



# CyclicAugment: Speech Data Random Augmentation with Cosine Annealing Scheduler for Automatic Speech Recognition

Zhihan Wang, Feng Hou\*, Yuanhang Qiu, Zhizhong Ma, Satwinder Singh, Ruili Wang\*

School of Mathematical and Computational Sciences, Massey University, New Zealand

{Z.Wang4, F.Hou, Y.Qiu1, Z.Ma, S.Singh4, Ruili.Wang}@massey.ac.nz

## Abstract

Recent speech data augmentation approaches use static augmentation operations or policies with consistency magnitude scaling. However, few work is done to explore the influence of the dynamic magnitude of augmentation policies. In this paper, we propose a novel speech data augmentation approach, CyclicAugment, to generate more diversified augmentation policies by dynamically configuring the magnitude of augmentation policies with a cosine annealing scheduler. We also propose additional augmentation operations to enlarge the diversity of augmentation policies. Motivated by learning rate warm restart and cyclical learning rates, we hypothesize that using dynamically configured magnitude for augmentation policies can also help escape local optima more efficiently than static augmentation policies with consistency magnitude scaling. Experimental results demonstrate that our approach is effective for escaping local optima. Our approach achieves 12%-35% relative improvement in word error rate (WER) over SpecAugment and RandAugment on the LibriSpeech 960h dataset, and achieves state-of-the-art result 7.1% in phoneme error rate (PER) on the TIMIT 5h dataset.

**Index Terms:** speech recognition, data augmentation, random augmentation, cosine annealing scheduler

## 1. Introduction

Data augmentation has been effective to improve the generalization of Automatic Speech Recognition (ASR) models, especially by generating synthetic speech data for low resource languages [1]. Typically, the raw waveform was manipulated directly to synthesise new data by injecting noisy audio signals [2], applying speed perturbation [3] or an acoustic room simulator [4] or normalising vocal tract length [5], and so on. Since the mel-spectrogram of speech data is more effective for representing emotion, pitch, and accent than raw waveform [6], augmentation approaches that directly operate on mel-spectrogram of speech data were proposed. These approaches, such as SpecAugment [7], SpecAugment++ [8], SpecSwap [9], SpecMix [10] and MixSpeech [11], treat mel-spectrogram as image and apply image data manipulation operations [12, 13]. However, these approaches use statically mixed augmentation operations (e.g., frequency masking, time masking and time warp) and cannot automatically optimize the augmentation policy.

In image processing domain, several image data augmentation approaches [14, 15, 16, 17] were proposed to automatically find the optimal data augmentation policy based on proxy tasks. This can overcome the performance limitation caused by cumbersome empirical exploration. However, these policy search based data augmentation approaches are computationally expensive, and lack of interpretation whether the optimal augmentation policies found by the proxy tasks were also the optimal

ones for the neural networks being trained. RandAugment [18] was proposed to reduce computational cost by randomly selecting image augmentation operations from a much smaller search space and only searching the strength of all the augmentation operations, i.e., the magnitude of a policy. With a much simplified algorithm, RandAugment even achieves marginal improvements over the above approaches that are computationally expensive. In the studies of SCADA (Stochastic, Consistent and Adversarial Data Augmentation) [19] for ASR, RandAugment was applied to the mel-spectrogram of speech data, and was shown to be slightly more effective than the SpecAugment approach.

Notably, the aforementioned speech and image data augmentation approaches explore augmentation operations or policies for reducing overfitting in the training of deep neural models. However, these approaches use a constant magnitude for augmentation policies; few work is done to explore the influence of the dynamic magnitude of augmentation policies.

In this paper, we propose a novel speech data augmentation approach, CyclicAugment, to generate more diversified augmentation policies by dynamically configuring the magnitude of augmentation policies with a cosine annealing scheduler. Motivated by learning rate warm restart of stochastic gradient descent [20] and cyclical learning rates [21], we hypothesize that training deep neural networks with dynamically configured magnitude for augmentation policies can also help escape local optima more efficiently than static augmentation policies with consistency magnitude scaling. We configure the search space of audio data augmentation operations similarly to RandAugment, but we propose additional augmentation operations to enlarge the diversity of augmentation policies. We evaluate our approach by training three popular end-to-end ASR models with our CyclicAugment approach. Experimental results demonstrate 12%-35% relative improvement in word error rate (WER) over SpecAugment and RandAugment on the LibriSpeech 960h dataset, and achieves state-of-the-art result 7.1% in phoneme error rate (PER) on the TIMIT 5h dataset. The performance gain found periodically in accordance with the cosine graph indicate that our approach is effective for escaping local optima when training an ASR model.

## 2. Speech Data Augmentation Operations

In this section, we present the definition of the individual data augmentation operations on the spectral-domain and the time-domain of speech data for our proposed CyclicAugment approach.

### 2.1. Spectral-Domain Augmentation Operations

We adopt three augmentation operations (i.e., frequency masking, time masking and time warping) proposed by SpecAug-

ment [7]. In addition, to introduce a more variety of mel-spectrogram variants, we design two augmentation operations for the mel-spectrogram: time shifting and loudness amplifying, which are modified from two augmentation approaches for raw waveform data [22]. The definition of each augmentation operation on the spectral-domain is described in detail as below.

**Frequency masking** is applied to  $f$  consecutive input frequency channels where  $[f_0, f_0 + f)$  are masked,  $f_0 \in [0, h - f]$  where  $h$  is the number of input frequency channels, and  $f \in [0, h \times V]$ , where  $V$  is used to adjust the frequency masking size. The masked portion of the mel-spectrogram are padded with 0 or the minimum value of the input data.

**Time masking** is applied to  $t$  consecutive input time steps where  $[t_0, t_0 + t)$  are masked,  $t_0 \in [0, w - t]$  and  $w$  is the number of time steps for the input mel-spectrogram, and  $t \in [0, w \times V]$ , where  $V$  is used for adjusting time masking size. The masked portion of the mel-spectrogram are padded with 0 or the minimum value of the input data.

**Time warping** is to warp the input mel-spectrogram on the time axis by  $t$  percent from the original point, where  $t \in [0, V]$ , and  $V$  is the time warping parameter which increases the speed of a voice. As the augmented input data on the time axis is shorter than the original input data after warping, the gap is filled with 0 or the minimum value of the input data.

**Time shifting** is to shift the input mel-spectrogram on the time axis by  $t$  percent from the original point, where  $t \in [-V, V]$ , and  $V$  is the time shifting parameter which forwards or backwards a speech with a random percentage. The gap left from the mel-spectrogram’s forward and backward shifting is filled with 0 or the minimum value of the input data.

**Loudness amplifying** is applied to  $t$  consecutive input time steps at  $[t_0, t_0 + t)$ ,  $t_0 \in [0, w - t)$  where  $w$  is the number of time steps for the input mel-spectrogram, and  $t \in [0, w \times 0.15]$ . The loudness of speech at  $[t_0, t_0 + t)$  is amplified by  $\lambda$ ,  $\lambda \in (0, V]$  where  $V$  is the extent of the loudness amplifying.

## 2.2. Time-Domain Augmentation Operations

For the raw waveform of speech data, we select 4 single augmentation operations of WaveAugment [23] developed in a public library for the time-domain speech data augmentation: pitch modification (*pitch*), additive noise (*add*), reverberation (*reverb*), and time drop (*tdrop*). We define their augmentation operations in detail as below.

**Pitch modification** is applied to the entire input time steps. The change in the pitch is uniformly sampled by  $p$ , and  $p \in [-V, +V]$ , where  $V$  is used to control pitch change.

**Reverberation** is applied to the entire input time steps with uniformly sampled room-size of  $r$ ,  $r \in [-V, +V]$ , where  $V$  is the reverberation strength.

**Additive noise** is applied to the entire input time steps by adding a Gaussian white noise with scale of  $w$ ,  $w \in [-V, +V]$ , where  $V$  is to control the noise scale.

**Time drop** is applied to  $t$  consecutive input time steps where  $[t_0, t_0 + t)$  are masked with 0 values,  $t_0 \in [0, w - t]$  and  $w$  is the number of time steps for the waveform of the input data, and  $t \in [0, w \times V]$ , where  $V$  is to adjust the time drop span.

## 3. Cyclic Augmentation Approach

In this section, we present the design of our random augmentation search space and introduce our approach of dynamically configuring the augmentation magnitude (i.e., representing the

magnitude scaling) with a cosine annealing scheduler. We use the magnitude to control the scale of augmentation policies applied to the speech data.

### 3.1. Speech Data Random Augmentation with Dynamic Magnitude

Based on the individual augmentation operations defined in Section 2, we configure their unified parameters  $V$  as shown in Table 1 for the scale of each augmentation operation for the speech data.

Table 1: Parameters ( $V$ ) for the spectral-domain and the time-domain augmentation operations.

	Operations	$V$
<b>Spectral-Domain</b>	Frequency masking	0.15
	Time masking	0.2
	Time warping	20
	Time shifting	5
	Loudness Amplifying	1.0
<b>Time-Domain</b>	Pitch modification	150
	Reverberation	50
	Additive noise	20
	Time drop	0.2

We merge the spectral-domain augmentation operations or the time-domain augmentation operations as shown in Table 1 in a search space ( $S$ ), based on the form of speech data for training an ASR model (i.e., either mel-spectrogram or raw waveform).

Algorithm 1 presents the process of the speech data random augmentation. The on-the-fly speech data is transformed by a number of ( $N$ ) randomly selected augmentation operations from the search space ( $S$ ), and then augmented with a unified magnitude ( $M$ ) during the training of ASR models. Algorithm 1 is a similar approach to RandAugment [18] applied to speech data for enlarging the diversity of speech data variants further.

**Algorithm 1:** Pseudo-code for speech data random augmentation

---

```

input : search space of augmentation operations  $S$ ,
        number of operations  $N$ , magnitude of the
        CyclicAugment  $M$ , input speech data  $x$ 
1 for  $n \leftarrow 1$  to  $N$  do
2    $s \leftarrow$  randomly select one operation from  $S$ 
3    $m \leftarrow M \times$  augmentation parameter  $V$  of  $s$ 
4    $y \leftarrow$  apply  $s$  to  $x$  with augmentation parameter
    $m$ 
5 end
6 return  $y$ 

```

---

### 3.2. Configuring Augmentation Magnitude Variable $M$

For the magnitude variable  $M$  representing magnitude scaling in Algorithm 1, we dynamically configure it with a cosine annealing scheduler as follows.

$$M = \alpha \left( \cos 2\pi \frac{T_{cur}}{T_{period}} + 1 \right) \quad (1)$$

where  $T_{cur}$  is the current training epoch value;  $T_{period}$  is the period of cosine function, and  $\alpha$  is a scalar value. Figure 1

shows an example of the magnitude  $M$  arranged by a cosine annealing scheduler with  $T_{period} = 4$  and  $\alpha = 0.75$ , where  $M$  varies between 0 and 1.5 during each period (i.e., 4 epochs) of the training of ASR models.

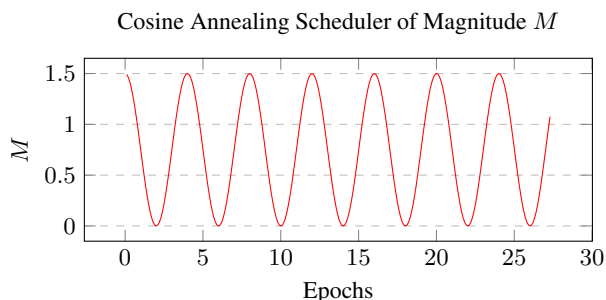


Figure 1: Example of variable  $M$  tuned with a cosine annealing scheduler during the training of ASR models.

## 4. Experiments and Results

### 4.1. LibriSpeech 960h Speech Recognition for Supervised Learning

#### 4.1.1. Models

We use two end-to-end ASR models, the LAS (Listen, Attend, and Spell) [24] and the Conformer [25] for supervised learning setting. We use a vocabulary size of 16000 Word Piece Model [26] to tokenize the text transcripts of the LibriSpeech 960h [27] dataset for training the two models.

For the LAS model, the input data is fed into a 6 convolutional layers of very deep Convolution Neural Network (VGG) [28] to extract the input features. The LAS encoder with 4 layers of bi-directional LSTMs of size 1024, is used to process the output of VGG and generate attention vectors. The attention vectors are then passed into the LAS decoder consisting of 2 layers of bi-directional LSTMs of size 1024 to generate the text predictions.

For the Conformer model, we use the medium size architecture [25] as follows. The input data is processed by convolution kernel with size of 32 for feature extraction. The Conformer encoder consists of 16 layers of conformer blocks with dimension size 256 and 4 attention heads. Its decoder has one layer of LSTM with cell size 640 for generating the text predictions.

For both models, we use a beam size of 8 for beam search to obtain the final transcripts of the text predictions. To get further performance improvements, we incorporate a 3-gram ARPA language model of LibriSpeech 960h dataset [27] by shallow fusion with the two ASR models.

#### 4.1.2. Spectral-Domain Augmentation Policies

We extract 80-dimension filter banks as the input mel-spectrogram data for the two ASR models. To demonstrate the performance of our approach, we compare our CyclicAugment approach with two baselines: SpecAugment [7] and RandAugment [18]. Their policy settings are as follows: (i) For SpecAugment, we apply each input mel-spectrogram with time masking, frequency masking and time warping from the spectral-domain operations in Table 1; (ii) We use RandAugment approach by transforming each input mel-spectrogram with 3 randomly selected individual spectral-domain augmentation operations in Table 1; (iii) We use our CyclicAugment

approach with  $N = 3$ , and a cosine annealing scheduler (i.e.,  $T_{period} = 4$ ,  $\alpha = 2.0$ ), that we randomly select three spectral-domain operations in Table 1 and configure their magnitude with a period of 4 epochs cosine annealing scheduler.

All the other hyperparameters of the above three augmentation policies are fixed without applying additional tuning to inspect the effectiveness of the different augmentation policies for the training of ASR models.

#### 4.1.3. Experiment Results

Both the LAS model and the Conformer model are trained from scratch on the LibriSpeech 960h datasets with the two baselines and our approach (i.e., SpecAugment, RandAugment and our CyclicAugment). Both models are trained for 200 epochs using the Adam optimizer [29] with a learning rate of  $1e-4$ .

The WER results in Table 2 show that ASR models trained with our CyclicAugment approach achieves 12%-35% relative improvements compared with that trained with SpecAugment and RandAugment. In addition, the RandAugment approach has marginal improvement (i.e., less than 5% relative improvement) over the SpecAugment approach. This aligns with the results in comparison with SCADA [19] approach.

Table 2: WER (%) of LAS and Conformer trained with the three augmentation policies assessed on LibriSpeech 960h test sets.

Model + Policy	No LM		With LM	
	Clean	Other	Clean	Other
LAS [24] + SpecAug [7]	8.1	16.2	6.7	14.1
LAS [24] + RandAug [18]	7.9	16.0	6.2	14.0
<b>LAS [24] + CyclicAug (ours)</b>	<b>6.9</b>	<b>14.3</b>	<b>5.1</b>	<b>12.0</b>
Conformer [25] + SpecAug [7]	7.3	12.2	4.7	9.2
Conformer [25] + RandAug [18]	7.0	11.2	4.3	8.8
<b>Conformer [25] + CyclicAug (ours)</b>	<b>5.5</b>	<b>9.3</b>	<b>3.6</b>	<b>6.2</b>

### 4.2. TIMIT 5h Phoneme Recognition for Semi-Supervised Learning

#### 4.2.1. Model

Semi-supervised learning pre-trains a model with large quantities of unlabeled data and then fine-tunes it with labeled data. For our semi-supervised learning setting, we fine-tune the pre-trained wav2vec2.0 [30] ASR models on TIMIT 5h phoneme recognition corpus [31]. Specifically, we use two LARGE size wav2vec2.0 models: LARGE LS-960 (pre-trained on LibriSpeech 960h [27]) and LARGE LV-60K (pre-trained on LibriVox of 60,000 hr unlabeled speech data [32]).

#### 4.2.2. Time-Domain Augmentation Policies

Both aforementioned wav2vec2.0 models are pre-trained with raw audio data rather than mel-spectrogram form. Thus, we fine-tune them with the time-domain augmentation operations in Table 1. When fine-tuning the wav2vec2.0 models, we apply three augmentation policies to the raw waveform speech data in comparison with their performance: (i) the Time drop operation in Table 1 implemented in the original wav2vec2.0 paper [30], (ii) the RandAugment [18] approach with 2 randomly selected time-domain operations in Table 1, (iii) our CyclicAugment with parameters set to  $N = 2$ , and the magnitude  $M$  is scheduled by a cosine annealing scheduler (i.e.,  $T_{period} = 4$ ,  $\alpha = 1.0$ ).

Both pre-trained models are fine-tuned for 100 epochs with Adam [29] in a learning rate of  $3e-5$ . For the TIMIT dataset,

we use 90% TRAIN set for fine-tuning the pre-trained models, 10% TRAIN set for validation, and the TEST set for phoneme predictions with a merged 39 phonemes configuration [33].

#### 4.2.3. Experiment Results

In Table 3, our proposed *CyclicAugment* approach has 8-12% relative gains measured in PER over the *Time drop* approach and the *RandAugment* approach. Further, our *CyclicAugment* approach achieves the state-of-the-art performance (i.e., 7.7% PER for the LARGE LS-960 model and 7.1% PER for the LARGE LV-60K model) compared with the 8.3% PER reported in the wav2vec2.0 paper [30] for the TIMIT dataset.

Table 3: *PER (%) achieved by fine-tuning LARGE LS-960 and LARGE LV-60K with three augmentation policies assessed on TIMIT test set.*

Model + Policy	TEST SET
LARGE LS-960 [30] + Time drop (original)	8.3
LARGE LS-960 [30] + Time drop	8.4
LARGE LS-960 [30] + RandAug [18]	8.2
<b>LARGE LS-960 [30] + CyclicAug (ours)</b>	<b>7.7</b>
LARGE LV-60K [30] + Time drop	8.0
LARGE LV-60K [30] + RandAug [18]	7.7
<b>LARGE LV-60K [30] + CyclicAug (ours)</b>	<b>7.1</b>

#### 4.3. Optimal Model Checkpoints with *CyclicAugment*

In this sub-section, we interpret the performance gains achieved by using our *CyclicAugment* approach to train an ASR model. Figure 2 shows *CyclicAugment* magnitude  $M$ , training loss and validation PER % evaluated on the TIMIT dataset [31] for our semi-supervised learning setting. Curves of different colours denote the three approaches used to fine-tune the LARGE LV-60K wav2vec2.0 pre-trained model: *Time drop*, *RandAugment* [18] and our *CyclicAugment* approach. Combining the training loss curves and validation PER % curves, we can see that training ASR models with the *Time drop* approach tends to be overfitting with its weaker magnitude, and training with the *RandAugment* approach delays overfitting with a stronger magnitude as seen in the performance gains over the *Time drop* approach. While for the training loss curve of using our *CyclicAugment* approach, a distinct pattern is the periodical high and low training loss synchronized with the periodical magnitude of augmentation operations (i.e., the cosine annealing scheduler of the magnitude  $M$ ). Our proposed dynamic magnitude of augmentation policies works in a mechanism similar to the learning rate warm restart of stochastic gradient descent [20] and cyclical learning rate [21], which is effective for escaping local optima. Thus, training an ASR model with our *CyclicAugment* approach can visit multiple local optima and have higher chance to converge to the global optima. Thus, good validation results can be found frequently at the troughs of the cosine annealing scheduler of *CyclicAugment*, and the changes around the crests indicate that the model is likely to escape the local optimal and start moving to the next one.

Recent speech data augmentation approaches (e.g., *SpecAugment* [7], *SpecSwap* [9], *WaveAugment* [23]) focus on exploring augmentation operations or policies with consistency magnitude mainly to reduce overfitting. Our proposed approach dynamically configure the magnitude of augmenta-

tion policies. It delays overfitting, and also can assist in escaping local optima during the model training with longer training epochs. Moreover, our *CyclicAugment* approach can be used jointly with adaptive learning rate optimizers (e.g., Adam [29] and Adadelta [34]) for training an ASR model. Training ASR models with our approach can converge quickly towards local optima with the adaptive learning rate optimizers, and periodically escape local optima and converge to a global optimum.

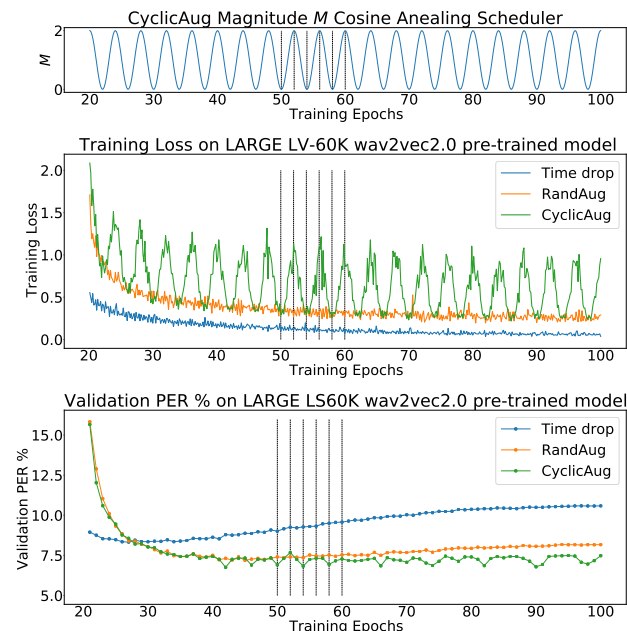


Figure 2: *CyclicAugment* magnitude  $M$ , training loss curve and validation PER % of using the three approaches to fine-tune the LARGE LV-60K wav2vec2.0 pre-trained model on TIMIT dataset.

## 5. Conclusions

In this paper, we propose a computationally cheap yet effective data augmentation approach, *CyclicAugment*, to generate diversified data augmentation policies for ASR. *CyclicAugment* uses a cosine annealing scheduler to dynamically configure the magnitude of augmentation policies. Experiment results show that our *CyclicAugment* approach is effective for both mel-spectrogram and raw waveform speech data in supervised learning and semi-supervised learning settings. Our proposed *CyclicAugment* approach can improve generalization, and also can help escape local optima in the training of ASR models.

In the future, we will try the slanted triangular learning rates [35] approach to tune the magnitude of augmentation policies. We will also test our *CyclicAugment* approach on other speech processing technologies, such as speech enhancement [36] and smoker identification [37].

## 6. Acknowledgements

This work is partly supported by the 2020 Catalyst: Strategic NZ-Singapore Data Science Research Programme Fund, MBIE, New Zealand, also by the New Zealand eScience Infrastructure (NeSI). Feng Hou\* and Ruili Wang\* are the corresponding authors.

## 7. References

- [1] A. Ragni, K. M. Knill, S. P. Rath, and M. J. F. Gales, “Data augmentation for low resource languages,” in *Proc. Interspeech 2014*, 2014, pp. 810–814.
- [2] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Sathesh, S. Sengupta, A. Coates, and A. Y. Ng, “Deep speech: Scaling up end-to-end speech recognition,” *arXiv preprint arXiv 1412.5567*, 2014.
- [3] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, “Audio augmentation for speech recognition,” in *Proc. Interspeech*, 2015, pp. 3586–3589.
- [4] C. Kim, A. Misra, K. Chin, T. Hughes, A. Narayanan, T. N. Sainath, and M. Bacchiani, “Generation of Large-Scale Simulated Utterances in Virtual Rooms to Train Deep-Neural Networks for Far-Field Speech Recognition in Google Home,” in *Proc. Interspeech*, 2017, pp. 379–383.
- [5] N. Jaitly and G. E. Hinton, “Vocal tract length perturbation (vtlp) improves speech recognition,” in *Proc. International Conference on Machine Learning (ICML)*, vol. 28, 2013.
- [6] E. Sejdić, I. Djurović, and J. Jiang, “Time–frequency feature representation using energy concentration: An overview of recent advances,” *Digital Signal Processing*, vol. 19, no. 1, pp. 153–183, 2009.
- [7] D. S. Park, W. Chan, Y. Zhang, C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition,” in *Proc. Interspeech*, 2019, pp. 2613–2617.
- [8] H. Wang, Y. Zou, and W. Wang, “SpecAugment++: A Hidden Space Data Augmentation Method for Acoustic Scene Classification,” in *Proc. Interspeech*, 2021, pp. 551–555.
- [9] X. Song, Z. Wu, Y. Huang, D. Su, and H. Meng, “Specswap: A simple data augmentation method for end-to-end speech recognition,” in *Proc. Interspeech*, 2020, pp. 581–585.
- [10] G. Kim, D. K. Han, and H. Ko, “SpecMix : A Mixed Sample Data Augmentation Method for Training with Time-Frequency Domain Features,” in *Proc. Interspeech*, 2021, pp. 546–550.
- [11] L. Meng, J. Xu, X. Tan, J. Wang, T. Qin, and B. Xu, “Mixspeech: Data augmentation for low-resource automatic speech recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 7008–7012.
- [12] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “mixup: Beyond empirical risk minimization,” in *Proc. International Conference on Learning Representations (ICLR)*, 2018.
- [13] S. Yun, D. Han, S. Chun, S. Oh, Y. Yoo, and J. Choe, “Cutmix: Regularization strategy to train strong classifiers with localizable features,” in *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 6022–6031.
- [14] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, “Learning transferable architectures for scalable image recognition,” in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 8697–8710.
- [15] E. D. Cubuk, B. Zoph, D. Mané, V. Vasudevan, and Q. V. Le, “Autoaugment: Learning augmentation strategies from data,” in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 113–123.
- [16] D. Ho, E. Liang, I. Stoica, P. Abbeel, and X. Chen, “Population based augmentation: Efficient learning of augmentation policy schedules,” in *Proc. International Conference on Machine Learning (ICML)*, 2019.
- [17] S. Lim, I. Kim, T. Kim, C. Kim, and S. Kim, “Fast autoaugment,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019, pp. 6665–6675.
- [18] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, “Randaugment: Practical automated data augmentation with a reduced search space,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020, pp. 18 613–18 624.
- [19] G. Wang, A. Rosenberg, Z. Chen, Y. Zhang, B. Ramabhadran, and P. Moreno, “Scada: Stochastic, consistent and adversarial data augmentation to improve asr,” in *Proc. Interspeech*, 2020.
- [20] I. Loshchilov and F. Hutter, “Sgdr: Stochastic gradient descent with warm restarts,” in *Proc. International Conference on Learning Representations (ICLR)*, 2017.
- [21] L. N. Smith, “Cyclical learning rates for training neural networks,” in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017, pp. 464–472.
- [22] L. R. Bahl, F. Jelinek, and R. L. Mercer, “A maximum likelihood approach to continuous speech recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-5, no. 2, pp. 179–190, 1983.
- [23] E. Kharitonov, M. Riviere, L. W. G. Synnaeve, P. Mazar, M. Douze, and E. Dupoux, “Data augmenting contrastive learning of speech representations in the time domain,” in *Proc. IEEE Spoken Language Technology Workshop (SLT)*, 2021, pp. 215–222.
- [24] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 4960–4964.
- [25] A. Gulati, J. Qin, C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, “Conformer: Convolution-augmented Transformer for Speech Recognition,” in *Proc. Interspeech 2020*, 2020, pp. 5036–5040.
- [26] M. Schuster and K. Nakajima, “Japanese and korean voice search,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 5149–5152.
- [27] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An asr corpus based on public domain audio books,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.
- [28] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *Proc. International Conference on Learning Representations (ICLR)*, 2015.
- [29] P. K. Diederik and B. Jimmy, “Adam: A method for stochastic optimization,” in *Proc. International Conference on Learning Representations (ICLR)*, 2015.
- [30] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020, pp. 12 449–12 460.
- [31] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zue, “Timit acoustic-phonetic continuous speech corpus,” *Linguistic Data Consortium*, 1992.
- [32] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P. E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fugien, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux, “Libri-light: A benchmark for asr with limited or no supervision,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7669–7673.
- [33] K.-F. Lee and H.-W. Hon, “Speaker-independent phone recognition using hidden markov models,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 11, pp. 1641–1648, 1989.
- [34] M. D. Zeiler, “ADDELTA: an adaptive learning rate method,” *arXiv preprint arxiv 1212.5701*, 2012.
- [35] J. Howard and S. Ruder, “Universal language model fine-tuning for text classification,” in *Proc. the Association for Computational Linguistics (ACL)*, 2018, pp. 328–339.
- [36] Y. Qiu, R. Wang, S. Singh, Z. Ma, and F. Hou, “Self-supervised learning based phone-fortified speech enhancement,” in *Proc. Interspeech*, 2021, pp. 211–215.
- [37] Z. Ma, Y. Qiu, F. Hou, R. Wang, J. T. Wai Chu, and C. Bullen, “Determining the best acoustic features for smoker identification,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 8177–8181.