is theoretically difficult to convert EL speech frame-by-frame like commonly-used voice changers due to the mismatch in temporal structure, there can be opportunities such as having a few second latency to give the network enough prior information to convert EL speech into healthy speech. This has been done in low-latency speech-to-speech translation such as Hibiki [110], which is based on the Moshi [111] architecture. In this framework, the network learns the alignments and only starts converting once it determines that it has recorded enough prior information. Typically, the translated speech is outputted starting after 2 seconds have been recorded. Moreover, since speech enhancement is a monotonous alignment (compared to speech-to-speech translation which has non-monotonous alignments between different languages), applying the same framework to the speech enhancement task should be possible and make it usable in a real-world setting.

In general, another problem in both tasks is that a speaker-dependent network needs to be trained, primarily due to the huge differences in pronunciation patterns between different EL speakers. Although this is the widely used method due to the difficulty of this problem, this also prohibits several EL speakers from using these models as a daily tool. Thus, speaker-independent frameworks where minimal amounts of data are only required, similar to how state-of-the-art networks function for healthy speakers, can help make these solutions mainstream and reasonable to use for any EL speaker. If a model can be robust enough to extract linguistic information at the same accuracy as with healthy speech, then this can be the first step in resolving this issue. This thesis somewhat tackled this framework in Chapter 4.2.2 with the recognition module being able to decode text from either healthy or EL speech, but there is still a gap in recognition performance between the two types of data. Future work can focus

on making this recognition module even more robust and ensuring that any type of speech can be projected into the same latent space (and thus be decoded at the same performance).

Acknowledgements

This thesis would not have been possible without the help of many people. All of them have given me mentorship, advice, encouragement, and motivation to power through the 5-year Ph.D. course.

I would primarily like to thank Prof. Tomoki Toda for accepting me to be part of the laboratory and guiding me for the past five years. I always felt that I improved myself as a student researcher every semester and it was all thanks to his guidance in my research direction. I learned to fail more times than I can count, but also learned how to always get back up on my feet. Thank you for trusting and believing in me. I would also like to thank the following laboratory seniors for all their help in my growth as a student researcher. Prof. Wen-Chin Huang, for helping me establish good research fundamentals, giving lots of advice in conducting my experiments, and always pushing me to do better and consistently improve my research skills. Ding Ma, for helping me in keeping sane with electrolaryngeal speech research, brainstorming new research ideas with me (and helping me read the kanji in restaurant menus). Ryuichi Yamamoto, for introducing me to the world of singing voice synthesis/conversion, as well as giving me lots of motivation and being my main inspiration to pursue the speech synthesis field. I would not be half the researcher that I am today without the help of these people.

Coming to Japan and being an international student in Japan without knowing anyone beforehand was a daunting task to say the least, thus I would like to thank the

entirety of Toda Laboratory, for helping me become familiar with Japanese culture and language. Moreover, a huge thank you to my family in the Philippines. In particular, my fiancée Shaina, and my closest friends Kyle, Gab, Jego, Nica, Julie, MM, Wacky, Patrick, Albert, Aljo, Geri, Bryan, my siblings Anna and Paul, and my parents, for all the support they gave over the course of my Ph.D. degree and cheering me on during several key moments.

Finally, I would like to convey my deepest gratitude to the Ministry of Education, Culture, Sports, Science and Technology for offering me a full scholarship to study in Japan and making all my dreams of doing deep learning research come true. It would have been entirely impossible for me to go and study in Japan without this scholarship, so everything started thanks to their generosity.

I would have never thought that I completely deserved all the trust that everyone has given to me. I hope to continue doing my best as I continue on to the next chapter of my research career.

References

[1] M. I. Singer and E. D. Blom, "An endoscopic technique for restoration of voice after laryngectomy," *Annals of Otology, Rhinology & Laryngology*, vol. 89, no. 6, pp. 529–533, 1980, PMID: 7458140.

[2] R. Kaye, C. G. Tang, and C. F. Sinclair, "The electrolarynx: Voice restoration after total laryngectomy," en, *Med Devices (Auckl)*, vol. 10, pp. 133–140, 2017.

[3] F. Rudzicz, A. K. Namasivayam, and T. Wolff, "The torgo database of acoustic and articulatory speech from speakers with dysarthria," *Language Resources and Evaluation*, vol. 46, no. 4, pp. 523–541, 2012.

[4] H. Kim, M. Hasegawa-Johnson, A. Perlman, J. Gunderson, T. S. Huang, K. Watkin, and S. Frame, "Dysarthric speech database for universal access research," in *Proc. Interspeech*, 2008, pp. 1741–1744.

[5] G. Schu, P. Janbakhshi, and I. Kodrasi, "On using the UA-Speech and TORGO databases to validate automatic dysarthric speech classification approaches," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 5 pages, 2023.

[6] I. Titze, "The physics of small-amplitude oscillation of the vocal folds," *The Journal of the Acoustical Society of America*, vol. 83, pp. 1536–52, 1988.

[7] J. C. Lucero, "Optimal glottal configuration for ease of phonation," *Journal of Voice*, vol. 12, no. 2, pp. 151–158, 1998.

[8] J. Koreman, "The effects of stress and f0 on the voice source," *Phonus*, vol. 1, pp. 105–120, 1995.

[9] S. Almaghrabi, S. Clark, and M. Baumert, "Bio-acoustic features of depression: A review," *Biomedical Signal Processing and Control*, vol. 85, p. 105 020, 2023.

[10] P. Lin, *BME 240 – Introduction to Clinical Medicine*, https://bme240.eng.uci.edu/students/06s/paytonl/index.html, 2006.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017, 11 pages.

[12] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Proc. Interspeech*, 2020, pp. 5036–5040.

[13] P. Guo, F. Boyer, X. Chang, T. Hayashi, Y. Higuchi, H. Inaguma, N. Kamo, C. Li, D. Garcia-Romero, J. Shi, J. Shi, S. Watanabe, K. Wei, W. Zhang, and Y. Zhang, "Recent developments on espnet toolkit boosted by conformer," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 5874–5878.

[14] S. Kim, T. Hori, and S. Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 4835–4839.

[15] T. Hori, S. Watanabe, Y. Zhang, and W. Chan, "Advances in Joint CTC-Attention Based End-to-End Speech Recognition with a Deep CNN Encoder and RNN-LM," in *Proc. Interspeech*, 2017, pp. 949–953.

[16] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. International Conference on Machine Learning*, 2006, pp. 369–376.

[17] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 4960–4964.

[18] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. International Conference on Machine Learning*, 2006, pp. 369–376.

[19] W.-C. Huang, T. Hayashi, Y.-C. Wu, H. Kameoka, and T. Toda, "Voice Transformer Network: Sequence-to-sequence voice conversion using transformer with text-to-speech pretraining," in *Proc. Interspeech*, 2020, pp. 4676–4680.

[20] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," *Proc. NeurIPS*, pp. 17 022–17 033, 2020.

[21] S. Becker and G. E. Hinton, "Self-Organizing Neural Network that Discovers Surfaces in Random-Dot Stereograms," *Nature*, vol. 355, no. 6356, pp. 161–163, 1992.

[22] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. Lecun, C. Moore, E. Säckinger, and R. Shah, "Signature Verification Using a "siamese" Time Delay Neural Network," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 07, no. 04, pp. 669–688, 1993.

[23] S.-W. Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin, T.-H. Huang, W.-C. Tseng, K.-t. Lee, D.-R. Liu, Z. Huang, S. Dong, S.-W. Li, S. Watanabe, A. Mohamed, and H.-Y. Lee, "SUPERB: Speech Processing Universal PERformance Benchmark," in *Proc. Interspeech*, 2021, pp. 1194–1198.

[24] Y.-A. Chung, W.-N. Hsu, H. Tang, and J. Glass, "An Unsupervised Autoregressive Model for Speech Representation Learning," in *Proc. Interspeech*, 2019, pp. 146–150.

[25] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 449–12 460, 2020.

[26] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, "Unsupervised Cross-Lingual Representation Learning for Speech Recognition," in *Proc. Interspeech*, 2021, pp. 2426–2430.

[27] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, J. Wu, L. Zhou, S. Ren, Y. Qian, Y. Qian, J. Wu, M. Zeng, X. Yu, and F. Wei, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[28] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, and Y. Bengio, "Multi-Task Self-Supervised Learning for Robust Speech Recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 6989–6993.

[29] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Proc. NeurIPS*, vol. 33, pp. 6840–6851, 2020.

[30] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," in *Proc. ICLR*, 28 pages, 2023.

[31] R. Gao, E. Hoogeboom, J. Heek, V. D. Bortoli, K. P. Murphy, and T. Salimans, "Diffusion meets flow matching: Two sides of the same coin," 2024.

[32] P. Stanislav, J. V. Psutka, and J. Psutka, "Recognition of the electrolaryngeal speech: Comparison between human and machine," in *Text, Speech, and Dialogue*, 2017, pp. 509–517.

[33] J. Shor, D. Emanuel, O. Lang, O. Tuval, M. Brenner, J. Cattiau, F. Vieira, M. McNally, T. Charbonneau, M. Nollstadt, A. Hassidim, and Y. Matias, "Personalizing ASR for Dysarthric and Accented Speech with Limited Data," in *Proc. Interspeech*, 2019, pp. 784–788.

[34] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 6645–6649.

[35] J. R. Green, R. L. MacDonald, P.-P. Jiang, J. Cattiau, R. Heywood, R. Cave, K. Seaver, M. A. Ladewig, J. Tobin, M. P. Brenner, P. C. Nelson, and K. Tomanek, "Automatic Speech Recognition of Disordered Speech: Personalized

Models Outperforming Human Listeners on Short Phrases," in *Proc. Interspeech*, 2021, pp. 4778–4782.

[36] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *Proc. International Conference on Machine Learning*, 28 pages, 2023.

[37] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[38] K. D. N, P. Wang, and B. Bozza, "Using Large Self-Supervised Models for Low-Resource Speech Recognition," in *Proc. Interspeech*, 2021, pp. 2436–2440.

[39] R. Al-Ghezi, Y. Getman, A. Rouhe, R. Hildén, and M. Kurimo, "Self-Supervised End-to-End ASR for Low Resource L2 Swedish," in *Proc. Interspeech*, 2021, pp. 1429–1433.

[40] Z. Jin, M. Geng, X. Xie, J. Yu, S. Liu, X. Liu, and H. Meng, "Adversarial Data Augmentation for Disordered Speech Recognition," in *Proc. Interspeech*, 2021, pp. 4803–4807.

[41] J. Harvill, D. Issa, M. Hasegawa-Johnson, and C. Yoo, "Synthesis of new words for improved dysarthric speech recognition on an expanded vocabulary," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 6428–6432.

[42] M. Soleymanpour, M. T. Johnson, R. Soleymanpour, and J. Berry, "Synthesizing dysarthric speech using multi-speaker tts for dysarthric speech recognition,"

in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2022, pp. 7382–7386.

[43] W.-C. Huang, B. M. Halpern, L. Phillip Violeta, O. Scharenborg, and T. Toda, "Towards identity preserving normal to dysarthric voice conversion," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2022, pp. 6672–6676.

[44] B. M. Halpern, W.-C. Huang, L. P. Violeta, R. van Son, and T. Toda, "Improving severity preservation of healthy-to-pathological voice conversion with global style tokens," in *Proc. ASRU*, 7 pages, 2023.

[45] W.-C. Huang, S.-W. Yang, T. Hayashi, and T. Toda, "A Comparative Study of Self-Supervised Speech Representation Based Voice Conversion," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1308–1318, 2022.

[46] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[47] H. Kameoka, W.-C. Huang, K. Tanaka, T. Kaneko, N. Hojo, and T. Toda, "Many-to-many voice transformer network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 656–670, 2021.

[48] T. Hayashi, W.-C. Huang, K. Kobayashi, and T. Toda, "Non-autoregressive sequence-to-sequence voice conversion," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 7068–7072.

[49] H. Kameoka, K. Tanaka, D. Kwaśny, T. Kaneko, and N. Hojo, "Convs2s-vc: Fully convolutional sequence-to-sequence voice conversion," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 28, pp. 1849–1863, 2020.

[50] J.-X. Zhang, L.-J. Liu, Y.-N. Chen, Y.-J. Hu, Y. Jiang, Z.-H. Ling, and L.-R. Dai, "Voice Conversion by Cascading Automatic Speech Recognition and Text-to-Speech Synthesis with Prosody Transfer," in *Proc. Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge*, 2020, pp. 121–125.

[51] W.-C. Huang, T. Hayashi, X. Li, S. Watanabe, and T. Toda, "On Prosody Modeling for ASR+TTS Based Voice Conversion," in *Proc. ASRU*, 2021, pp. 642–649.

[52] H. Liu, Q. Zhao, M. Wan, and S. Wang, "Enhancement of electrolarynx speech based on auditory masking," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 5, pp. 865–874, 2006.

[53] K. Nakamura, T. Toda, H. Saruwatari, and K. Shikano, "Electrolaryngeal speech enhancement based on statistical voice conversion," in *Proc. Interspeech*, 2009, pp. 1431–1434.

[54] K. Nakamura, T. Toda, H. Saruwatari, and K. Shikano, "Speaking-aid systems using gmm-based voice conversion for electrolaryngeal speech," *Speech Communication*, vol. 54, no. 1, pp. 134–146, 2012.

[55] P. Malathi, G. R. Sureshw, and M. Moorthi, "Enhancement of electrolaryngeal speech using frequency auditory masking and gmm based voice conversion," in *Proc. International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics*, 4 pages, 2018.

[56] Z. Qian, H. Niu, L. Wang, K. Kobayashi, S. Zhang, and T. Toda, "Mandarin electro-laryngeal speech enhancement based on statistical voice conversion and manual tone control," in *Proc. APSIPA ASC*, 2021, pp. 546–552.

[57] M.-C. Yen, W.-C. Huang, K. Kobayashi, Y.-H. Peng, S.-W. Tsai, Y. Tsao, T. Toda, J.-S. R. Jang, and H.-M. Wang, "Mandarin electrolaryngeal speech voice conversion with sequence-to-sequence modeling," in *Proc. ASRU*, 2021, pp. 650–657.

[58] K. Kobayashi, T. Hayashi, and T. Toda, "Low-latency electrolaryngeal speech enhancement based on fastspeech2-based voice conversion and self-supervised speech representation," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 5 pages, 2023.

[59] D. Ma, L. P. Violeta, K. Kobayashi, and T. Toda, "Two-stage training method for Japanese electrolaryngeal speech enhancement based on sequence-to-sequence voice conversion," in *Proc. SLT*, 2023, pp. 949–954.

[60] W.-C. Huang, T. Hayashi, Y.-C. Wu, H. Kameoka, and T. Toda, "Pretraining techniques for sequence-to-sequence voice conversion," *IEEE/ACM TASLP*, vol. 29, pp. 745–755, 2021.

[61] D. Ma, L. P. Violeta, K. Kobayashi, and T. Toda, "Pretraining and fine-tuning techniques for electrolaryngeal speech enhancement based on sequence-to-sequence voice conversion," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 3189–3201, 2025.

[62] Y. Wang, X. Wu, D. Wang, L. Meng, and H. Meng, "UNIT-DSR: Dysarthric Speech Reconstruction System Using Speech Unit Normalization," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2024, pp. 12 306–12 310.

[63] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM TASLP*, vol. 29, pp. 3451–3460, 2021.

[64] R. D. Kent, "Research on Speech Motor Control and Its Disorders: A Review and Prospective," *Journal of Communication Disorders*, vol. 33, no. 5, pp. 391–428, 2000.

[65] M. Moore, H. Venkateswara, and S. Panchanathan, "Whistle-blowing ASRs: Evaluating the Need for More Inclusive Speech Recognition Systems," in *Proc. Interspeech*, 2018, pp. 466–470.

[66] A. Graves and N. Jaitly, "Towards End-to-End Speech Recognition with Recurrent Neural Networks," in *Proc. ICML*, 2014, pp. 1764–1772.

[67] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *International conference on artificial neural networks*, 2018, pp. 270–279.

[68] J. Phang, T. Févry, and S. R. Bowman, "Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks," 12 pages, 2018.

[69] S. Garg, T. Vu, and A. Moschitti, "Tanda: Transfer and adapt pre-trained transformer models for answer sentence selection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 7780–7788.

[70] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Min-*

neapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), 2019, pp. 4171–4186.

[71] I. Papadimitriou and D. Jurafsky, "Learning Music Helps You Read: Using transfer to study linguistic structure in language models," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 6829–6839.

[72] C.-H. Chiang and H.-y. Lee, "On the transferability of pre-trained language models: A study from artificial datasets," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, 2022, pp. 10 518–10 525.

[73] R. Ri and Y. Tsuruoka, "Pretraining with artificial language: Studying transferable knowledge in language models," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022, pp. 7302–7315.

[74] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 5206–5210.

[75] P. Gage, "A new algorithm for data compression," *C Users J.*, vol. 12, no. 2, pp. 23–38, 1994.

[76] S. Liu, M. Geng, S. Hu, X. Xie, M. Cui, J. Yu, X. Liu, and H. Meng, "Recent progress in the cuhk dysarthric speech recognition system," *IEEE/ACM TASLP*, vol. 29, pp. 2267–2281, 2021.

[77] L. Wu, D. Zong, S. Sun, and J. Zhao, "A Sequential Contrastive Learning Framework for Robust Dysarthric Speech Recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 7303–7307.

[78] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, "Neural speech synthesis with transformer network," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 6706–6713.

[79] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: End-to-End Speech Processing Toolkit," in *Proc. Interspeech*, 2018, pp. 2207–2211.

[80] S. Watanabe, F. Boyer, X. Chang, P. Guo, T. Hayashi, Y. Higuchi, T. Hori, W.-C. Huang, H. Inaguma, N. Kamo, S. Karita, C. Li, J. Shi, A. S. Subramanian, and W. Zhang, "The 2020 ESPnet Update: New Features, Broadened Applications, Performance Improvements, and Future Plans," in *Proc. IEEE Data Science and Learning Workshop (DSLW)*, 5 pages, 2021.

[81] S. Ando and H. Fujihara, "Construction of a large-scale japanese asr corpus on tv recordings," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 6948–6952.

[82] K. Maekawa, "Corpus of spontaneous japanese: Its design and evaluation," in *ISCA/IEEE Workshop on Spontaneous Speech Processing and Recognition*, 5 pages, 2003, paper MMO2.

[83] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, *et al.*, "Libri-light: A benchmark for asr with limited or no supervision," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 7669–7673.

[84] W.-N. Hsu, A. Sriram, A. Baevski, T. Likhomanenko, Q. Xu, V. Pratap, J. Kahn, A. Lee, R. Collobert, G. Synnaeve, and M. Auli, "Robust wav2vec 2.0:

Analyzing Domain Shift in Self-Supervised Pre-Training," in *Proc. Interspeech*, 2021, pp. 721–725.

[85] R. Doshi, Y. Chen, L. Jiang, X. Zhang, F. Biadsy, B. Ramabhadran, F. Chu, A. Rosenberg, and P. J. Moreno, "Extending parrotron: An end-to-end, speech conversion and speech recognition model for atypical speech," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 6988–6992.

[86] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 6199–6203.

[87] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *JMLR*, vol. 9, no. 86, pp. 2579–2605, 2008.

[88] R. Sonobe, S. Takamichi, and H. Saruwatari, "Jsut corpus: Free large-scale japanese speech corpus for end-to-end speech synthesis," 4 pages, 2017.

[89] B. Sisman, J. Yamagishi, S. King, and H. Li, "An overview of voice conversion and its challenges: From statistical modeling to deep learning," *IEEE/ACM TASLP*, vol. 29, pp. 132–157, 2021.

[90] R. L. MacDonald, P.-P. Jiang, J. Cattiau, R. Heywood, R. Cave, K. Seaver, M. A. Ladewig, J. Tobin, M. P. Brenner, P. C. Nelson, J. R. Green, and K. Tomanek, "Disordered Speech Data Collection: Lessons Learned at 1 Million Utterances from Project Euphonia," in *Proc. Interspeech*, 2021, pp. 4833–4837.

[91] R. Doshi, Y. Chen, L. Jiang, X. Zhang, F. Biadsy, B. Ramabhadran, F. Chu, A. Rosenberg, and P. J. Moreno, "Extending Parrotron: An End-to-End, Speech

Conversion and Speech Recognition Model for Atypical Speech," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 6988–6992.

[92] Z. Chen, B. Ramabhadran, F. Biadsy, X. Zhang, Y. Chen, L. Jiang, F. Chu, R. Doshi, and P. J. Moreno, "Conformer Parrotron: A Faster and Stronger End-to-End Speech Conversion and Recognition Model for Atypical Speech," in *Proc. Interspeech*, 2021, pp. 4828–4832.

[93] S. Liu, J. Zhong, L. Sun, X. Wu, X. Liu, and H. Meng, "Voice conversion across arbitrary speakers based on a single target-speaker utterance," in *Proc. Interspeech*, 2018, pp. 496–500.

[94] S. Liu, Y. Cao, D. Wang, X. Wu, X. Liu, and H. Meng, "Any-to-many voice conversion with location-relative sequence-to-sequence modeling," *IEEE/ACM TASLP*, vol. 29, pp. 1717–1728, 2021.

[95] B. van Niekerk, M.-A. Carbonneau, J. Zaïdi, M. Baas, H. Seuté, and H. Kamper, "A comparison of discrete and soft speech units for improved voice conversion," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2022, pp. 6562–6566.

[96] L. P. Violeta and T. Toda, "An Analysis of Personalized Speech Recognition System Development for the Deaf and Hard-of-Hearing," in *Proc. APSIPA ASC*, 2023, pp. 1862–1867.

[97] J. Tobin and K. Tomanek, "Personalized Automatic Speech Recognition Trained on Small Disordered Speech Datasets," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2022, pp. 6637–6641.

[98] T. Okamoto, Y. Shiga, and H. Kawai, *Hi-Fi-CAPTAIN: High-fidelity and high-capacity conversational speech synthesis corpus developed by NICT*, https://ast-astrec.nict.go.jp/en/release/hi-fi-captain/, 2023.

[99] S. Kim, H. Kim, and S. Yoon, "Guided-TTS 2: A diffusion model for high-quality adaptive text-to-speech with untranscribed data," *arXiv preprint arXiv:2205.15370*, 2022, 16 pages.

[100] J. Ho and T. Salimans, "Classifier-free diffusion guidance," in *Proc. NeurIPS*, 8 pages, 2021.

[101] R. Badlani, A. Łańcucki, K. J. Shih, R. Valle, W. Ping, and B. Catanzaro, "One tts alignment to rule them all," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2022, pp. 6092–6096.

[102] K. J. Shih, R. Valle, R. Badlani, A. Lancucki, W. Ping, and B. Catanzaro, "RAD-TTS: Parallel Flow-Based TTS with Robust Alignment Learning and Diverse Synthesis," in *Proc. ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models*, 8 pages, 2021.

[103] W.-C. Huang, K. Kobayashi, and T. Toda, "AAS-VC: On the Generalization Ability of Automatic Alignment Search based Non-autoregressive Sequence-to-sequence Voice Conversion," *arXiv preprint arXiv:2309.07598*, 2023, 5 pages.

[104] T. Saeki, D. Xin, W. Nakata, T. Koriyama, S. Takamichi, and H. Saruwatari, "UTMOS: UTokyo-SaruLab System for VoiceMOS Challenge 2022," in *Proc. Interspeech*, 2022, pp. 4521–4525.

[105] S. Takamichi, K. Mitsui, Y. Saito, T. Koriyama, N. Tanji, and H. Saruwatari, "JVS corpus: free Japanese multi-speaker voice corpus," *arXiv preprint arXiv:1908.06248*, 2019, 4 pages.

[106] C. Veaux, J. Yamagishi, and K. MacDonald, "CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit," *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2012.

[107] J. Liu, C. Li, Y. Ren, F. Chen, P. Liu, and Z. Zhao, "Diffsinger: Singing voice synthesis via shallow diffusion mechanism," *arXiv preprint arXiv:2105.02446*, vol. 2, 2021.

[108] Y. Yan, X. Tan, B. Li, T. Qin, S. Zhao, Y. Shen, and T.-Y. Liu, "Adaspeech 2: Adaptive text to speech with untranscribed data," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 6613–6617.

[109] S. Mehta, R. Tu, J. Beskow, É. Székely, and G. E. Henter, "Matcha-tts: A fast tts architecture with conditional flow matching," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, 2024, pp. 11 341–11 345.

[110] T. Labiausse, L. Mazaré, E. Grave, A. Défossez, and N. Zeghidour, "High-fidelity simultaneous speech-to-speech translation," in *Proc. International Conference on Machine Learning*, 12 pages, 2025.

[111] A. Défossez, L. Mazaré, M. Orsini, A. Royer, P. Pérez, H. Jégou, E. Grave, and N. Zeghidour, "Moshi: A speech-text foundation model for real-time dialogue," Tech. Rep., 2024, 67 pages.

List of Publications

Journal Papers

[1] **L. P. Violeta**, W. C. Huang, D. Ma, R. Yamamoto, K. Kobayashi and T. Toda, "Resolving Domain Mismatches in Electrolaryngeal Speech Enhancement With Linguistic Intermediates," in IEEE Journal of Selected Topics in Signal Processing, vol. 19, pp. 827–839, 2025

[2] **L. P. Violeta**, D. Ma, W. C. Huang and T. Toda, "Pretraining and Adaptation Techniques for Electrolaryngeal Speech Recognition," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 32, pp. 2777–2789, 2024

[3] B. M. Halpern, W. C. Huang, **L.P. Violeta**, T. Toda, "Severity-Controllable Pathological Text-to-Speech Synthesis for Clinical Applications", in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 34, pp. 573–582, 2026

[4] D. Ma, **L.P. Violeta**, K. Kobayashi, T. Toda, "Pretraining and Fine-Tuning Techniques for Electrolaryngeal Speech Enhancement Based on Sequence-to-Sequence Voice Conversion", in IEEE Transactions on Audio, Speech and Language Processing, vol. 33, pp. 3189–3201, 2025

International Conferences

[1] **L.P. Violeta**, X. Zhang, J. Shi, Y. Yasuda, W.-C. Huang, Z. Wu, T. Toda, "The Singing Voice Conversion Challenge 2025: From Singer Identity Conversion To Singing Style Conversion" Proc. ICASSP, pp. **–**, 2026 (Accepted).

[2] **L.P. Violeta**, W.-C. Huang, T. Toda, "Serenade: A Singing Style Conversion Framework Based On Audio Infilling" in Proc. EUSIPCO, pp. 411–415, 2025

[3] **L. P. Violeta**, W. C. Huang, D. Ma, R. Yamamoto, K. Kobayashi and T. Toda, "Electrolaryngeal Speech Intelligibility Enhancement through Robust Linguistic Encoders" in Proc. ICASSP, pp. 10961–10965, 2024

[4] **L. P. Violeta** and T. Toda, "An analysis of personalized speech recognition system development for the deaf and hard-of-hearing" in Proc. APSIPA ASC, pp. 1862–1867, 2023

[5] **L. P. Violeta**, D. Ma, W. C. Huang and T. Toda, "Intermediate fine-tuning using imperfect synthetic speech for improving electrolaryngeal speech recognition" in Proc. ICASSP, 5 pages, 2023

[6] **L.P. Violeta**, W.-C. Huang, T. Toda, "Investigating self-supervised pretraining frameworks for pathological speech recognition," Proc. INTERSPEECH, pp. 41–45, Incheon, Korea, Sep. 2022.

[7] B.M. Halpern, T. Tienkamp, W.-C. Huang, **L.P. Violeta**, T. Rebernik, S. de Visscher, M. Witjes, M. Wieling, D. Abur, T. Toda, "Quantifying the effect of speech pathology on automatic and human speaker verification", in Proc. Interspeech, pp. 3015–3019, 2024

[8] W.-C. Huang, **L.P. Violeta**, S. Liu, J. Shi, T. Toda, "The Singing Voice Conversion Challenge 2023", in Proc. ASRU, 6 pages, 2023

[9] R. Yamamoto, R. Yoneyama, **L.P. Violeta**, W.-C. Huang, T. Toda, "A comparative study of voice conversion models with large-scale speech and singing data: The T13 systems for the singing voice conversion challenge 2023", in Proc. ASRU. 6 pages, 2023

[10] B.M. Halpern, W.-C. Huang, **L.P. Violeta**, RJJH van Son, T. Toda, "Improving severity preservation of healthy-to-pathological voice conversion with global style tokens", in Proc. ASRU, 6 pages, 2023

[11] W.-C. Huang, B.M. Halpern, **L.P. Violeta**, O. Scharenborg, T. Toda, "Towards identity preserving normal to dysarthric voice conversion," IEEE ICASSP, pp. 6672–6676, Singapore, May 2022.

[12] D. Ma, **L.P. Violeta**, K. Kobayashi, T. Toda, "Two-stage training method for japanese electrolaryngeal speech enhancement based on sequence-to-sequence voice conversion", Proc. IEEE SLT, Doha, Qatar, Jan. 2021.

Awards

[1] Travel grant, ISCA and Interspeech 2022

[2] Ph.D. Fellow, Nagoya University Interdisciplinary Frontier Fellowship

[3] Ministry of Education, Culture, Sports, Science and Technology Full Scholarship

Organizing Committee

[1] The Singing Voice Conversion Challenge 2025

[2] The Singing Voice Conversion Challenge 2023