Title

Towards Better Automatic Speech Recognition Systems for Children

Author

Zhu, Yunzheng

Publication Date

2023

UNIVERSITY OF CALIFORNIA

Los Angeles

Towards Better Automatic Speech Recognition Systems for Children

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Electrical and Computer Engineering

by

Yunzheng Zhu

2023

ABSTRACT OF THE THESIS

Towards Better Automatic Speech Recognition Systems for Children

by

Yunzheng Zhu
Master of Science in Electrical and Computer Engineering
University of California, Los Angeles, 2023
Professor Abeer A. Alwan, Chair

This thesis aims to achieve better automatic speech recognition (ASR) for children. The most challenging problem is a lack of transcribed available databases, and thus this problem could be regarded as a low-resource task. We introduce this problem from three aspects. First, compared to adults' speech, larger intra- and inter-speaker variabilities in children's speech exacerbate the low-resource problem due to the different growth patterns of children's vocal tracts. Second, there are children's speech data on the Internet untranscribed. Thus, exploring how to utilize such untranscribed data to improve children's ASR is significant but challenging. Last, lacking training data will cause inadequate training when using random model initialization. Therefore, finding a good model initialization is important for training a robust children's ASR model in a low-resource setting.

We improve the performance of children's ASR systems from the aforementioned three aspects. First, we compare multiple effective data augmentation methods for children's ASR. On the OGI Kids' Corpus, we can achieve a WER reduction of around 10 % for the HMM-BLSTM modeling-based hybrid ASR system and around 25 % for the Connectionist Temporal Classification - Attention-based Encoder-Decoder (CTC-AED) modeling-based end-to-end ASR system. Second, unsupervised pre-training and semi-supervised learning are used as two effective methods for utilizing untranscribed data. Using one of the unsupervised pre-training methods, bidirectional autoregressive predictive coding, and 3 iterations of semi-supervised learning could bring a WER reduction of 9.6 % with 60 hours of untranscribed data. Third, model-agnostic meta-learning (MAML) based meta-initialization (MI) was used to find a good model initialization. However, MI is vulnerable to overfitting on training tasks (learner overfitting). To alleviate learner overfitting, an age-based task-level augmentation method is proposed. After using task-level augmentation methods with MI, the children's ASR system is able to achieve a WER reduction of 51 % in kindergarten-aged speech over no augmentation or initialization.

The thesis of Yunzheng Zhu is approved.

Gregory J. Pottie

Achuta Kadambi

Abeer A. Alwan, Committee Chair

University of California, Los Angeles

2023

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First of all, I would like to express my gratitude and respect for my advisor, Prof. Abeer Alwan. I really appreciate your guidance and advice in advancing my ability in research development. It was a great opportunity for me in joining your research lab and learned how to think, resolve, and express research problems, particularly in speech. It was really difficult time for me to understand most of the materials in the beginning. Under your supervision, I was able to think and practice more in the process of doing research. I achieved more than I expected during my Master's studies.

I would like to express my appreciation to my Master's committee, Prof. Gregory Pottie, and Prof. Achuhta Kadambi. Thank you all for agreeing to be on my committee. and thank you for giving me this opportunity to complete this thesis as an important step during my graduate studies.

I would also like to express my appreciation to Prof. Jiann-Wen Woody Ju, and thank you for your encouragement. Your words really motivated me to keep seeking my potential.

Then, I would like to thank my former and current labmates in SPAPL group. Everyone here is ambitious and inspiring. I would like to particularly thank Dr. Gary Yeung for helping me in using his F0-norm-based methods in my thesis. Thank you all for offering help, Jinhan Wang, Vijay Ravi, Amber Afshan, Alexander Johnson, Morgan Tinkler, and Vishwas Shetty. Lastly, I would like to particularly express my gratitude to Ruchao Fan for his advice and help. Ruchao Fan helped me not only in research experiments but also in real life. Thinking about research problems is equivalent to thinking about problems in real life. It truly inspired me and developed my mind in viewing research problems as more than just a job. It was a great time working with everyone here in SPAPL group.

Finally, I would like to thank my parents for showing their support for my graduate studies at UCLA. Prof. Hehua Zhu and Mrs. Wei Wu is always encouraging me in graduate studies and helping me go through difficulties.

Chapter 3 is a slightly modified version of [WZF21] published in Interspeech 2021 and has been reproduced here with the permission of the copyright holder. Chapter 4 is a slightly modified version of [ZFA22] published in ICASSP 2022 and has been reproduced here with the permission of the copyright holder.

CHAPTER 1

Introduction

Automatic Speech Recognition (ASR) for adults has been well-developed over the years due to the sufficient amount of adult speech data [HDY12]. However, children's ASR still remains a challenging problem. In this chapter, three common problems: speaker variations in children's speech, utilizing untranscribed children's speech data, and finding a good model initialization for children's ASR in a low-resource setting are discussed.

1.1 Problem 1: Variability in Children's Speech

First, due to a lack of large, publicly-available databases, it is hard to establish a good ASR system for children's speech. Second, acoustic features are speaker-dependent, particularly for children's speech. Compared to adults, children's vocal tracts are shorter and change with age. Hence, the resonances of the vocal tracts, or formant frequencies, are continuously shifting [MH12]. In previous research, various methods are proposed to alleviate the above problems [GG03,SHS03,SPL14]. Vocal tract length normalization (VTLN) [LR98] aims at alleviating the variabilities by utilizing a warping in the frequency domain [GGB07,PN03,SPL14]. Maximum likelihood linear regression (MLLR) [LW95], maximum a posterior (MAP) [GL94], and speaker adaptation training (SAT) [Woo01] are found to be helpful adaptation methods for children's speech [GWL14,SPL14]. To address the issue of limited data resources, multiple data augmentation methods are applied [FBL16, CNW20]. Most of the data augmentation methods are in-domain, which applies directly to the training data of the target task whereas out-of-domain data augmentation applies to the upstream task instead of the downstream target task. In our experiments, we investigated in-domain data augmentation only. The data augmentation methods we used are speed perturbation [KPP15], vocal tract length perturbation (VTLP) [JH13], SpecAugment (SpecAug) [Par19], volume perturbation [CNW20], pitch perturbation [CNW20], and F0-based normalization [YFA21a].

1.2 Problem 2: How to Utilize Untranscribed Children's Speech Data

Compared to transcribed speech data, un-transcribed speech data is easier to obtain from various sources. Transcribing speech data is time-consuming and costly. Thus, the exploitation of available limited speech data has been explored [IPM14, TSC13]. To leverage a large amount of un-transcribed speech data, unsupervised learning with pre-training is an effective method in utilizing the pre-trained model parameters as acoustic modeling initialization for model parameters in the downstream task [CHT19, OLV18, RFA20, FAA21]. Another effective method is semi-supervised learning [LGA02, SXK19, WMG20, KMD20, DMB19, CHC12]. In semi-supervised learning, pseudo-labels for the un-transcribed speech data are generated from a limited amount of transcribed speech data for further training.

1.3 Problem 3: How to Find a Good Model Initialization for Children's ASR in a Low-Resource Setting

As mentioned earlier, children's ASR is difficult due to the small size of publicly available databases. With such insufficient training data, the trained acoustic model is unable to generalize well due to optimizing to local minima during training. One effective solution is data augmentation, as mentioned in Section 1.1. Another possible solution, similar to the aforementioned unsupervised pre-training in Section 1.2, is to learn a good model initialization first, then adapt to the downstream task. Conventionally, supervised pre-training [TWM17] is a method with such a purpose. However, supervised pre-training is not able to adapt fast to an unseen task since it does not have a self-adaptation process in the pre-training phase. Thus, to alleviate such a problem, a self-adapted initialization, model-agnostic meta-learning (MAML) [FAL17, NAS18] based meta-initialization (MI) is used in the pre-training phase.

In this thesis, several difficulties in children's ASR are investigated and addressed with multiple methodologies. In Chapter 2, speaker variations in children's speech are alleviated with various data augmentation methods. In Chapter 3, to utilize untranscribed children's speech data, unsupervised learning with pre-train and semisupervised learning are used with various data augmentation methods. In Chapter 4, to find a good model initialization for children's ASR in a low-resource setting, a MAML-based MI is used with the proposed task-level augmentation. Chapter 5 concludes the paper and proposes future work.

CHAPTER 2

Towards Better children's Automatic Speech Recognition with Data Augmentation

In this chapter, we present our study on using different data augmentation for improving the robustness of children's Automatic Speech Recognition (ASR) systems.

2.1 Background

As mentioned earlier, there is a lack of large publically available children's speech databases, and one method to address that is data augmentation techniques. Multiple low-cost data augmentation methods are investigated, such as speed perturbation, vocal tract length perturbation (VTLP), SpecAugment (SpecAug), volume perturbation, pitch perturbation, and F0-based normalization.

Data augmentation has shown to result in performance improvement in many ASR tasks [CGK15,KPP15,TGN14,YFA21b,Par19,JH13,CNW20,WMG20]. Most state-of-the-art ASR systems consist of two types of NN-based systems: hybrid and end-to-end. Even for the traditional Hidden Markov Models-based (HMM-based) ASR systems, data augmentation improves adult speech [KPP15, Par19]. Another study showed the advantage of using data augmentation to improve the performance of children's ASR using neural network systems [CNW20]. However, the previous studies did not consider carefully the ASR of kindergarten-aged children (5 - 6 years old), which has been shown to be a particularly hard task compared to older children [YA18]. In order to verify the performance improvement in children's speech

(5 - 16 years old) with NN-based ASR systems, such as hybrid, and end-to-end ASR systems, experiments are conducted using data augmentation methods for both.

2.2 Data Augmentation Methods

In this section, data augmentation methods are discussed, such as speed perturbation, VTLP, SpecAug, and volume perturbation, along with some pitch warping-based data augmentation, such as pitch perturbation, and F0-based perturbation.

2.2.1 Vocal Tract Length Perturbation (VTLP)

Vocal tract length perturbation (VTLP) was the first successful augmentation method in the speech domain [JH13]. The key idea is inherited from vocal tract length normalization (VTLN) [LR98]. Similar to VTLN, VTLP linearly warps the frequency axis, in this case, of the acoustic features with a warping factor α. However, different from VTLN, which targets removing inter-speaker variations, the purpose of VTLP is to add variations to the entire input speech data by generating multiple folds of data with different warping factors.

The general approach of generating the warped frequency f' from the original frequency f is:

$$f' = \begin{cases} f\alpha & f \leq F_{high}\frac{min(\alpha,1)}{\alpha} \\ S/2 - \frac{S/2 - F_{high}min(\alpha,1)}{S/2 - F_{high}\frac{min(\alpha,1)}{\alpha}}(S/2 - f) & otherwise \end{cases} \tag{2.1}$$

where S is the sampling frequency, F_{high} is the boundary frequency empirically chosen that falls above the highest significant formant in speech. Based on Eq. (2.1), the center frequencies $f(i)$ for $1 \leq i \leq N$ filter-banks are warped to $f(i)'$ as the new center frequencies, and the triangular filter banks are generated at those new center frequencies $f(i)'$.

2.2.2 Speed Perturbation

Speed perturbation is one of the most effective augmentation methods in speech [KPP15]. As a warping-based method, a signal $x(t)$ is warped in the time domain by a factor α, generating a warped signal $x(\alpha t)$. However, speed perturbation considers the changes in duration or the number of frames.

2.2.3 SpecAugment (SpecAug)

SpecAugment is another widely-used augmentation method. It is an online augmentation method, which does not require additional computation costs. The idea composes of three augmentation techniques on the acoustic features: time warping, frequency masking, and time masking [Par19], where the two masking policies are adopted from the random masking algorithm Cutout [DT17] from computer vision.

Suppose we have τ time steps of the log Mel spectrograms, time warping is a strategy that warps a random point along the horizontal line passing through the center of the image within the time steps $(W, \tau - W)$, either to the left or right by a distance chosen from the uniform distribution of $(0, W)$ where W is the time warp factor along that line.

For frequency masking, it is applied by masking out f consecutive Mel frequency channels $[f_0, f_0 + f]$, where f is chosen from a uniform distribution $(0, F)$ such that F is the frequency mask parameter, and f_0 is chosen from $[0, v - f]$ where v is the number of Mel frequency channels.

Similar to frequency masking, time masking is applied by masking out t consecutive time frames $[t_0, t_0 + t]$, where t is chosen from a uniform distribution $(0, T)$ such that T is the time mask parameter and t_0 is chosen from $[0, \tau - t]$.

Both time and frequency masking have shown their effectiveness, whereas time warping has shown minor improvement [Par19]. Thus, time warping is not further investigated.

2.2.4 Volume Perturbation

Volume perturbation is an augmentation method that generates multiple folds of speech data with a warping factor α on the amplitude of the entire audio signal. The key idea is to increase the robustness of an ASR system with volume variations [GY19, CNW20].

2.2.5 Pitch Perturbation

Pitch perturbation is an augmentation method that shifts the pitch frequency of audio signals [CNW20]. The `pitch` function in SoX audio manipulation tool is used to shift the pitch frequencies within a warping factor [SoX]. For each utterance, we randomly select the warping factor from the search range for each utterance to increase the robustness of pitch variations.

2.2.6 F0-based Perturbation

F0-based perturbation is also an augmentation method that originated from F0-based normalization [YFA21b]. The key idea is based on the tonotopic distances between f_0 and each formant frequency. To measure this tonotopic distance consistently across different productions of the same vowel, normalization is applied to the formants of the vowel, with a default f_0 value, expressed as:

$$F(n)_{norm} = F(n)_{orig} - (f_{0,utt} - f_{0,def}), n \in \{1, 2, 3, \ldots\} \tag{2.2}$$

where $f_{0,utt}$ is the f_0 of the utterance, $f_{0,def}$ is a predetermined value of f_0 to represent a default speaker, $F(n)_{org}$ is the n^{th} formant after normalizing to $f_{0,def}$, and all frequencies in Eq. (2.2) are measured in the perceptual scale. Here, the Mel scale is used as the perceptual scale of choice.

To make Eq. (2.2) more reliable for children or speakers with high f_0 and formant values, the complication of formant estimations is disregarded. Instead, the entire spectrum is:

$$f_{norm} = f_{orig} - (f_{0,utt} - f_{0,def}) \tag{2.3}$$

where f_{orig} is the frequency of the original spectrum and f_{norm} is the corresponding frequency in the normalized spectrum. This represents that the frequency content in f_{orig} is shifted to f_{norm}. However, different from VTLN, the warping function is different, where VTLN is implemented by a piece-wise linear function, such as F3 normalization [CA06] and SGR normalization [GPY15], but F0-based normalization is nonlinear due to the shift in Mel scale. By applying F0-based normalization, inter-speaker variability is reduced.

F0-based perturbation can be used to create several copies of the original training data by performing the F0-based normalization. By extracting features using Eq. (2.3) multiple times with different $f_{0,def}$, different folds of speech are generated with variability in terms of tonotopic distances. Similar to F0-normalization, inter-speaker variabilities are reduced with this F0-based perturbation.

2.3 ASR Systems

Traditional ASR systems focus on maximizing a posterior probability, which is to find the best transformation from a sequence of n acoustic feature representations $X = (x_1, x_2, \cdots, x_n)$ into a sequence of n words $W = (w_1, w_2, \cdots, w_n), w_i \in V$ where V is the vocabulary. Then the expression for achieving the most probable sequence of n words $W^* = (w_1^*, w_2^*, \cdots, w_n^*)$ is:

$$\begin{aligned} W^* &= \arg\max_W P(W|X) \\ W &\in V^* \end{aligned} \tag{2.4}$$

where $\arg\max_W$ is the search space of the vocabulary V. The optimal model is achieved by finding the best posterior distribution $P(W|X)$ in Eq. (2.4). To be specific, optimizing an ASR model can be decomposed into the following steps: acoustic feature extraction from speech signals, acoustic modeling, language modeling, and decoding the sequence of words.

The Hidden Markov Model-Gaussian Mixture Model (HMM-GMM) was the representative traditional ASR model since the 80s. With the rise of deep learning in ASR, ASR systems have improved significantly. Most of the recent state-of-the-art ASR systems are NN-based. Two types of NN-based ASR systems emerged, hybrid and end-to-end (E2E) ASR systems [ODK22, DAR21, SLY11, BM12, SVK21]. Similar to an HMM-GMM-based ASR system, a hybrid system is composed of multiple sub-systems, such as acoustic model (AM), language model (LM), and pronunciation lexicon model (PLM), where each can be trained with a NN. It has the advantage of tunability and explainability since it trains every single module of the system separately. An end-to-end ASR system is a system with only one model, but applying an additional LM is helpful for resolving the grammar and spelling issues in CTC [Nak19]. Since the end-to-end ASR system is developed with one model only, it has the advantage of low training complexity.

2.3.1 Feature Extraction

Mel-frequency cepstral coefficients (MFCCs) [ZZS01] are widely used for HMMs [RM07, SS14, DPG09]. However, MFCC features are not preferred for deep learning-based acoustic modeling because MFCC features are decorrelated, but for deep learning acoustic modeling, data are correlated with a particular distribution [Moh14]. Instead, deep learning-based acoustic modeling use Mel filterbanks (MFbank) [SP03], which result in lower errors [DLH13]. The steps for extracting the MFCC features are shown below:

Step 1. Pre-emphasising to boost the energy in high frequencies.

Step 2. For windowing, common window length and window shift are 25 ms and 10 ms, respectively, which are just enough for capturing the stationary and dynamic information of current and neighboring frames. Hamming and Hanning windows are commonly used.

Step 3. DFT is applied to extract frequency information.

Step 4. Triangular Mel-scale filter banks and a logarithmic are applied to transform the outputs to log Mel-scale ceptrum. This scale better mimicks human perception.

Step 5. An inverse DFT (or DCT) is applied.

Step 6. A final liftering (as a synonym to filtering but in the cepstral domain) is applied to filter out only the necessary amount of coefficients (usually 13).

Note, if we are only extracting the MFbank features, we would stop at step 4 above.

2.3.2 Acoustic Modeling

An acoustic model (AM) for ASR could be either Hidden Markov Models (HMM) or deep neural networks (DNN). The goal for AM is to map the acoustic frames x_t to the phonetic state of the subsequent f_t:

$$\arg\max P(f_t|x_t) \tag{2.5}$$

For most of the DNN modeling, the ground truth of Eq. (2.5) are sequences of phonetic states at the frame level $f_{1:n}$ generated by a pre-trained HMM-GMMs, where GMMs model the acoustic frames, and HMMs predict the most probable phonetic states in a sequence. During DNN acoustic model training, the objective is to classify the frame-level phonetics optimized by a cross-entropy loss.

2.3.3 Language Modeling

The goal of a language model (LM) $P(W)$ is to find the most probable sequences of words. To be specific, it is to optimize the model $P(w|w_{pre})$ given previously recognized words w_{pre}.

Traditionally, N-gram LMs were developed to model the transition probabilities between words. However, it was found that RNN-based language modeling was better at capturing words out of the N-range, thus RNN-based N-gram language models were developed [CNB17].

2.3.4 Hybrid System

Before the widespread use of neural networks, the hidden Markov model (HMM) was the most reliable model for continuous speech recognition. An HMM-based system is composed of an AM, a PLM, and an LM. Each model needs to be estimated separately and then composited together for predicting the most probable word sequence given the acoustic observations with Weighted Finite-State Transducer (WFST) [MPR02]. Suppose we have $X = (x_1, x_2, \cdots, x_n)$ as the sequence of acoustic feature vectors (acoustic observations), and $W = (w_1, w_2, \cdots, w_n)$ as the word sequence. Then the expression for achieving the most probable word sequence $W^* = (w_1^*, w_2^*, \cdots, w_n^*)$ is:

$$\begin{aligned} W^* &= \arg\max_W P(W|X) \\ &= \arg\max_W P(X|W)P(W) \end{aligned} \tag{2.6}$$

where $\arg\max_W$ is the search space of the vocabulary, $P(W)$ represents the LM, and $P(X|W)$ represents the generative AM. Also, the goal of Eq. (2.6) is to find the best path of states that maximizes the likelihood of observable X, as expressed below:

$$P(x, a) = \prod_{t=1}^{T} p_{a_t} \sum_{j}^{m} C_{a_t,j} \prod_{d} \mathcal{N}(x_{t,d}; \mu_{a_t,j,d}, \sigma^2_{a_t,j,d}) \tag{2.7}$$

where x is the observations, a is the path, p_{a_t} is the probability of path a, $C_{a_t,j}$ is the weight for the j-th GMM component, $\mu_{a_t,j,d}$ and $\sigma^2_{a_t,j,d}$ are the mean and variance for the gaussian distribution, respectively, for path a_t with j-th GMM component and d as dimension. Thus, the best path is determined by the PLM and the m-component GMM. The PLM models the sequence of phones of the words. This step is usually referenced from a mapping table, which is a vocabulary table that maps each grapheme to the corresponding phoneme. For example, the word "apple" is mapped to "æpəl". Based on the mapping table, the PLM is modeled implicitly with the HMM. With the trained HMM model, an alignment of the phones and the audio frames (observables) is performed. Then, the final likelihood of a phoneme observation x given a state is modeled with the m-component GMM. Compared to a monophone model, a triphone model is usually used for better modeling coarticulation between phonemes, but this would increase the complexity of the GMM model. Thus, a forced alignment process between phonemes and HMM states, Viterbi decoding algorithm [For73], is applied, which reduces the complexity by computing the current maximum path in a one-time step. Then, the current maximum path is used for computing future maximum paths. Finally, each HMM model can be concatenated and looped back to handle continuous speech.

With the development and competition of the neural networks (NN) [Abd94, HH02], NN-based models commonly take place of GMMs with a much better performance in ASR modeling due to neural network's ability to incorporate nonlinear functions. Since DNN is not able to generate a conditional probability initially, it uses the posterior probability of the HMM state and replaces it with the GMM observation probability. This turns the traditional HMM-GMM model into the HMM-DNN hybrid model. Since each training needs an alignment of the inputs and the target states, a frame-by-frame force alignment is performed based on a prior HMM-GMM model. Later on, various architectures of HMM-DNN-based hybrid systems are developed with NN-based architectures, such as convolutional neural network (CNN) [CL13], time-delay neural network (TDNN), recurrent neural network (RNN) and long-short term memory (LSTM) [STB10], convolutional LSTM deep neural network (CLDNN) [SVS15]. And with the rapid emergence of the attention-based mechanism in NLP [VSP17], transformer-based NN architectures are developed [WML20, GQC20].

As mentioned in Section 2.3.3, n-gram LM is a common method applied to further improve performance by modeling the transition probabilities between words. To decode a model based on AM, PLM, and LM, a WFST is applied. WSFT has the ability to represent multiple transducers and optimize them well. For ASR, four transducers are introduced, such as word-level grammar (G), pronunciation lexicon (L), context-dependency (C), and an HMM (H). The inputs and outputs for each transducer are listed in Table 2.1 below.

Table 2.1: WFST Transducer mapping (G stands for word-level grammar, L stands for pronunciation lexicon, C stands for context-dependency, and H stands for HMM).

transducer	input sequence	output sequence
G	words	words
L	phones	words
C	C phones	phones
H	HMM states	C phones

The general idea of optimization in WFST is the composition of the four transducers with the following process:

$$H \circ C \circ L \circ G = \min(\det(H \circ \min(\det(C \circ \min(\det(L \circ G)))))) \tag{2.8}$$

The phonemes are expected to generate from the HMM states by Eq. (2.8). However, since the graph for WFST is too large to search for large vocabulary continuous speech recognition (LVCSR) [You96], a beam search [TN03], and the Viterbi algorithm is usually applied on top of WFST for constraining the search space to reduce the complexity.

2.3.5 End-to-End System

The previous section has shown how a hybrid ASR system can develop the mapping from acoustic features to HMM states in AM, from HMM states to words in PLM, and the transitions between words in LM. However, there are several problems in hybrid ASR systems. First, hybrid ASR requires multiple steps for training. As mentioned in Sec. 2.3, in order to train a DNN-based AM, the HMM-GMM model is required to train beforehand. The HMM states and the phonetic alignments are generated from the HMM-GMM model as the targets for DNN-based AM training. Second, the PLM is based on the handcrafted dictionary (usually represented by a word-to-phoneme mapping table), which is time-consuming to build. Third, although the finite state transducers could efficiently integrate all different modules in a hybrid ASR system together, designing well-optimized transducers to perform such integration would be complex. Lastly, the optimizations of each module are depending on different objectives. An implicit incoherence of each module is a possible drawback to the general objective of the entire hybrid system. Thus, the above issues always hinder researchers from efficiently and effectively developing applications for new languages on top of the hybrid ASR system. In this section, to address the above issues in hybrid ASR systems, a simplified ASR system, end-to-end (E2E) ASR system is discussed. The E2E ASR system contains only one NN architecture in its pipeline. The joint training of all modules into one enables the model to highly optimize based on the global objective and it directly maps the input acoustic features to the language representations without the requirement of PLM represented by a ground truth dictionary. For example, suppose we have $X = (x_1, x_2, \cdots, x_n)$ as the sequence of acoustic feature vectors (acoustic observations), and $W = (w_1, w_2, \cdots, w_n)$ as the word sequence. Then the expression for achieving the most probable word sequence $W^* = (w_1^*, w_2^*, \cdots, w_n^*)$ is:

$$W^* = \arg\max_W P(W|X) \tag{2.9}$$

As can be seen, Eq. (2.9) is not much different than Eq. (2.4). However, instead of factorizing $P(W|X)$ into a language model $P(W)$ and an acoustic model likelihood $P(X|W)$, the E2E system is trying to directly model the transformation between input acoustic features and output word sequences with one single model. Another difference in the E2E system is that it uses soft alignment by corresponding the audio signals to all possible states with a certain probability distribution instead of a distinct forced alignment [GJ14]. Common E2E models are of three types: connectionist temporal classification (CTC) [GFG06], attention-based encoder-decoder (AED) [CJL16], and recurrent neural network transducer (RNN-T) [BCC17].

CTC assumes the independency of the output labels at each time step, which means it enumerates all possible hard alignments and generates a soft alignment by combining those hard alignments. This step is usually decomposed into two sub-processes: path probability calculation and path aggregation. After passing the input sequence $X = \{x_1, \cdots, x_T\}$ into the encoder of CTC, a feature sequence is generated with the same length as the input sequence $F = \{f_1, \cdots, f_T\}$. Each unit of the feature sequence is a vector that is in the dimension of one more than the number of elements in vocabulary (the extra element represents a blank label "-"). The features are then passed to the CTC decoder with a softmax operation, generating a probability distribution sequence $Y = \{y_1, \cdots, y_T\}$ of the same length, where each unit $y_t = \{y_t^1, \cdots, y_t^{|V+1|}\}$ contains the probabilities of V wordpieces and one additional blank label at a time step t. Suppose the entire vocabulary $V^{'} = V \cup \{b\}$, $V^{'T}$ is the collection of all sequences of length T for vocabulary V'. The conditional probability distribution of one sequence π in the collection of $V^{'T}$ is:

$$p(\pi|X) = \sum_{t=1}^{T} y_t^{\pi_t}, \forall \pi \in V^{'T} \tag{2.10}$$

where π_t is the label at position t of sequence π. Each element π in $V^{'T}$ is one path. At this stage, the mapping from one input sequence X to one path π of the same length could be regarded as a hard-aligning process. However, since the input speech sequence is usually longer than its corresponding transcription, a many-to-one mapping step is needed to aggregate multiple paths into a shorter label sequence. This mapping consists of a contiguous label merging, which merges the identical labels appearing consecutively, and blank label deletion. For example, two different lengths of 7 paths "cc-aa-t-" and "c-aa-tt-" are aggregated to be "c-a-t-", and then blank labels "-" are deleted to generate "cat" of length 3 as the final sequence. As shown in the example, there are many possible paths of the one single word "cat", an aggregation step is required to merge them and calculate the probability $P(L|X)$ of all possible paths for the label sequence L:

$$P(L|X) = \sum_{\pi \in B^{-1}(L)} p(\pi|X) \tag{2.11}$$

where $B^{-1}(L)$ represents all paths in $V^{'T}$. Since it is hard to determine the number of paths from $V^{'T}$ is in $B^{-1}(L)$, dynamic programming is used to calculate the aggregation probability [GJ14, GFG06]. Since the alignment between input speech signals and output transcription is not depending on one certain path, the entire alignment process is usually regarded as a soft alignment.

However, the CTC model remains with the issue of independence of output sequences. One solution is to incorporate the attention mechanism for the CTC model (CTC-AED). Thus, the final objective function of the CTC-AED model is to estimate the posterior probability $P(L|X)$:

$$P(L|X) = \prod_{l} P(y_l|y_{1:l-1}, X) \tag{2.12}$$

where l is the output label index. The objective is to minimize the $-\ln P(L|X)$. The attention mechanism is to find the alignment between each element of the output sequence and the hidden states generated by the encoder. The decoder calculates the weight score between its hidden states with the states generated by the encoder at each input time. Then the formed temporal alignment distribution is used to extract an average of the corresponding encoder hidden states. This attention mechanism does not require any conditional independence assumptions, which address the problem of CTC. The attention mechanism can be divided into three types: content-based, location-based, and hybrid. The differences are from the weight score calculation. Content-based uses only the feature sequence and the previous hidden state to calculate the weight score at each position. However, it does not incorporate position information in the calculation. Thus, the location-based method addresses this issue by using the previous weight score as the location information at each step to calculate the current weight score. However, it is not using the input feature sequence, which lacks the information from input features. Thus, to combine the advantages of the above two mechanisms, the hybrid attention mechanism is usually applied. Thus, the hybrid CTC-AED model was developed and used in our experiments [WHK17a].

Other solutions to address the independence issue include incorporating LM [Nak19] and using a predictor and joiner combined RNNT architecture. The predictor is developed by a language model for contextual information, and the joiner is developed to combine the acoustic and context information. The one-to-many mapping is developed by omitting multiple tokens at a single acoustic unit. However, those solutions are not investigated in this thesis.

2.4 Experiments

2.4.1 Database

The database used in the study is the OGI Kids' Speech Corpus [SHC00]. The corpus is composed of both scripted and spontaneous speech from 1100 children from kindergarten through grade 10 (approximately from 5 to 16 years old), approximately 100 children per grade. The specific distribution of the data with respect to age is shown in Table 2.2. In this study, the total amount of scripted speech is approximately 70 hours for 71,999 utterances, and for spontaneous speech, it is approximately 30 hours for 1,101 utterances, combined to be approximately 100 hours for 73,100 utterances.

Table 2.2: Number of utterances in the OGI Kids' Speech Corpus, separated by grade level and speech style (scripted and spontaneous)

Grade	K	1	2	3	4	5	6	7	8	9	10
Scripted	3270	5866	7595	7596	6063	6649	7575	6539	6694	7181	6971
Spon.	87	87	114	114	91	98	111	96	99	102	102

2.4.2 Data Preprocessing

2.4.2.1 Feature Extraction

The features for the baseline HMM-GMM were MFCCs. The features were extracted with a frame length of 25 ms and a frame shift of 10 ms, 512-point DFT, 23 Mel filters, a pre-emphasis coefficient of 0.97, and a lifter coefficient of 22. For each frame, the first 13 MFCCs were kept for a 13-dimensional feature set.

MFCCs were widely used in traditional ASR-based HMM-GMM models and they are good representations of the perceptually relevant aspects of the short-term speech spectra. However, for the deep learning-based ASR model, it was found that the decorrelation by the discrete cosine transform used in the computation of MFCCs results in lower performance than just using Mel-Fbank features [DLH13]. Thus, the features used for both the AM in the deep learning-based hybrid model and end-to-end model training were Mel-Fbanks. The features were extracted with a frame length of 25 ms and frame shift of 10 ms, 512-point DFT, 23 Mel filters, pre-emphasis coefficient of 0.97. For each frame, 80-dimensional Mel-Fbank features were extracted. For the HMM-BLSTM model, an additional frame of features after each frame is then appended to form a 160-dimensional input. However, for the end-to-end model, 80-dimensional Mel-Fbank features were used. (find the study that did comparisons of features in terms of dimension)

2.4.2.2 Data Split

The data was split into a training set (70%), a development set (8%), and a test set (22%). The detailed statistics are shown in Table 2.3.

Table 2.3: Statistics of the Scripted, Spontaneous, and Combined datasets for OGI Corpus. The duration of utterances is analyzed. Minimum, maximum, and average(standard deviation) are measured in seconds. Durations are reported with a (min., max.) range.

Data	**Train (70%)**			**Dev (8%)**			**Eval (22%)**		
	# of Utt.	Duration	Avg.(Std.)	# of Utt.	Duration	Avg.(Std.)	# of Utt.	Duration	Avg.(Std.)
Scripted	50439	(1.21, 9.67)	3.49(1.55)	5482	(1.21, 9.65)	3.38(1.50)	16078	(1.21, 9.67)	3.52(1.57)
Spon.	774	(14, 479)	101.19(35.78)	83	(15, 157)	93.52(29.33)	244	(11, 225)	98.68(31.67)
Combine	63199	(1, 14.71)	3.70(1.83)	6755	(1, 14.1)	3.64(1.80)	16322	(1.21, 225)	4.95(12.28)

2.4.2.3 Speech Segmentation

From Table 2.3, we observe that the OGI spontaneous data contains utterances that are 100 seconds in duration. With such large utterance durations, the HMM-GMM model is unable to perform an accurate alignment of speech with its corresponding transcriptions. Moreover, training with long utterances is infeasible due to the limitation of computational power. In order to solve this problem, segmentation is applied to the audio signals and the corresponding transcription files of the entire database.

The segmentation process is conducted using the Kaldi toolkit. The segmentation is applied to the audio signals while retrieving their corresponding transcriptions. The transcript alignment was decoded with a biased language model (LM) trained on the raw transcripts. After the raw audio signals are segmented, the best-matching sub-sequence of words is obtained from the raw transcriptions [MPK17]. We use the segmented audio signals and the corresponding sub-sequence transcriptions for the rest of the experiments on the OGI spontaneous data. Details of how to choose the configuration parameters for segmentation are explained in Section 2.5.1.

2.4.3 Model Setup

2.4.3.1 Hybrid Model - Acoustic Model (AM)

The acoustic model training experiments are conducted on pykaldi2 [LXC19]. Since the HMM-BLSTM is effective in sequence data training [HS97, ZDV17], it is chosen as the acoustic model in all experiments. The model has 4 BLSTM layers with 512 hidden units in each direction per layer. The last layer transforms the outputs of the BLSTM to a probability distribution of 1,360, 1,408, and 3,760 states for the scripted, spontaneous, and combined(scripted + spontaneous) parts of the OGI corpus, respectively, generated by the HMM model. The dropout rate for the network is empirically set to 0.2 for all experiments. HMM models with the same hyperparameters are trained for scripted and spontaneous data, and for the combined dataset, larger hyperparameters are used.

2.4.3.2 End-to-End Model

The end-to-end model training is conducted on the ESPnet, a PyTorch-based end-to-end speech processing toolkit. The CTC-AED-based end-to-end model is comprised of a transformer block with a 12-layer encoder, and a 6-layer decoder. The dimension of attention and feed-forward layer is set to 256 (4 heads) and 2048, respectively. Based on the vocabulary from only the training set, a unigram algorithm is applied to generate the number of word-pieces for the scripted, spontaneous, and combined corpus, which are 564, 3,861, and 3,766, respectively. The dropout rate in the network is empirically set to 0.1.

2.4.3.3 Language Model Setup

For the hybrid system, the lexicon and language models from the original Librispeech corpus are used, where the 14M tri-gram (tgsmall) language model is used for decoding, and the 1.3G 4-gram (fglarge) language model is used for rescoring. For the end-to-end system, all experiments are conducted with no external language models.

2.4.4 Data Augmentation Setup

To compare different augmentation strategies fairly, all training data are increased by 3 folds with offline augmentation. For speed perturbation and VTLP, the warping factors are (0.9, 1.0, 1.1) since such a small range results in reasonable warpings. In VTLP, the boundary frequency (F_{high}) is chosen to be 7,800 Hz. For volume perturbation, the warping factors are randomly chosen from 0.125 to 2. For pitch perturbation, the warping factors are randomly chosen from -160 to 160, where each unit is 100th of a semitone. For F0-based perturbation, $f_{0,utt}$ is chosen to be the median of all training utterances using the multi-band summary correlogram (MBSC) pitch detection algorithm [TA13], $f_{0,def}$ are chosen to be (80, 100, 120) Mels, which is (85.93, 100, 114.32) Hz. Lastly, for SpecAug, since it is an online augmentation strategy with time and frequency masking, no fold increase is applied. Two maximum width of 5 masks and two maximum width of 8 masks are empirically chosen for the frequency and time channels, respectively.

2.5 Results and Discussions

2.5.1 Speech Segmentation

Table 2.4: Segmentation experiments with different configuration thresholds applied to the Spontaneous dataset of the OGI Corpus. The second, third, fourth, fifth, and sixth columns represent the segmentation configurations. The last two columns are the % WER for the development set and evaluation set based on the training on segmented data using HMM-GMM and HMM-BLSTM models. The best performance is bold-faced. 4-gram language model rescoring is applied to all systems.

Exp #	**Segmentation Configuration (higher bound)**					**HMM-GMM**		**HMM-BLSTM**	
	dur	*seg len merge*	*bad ppt*	*wer*	*seg len split*	**Dev**	**Eval**	**Dev**	**Eval**
a	30	15	0.75	50	30	50.94	50.35	37.67	36.84
b	30	30	0.75	50	30	51.95	51.54	38.25	37.52
c	30	45	0.75	50	30	51.73	51.16	38.39	37.50
d	30	60	0.75	50	30	51.73	51.16	38.39	37.50
e	30	15	0.95	50	30	51.47	50.50	37.07	36.75
f	30	15	0.55	50	30	51.65	51.01	37.54	37.03
g	30	15	0.75	60	30	51.49	50.52	37.50	36.97
h	30	15	0.75	40	30	51.27	50.71	37.35	36.70
i	30	15	0.75	30	30	52.22	51.17	37.68	37.00
j	**30**	**15**	**0.75**	**40**	**45**	**50.94**	**50.35**	**36.94**	**36.48**
k	30	15	0.75	40	15	51.82	50.97	37.37	36.41
l	30	15	0.75	40	60	50.94	50.35	36.94	36.48
m	20	15	0.75	40	45	53.55	52.85	38.45	37.68
n	40	15	0.75	40	45	51.57	50.32	37.52	36.84

The segmentation experiments are conducted using Kaldi. The biased LM is trained from 960 hours of LibriSpeech. As shown in Table 2.4, the higher bounds of the five factors are grid searched: *dur*, *seg len merge*, *bad ppt*, *wer*, and *seg length split*. *dur* is to control the maximum duration of the output segments. *seg length merge* is to control the length of the segments before merging them together. The algorithm will keep splitting the long segments into segments that are smaller than the size chosen for *seg length merge*. *bad ppt* is to control the proportion of the maximum length of silence, junk, and incorrect words into a merged segment that is less than the fraction of the total length of the merged segment. *wer* is to control the maximum word error rate (WER) of merged segments compared to the original speech data when merging together. And *seg len split* is to split the segments longer than this size into smaller segments. Among these five factors, only *dur* is a factor when splitting the input into uniform segments, and the other four are the factors when creating segmented speech data with its corresponding annotations. After performing the segmentation on the OGI spontaneous speech data with each set of configuration parameters, an HMM-GMM and an HMM-BLSTM model are trained based on the segmented speech data. The performance of each set of configuration parameters is evaluated with WERs after decoding from the model trained with the segmented dataset in the HMM-GMM and HMM-BLSTM systems, respectively. After grid searching with 14 different configurations as shown in Table 2.4, it is shown that the configuration parameters of experiment j has the best performance in WER on both the development and evaluation dataset on HMM-GMM and HMM-BLSTM models. Thus, the configuration parameters of experiment j are chosen for the OGI spontaneous data.

2.5.2 Baseline Hybrid and End-to-End ASR models

The baseline experiments are conducted on the HMM-GMM, HMM-BLSTM, and CTC-AED models. HMM-GMM is our baseline model [SHC00]. HMM-BLSTM was the SOTA method (at the time of the experiments) in a hybrid ASR system that uses a deep learning model

Table 2.5: % Word Error Rate (WER) for HMM-GMM, HMM-BLSTM, CTC-AED on the OGI Scripted (Scripted), Spontaneous (Spon.), and Combined (Combine) datasets. 4-gram language model rescoring is applied for all systems.

Data	**Model**	**Dev**			**Test**		
		Scripted	Spon.	Combine	Scripted	Spon.	Combine
Scripted	HMM-GMM	30.67	72.58	51.17	35.17	71.95	52.89
	HMM-BLSTM	9.57	86.98	45.88	10.69	87.06	45.78
	CTC-AED	1.90	98.70	46.90	2.80	98.90	46.90
Spon.	HMM-GMM	61.84	53.90	59.96	64.03	53.76	61.09
	HMM-BLSTM	51.79	39.72	47.07	54.12	39.06	48.07
	CTC-AED	103.20	84.90	93.30	104.30	89.20	95.40
Combine	HMM-GMM	26.24	52.14	40.06	30.93	51.24	41.98
	HMM-BLSTM	8.31	40.43	23.38	9.04	39.31	22.92
	CTC-AED	1.80	82.20	39.00	2.70	82.90	39.30

for the acoustic model training [YFA21a, FAA21]. Moreover, the CTC-AED ASR model was a very common and effective one in ASR training [WHK17b], as well as in children's ASR [NLP20]. Performance in the speech data of different speaking styles, scripted and spontaneous, is also evaluated using the OGI Kids' speech, for both in-domain and out-of-domain speech tasks.

2.5.3 Speaking-Style Experiments

In this part, we further investigated the scripted-spontaneous speaking-style mismatch problem [LLH16]. Experiments for both in-domain, out-of-domain, and combined speaking styles are conducted respectively.

2.5.3.1 In-domain Speaking-Style

First, let's take a look at the hybrid ASR system performance for the in-domain speaking style experiments from Table 2.5 (column scripted-row scripted and column spontaneous-row spontaneous). When training with scripted speech/testing with scripted speech, the traditional HMM-GMM can achieve 30.67 % and 35.17 % WER for the scripted-to-scripted in-domain task on development and test set, respectively. However, for training with spontaneous speech/testing with spontaneous speech, the performance is much worse, in 53.90 % and 53.76 % WER, respectively.

HMM-BLSTM outperforms the results from the HMM-GMM in all experiments trained and evaluated on the speech of the same speaking style. HMM-BLSTM achieves nearly 70 % and 27 % WER improvement for both development and evaluation set on scripted and spontaneous data, respectively.

For the end-to-end system, CTC-AED is applied. As shown in Table 2.5, it outperforms significantly the HMM-BLSTM in the train on scripted/test on scripted by nearly 70 % WER relative improvement for both the development and test scripted data. We hypothesize this is due to the equivalent format in most of the scripted utterances, which is much easier to generalize by an end-to-end ASR system. However, for experiments training on spontaneous speech, the performance degrades due to the long-form utterances in the development and evaluation set, commonly seen as an utterance length mismatch problem in the end-to-end system. This problem could be resolved by either increasing data diversity through multidomain training or simulating long-form characteristics during training [NPC19]. This will be addressed in future work.

2.5.3.2 Out-of-domain Speaking-Style

Out-of-domain speaking-style experiments are also conducted, as shown in Table 2.5 (column spontaneous-row scripted and column scripted-row spontaneous). When training with the

HMM-GMM model, both training on scripted speech /testing on spontaneous speech, and training on spontaneous speech/testing on scripted speech have a much worse performance compared to the in-domain results (training on spontaneous speech/testing on spontaneous speech and training on scripted speech/testing on scripted speech). This problem is commonly termed a speaking style mismatch. The same problem still exists when training with the deep learning-based HMM-BLSTM hybrid model and the CTT-AED end-to-end model. In the next section, a simple data style combination is adopted to increase the robustness of the model.

2.5.3.3 Combination of different Speaking-Style

For the speaking-style combination, a simple combination of the scripted and spontaneous speech data is performed and trained with the three models: HMM-GMM, HMM-BLSTM, and CTC-AED. Also, as mentioned in Sec. 2.4, due to the constraint of the GPU computation resources, segmentation is applied to the spontaneous speech data before combining it with the original scripted speech data. As shown in Table 2.5, the model trained with the combined data is tested not only on the combined data but also on the scripted and spontaneous data individually. It is initially observed that incorporating training speech data of the multiple speaking styles could improve the performance of the model for all three models (HMM-GMM, HMM-BLSTM, and CTC-AED), but it could also be due to the increase of the training data size. Thus, further experiments are needed. Second, a further improvement is observed when testing the scripted, and spontaneous data separately for all three models as well, achieving nearly 10 % WER improvement on scripted, and about 3 % WER improvement on spontaneous. This validates that incorporating the speaking style into the training data improves performance.

Table 2.6: % Word Error Rate (WER) for HMM-GMM, HMM-BLSTM, CTC-AED on the OGI Scripted (Scripted), Spontaneous (Spon.), and Combined (Combine) datasets. 4-gram language model rescoring is applied for all systems.

Data	**Model**	**Dev**			**Test**		
		Scripted	Spon.	Combine	Scripted	Spon.	Combine
Scripted	HMM-GMM	26.84	70.43	48.17	31.22	68.86	49.25
	HMM-BLSTM	7.76	86.18	44.67	8.81	86.13	44.53
Spon.	HMM-GMM	56.93	50.94	56.10	59.40	50.35	56.72
	HMM-BLSTM	47.66	36.94	43.98	49.34	36.48	44.30
Combine	HMM-GMM	23.03	49.29	36.95	27.74	48.19	38.74
	HMM-BLSTM	6.43	38.51	21.52	7.18	36.99	20.09

2.5.4 Language Model Rescoring

For the hybrid systems, a four-gram language model of LibriSpeech is applied and the results are shown in Table 2.6. After rescoring with the 4-gram language model of LibriSpeech, all settings can achieve nearly 10 % to 20 % improvement in WER. Due to time constraints, no external language model is applied for the end-to-end systems.

2.5.5 Hybrid and End-to-End ASR models with Data Augmentation

Table 2.7: % Word Error Rate (WER) for applying different data augmentation methods to the hybrid ASR system (HMM-BLSTM modeling) on OGI Scripted (Scripted), Spontaneous (Spon.), and Combined (Combine) datasets. 4-gram language model rescoring is applied for all systems.

Train Data	**Aug. Technique**	**Dev**			**Test**		
		Scripted	Spon.	Combine	Scripted	Spon.	Combine
Scripted	No Aug.	7.76	86.18	44.67	8.81	86.13	44.53
	Speed	7.08	90.18	47.07	7.91	90.60	47.71
	VTLP	7.12	89.91	47.26	8.13	90.58	48.07
	SpecAug	7.67	88.41	46.45	8.66	88.47	46.91
	Volume	7.28	88.83	47.12	8.30	88.65	47.49
	Pitch	7.17	89.37	46.80	8.07	89.82	47.39
	F0-norm	7.34	90.85	46.52	8.21	90.81	46.27
Spon.	No Aug.	47.66	49.34	36.94	36.48	43.98	44.30
	Speed	46.78	48.60	34.46	34.23	41.93	42.71
	VTLP	47.61	50.08	35.08	35.08	42.70	43.95
	SpecAug	47.74	49.28	36.95	36.32	43.96	44.00
	Volume	49.01	50.90	36.58	35.74	44.32	44.61
	Pitch	48.21	50.73	36.33	36.12	43.90	44.84
Combine	No Aug.	6.43	38.51	21.52	7.18	36.99	20.09

Table 2.8: % Word Error Rate (WER) for applying different data augmentation methods to the end-to-end ASR system (CTC-AED modeling) on OGI Scripted (Scripted), Spontaneous (Spon.), and Combined (Combine) datasets. 4-gram language model rescoring is applied for all systems.

Train Data	**Aug. Technique**	**Dev**			**Test**		
		Scripted	Spon.	Combine	Scripted	Spon.	Combine
Scripted	No Aug.	1.90	98.70	46.90	2.80	98.90	46.90
	Speed	1.50	97.30	46.30	2.10	97.40	46.10
	VTLP	1.70	99.10	47.10	2.70	99.30	47.00
	SpecAug	1.60	99.10	47.10	2.30	99.40	46.90
	Volume	1.90	99.20	47.20	2.60	99.40	47.00
	Pitch	1.50	98.90	46.90	2.00	98.80	46.50
	F0-norm	1.40	97.60	47.30	2.10	97.80	47.30
Spon.	No Aug.	103.20	84.90	93.30	104.30	89.20	95.40
Combine	No Aug.	1.80	82.20	39.00	2.70	82.90	39.30

A further experiment on the effectiveness of different data augmentation methods is investigated, such as speed perturbation, VTLP, SpecAug, volume perturbation, pitch perturbation, and F0-normalization. As shown in Tables 2.7 and 2.8, improvements are observed in all the data augmentation methods compared to using no augmentation on both development and test scripted data, where around 10 % WER reduction is observed for the HMM-BLSTM based hybrid ASR system and around 25 % WER reduction is observed for CTC-AED based end-to-end ASR system. For spontaneous speech, some improvement could be observed for the HMM-BLSTM based hybrid ASR system by using data augmentation techniques, such as speed perturbation and SpecAug. It is observed from Table 2.7 that speed perturbation can still achieve the best performance overall in all scenarios. Also, we can observe that after applying data augmentation to an out-of-domain task, there is nearly no improvement in the testing performance, even when the testing data is combined with the in-domain data (combined data). From Table 2.8, a similar observation can be achieved as observed from Table 2.7. However, due to various problems in training spontaneous speech data on an end-to-end system, such as GPU computation constraints and length-mismatch for segmented speech, the data augmentation experiments are not investigated on spontaneous speech and combined speech data.

2.6 Summary and Conclusion

This chapter has presented the performance of different data augmentation methods applied to the children's ASR for two neural network-based ASR systems, the HMM-BLSTM-based hybrid ASR system, and the CTC-AED-based end-to-end ASR system. Data augmentation is an efficient and low-cost method for improving the robustness of children's ASR, as a low-resource task. Most of the data augmentation methods are able to address the speaker variations of the children's speech by warping the speech signal. In particular, frequency-based warping methods, such as vocal tract length perturbation, speed perturbation, and SpecAugment, and pitch-based warping methods, such as pitch perturbation, and F0-norm, are improving the robustness of children's ASR by around 10 % WER reduction on HMM-BLSTM and 25 % WER reduction on CTC-AED.

In addition, some of the speaking style experiments and hybrid and end-to-end ASR systems comparison experiments are conducted. For scripted speech, CTC-AED can achieve better performance than HMM-BLSTM; however, for spontaneous speech, CTC-AED is not able to train well, which is possibly due to the constraint of the data size and the long-form utterances. To address the above two possible issues for CTC-AED-based end-to-end ASR systems, further investigations are needed. Combining both scripted and spontaneous speech not only increases the amount of data in training but also fuses the speech style, which relieves the speaking style mismatch. Furthermore, data augmentation methods are generally helpful in improving the performance of the children's ASR on both HMM-BLSTM and CTC-AED-based ASR systems. However, most of the data augmentation methods require much more computation resources compared to no augmentation, which could be a constraint.

CHAPTER 3

Towards Better Children's Automatic Speech Recognition with Untranscribed Data

In this chapter, our study [WZF21] for low-resource non-native children's ASR in German is investigated. We explored the usage of the untranscribed data with unsupervised pre-training and semi-supervised learning (SSL) to improve the robustness of children's ASR, in addition to various augmentation methods.

3.1 Background

To address the low-resource issue of the speech data, particularly the non-native children's data, data augmentation has verified its effectiveness in improving children's ASR by increasing the variability of the data as shown in Chapter 2. In this chapter, in addition to data augmentation, utilizing the available untranscribed data to improve ASR performance is investigated. Two well-known effective approaches for utilizing untranscribed data, unsupervised pre-training and semi-supervised learning are investigated. Lastly, since many unprocessed recordings contain non-speech segments within utterances, a non-speech state discriminative loss (NSDL) is proposed to discriminate the non-speech segments from the speech part of the utterance during acoustic model training.

3.2 Data Augmentation

As explained in Section 2.2 of Chapter 2, data augmentation is effective in mitigating the low resource issue for data, such as children's speech, and increasing the robustness of the model. It is applied in different stages of the training pipeline. Speed perturbation, vocal tract length perturbation (VTLP), volume perturbation, noise augmentation, and pitch perturbation are used for the transcribed speech data, whereas speed perturbation and noise augmentation are used for the untranscribed speech data. For noise augmentation, the removed non-speech utterances are used as foreground and background noise to augment the original audio with various random SNRs. Speed perturbation uses a conventional method with 3-way warping factor of 0.9, 1, and 1.1. Volume perturbation, pitch perturbation, and VTLP are implemented using Kaldi scripts with random warping factors. We also tried SpecAug for our experiments, but the technique does not improve performance. We will detail the data augmentation schemes for each training stage in Section 3.6.

3.3 Usage of Untranscribed Data

Due to the ease of acquirement of the untranscribed data, as well as children's speech data, unsupervised pre-training and semi-supervised learning are used to tackle the issue with a limited amount of transcribed speech data.

3.3.1 Unsupervised Pre-training

For unsupervised pre-training, the speech itself is regarded as the supervision, such as autoregressive predictive coding (APC) [CHT19], contrastive predictive coding (CPC) [OLV18], and bi-directional autoregressive coding (Bi-APC) [FAA21]. The core mechanism of all three unsupervised pre-training methods is to predict a future frame given all the past frames. Moreover, Bi-APC is able to perform the prediction on the past frames given all the future frames, in a bi-directional fashion. This behavior perfectly matches with the state-of-the-art (SOTA) bi-directional models, such as BLSTM and Bi-Transformer. Also, bi-directional autoregressive predictive coding (Bi-APC) has shown its effectiveness in children's speech in the previous work [FAA21].

3.3.2 Semi-Supervised Learning (SSL)

Similar to unsupervised pre-training, semi-supervised learning also has shown its effectiveness in leveraging the untranscribed data [LGA02, SXK19, WMG20, KMD20, DMB19, CHC12], though not in children's speech. The mechanism is to generate the pseudo labels for untranscribed data based on the transcribed data training. To ensure the quality of the pseudo labels, an incremental SSL is applied for filtering out the utterances with log-likelihood lower than the threshold, as a comparison to the conventional one-shot SSL.

3.4 Non-Speech State Discriminative Loss (NSDL)

Table 3.1: Statistic of non-speech and speech states in the 5-hour transcribed data.

HMM States	**Total**	**Avg (Utt)**	**Std (Utt)**
Non-speech	$11\text{x}10^5$	817	875
Speech	$5\text{x}10^5$	352	519

Table 3.1 shows the statistics of non-speech and speech states, using forced alignment from the trained GMM model, in the dataset. This shows the dominance of the long-duration non-speech segments within speech utterances in the dataset. As shown in Table 3.1, the number of non-speech states is more than twice the number of the speech states, which is a possible cause of the overfitting to the non-speech states, resulting in more deletion errors. Thus, by separating the probability distribution of all pdf-ids into two parts, NSDL is proposed to balance the data, in terms of speech and non-speech, during acoustic model

training, and better discriminate the speech and non-speech states.

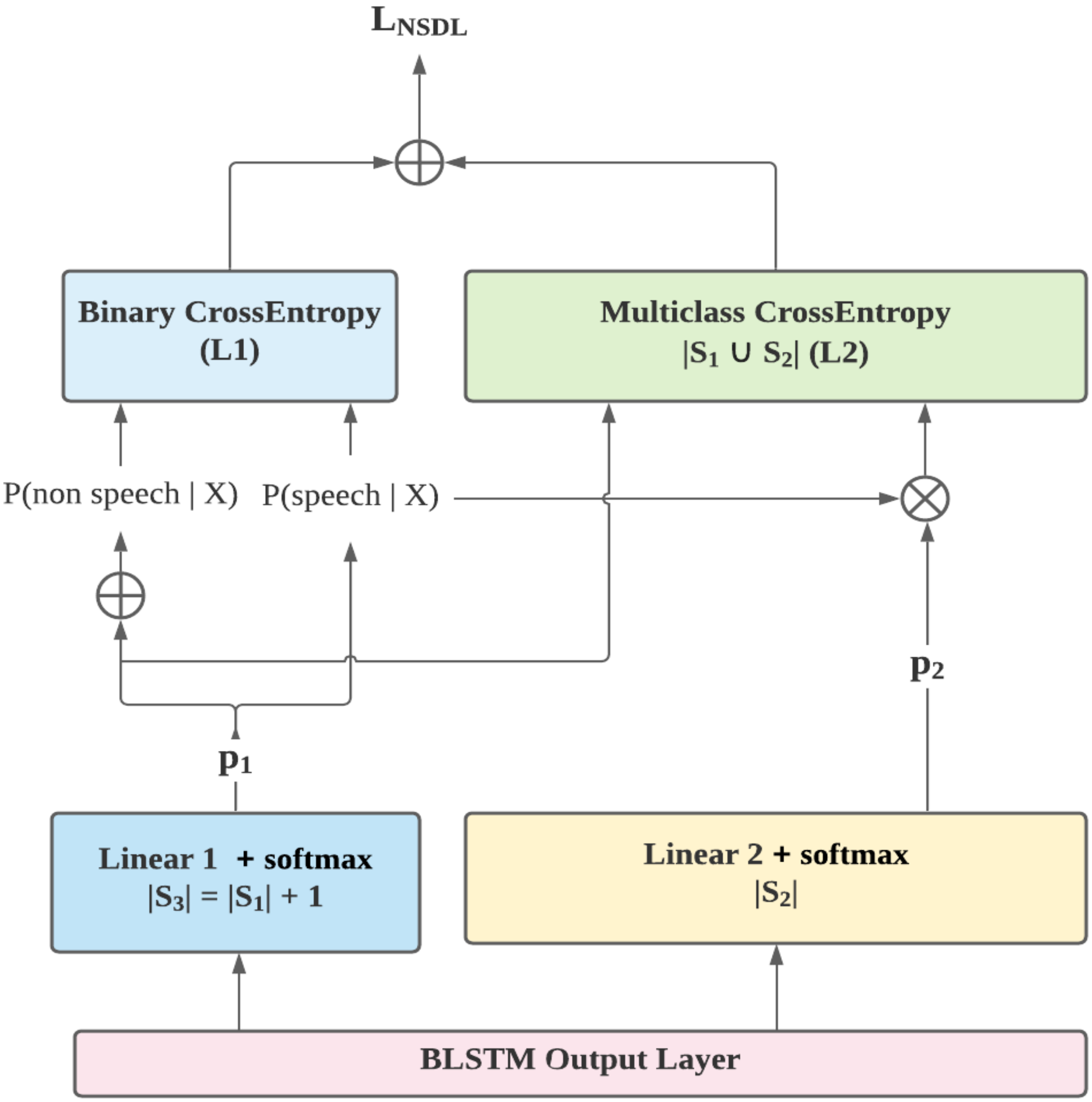

Figure 3.1: A Schematic Diagram of NSDL

The schematic diagram of NSDL is shown in Figure 3.1. Suppose the sets of non-speech states and speech states are S_1 and S_2, respectively. The original acoustic model outputs a probability distribution over all states in $S_1 \cup S_2$. Instead of doing a single classification task among all HMM states, the last BLSTM layer output is fed into two separate fully connected layers to obtain two probability distributions p_1 and p_2 over states in $S_3 = \{S_1, \mathit{speech}\}$ and S_2, respectively. The symbol *speech* is a placeholder reserved to represent a general speech state, regardless of specific phoneme states. The probability of a state belonging to *speech* state is denoted as P_{speech}. We can also obtain $P_{\text{non-speech}}$ by summing up all non-speech states in S_1. The two probabilities are used to construct a speech/non-speech binary classifier and are shown in the left path of Figure 3.1. The output probability distributions p_1 and p_2 are combined to construct the original HMM states classification problem as plotted in the right path of Figure 3.1. Let X be the input sequence and $Y = \{y_1, ..., y_t, ..., y_T\}$ represents the results from forced alignment, the ground truth for binary classification is $b_t = \mathbb{1}(y_t \in S_2)$. The original acoustic model training is reformulated into a multi-task training as:

$$L_{\text{NSDL}} = L_1 + \lambda L_2 \tag{3.1}$$

in which

$$\begin{aligned}
L_1 &= -\sum_{t=1}^{T}(\mathbb{1}(b_t == 1)\log(p_1(\text{speech}|X)) + \\
&\qquad \mathbb{1}(b_t == 0)\log(P(\text{non-speech}|X))) \\
L_2 &= -\sum_{t=1}^{T}\log(P(s = y_t|X)) \\
P(\text{non-speech}|X) &= \sum_{s \in S_1} p_1(s|X) \\
P(s|X) &= \begin{cases} p_1(s|X) & s \in S_1 \\ p_1(\text{speech}|X)p_2(s|X) & s \in S_2 \end{cases}
\end{aligned} \tag{3.2}$$

where L_1 is the loss function for the binary classifier for non-speech and speech states, and L_2 is the cross entropy loss for the classifier among all HMM states. λ denotes the task ratio for L_2, which is empirically set to 1.

In addition, we apply weights to the loss with respect to non-speech/speech classes in L_2 to further alleviate the problem that the model is more likely to classify frames as non-speech. The weights of non-speech states and speech states are set to 0.9 and 1. Hence, the model will assign a larger penalty when it misclassifies a speech frame as a non-speech frame.

3.5 Experiments Setup

3.5.1 Database

The TLT2021 Challenge Corpus is a corpus that contains 5 hours of transcribed training data from 296 students, about 1,445 utterances, and 1 hour of transcribed development data from 72 pupils, about 339 utterances, and 60 hours of untranscribed training data from 124 pupils, about 10047 utterances. Two text sources are provided: (1) manual transcriptions for 5 hours of the training data, and (2) written data extracted from sentences written by the pupils. Participants' ages range from 9 to 16.

3.5.2 Data Preprocessing

Before training the baseline, we observed that there are some non-speech utterances in the provided transcribed dataset. Hence, we remove non-speech utterances that have one or more of the following transcriptions, @sil, @noise, @laughs, @hes. As a result, 211 non-speech utterances out of 1445 utterances are eliminated in the transcribed training data.

3.5.3 Feature Extraction

For the baseline GMM model, 39-dimensional Mel-frequency cepstral coefficients (MFCCs), including first and second derivatives, are extracted with a frame length of 25 ms and a frame shift of 10 ms, 512-point DFT, 23 Mel filters, pre-emphasis coefficient of 0.97, and lifter coefficient of 22. For the TDNN model training, 40-dimensional high-resolution MFCCs are extracted with the same settings except for Mel filters of 40. For the HMM-BLSTM-based model training, 80-dimensional log-Mel-filter banks are first extracted every 10 ms with a frame length of 25 ms. An additional frame of features after each frame is then appended to form a 160-dimensional input.

3.5.4 Acoustic Model Setup

For the TDNN model, a 3-way speed perturbation is applied to both the transcribed and untranscribed datasets with factors of 0.9, 1.0, and 1.1 to form the inputs of the models. A TDNN model is first trained with the 3-fold transcribed data, and then fine-tuned with the weighted combination of the 3-fold transcribed and 3-fold untranscribed data using one-shot SSL [GMF21].

The HMM-GMM model is trained with data after preprocessing to obtain the frame-level alignment for the DNN-based acoustic model training, particularly for the HMM-BLSTM model. The HMM-BLSTM model consists of 4 BLSTM layers with 512 hidden units in each direction, followed by a layer transforming outputs of BLSTM to a probability distribution of 2392 states from HMM model. The dropout rate in the network is set to 0.3. The model is first trained with a sequence-wise mechanism where each utterance is fed into the model as one instance (1169 frames on average). However, as shown in Table 3.1, the variance of the duration in utterances is large, leading to many unnecessary paddings. Moreover, BLSTM does not model sentences over 3,000 frames well. To mitigate the gradient vanishing problem in BLSTMs for modeling long-duration utterances and accelerating the training, the model is then trained in a chunk-wise mechanism where one instance is a segment consisting of 300 frames with 10 appended neighboring frames. The appended frames are only used in training for accumulating the BLSTM hidden states.

3.5.5 Language Model Setup

The baseline language model is based on the provided multi-lingual lexicon for both English and Italian words since the mother tongue of the children is either English or Italian [GMF21].

An additional RNNLM [MKD11, Mik12] is trained as the second-pass rescoring with the provided scripts in Kaldi. The model consists of two LSTM layers with a hidden size of 128. The dimension of word embedding is 256. A grid search is performed on the language model weights of 0.25, 0.3, and 0.35, and n-gram order of 2 and 3. The best settings for the development set in each experiment are used to rescore the lattice of the evaluation set.

3.6 Results and Discussions

Experiments are conducted using Pykaldi2 [LXC19] for acoustic model training and Kaldi [Pov11] for decoding. The results of the ASR experiments using different methods are shown in Table 3.2.

Table 3.2: % Word Error Rate (WER) for different systems on the TLT-2021 development and evaluation sets, including chunk-wise modeling, NSDL, Bi-APC, different iterations of SSL, and LM rescoring (in parentheses). The second and third columns represent augmentation schemes for the transcribed and untranscribed datasets, respectively. SP: speed perturbation, Pit.: pitch perturbation, Vol.: volume perturbation, VTLP: vocal tract length perturbation, and Noise: noise perturbation. The last line (+dev) indicates that the development data is included in the training for the open track case. The best performance in each case is bold-faced.

Model	**Transcribed Data**	**Untranscribed Data**	**Dev(%)**	**Eval(%)**
Official Baseline	3x SP	–	52.38	45.21
		Proposed Systems		
BLSTM Baseline	3x SP	–	57.54	56.63
+ chunk	3x SP	–	54.27	50.09
+ NSDL	3x SP	–	51.52	48.54
+ Bi-APC	3x SP	3x SP	49.91	44.60
+ SSL iter1 (0.35)	2x Pit. 3x Vol. 2x VTLP 3x SP	3x SP	49.91	43.55
+ SSL iter2 (0.3)	2x Pit. 3x Vol. 2x VTLP 3x SP	3x SP	49.05	41.11
+ SSL iter3 (0.28)	2x Pit. 3x Vol. 2x VTLP 3x SP	3x SP	**47.10(46.30)**	41.23(41.05)
	2x Pit. 3x Vol. 2x VTLP 3x SP	2x Noise 3x SP	47.79(46.99)	41.23(40.81)
	2x Pit. 3x Vol. 2x Noise 3x SP	3x SP	47.22(46.59)	41.28(39.86)
	2x Pit. 3x Vol. 2x VTLP 2x Noise 3x SP	3x SP	48.02(46.61)	**40.87(39.68)**
+ dev	2x Pit. 3x Vol. 2x VTLP 2x Noise 3x SP	3x SP	–	39.92(38.85)

3.6.1 HMM-BLSTM Baseline and chunk-wise training

A developed HMM-DNN hybrid system with BLSTM modeling is applied as our baseline. It can achieve 57.54 % and 56.63 % WER for development and evaluation datasets, respectively. Since this performs much worse than the official baseline, we experiment with chunk-wise training instead of sequence-wise training, which uses a chunk size of 300 frames, and left and right context chunks of 10 frames each. Table 3.2, shows that the chunk-wise training improves the performance over the sequence-wise training mechanism by 5.68% and 11.55% in relative WER for the development and evaluation datasets, respectively.

3.6.2 Non-Speech State Discriminative Loss (NSDL)

When using our proposed NSDL method, we further achieve a relative improvement of 5.07 % and 3.09 % on the development and evaluation datasets, respectively, compared with the chunk-wise training mechanism. A more fair baseline for NSDL is using the default VAD in Kaldi to filter out the silence frames, which has a WER of 54.33 % for the development set. The result is even worse than the model without VAD. The reason may be that many non-speech states are not silenced states, but laugh, hesitation and noise. The proposed NSDL method is more suitable in this case. Starting from Section 3.3, we use the untranscribed data to further improve the performance. The relative WER will be in reference to the official baseline.

3.6.3 Bidirectional Autoregressive Predicting Coding

The first method we used to train the 60 hours of untranscribed data is Bi-APC. Using the same data augmentation strategy for transcribed data, we apply 3-way speed perturbation to the untranscribed data. As shown in Table 3.2, better performance is obtained by using Bi-APC. Both results of WER for the development and evaluation data are further improved, and relative improvements of 4.72% and 1.35%, respectively, are observed, compared to the official baseline system.

3.6.4 Semi-supervised Learning

Incremental SSL and various augmentation strategies are applied. As shown in Table 3.2, the first iteration of SSL chose 0.35 as the log-likelihood threshold, by applying a 10-fold (2 pitch + 3 volume + 2 VTLP + 3 speed) augmentation on the transcribed data and 3-

way speed perturbation on untranscribed data. The WER does not improve significantly. The second iteration with a 0.3 threshold, with the same augmentation strategies used in the first iteration, results in a relative improvement of 6.36% and 9.07% for the development and evaluation datasets, respectively, compared to the official baseline. When applying the third iteration with a threshold of 0.28 and the same augmentation strategies, the performance is only improved for the development data with the WER of 47.10%. Thus, another SSL iteration is not applied but implements different augmentation strategies at this stage. When we apply one iteration of SSL to the BLSTM baseline, the performance still improves from 57.54% to 56.86% in WER for the development set.

Replacing 2-fold VTLP with 2-fold noise perturbation for the transcribed data and keeping the untranscribed data augmentation to be 3-fold speed perturbation results in performance degradation. Adding only 2-fold noise perturbation to the untranscribed dataset does not improve the performance. However, 12-fold (2 pitch + 3 volume + 2 VTLP + 2 Noise + 3 speed) augmentation results in better performance for the evaluation dataset, though slightly worse for the development dataset.

3.6.5 Language Model Rescore

RNNLM rescoring (in parentheses in Table 3.2) is applied to the output of the third iteration of SSL, which results in a relative improvement of approximately 2% for both the development and evaluation datasets. The best performance is achieved with a 12-fold augmentation, resulting in a WER of 39.68 %, a 12.23% relative improvement over the official baseline system. If we use the development set in training, then the WER for the evaluation set improves to 38.85%.

3.7 Summary and Conclusion

In this chapter, a non-native children's ASR system is developed by utilizing untranscribed speech data with limited transcribed speech data. To compensate for long-duration non-speech segments within speech utterances, a novel non-speech discriminative loss is proposed in the acoustic model training phase to enable the classification of the speech/non-speech states. By utilizing the 60 hours of untranscribed speech data, Bi-APC pre-training, and incremental semi-supervised learning are combined and validated the effectiveness of those methods in addition to the various data augmentation methods, achieving a relative 9.6 % WER reduction over training with 5 hours of transcribed speech data only. The final system is able to achieve a 39.68% WER on the evaluation set.

CHAPTER 4

Towards Better Meta-Initialization with Task Augmentation for Kindergarten-aged speech Recognition

In the previous chapters, we have shown that data augmentation and using untranscribed data with unsupervised pre-training and semi-supervised learning can help improve the performance of the children's ASR models. In this chapter, our study [ZFA22] is presented for addressing the model initialization problem for training a children's ASR system, a model-agnostic meta-learning (MAML) [FAL17, NAS18] based approach meta-initialization (MI) is introduced as a possible solution.

4.1 Background

Meta-learning allows for fast adaption from different tasks to an unseen task, and is referred to as meta-initialization (MI) [AES19, Wan21]. The idea is to learn a good model initialization from different training tasks. It has been shown to be effective in cross-accent [al20a] and multi-lingual ASR [HCL20] as well as in other fields such as computer vision [SSZ17], neural machine translation [GWC18], and speaker adaptive training [KFB18]. However, MI is also vulnerable to learner overfitting [Yao21, NGS21], which happens when the model overfits the training tasks and is unable to generalize to the testing task.

To address the issue of learner overfitting, several task augmentation-based mechanisms were proposed. Liu et al. treated each rotation of an image as a new task for image classification tasks [LCL20], and Murty et al. proposed *DRECA* that uses latent reasoning categories to form new tasks for natural language processing tasks [MHM21]. To our knowledge, no study has addressed the issue of learner overfitting in ASR before.

In this chapter, we discover how meta-learning and task-based augmentation algorithms can apply to kindergarten children's ASR. In MI, the tasks are defined according to the development of children's vocal tract because it varies by the child's age.

Although a promising improvement is observed with the MI for kindergarten-aged speech, learner overfitting occurs. To alleviate learner overfitting, we propose a task augmentation mechanism for children's ASR by simulating new tasks using speed perturbation, and spectral shifting-based data augmentation methods, VTLP, because of the characteristics of each task (vocal tract differences).

The remainder of this chapter is organized as follows: Section 4.2 presents the meta-initialization and task augmentation approaches for the low-resource kindergarten-aged ASR. Section 4.3 describes the experimental setup, followed by results and discussion in Section 4.4. Section 4.5 concludes this thesis.

4.2 Methods

For a data-sufficient task, traditional machine learning can generalize well for in-domain data using random parameter initialization. However, when data are scarce, random initialization might overfit to the training data easily, and hence good starting points for training are essential for better model generalization. Previously, it has been shown that supervised pre-training can provide a good starting point for training in low-resource tasks [TWM17]. As mentioned earlier, the aim of a meta-learning application is to provide good initialization for low-resource tasks by quickly adapting the knowledge learned from the different available tasks to the unseen task, which is referred to as meta-initialization (MI). However, meta-initialization can be at risk of overfitting to the training tasks; this is referred to as learner overfitting [RIJ20]. In this section, we show how to use MI for ASR of children's speech and describe the proposed task-level augmentation method for solving the learner overfitting problem.

4.2.1 Supervised Pretraining

Supervised pretraining has shown its effectiveness in children's ASR [SG20]. Suppose we have the model outputs $Y = (y_1, y_2, \ldots, y_n)$ and the corresponding frame-level label $\hat{Y} = (\hat{y_1}, \hat{y_2}, \cdots, \hat{y_n}$ where frame $T = 1, 2, \cdots, n$ generated from forced alignment, the objective function of cross-entropy loss is expressed as:

$$L = -\sum_{t=1}^{T}\sum_{c=1}^{C}\hat{y_t^c}\log(y_t^c) \tag{4.1}$$

where C is the number of output categories (HMM states). As shown in Eq. (4.1), the trained weights are further utilized as the initialization for the downstream children's acoustic model training. However, the trained weights of the last feed-forward layer are not used due to the state mismatch from different models.

4.2.2 Meta Initialization (MI) with Meta Learning

Meta-learning is defined as a "learning to learn" method where the goal is to design a strategy to better choose a system's hyperparameters and learning algorithm. Learning model initialization or meta-initialization (MI) is also one of the most important components in meta-learning. Suppose we have a set of training tasks $\mathbb{G} = \{G_1, G_2, \ldots, G_i, \ldots, G_n\}$ and a target test task T. The idea is to simulate the adaptation stage during training and minimize an objective function. Note that the objective function is based on the adapted model so that the model before adaptation can be regarded as a good model initialization for the adaptation stage. For each training task G_i, the data are split into a support set G_i^{sup} that is used in the inner loop for the adaptation stage, and a query set G_i^{que} for evaluating the effectiveness of the model after the task's adaptation stage. A better initial model before adaptation leads to better performance. The loss function based on the query set is used in the outer loop to calculate the final objective function.

Suppose that the model parameters in the inner-loop are θ_j at step j, the audio samples in the support set of each training task G_i^{sup} are used to simulate the adaptation stage. The model is updated as follows:

$$\phi_{ji} = \theta_j - \alpha \nabla_{\theta_j} L(f(X_i^{sup}; \theta_j), Y_i^{sup}) \tag{4.2}$$

where X_i^{sup} and Y_i^{sup} are data samples and corresponding labels in the support set of task i, respectively. ϕ_{ji} is the model parameter updated for task i and step j. f is the forward computation of the model. L is the cross-entropy loss used in acoustic modeling, and α is the learning rate for the inner-loop optimizer. ∇ is the nabla operator for computing the gradient of θ_j.

In the outer-loop, we quantify how the adaptation behaves in the inner loop by a summation of the loss function for the query set of each task. The summation is referred to as the meta-objective function:

$$\sum_{G_i} L(f(X_i^{que}; \phi_{ji}), Y_i^{que}) \tag{4.3}$$

where X_i^{que} and Y_i^{que} are data samples and corresponding labels in the query set of task i, respectively. By minimizing the above objective function with respect to θ_j, we can find a model that is suitable for adaptation, and hence the model can be regarded as a good initialization. Based on Eq. (4.2), after the optimization of the inner loop is completed, the initialization would be the focus of the algorithm.

$$\theta_{j+1} \leftarrow \theta_j - \beta \nabla'_{\theta_j} \sum_{G_i} L(f(X_i^{que}; \phi_{ji}), Y_i^{que})) \tag{4.4}$$

where β is the learning rate for the outer-loop optimizer, and ∇'_{θ_j} indicates that only first-order MAML [NAS18] is used since the second-order derivative is computationally expensive and it does not affect the results significantly. After enough training steps, N, the final model θ_N is regarded as the learned initialization for the unseen test task.

4.2.3 Data Augmentation

As explained in the previous chapters, data augmentation is an effective technique to improve the ASR model performance by increasing the data variability, particularly for low-resource tasks, such as children's ASR. It can also mitigate the overfitting problem commonly existing in traditional machine learning algorithms.

In this experiment, data augmentation is also applied at the adaptation phase, where the training data is augmented with additional folds. For example, if all the training data for the adaptation are augmented with two warping factors 0.9 and 1.1, the final training data will be 3-fold. Another common augmentation technique is online-based. An online-based augmentation modifies the training data every epoch without increasing the number of training utterances, which is usually applied for SpecAug [Par19]. The comparison of the data augmentation methods is shown in Section 2.2.

4.2.4 Age-based Task Augmentation for MI

Different from overfitting in traditional machine learning algorithms, there are two other overfitting problems in MI, which are memorization overfitting [al20b] and learner overfitting. The memorization overfitting happens when the θ_{j+1} memorizes all tasks and does not rely on support sets for inner-loop adaptation. The learner overfitting happens when the θ_{j+1} is unable to generalize well on the test task T. The memorization can be well mitigated by randomly sampling the support set and query set at each step during training since each sample has the opportunity to participate in either inner loop updates or outer loop updates. In terms of learner overfitting, a common strategy is to use task augmentation to increase the model generalization for the test task. However, task augmentation has not been explored in ASR, to our knowledge, before.

We propose an age-based task augmentation framework to alleviate the problem of learner overfitting in kindergarten-aged speech recognition. The higher degree of inter-speaker variability of children's speech is mainly due to the different growth patterns of children. These differences result in shifts in the fundamental frequency (F_0) and formant frequencies (F_1, F_2, F_3, etc.) in kids' speech as they grow. Hence, we perform the augmentation by simulating new tasks of children's speech using time and frequency warping methods, such as VTLP and speed perturbation. For example, the task for each age $G_i(G_i^{1.0})$ is augmented with two new tasks with two warping factors 0.9 ($G_i^{0.9}$) and 1.1 ($G_i^{1.1}$). We compare the two methods in Section 4.3.

4.3 Experiments

Experiments are conducted using the Kaldi toolkit [Pov11] for feature extraction and WFST-based decoding and Pykaldi2 [LXC19] for acoustic model training.

4.3.1 Database

The database for the experiments is the scripted part of OGI Kids' Speech Corpus [SHC00]. The Corpus contains kids' speech in eleven age groups from kindergarten, grade 1 (G1) to grade 10 (G10). Each age group has approximately 100 speakers saying single words, sentences, and digit strings. The dataset is randomly split into 70 % training data, 8 % development data, and 22 % test data without speaker overlap for each age as in [FAA21]. The kindergarten-aged task is regarded as the meta-testing task for fine-tuning. G1 speech data, which corresponds to the closest age to kindergarten speech, are used for the validation task in meta-learning. Other tasks with kids' speech from G2 to G10 are used as the training tasks, which is similar to pre-training, for obtaining a model initialization. For meta-training and meta-validation tasks, training and development sets are combined for sampling the support and query sets. Note that the training data for kindergarten-aged speech is approximately 4 hours and the training data for the meta-initialization stage is about 45 hours.

4.3.2 Acoustic Model Setup

First, an HMM-GMM model is trained with all the data in the meta-training tasks to obtain frame-level alignment for the DNN-based acoustic model training. 80-dimensional log-Mel-filter bank features are extracted every 10 ms with a 25 ms window. An additional frame of features after each frame is appended to form a 160-dimensional input [al15]. The model has 4 BLSTM layers with 512 hidden units in each direction. The last layer transforms the outputs of BLSTM to a probability distribution of the 1360 states from the HMM model. For the baseline and the adaptation of kindergarten-aged task, the training process takes 15 iterations. An Adam optimizer with a multi-step scheduler is applied, where the learning rate is initially set to $1e^{-5}$ for the first two iterations and decayed with a ratio of 0.1 till the last iteration.

4.3.3 Meta Initialization (MI) Setup

Table 4.1: % Word error rate (WER) for Data Augmentation (Data Aug) mechanisms on baseline system, meta-initialization (MI), and the proposed task augmentation (Task Aug) mechanisms for MI with vocal tract length perturbation (VTLP) and speed perturbation (SP) on the Kindergarten-aged development and test sets. SPT stands for supervised pre-training. Raw Aug stands for augmentation within each task without creating new tasks.

Model	Data Aug Type	MI Aug Type	Dev	Test
Baseline	-	-	53.17	55.01
+ Data Aug	SP	-	46.13	43.75
	VTLP	-	45.42	46.05
	SpecAug	-	56.69	53.70
+ SPT [TWM17]	-	-	36.27	29.06
+ MI	-	-	35.21	30.68
+ Raw Aug	-	SP	36.62	28.00
	-	VTLP	36.27	30.06
+ Task Aug	-	SP	**34.86**	**27.50**
	-	VTLP	**34.86**	29.06

In MI, the support set and query set are randomly sampled with a batch size of 16 for each age of G2 to G10 during training. The same frame-level alignment and BLSTM model configuration are used as mentioned in Section 4.3.2.

The number of iterations for MI training is empirically set to 6,800. Separate optimizers are applied to the outer-loop and inner-loop optimization. The inner loop uses an SGD optimizer with a fixed learning rate of $2e^{-4}$. The outer loop uses an Adam optimizer with a multi-step scheduler, where the learning rate is stabilized to $2e^{-4}$ for the first 2,000 iterations and decayed with a ratio of 0.15 to $3e^{-5}$ till the last iteration. All the parameters trained from MI are used as the initialization for the training in the adaptation stage.

4.3.4 Task Augmentation Setup

For age-based task augmentation during the MI stage, speed perturbation and vocal tract length perturbation (VTLP) are used with the warping factors of 0.9, 1.0, and 1.1, according to our preliminary results [al21b,al21a], and hence the number of tasks is increased by 3 folds. Thus, we adopt an online augmentation mechanism where at each iteration the warping factor is randomly selected from (0.9, 1.0, 1.1).

4.3.5 Raw Augmentation Setup

For a fair comparison with the task augmentation setup in Section 4.3.4, speed perturbation and vocal tract length perturbation (VTLP) are used with warping factors of 0.9, 1.0, and 1.1 as well; however, it is applied directly to the original meta-training data from all the grades.

4.3.6 Data Augmentation for Adaptation Setup

During the adaptation stage, speed perturbation and VTLP are used with the same warping factors (0.9, 1.0, 1.1) as task augmentation. For SpecAug, a maximum width of 5 frequency channels are masked twice, and a maximum width of 8 time channels are masked twice as well. The width of the frequency and time channel is chosen empirically.

4.4 Results and Discussions

An HMM-DNN hybrid system with BLSTM modeling is used as our baseline. As shown in Table 4.2, the development and test set of kindergarten speech have a WER of 53.17% and

55.01%, respectively, without any prior knowledge. The baseline WER is similar to that reported in [al21b] for a small size (5 hours) kids dataset.

Table 4.2: % Word error rate (WER) for Data Augmentation (Data Aug) mechanisms on baseline system, meta-initialization (MI), and the proposed task augmentation (Task Aug) mechanisms for MI with vocal tract length perturbation (VTLP) and speed perturbation (SP) on the Kindergarten-aged development and test sets. SPT stands for supervised pre-training. Raw Aug stands for augmentation within each task without creating new tasks.

Model	Data Aug Type	MI Aug Type	Dev	Test
Baseline	-	-	53.17	55.01
+ Data Aug	SP	-	46.13	43.75
	VTLP	-	45.42	46.05
	SpecAug	-	56.69	53.70
+ SPT [TWM17]	-	-	36.27	29.06
+ MI	-	-	35.21	30.68
+ Raw Aug	-	SP	36.62	28.00
	-	VTLP	36.27	30.06
+ Task Aug	-	SP	**34.86**	**27.50**
	-	VTLP	**34.86**	29.06

4.4.1 Meta Initialization

The results of MI and the proposed task augmentation methods are shown in Table 4.2. As we can observe from the table, using data augmentation (Data Aug) methods can improve performance over baseline. The relative improvement in WER for speed perturbation (SP) and VTLP are around 20%. When training with an initialization through meta-learning, the WER of the kindergarten-aged test set is decreased from 55.01% to 30.68%, a larger relative WER improvement than the data augmentation strategies. For a fair comparison, we used the supervised pre-training method (SPT) to directly train the acoustic model with data from G2-G10 as the starting point. We can see from the table that MI is slightly worse than SPT on the test set.

4.4.2 Raw Augmentation v.s. Task Augmentation

The proposed task augmentation methods are used to address the overfitting problem and we observe a significant improvement over the MI without augmentation. From Table 4.2, we found that SP is better than VTLP as a method to simulate new tasks. For a fair comparison, we also experimented with augmentation that is not task-dependent. In raw augmentation (Raw Aug), warping is applied to the original data. The results validate the effectiveness of the proposed task augmentation (Task Aug) method, which achieves a WER of 27.5% on the kindergarten test set. SpecAug is not used in task augmentation since it randomly masks out time or frequency channels. Such masking is not consistent for the data in one task that is regarded as a new task after augmentation.

4.4.3 Impact of Augmented Tasks

The task augmentation in Table 4.2 is using speech data from all ages in the training set to augment a new ASR task. To obtain an insight into the impact of the augmented tasks on WER performance, we add the number of augmented tasks incrementally according to age. For example, as shown in Fig.4.1, the number of tasks is added in either an increasing order (from G2 to G10) or a decreasing order (from G10 to G2). Our goal is to investigate which subset of the data is more important for augmentation.

Since SP outperforms VTLP in the previous experiments, SP is explored. As shown in Fig.4.1, including more augmented tasks in either the forward order or reverse order results

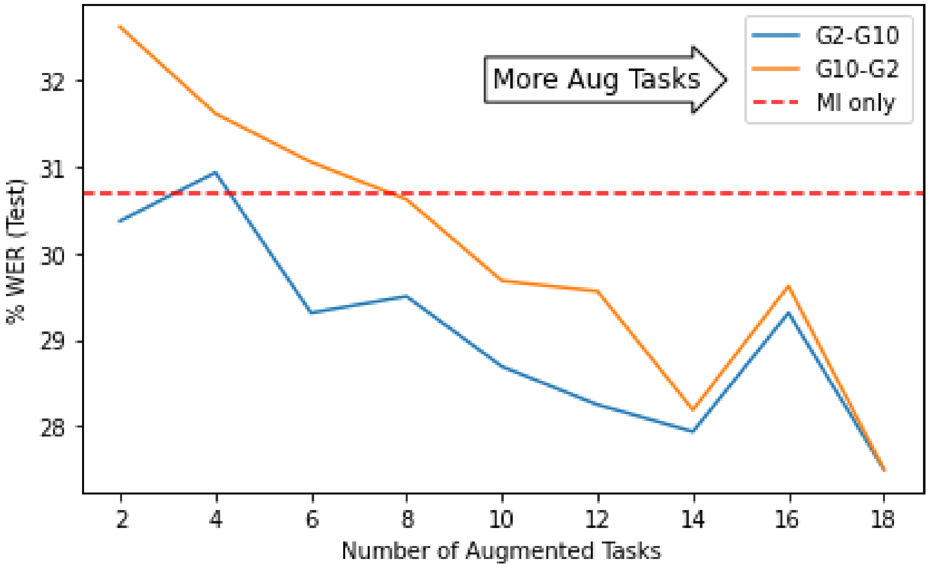

Figure 4.1: % WER Results of task augmentation mechanism using speed perturbation (SP) versus the number of augmentation tasks for MI on the Kindergarten test set. The tasks are added either from G2 to G10 (in blue), or from G10 to G2 (in orange). The dashed line (in red) is MI without any task augmentation mechanism.

in improved performance. However, the reverse order generally performs worse than the forward order by 1% WER for the kindergarten-aged test set, which means creating new tasks that are similar to the target task is effective in addressing the learner overfitting problem. With all tasks being augmented, the final performance has a 10% relative WER improvement over MI without task augmentation.

4.4.4 Data Augmentation for Adaptation

Table 4.3: % Word error rate (WER) for data augmentation during the adaptation stage with SpecAug, vocal tract length perturbation (VTLP), and speed perturbation (SP) on the Kindergarten development and test sets.

Aug Type (in adaptation stage)	Dev	Test
No Aug	34.86	27.50
SpecAug	32.75	**27.01**
VTLP	**32.39**	28.13
SP	33.45	27.75

The task we are focusing on is a low-resource one (kindergarten ASR). Hence, data augmentation methods are further used during the adaptation stage of the kindergarten-aged task. SP, SpecAug, and VTLP are compared in the experiments. The results are shown in Table 4.3. Although all three strategies can improve the performance on the development set, only SpecAug achieves a slightly better performance on the test set. The reasons why VTLP and SP did not achieve better results will be explored in future work.

4.5 Summary and Conclusion

In this chapter, to deal with the data scarcity of children's speech, particularly kindergarten-aged, meta-initialization is used to find a good starting point for training the acoustic model. To mitigate the overfitting problem in meta-initialization, particularly learner overfitting, an age-based task augmentation mechanism is proposed to simulate new ages using time and frequency warping methods. The data augmentation strategies using speed perturbation and VTLP that are also used in the task augmentation stage are not helpful in the adaptation stage. SpecAug used in the adaptation stage resulted in a small WER improvement, and the final system achieved a 51% relative WER improvement over the baseline (no augmentation and no adaptation). In the future, we will explore the use of the proposed algorithm in other low-resource tasks for both adults' and children's ASR.

CHAPTER 5

Conclusions and Future Work

Children's automatic speech recognition (ASR) is a comparatively harder problem than adults' ASR, due to a lack of transcribed available databases. The approaches proposed to improve the children's ASR:

1. Data augmentation is a low-cost and effective method for improving children's ASR given limited data. Several augmentation techniques were introduced in Chapter 2: speed perturbation, VTLP, SpecAug, pitch perturbation, and F0-based normalization. In Chapter 3, volume perturbation is introduced. These techniques improved ASR within 10 - 25 % WER. On average, speed perturbation achieved the best WER reduction.

2. Both unsupervised pre-training and incremental semi-supervised learning are used to utilize untranscribed children's speech data. Bidirectional autoregressive predictive coding (Bi-APC) is an effective unsupervised pretraining method to learn acoustic modeling parameters from untranscribed data, and incremental semi-supervised learning well-utilizes the untranscribed speech data by filtering out low log-likelihood utterances during decoding for multiple iterations. Both Bi-APC and incremental SSL are introduced in Chapter 3.

3. A meta-initialization technique is used to find a good model initialization for kindergarten-aged ASR by treating each age group of children as one task. The upstream framework is based on the model-agnostic meta-learning mechanism, and age-based task augmentation is proposed to alleviate the learner overfitting in MI, the final performance is able to achieve much better performance compared to baseline ASR modeling with no augmentation or

initialization. Details are provided in Chapter 4.

However, some challenges remain. In Chapter 2, first, it is found that speaking style mismatch is not fully alleviated by combining scripted and spontaneous data. Further explorations are needed: for example, the variable frame rate (VFR) technique could be used for compensating speaking style effects [AGP20]. Another challenge is that for spontaneous speech, CTC-AED is not able to train well due to multiple possible constraints, such as data size and long-form utterances. To address these challenges for CTC-AED-based end-to-end ASR systems, further investigations are needed. In Chapter 4, using the meta-learning technique with respect to each age group has shown improvement. Further experiments are needed to investigate the effect of reducing the number of samples for each inner-loop task as mentioned in [FAL17].

REFERENCES

[Abd94] Herve Abdi. “A neural network primer.” *Journal of Biological Systems*, **2**(03):247–281, 1994.

[AES19] Antreas Antoniou, Harrison Edwards, and Amos J. Storkey. “How to train your MAML.” In *7th ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019.

[AGP20] Amber Afshan, Jinxi Guo, Soo Jin Park, Vijay Ravi, Alan McCree, and Abeer Alwan. “Variable frame rate-based data augmentation to handle speaking-style variability for automatic speaker verification.” *arXiv preprint arXiv:2008.03616*, 2020.

[al15] H. Sak et al. “Fast and Accurate Recurrent Neural Network Acoustic Models for Speech Recognition.” *Proc. Interspeech 2015*, 2015.

[al20a] Genta Indra Winata et al. “Learning Fast Adaptation on Cross-Accented Speech Recognition.” In *Proc. Interspeech 2020*, 2020.

[al20b] Mingzhang Yin et al. “Meta-Learning without Memorization.” In *ICLR*, 2020.

[al21a] J. Wang et al. “Low Resource German ASR with Untranscribed Data Spoken by Non-Native Children — INTERSPEECH 2021 Shared Task SPAPL System.” In *Proc. Interspeech 2021*, 2021.

[al21b] R. Gretter et al. “ETLT 2021: Shared Task on Automatic Speech Recognition for Non-Native Children’s Speech.” In *Proc. Interspeech 2021*, 2021.

[BCC17] Eric Battenberg, Jitong Chen, Rewon Child, Adam Coates, Yashesh Gaur Yi Li, Hairong Liu, Sanjeev Satheesh, Anuroop Sriram, and Zhenyao Zhu. “Exploring neural transducers for end-to-end speech recognition.” In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 206–213. IEEE, 2017.

[BM12] Herve A Bourlard and Nelson Morgan. *Connectionist speech recognition: a hybrid approach*, volume 247. Springer Science & Business Media, 2012.

[CA06] Xiaodong Cui and Abeer Alwan. “Adaptation of children’s speech with limited data based on formant-like peak alignment.” *Computer speech & language*, **20**(4):400–419, 2006.

[CGK15] Xiaodong Cui, Vaibhava Goel, and Brian Kingsbury. “Data augmentation for deep neural network acoustic modeling.” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **23**(9), 2015.

[CHC12] Xiaodong Cui, Jing Huang, and Jen-Tzung Chien. "Multi-view and multi-objective semi-supervised learning for hmm-based automatic speech recognition." *IEEE Transactions on Audio, Speech, and Language Processing*, **20**(7):1923–1935, 2012.

[CHT19] Yu-An Chung, Wei-Ning Hsu, Hao Tang, and James Glass. "An Unsupervised Autoregressive Model for Speech Representation Learning." *Proc. Interspeech 2019*, pp. 146–150, 2019.

[CJL16] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition." In *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 4960–4964. IEEE, 2016.

[CL13] I-Fan Chen and Chin-Hui Lee. "A hybrid HMM/DNN approach to keyword spotting of short words." In *INTERSPEECH*, pp. 1574–1578, 2013.

[CNB17] Ciprian Chelba, Mohammad Norouzi, and Samy Bengio. "N-gram language modeling using recurrent neural network estimation." *arXiv preprint arXiv:1703.10724*, 2017.

[CNW20] Guoguo Chen, Xingyu Na, Yongqing Wang, Zhiyong Yan, Junbo Zhang, Sifan Ma, and Yujun Wang. "Data Augmentation For Children's Speech Recognition–The "ETHIOPIAN" System For The SLT 2021 Children Speech Recognition Challenge." *arXiv preprint arXiv:2011.04547*, 2020.

[DAR21] Anirban Dutta, Gudmalwar Ashishkumar, and Ch V Rama Rao. "Performance analysis of ASR system in hybrid DNN-HMM framework using a PWL euclidean activation function." *Frontiers of Computer Science*, **15**(4):1–11, 2021.

[DLH13] Li Deng, Jinyu Li, Jui-Ting Huang, Kaisheng Yao, Dong Yu, Frank Seide, Michael Seltzer, Geoff Zweig, Xiaodong He, Jason Williams, et al. "Recent advances in deep learning for speech research at Microsoft." In *2013 IEEE international conference on acoustics, speech and signal processing*, pp. 8604–8608. IEEE, 2013.

[DMB19] Subhadeep Dey, Petr Motlicek, Trung Bui, and Franck Dernoncourt. "Exploiting Semi-Supervised Training Through a Dropout Regularization in End-to-End Speech Recognition." *Proc. Interspeech 2019*, pp. 734–738, 2019.

[DPG09] Cong-Thanh Do, Dominique Pastor, and André Goalic. "On the recognition of cochlear implant-like spectrally reduced speech with MFCC and HMM-based ASR." *IEEE Transactions on Audio, Speech, and Language Processing*, **18**(5):1065–1068, 2009.

[DT17] Terrance DeVries and Graham W Taylor. "Improved regularization of convolutional neural networks with cutout." *arXiv preprint arXiv:1708.04552*, 2017.

[FAA21] Ruchao Fan, Amber Afshan, and Abeer Alwan. "Bi-apc: Bidirectional autoregressive predictive coding for unsupervised pre-training and its application to children's asr." In *ICASSP*. IEEE, 2021.

[FAL17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. "Model-agnostic meta-learning for fast adaptation of deep networks." In *ICML*. PMLR, 2017.

[FBL16] Joachim Fainberg, Peter Bell, Mike Lincoln, and Steve Renals. "Improving Children's Speech Recognition Through Out-of-Domain Data Augmentation." In *Interspeech*, pp. 1598–1602, 2016.

[For73] G David Forney. "The viterbi algorithm." *Proceedings of the IEEE*, **61**(3):268–278, 1973.

[GFG06] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks." In *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, 2006.

[GG03] Diego Giuliani and Matteo Gerosa. "Investigating recognition of children's speech." In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, volume 2, pp. II–137. IEEE, 2003.

[GGB07] Matteo Gerosa, Diego Giuliani, and Fabio Brugnara. "Acoustic variability and automatic recognition of children's speech." *Speech Communication*, **49**(10-11):847–860, 2007.

[GJ14] Alex Graves and Navdeep Jaitly. "Towards end-to-end speech recognition with recurrent neural networks." In *International conference on machine learning*, pp. 1764–1772. PMLR, 2014.

[GL94] J-L Gauvain and Chin-Hui Lee. "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains." *IEEE transactions on speech and audio processing*, **2**(2):291–298, 1994.

[GMF21] Roberto Gretter, Marco Matassoni, Daniele Falavigna, A Misra, Chee Wee Leong, Katherine Knill, and Linlin Wang. "ETLT 2021: Shared Task on Automatic Speech Recognition for Non-Native Children's Speech." 2021.

[GPY15] Jinxi Guo, Rohit Paturi, Gary Yeung, Steven M Lulich, Harish Arsikere, and Abeer Alwan. "Age-dependent height estimation and speaker normalization for children's speech using the first three subglottal resonances." In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[GQC20] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. "Conformer: Convolution-augmented transformer for speech recognition." *arXiv preprint arXiv:2005.08100*, 2020.

[GWC18] Jiatao Gu, Yong Wang, Yun Chen, Victor O. K. Li, and Kyunghyun Cho. "Meta-Learning for Low-Resource Neural Machine Translation." In *Proceedings of the 2018 EMNLP*, Brussels, Belgium, October-November 2018. ACL.

[GWL14] Sharmistha S Gray, Daniel Willett, Jianhua Lu, Joel Pinto, Paul Maergner, and Nathan Bodenstab. "Child automatic speech recognition for US English: child interaction with living-room-electronic-devices." In *WOCCI*, pp. 21–26, 2014.

[GY19] Ramazan Gokay and Hulya Yalcin. "Improving Low Resource Turkish Speech Recognition with Data Augmentation and TTS." In *2019 16th International Multi-Conference on Systems, Signals Devices (SSD)*, pp. 357–360, 2019.

[HCL20] Jui-Yang Hsu, Yuan-Jui Chen, and Hung-yi Lee. "Meta learning for end-to-end low-resource speech recognition." In *ICASSP*. IEEE, 2020.

[HDY12] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups." *IEEE Signal processing magazine*, **29**(6):82–97, 2012.

[HH02] Yu Hen Hu and Jeng-Neng Hwang. "Handbook of neural network signal processing.", 2002.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory." *Neural computation*, **9**(8):1735–1780, 1997.

[IPM14] David Imseng, Blaise Potard, Petr Motlicek, Alexandre Nanchen, and Hervé Bourlard. "Exploiting un-transcribed foreign data for speech recognition in well-resourced languages." In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2322–2326. IEEE, 2014.

[JH13] Navdeep Jaitly and Geoffrey E Hinton. "Vocal tract length perturbation (VTLP) improves speech recognition." In *Proc. ICML Workshop on Deep Learning for Audio, Speech and Language*, volume 117, 2013.

[KFB18] Ondřej Klejch, Joachim Fainberg, and Peter Bell. "Learning to adapt: a meta-learning approach for speaker adaptation." *arXiv preprint arXiv:1808.10239*, 2018.

[KMD20] Banriskhem Khonglah, Srikanth Madikeri, Subhadeep Dey, Hervé Bourlard, Petr Motlicek, and Jayadev Billa. "Incremental semi-supervised learning for multi-genre speech recognition." In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7419–7423. IEEE, 2020.

[KPP15] Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. "Audio augmentation for speech recognition." In *Interspeech 2015*, 2015.

[LCL20] Jialin Liu, Fei Chao, and Chih-Min Lin. "Task augmentation by rotating for meta-learning." *arXiv preprint arXiv:2003.00804*, 2020.

[LGA02] Lori Lamel, Jean-Luc Gauvain, and Gilles Adda. "Lightly supervised and unsupervised acoustic model training." *Computer Speech & Language*, **16**(1):115–129, 2002.

[LLH16] Tan Lee, Yuanyuan Liu, Pei-Wen Huang, Jen-Tzung Chien, Wang Kong Lam, Yu Ting Yeung, Thomas KT Law, Kathy YS Lee, Anthony Pak-Hin Kong, and Sam-Po Law. "Automatic speech recognition for acoustical analysis and assessment of cantonese pathological voice and speech." In *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 6475–6479. IEEE, 2016.

[LR98] Li Lee and Richard Rose. "A frequency warping approach to speaker normalization." *IEEE Transactions on speech and audio processing*, **6**(1):49–60, 1998.

[LW95] Christopher J Leggetter and Philip C Woodland. "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models." *Computer speech & language*, **9**(2):171–185, 1995.

[LXC19] Liang Lu, Xiong Xiao, Zhuo Chen, and Yifan Gong. "Pykaldi2: Yet another speech toolkit based on kaldi and pytorch." *arXiv preprint arXiv:1907.05955*, 2019.

[MH12] Ryoko Mugitani and Sadao Hiroya. "Development of vocal tract and acoustic features in children." *Acoustical Science and Technology*, **33**(4):215–220, 2012.

[MHM21] Shikhar Murty, Tatsunori Hashimoto, and Christopher D Manning. "Dreca: A general task augmentation strategy for few-shot natural language inference." In *Proc. 2021 NAACL: Human Language Technologies*, 2021.

[Mik12] Tomáš Mikolov et al. "Statistical language models based on neural networks." *Presentation at Google, Mountain View, 2nd April*, **80**:26, 2012.

[MKD11] Tomas Mikolov, Stefan Kombrink, Anoop Deoras, Lukar Burget, and Jan Cernocky. "RNNLM-recurrent neural network language modeling toolkit." In *Proc. of the 2011 ASRU Workshop*, pp. 196–201, 2011.

[Moh14] Abdel-rahman Mohamed. *Deep Neural Network Acoustic Models for ASR.* PhD thesis, University of Toronto, 2014.

[MPK17] Vimal Manohar, Daniel Povey, and Sanjeev Khudanpur. "JHU Kaldi system for Arabic MGB-3 ASR challenge using diarization, audio-transcript alignment and transfer learning." In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 346–352. IEEE, 2017.

[MPR02] Mehryar Mohri, Fernando Pereira, and Michael Riley. "Weighted finite-state transducers in speech recognition." *Computer Speech & Language*, **16**(1):69–88, 2002.

[Nak19] Tomohiro Nakatani. "Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration." In *Proc. Interspeech*, 2019.

[NAS18] Alex Nichol, Joshua Achiam, and John Schulman. "On first-order meta-learning algorithms." *arXiv preprint arXiv:1803.02999*, 2018.

[NGS21] Renkun Ni, Micah Goldblum, Amr Sharaf, Kezhi Kong, and Tom Goldstein. "Data augmentation for meta-learning." In *ICML*. PMLR, 2021.

[NLP20] Si-Ioi Ng, Wei Liu, Zhiyuan Peng, Siyuan Feng, Hing-Pang Huang, Odette Scharenborg, and Tan Lee. "The cuhk-tudelft system for the slt 2021 children speech recognition challenge." *arXiv preprint arXiv:2011.06239*, 2020.

[NPC19] Arun Narayanan, Rohit Prabhavalkar, Chung-Cheng Chiu, David Rybach, Tara N Sainath, and Trevor Strohman. "Recognizing long-form speech using streaming end-to-end models." In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 920–927. IEEE, 2019.

[ODK22] Mamyrbayev Orken, Oralbekova Dina, Alimhan Keylan, Turdalykyzy Tolganay, and Othman Mohamed. "A study of transformer-based end-to-end speech recognition system for Kazakh language." *Scientific Reports*, **12**(1):1–11, 2022.

[OLV18] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. "Representation learning with contrastive predictive coding." *arXiv preprint arXiv:1807.03748*, 2018.

[Par19] Daniel S Park et al. "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition." *Proc. Interspeech 2019*, 2019.

[PN03] Alexandros Potamianos and Shrikanth Narayanan. “Robust recognition of children’s speech.” *IEEE Transactions on speech and audio processing*, **11**(6):603–616, 2003.

[Pov11] Daniel Povey et al. “The Kaldi speech recognition toolkit.” In *IEEE 2011 workshop on automatic speech recognition and understanding*, number CONF. IEEE Signal Processing Society, 2011.

[RFA20] Vijay Ravi, Ruchao Fan, Amber Afshan, Huanhua Lu, and Abeer Alwan. “Exploring the Use of an Unsupervised Autoregressive Model as a Shared Encoder for Text-Dependent Speaker Verification.” *Proc. Interspeech 2020*, pp. 766–770, 2020.

[RIJ20] Janarthanan Rajendran, Alexander Irpan, and Eric Jang. “Meta-Learning Requires Meta-Augmentation.” In *Adv. Neural Inf. Process. Syst.*, volume 33. Curran Associates, Inc., 2020.

[RM07] Richard Rose and Parya Momayyez. “Integration of multiple feature sets for reducing ambiguity in ASR.” In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP’07*, volume 4, pp. IV–325. IEEE, 2007.

[SG20] Prashanth Gurunath Shivakumar and Panayiotis Georgiou. “Transfer learning from adult to children for speech recognition: Evaluation, analysis and recommendations.” *Computer speech & language*, **63**:101077, 2020.

[SHC00] Khaldoun Shobaki, John-Paul Hosom, and Ronald Cole. “The OGI kids’ speech corpus and recognizers.” In *Proc. of ICSLP*, 2000.

[SHS03] Georg Stemmer, Christian Hacker, Stefan Steidl, and Elmar Nöth. “Acoustic normalization of children’s speech.” In *IN PROC. EUROPEAN CONF. ON SPEECH COMMUNICATION AND TECHNOLOGY*, 2003.

[SLY11] Frank Seide, Gang Li, and Dong Yu. “Conversational speech transcription using context-dependent deep neural networks.” In *Twelfth annual conference of the international speech communication association*, 2011.

[SoX] “Sound exchange: Homepage.”.

[SP03] Ben J Shannon and Kuldip K Paliwal. “A comparative study of filter bank spacing for speech recognition.” In *Microelectronic engineering research conference*, volume 41, pp. 310–12. Citeseer, 2003.

[SPL14] Prashanth Gurunath Shivakumar, Alexandros Potamianos, Sungbok Lee, and Shrikanth S Narayanan. “Improving speech recognition for children using acoustic adaptation and pronunciation modeling.” In *WOCCI*, pp. 15–19, 2014.

[SS14] Seyed Reza Shahamiri and Siti Salwah Binti Salim. “Artificial neural networks as speech recognisers for dysarthric speech: Identifying the best-performing set of MFCC parameters and studying a speaker-independent approach.” *Advanced Engineering Informatics*, **28**(1):102–110, 2014.

[SSZ17] Jake Snell, Kevin Swersky, and Richard Zemel. “Prototypical Networks for Few-shot Learning.” In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Adv. Neural Inf. Process. Syst.*, volume 30. Curran Associates, Inc., 2017.

[STB10] Yang Sun, Louis Ten Bosch, and Lou Boves. “Hybrid HMM/BLSTM-RNN for robust speech recognition.” In *International Conference on Text, Speech and Dialogue*, pp. 400–407. Springer, 2010.

[SVK21] Peter Smit, Sami Virpioja, and Mikko Kurimo. “Advances in subword-based HMM-DNN speech recognition across languages.” *Computer Speech & Language*, **66**:101158, 2021.

[SVS15] Tara N Sainath, Oriol Vinyals, Andrew Senior, and Haşim Sak. “Convolutional, long short-term memory, fully connected deep neural networks.” In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 4580–4584. IEEE, 2015.

[SXK19] Gabriel Synnaeve, Qiantong Xu, Jacob Kahn, Tatiana Likhomanenko, Edouard Grave, Vineel Pratap, Anuroop Sriram, Vitaliy Liptchinsky, and Ronan Collobert. “End-to-end ASR: from supervised to semi-supervised learning with modern architectures.” *arXiv preprint arXiv:1911.08460*, 2019.

[TA13] Lee Ngee Tan and Abeer Alwan. “Multi-band summary correlogram-based pitch detection for noisy speech.” *Speech communication*, **55**(7-8):841–856, 2013.

[TGN14] Zoltán Tüske, Pavel Golik, David Nolden, Ralf Schlüter, and Hermann Ney. “Data augmentation, feature combination, and multilingual neural networks to improve ASR and KWS performance for low-resource languages.” In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[TN03] Christoph Tillmann and Hermann Ney. “Word reordering and a dynamic programming beam search algorithm for statistical machine translation.” *Computational linguistics*, **29**(1):97–133, 2003.

[TSC13] Samuel Thomas, Michael L Seltzer, Kenneth Church, and Hynek Hermansky. “Deep neural network features and semi-supervised training for low resource speech recognition.” In *2013 IEEE international conference on acoustics, speech and signal processing*, pp. 6704–6708. IEEE, 2013.

[TWM17] Rong Tong, Lei Wang, and Bin Ma. "Transfer learning for children's speech recognition." In *2017 IALP*. IEEE, 2017.

[VSP17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." *Advances in neural information processing systems*, **30**, 2017.

[Wan21] Disong Wang et al. "Improved End-to-End Dysarthric Speech Recognition via Meta-learning Based Model Re-initialization." In *2021 12th ISCSLP*. IEEE, 2021.

[WHK17a] Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R. Hershey, and Tomoki Hayashi. "Hybrid CTC/Attention Architecture for End-to-End Speech Recognition." *IEEE Journal of Selected Topics in Signal Processing*, **11**(8):1240–1253, 2017.

[WHK17b] Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R Hershey, and Tomoki Hayashi. "Hybrid CTC/attention architecture for end-to-end speech recognition." *IEEE Journal of Selected Topics in Signal Processing*, **11**(8):1240–1253, 2017.

[WMG20] Felix Weninger, Franco Mana, Roberto Gemello, Jesús Andrés-Ferrer, and Puming Zhan. "Semi-supervised learning with data augmentation for end-to-end ASR." *arXiv preprint arXiv:2007.13876*, 2020.

[WML20] Yongqiang Wang, Abdelrahman Mohamed, Due Le, Chunxi Liu, Alex Xiao, Jay Mahadeokar, Hongzhao Huang, Andros Tjandra, Xiaohui Zhang, Frank Zhang, et al. "Transformer-based acoustic modeling for hybrid speech recognition." In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6874–6878. IEEE, 2020.

[Woo01] Phil C Woodland. "Speaker adaptation for continuous density HMMs: A review." In *ISCA Tutorial and Research Workshop (ITRW) on Adaptation Methods for Speech Recognition*, 2001.

[WZF21] Jinhan Wang, Yunzheng Zhu, Ruchao Fan, Wei Chu, and Abeer Alwan. "Low Resource German ASR with Untranscribed Data Spoken by Non-native Children–INTERSPEECH 2021 Shared Task SPAPL System." *arXiv preprint arXiv:2106.09963*, 2021.

[YA18] Gary Yeung and Abeer Alwan. "On the difficulties of automatic speech recognition for kindergarten-aged children." *Interspeech 2018*, 2018.

[Yao21] Huaxiu Yao et al. "Improving generalization in meta-learning via task augmentation." In *ICML*. PMLR, 2021.

[YFA21a] Gary Yeung, Ruchao Fan, and Abeer Alwan. "Fundamental frequency feature normalization and data augmentation for child speech recognition." In *ICASSP*. IEEE, 2021.

[YFA21b] Gary Yeung, Ruchao Fan, and Abeer Alwan. "Fundamental Frequency Feature Normalization and Data Augmentation for Child Speech Recognition." In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6993–6997, 2021.

[You96] Steve Young. "A review of large-vocabulary continuous-speech." *IEEE signal processing magazine*, **13**(5):45, 1996.

[ZDV17] Albert Zeyer, Patrick Doetsch, Paul Voigtlaender, Ralf Schlüter, and Hermann Ney. "A comprehensive study of deep bidirectional LSTM RNNS for acoustic modeling in speech recognition." In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2462–2466, 2017.

[ZFA22] Yunzheng Zhu, Ruchao Fan, and Abeer Alwan. "Towards Better Meta-Initialization with Task Augmentation for Kindergarten-aged Speech Recognition." In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8582–8586. IEEE, 2022.

[ZZS01] Fang Zheng, Guoliang Zhang, and Zhanjiang Song. "Comparison of different implementations of MFCC." *Journal of Computer science and Technology*, **16**(6):582–589, 2001.