INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

MCR-Data2vec 2.0: Improving Self-supervised Speech Pre-training via Model-level Consistency Regularization

Ji Won Yoon[1], *Seok Min Kim*[1], *Nam Soo Kim*[1]

[1]Department of ECE and INMC, Seoul National University, Seoul, Republic of Korea
{jwyoon, smkim}@hi.snu.ac.kr, nkim@snu.ac.kr

Abstract

Self-supervised learning (SSL) has shown significant progress in speech processing tasks. However, despite the intrinsic randomness in the Transformer structure, such as dropout variants and layer-drop, improving the model-level consistency remains under-explored in the speech SSL literature. To address this, we propose a new pre-training method that uses consistency regularization to improve Data2vec 2.0, the recent state-of-the-art (SOTA) SSL model. Specifically, the proposed method involves sampling two different student sub-models within the Data2vec 2.0 framework, enabling two output variants derived from a single input without additional parameters. Subsequently, we regularize the outputs from the student sub-models to be consistent and require them to predict the representation of the teacher model. Our experimental results demonstrate that the proposed approach improves the SSL model's robustness and generalization ability, resulting in SOTA results on the SUPERB benchmark.

Index Terms: Self-Supervised Learning, Speech Pre-Training, Consistency Regularization

1. Introduction

Self-supervised learning (SSL) has shown great promise in speech processing [1, 2, 3, 4]. It uses large amounts of unlabeled speech audio data to learn general speech representations, which can benefit various downstream tasks by fine-tuning.

Recently, research in speech SSL algorithms has focused on data augmentation for a noisy scenario to improve the pre-training of SSL models [3, 5, 6, 7, 8, 9]. WavLM [3] employs a masked speech denoising and prediction framework to pre-train speech representations. Some inputs are artificially simulated to be noisy or overlapped with masks, and the model predicts pseudo-labels of the original speech on the masked region. CCC-Wav2vec 2.0 [5] introduces an augmentation of the original sample and uses its representations to add an extra cross-contrastive loss to the Wav2vec 2.0 [2] framework. Robust Data2vec [9] improves the noise robustness of the Data2vec [4] by allowing the model to have consistent predictions for both original and noisy speech. These studies mainly aim to make the SSL model more robust to augmentations, which in turn helps learn better representations.

While augmentation-based approaches are certainly helpful to improve the SSL model's robustness against the data variation, they may not consider randomness in the model architecture. For example, most recent SSL models are based on the Transformer [10] structure, which has intrinsic randomness due to multiple dropout variants, such as standard dropout [11] for each module, LayerDrop [12, 13], etc. During the pre-training stage, these dropout variants randomly discard a portion of layers or neurons, selecting a random *sub-model* at each iteration. However, when fine-tuning on a downstream task, all or a part of the pre-trained SSL model's parameters are commonly frozen [14, 15, 16, 17, 18, 19], causing inconsistency between the pre-training and fine-tuning. For ease of understanding, assuming that we freeze all of the SSL model's parameters for the downstream task, the pre-training is performed on a *sub-model* that includes sources of randomness. In contrast, the fine-tuning is conducted on a single *full model* without randomness. This difference in the amount of randomness between the pre-training and fine-tuning can create a gap between the two stages, leading to performance degradation when applying the SSL model to downstream tasks.

To address this issue, one possible solution is to encourage the sub-models to produce consistent outputs, regardless of the sources of randomness. If the sub-models are less affected by the randomness, the gap between the sub-model (pre-training) and the full model (fine-tuning) can be alleviated. Previous studies [20, 21, 22, 23] have attempted to improve model-level consistency by regularizing the output predictions of sub-models to be consistent. However, since these techniques are mainly designed for supervised or semi-supervised settings, it is difficult to directly apply them to fully unsupervised pre-training. Therefore, it is necessary to design a new consistency regularization framework that can be applied during the pre-training stage of speech SSL.

In this paper, we introduce MCR-Data2vec 2.0, a new pre-training method for improving the model-level consistency of Data2vec 2.0 [24], which is the current state-of-the-art (SOTA) SSL model. Based on the teacher-student scheme of the Data2vec 2.0 framework, the proposed method randomly samples two different student sub-models, enabling two output variants derived from a single input without additional parameters. Due to the dropout variants, the two sub-models are based on different subsets of layers or neurons. Then, we regularize the outputs of the student sub-models to be consistent with each other and require them to predict the same target representation of the teacher model. Regularizing the student sub-models to produce similar outputs makes them less affected by randomness. Thus, MCR-Data2vec 2.0 can effectively reduce the gap between the pre-training and fine-tuning stages, improving the overall quality of the SSL model.

Through extensive experiments on SUPERB [25, 26], we confirm that the MCR-Data2vec 2.0 effectively improves the SSL model's robustness and generalization ability. It achieves SOTA performance on multiple subtasks of SUPERB, including Phoneme Recognition (PR), Automatic Speech Recognition (ASR), Keyword Spotting (KS), Intent Classification (IC), Slot Filling (SF), Emotion Recognition (ER). Furthermore, MCR-Data2vec 2.0 yields considerable performance improvements

10.21437/Interspeech.2023-1579

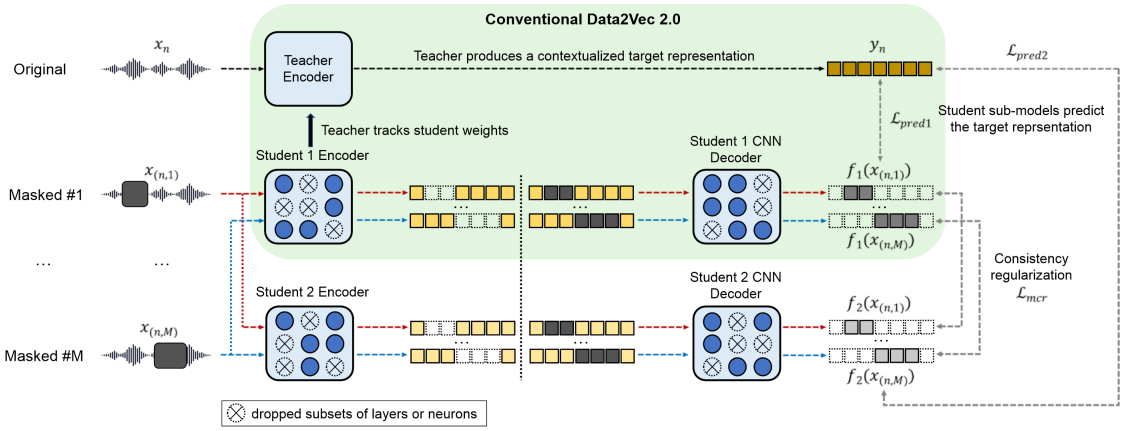


Figure 1: *An overview of MCR-Data2vec 2.0. In the student encoder and decoder, a crossed-out dotted circle represents dropped subsets of layers or neurons due to the dropout variants. First, a contextualized representation y_n is generated from the teacher model based on the unmasked training sample x_n. The teacher model's weights are a moving average of the student model's weights. MCR-Data2vec 2.0 randomly samples two different student sub-models, resulting in two predictions $f_1(x_{(n,m)})$ and $f_2(x_{(n,m)})$ for the same masked input $x_{(n,m)}$. The masked portions of the training sample are not encoded [24]. To improve model-level consistency, the outputs from the student sub-models are regularized to be consistent ($\mathcal{L}_{mcr}$), and the student sub-models predict the same contextualized target representation for various masked versions of the training example ($\mathcal{L}_{pred1}$ and $\mathcal{L}_{pred2}$). $\mathcal{L}_{pred1}$ corresponds to the training objective of the original Data2vec 2.0.*

for speaker and generation-related tasks compared to the original Data2vec 2.0.

To summarize, the main contributions of this paper are:

- We propose MCR-Data2vec 2.0, a new consistency regularization framework designed to improve the recent Data2vec 2.0. *To the best of our knowledge, this is the first attempt to apply the model-level consistency regularization during the pre-training of speech SSL.*
- By regularizing the outputs of student sub-models to be consistent, the proposed method can reduce the gap between the pre-training and fine-tuning stages.
- Through experimental results on SUPERB, we validate that MCR-Data2vec 2.0 significantly improves the overall quality of Data2vec 2.0. Compared to recent SSL models, our framework achieves promising results on various downstream tasks.

2. Background: Data2vec 2.0

As aforementioned, we use Data2vec 2.0 [24] as our backbone model since it is one of the most recent SOTA SSL models. Before we describe the proposed method, it might be beneficial to review some properties of Data2vec 2.0.

2.1. Teacher-student Setup

Similar to Data2vec [4], Data2vec 2.0 follows the teacher-student scheme, where the weights of the teacher Δ are an exponentially moving average (EMA) of the student encoder θ [27]: $\Delta \leftarrow \tau\Delta + (1-\tau)\theta$. The parameter τ linearly increases from a starting value τ_0 to a final value τ_e over τ_n updates, after which the value is kept constant [4, 24].

2.2. Contextualized Target Prediction

As shown in Figure 1 (green highlight), Data2Vec 2.0 is based on multi-mask training that considers M different masked versions of the training sample, similar to Masked Autoencoders (MAE) [28]. Given a training dataset $\mathcal{D} = \{x_1, x_2, ..., x_N\}$ where N is the number of training data samples, each sample x_n is masked M times. The m-th masked version of sample x_n is represented as $x_{(n,m)}$. y_n denotes a contextualized target representation from the teacher model based on the original sample x_n.

In the Data2vec 2.0, the teacher model consumes the unmasked training sample x_n to produce the target representation y_n. Specifically, the output of the top K blocks of the teacher model is averaged to construct y_n. This target representation is contextualized since the teacher model uses a self-attention mechanism in the Transformer architecture [10]. Each masked version of sample is fed into the student model, which predicts the same target representation y_n for the different masked versions. When passing the masked sample $x_{(n,m)}$, the student does not encode masked tokens to further improve the model efficiency. During the pre-training, Data2vec 2.0 is trained to minimize the l_2 loss ($\mathcal{L}_{pred1}$ in Figure 1) between the student's prediction and the contextualized target representation.

3. MCR-Data2vec 2.0

In this section, we introduce our MCR-Data2vec 2.0 framework and the training algorithm. Based on Data2vec 2.0, we propose a new model-level consistency regularization to improve the pre-training stage of SSL. The overall framework of our regularization method is shown in Figure 1.

3.1. Student Sub-model Sampling

Unlike Data2vec 2.0, MCR-Data2vec 2.0 samples two different student sub-models f_1 and f_2 from a full single student model f. This sub-model sampling allows two output variants to be derived from a single input without requiring additional parameters or model structure changes.

Although it is possible to use multiple sub-models for train-

ing, this can be computationally expensive and time-consuming. We experimentally confirm that MCR-Data2vec 2.0 can achieve considerable improvements using only two sub-models.

3.2. Model-level Consistency Regularization

To train our network, we pass each masked input $x_{(n,m)}$ through the network twice, resulting in two predictions denoted as $f_1(x_{(n,m)})$ and $f_2(x_{(n,m)})$. Note that the two forward passes are based on different sub-models due to the dropout variants. Thus, the predictions $f_1(x_{(n,m)})$ and $f_2(x_{(n,m)})$ are different for the same masked input $x_{(n,m)}$.

Firstly, the MCR-Data2vec 2.0 requires the student sub-models to predict the same target representation y_n of the teacher model. We minimize the l_2 loss between the target representation and the predictions, which can be formulated as:

$$\begin{aligned}\mathcal{L}_{pred}^{(n,m)} &= \mathcal{L}_{pred1}^{(n,m)} + \mathcal{L}_{pred2}^{(n,m)} \\ &= (y_n - f_1(x_{(n,m)}))^2 + (y_n - f_2(x_{(n,m)}))^2 \end{aligned} \quad (1)$$

where $\mathcal{L}_{pred1}^{(n,m)}$ corresponds to the training objective of the original Data2vec 2.0.

Also, the MCR-Data2vec 2.0 aims to encourage the student sub-models to produce consistent outputs, regardless of randomness in the model architecture. Since the target of the sub-models is the contextualized representation, both $f_1(x_{(n,m)})$ and $f_2(x_{(n,m)})$ are continuous. Thus, we minimize the l_2 loss between $f_1(x_{(n,m)})$ and $f_2(x_{(n,m)})$ to perform the consistency regularization. The proposed regularization objective $\mathcal{L}_{mcr}^{(n,m)}$ for the masked sample x_n^m can be computed as follows:

$$\mathcal{L}_{mcr}^{(n,m)} = (f_1(x_{(n,m)}) - f_2(x_{(n,m)}))^2. \quad (2)$$

Although Eq. (1) aims to make two different sub-models predict the same target y_n, relying solely on this objective function may not sufficiently reduce the variance (difference) between the outputs of the sub-models. By minimizing the variance between the outputs of different sub-models via Eq. (2), we can further improve the model's consistency.

3.3. Training Objective

The final objective $\mathcal{L}_{total}^{(n,m)}$ for the sample $x_{(n,m)}$ is given as

$$\mathcal{L}_{total}^{(n,m)} = \mathcal{L}_{pred}^{(n,m)} + \lambda \mathcal{L}_{mcr}^{(n,m)} \quad (3)$$

where λ is a tunable parameter, and we experimentally set λ to 1.

As mentioned earlier, there is the gap between the pre-training and fine-tuning stages due to the difference in the amount of randomness. By regularizing the outputs from the two sub-models to be consistent, the sub-models are less affected by the randomness. Thus, MCR-Data2vec 2.0 can effectively reduce the inconsistency between the pre-training and fine-tuning, improving the overall quality of the SSL model.

4. Experimental Setup

4.1. Pre-training Setup

We conducted experiments using the Base model configuration, which includes 12 Transformer blocks, 768-dimensional hidden states, and 8 attention heads, resulting in a total of 93.78 M parameters for MCR-Data2vec 2.0. To pre-train the Data2vec 2.0 Base and MCR-Data2vec 2.0 Base, we used 960 hours of audio from LibriSpeech [29] and trained it for 400K updates on 8 Quadro RTX 8000 48GB GPUs. The models were implemented using fairseq toolkit [30]. For all other training configurations, we followed the hyperparameters outlined in Data2vec 2.0 [24]. The multi-masking strategy for the MCR-Data2vec 2.0 was also identical to Data2vec 2.0.

4.2. Universal Representation Evaluation

To evaluate the effectiveness of our proposed approach, we conducted experiments using SUPERB [25, 26], a standardized benchmark for pre-trained models in a range of speech tasks. We evaluated our model on eleven subtasks of SUPERB, including Phoneme Recognition (PR), Automatic Speech Recognition (ASR), Keyword Spotting (KS), Query by Example Spoken Term Detection (QbE), Intent Classification (IC), Slot Filling (SF), Emotion Recognition (ER), Speaker Identification (SID), Automatic Speaker Verification (ASV), Speech Enhancement (SE), and Speech Separation (SS). We followed the guidelines created by SUPERB. Firstly, we employed the same downstream models that were utilized by SUPERB for each task. Secondly, we froze the pre-trained models during the fine-tuning. Finally, the downstream models processed the weighted sum of hidden state results obtained from each layer of the pre-trained model. For ASR, ASV, SE, and SS, we followed the official configurations of SUPERB. For the other downstream tasks, we followed the fine-tuning hyperparameter settings of WavLM [3]. During the fine-tuning, we trained the models on a single Quadro RTX 8000 48GB GPU.

5. Experimental Result

5.1. Main Results

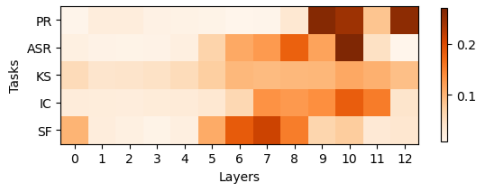
We compared the MCR-Data2vec 2.0 with previous SOTA approaches, including HuBERT [1], Wav2vec 2.0 [2], WavLM [3], Data2vec [4], CCC-Wav2vec 2.0 [5], and Data2vec 2.0 [24]. Table 1 summarizes the results on SUPERB benchmark. Even though Data2vec 2.0 achieved promising results compared to previous SSL methods, MCR-Data2vec 2.0 outperformed the original Data2vec 2.0 in most configurations, except for the QbE task. For the QbE, the performance of the MCR-Data2vec 2.0 was worse than that of the Data2vec 2.0. Considering that the HuBERT Large model performed worse than the HuBERT Base model on the QbE task [25], higher performance on QbE did not always indicate a better SSL model.

As shown in Table 1, the proposed approach yielded SOTA performances for PR, ASR, KS, IC, SF, and ER tasks, while Data2vec 2.0 had the second-best results. By simply regularizing the output of student sub-models to be consistent, the proposed framework effectively improved the overall quality of the original Data2vec 2.0. From the results, it is verified that reducing the gap between the pre-training and fine-tuning stages is important in training the SSL model.

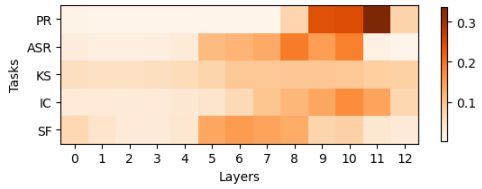
Additionally, the proposed method showed significant improvements over the Data2vec 2.0 for speaker and generation tasks, achieving the second-best performances for SID, SE, and SS tasks. In the case of STOI for SE, MCR-Data2vec 2.0 yielded the best performance along with CCC-Wav2vec 2.0. It is important to note that WavLM and CCC-Wav2vec 2.0, which performed well on such tasks, benefited from data augmentation for the noisy scenario during the pre-training. WavLM [3] manually simulated noisy/overlapped speech as inputs, and CCC-Wav2vec 2.0 [5] employed three different data augmen-

Method	# Params	Content				Semantics			ParaL	Speaker		Generation		
		PR	ASR	KS	QbE	IC	SF		ER	SID	ASV	SE		SS
		PER ↓	WER ↓	Acc ↑	MTWV ↑	Acc ↑	F1 ↑	CER ↓	Acc ↑	Acc ↑	EER ↓	PESQ ↑	STOI ↑	SI-SDRi ↑
HuBERT [1]	94.70 M	5.41	6.42	96.30	0.0736	98.34	88.53	25.20	64.92	81.42	5.11	2.58	93.90	9.36
Wav2vec 2.0 [2]	95.04 M	5.74	6.43	96.23	0.0233	92.35	88.30	24.77	63.43	75.18	6.02	2.55	93.90	9.77
WavLM [3]	94.70 M	4.84	6.21	96.79	**0.0870**	98.63	89.38	22.86	65.94	**84.51**	**4.69**	2.58	94.01	10.37
Data2vec [4]	93.75 M	4.69	4.94	96.56	0.0576	97.63	88.59	25.27	66.27	70.21	5.77	2.96	94.83	9.78
CCC-Wav2vec 2.0 [5]	95.04 M	5.95	6.30	96.72	0.0673	96.47	88.08	24.34	64.17	72.84	5.61	**3.06**	**94.94**	**10.86**
Our implementation														
Data2vec 2.0 [24]	93.78 M	3.64	4.81	96.89	0.0841	99.00	89.67	22.09	66.66	81.43	5.59	2.98	94.85	10.41
MCR-Data2vec 2.0 (Ours)	93.78 M	**3.37**	**4.68**	**97.05**	0.0595	**99.21**	**90.04**	**21.73**	**66.99**	82.40	5.36	3.01	**94.94**	10.62

Table 1: *Performance comparison on the SUPERB benchmark. ParaL represents Paralinguistics aspect of speech. All models were trained using 960 hours of LibriSpeech and based on the Base setting, consisting of 12 Transformer blocks. We marked the best in* ***bold*** *and the second-best with underline.*



(a) *Data2vec 2.0*



(b) *MCR-Data2vec 2.0*

Figure 2: *Weight analysis on content and semantic tasks.*

tations to make the SSL model more robust to augmentations. However, MCR-Data2vec 2.0 achieved comparable results on speaker and generation tasks without using such data augmentation techniques. This means that our model-level consistency regularization was supportive in improving the generalization ability and robustness of the SSL model.

5.2. Weight Analysis

Following the SUPERB guidelines, we computed a weighted sum of the representations obtained from each layer of the pre-trained model and then fed it to the downstream models. Since MCR-Data2vec 2.0 showed considerable performance improvements, especially for content and semantic tasks, we analyzed the contribution patterns of the proposed approach. Figure 2 depicts the weights of the different layers of Data2vec 2.0 and MCR-Data2vec 2.0 for PR, ASR, KS, IC, and SF tasks, where the higher weight means that the corresponding layer contributed more to achieving the specific task. From the results, we found that the patterns of MCR-Data2vec 2.0 were similar to those of Data2vec 2.0. Even though WavLM [3] reported that the content and semantic information were primarily encoded in the top layers, both SSL models tended to leverage information from both middle and top layers. In Data2vec 2.0 framework, the use of information from broader layers could be an important factor in achieving better performance for content and semantic tasks. In addition, we can observe that the contribution patterns of MCR-Data2vec 2.0 were not predominantly concentrated on a specific layer compared to those of Data2vec 2.0. For example, for the ASR task, Data2vec 2.0 heavily relied

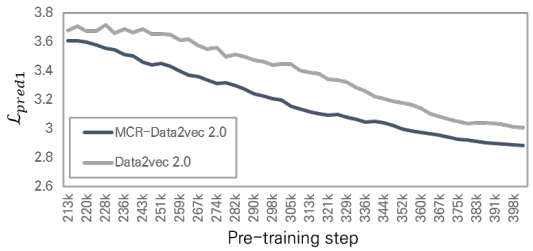


Figure 3: *$\mathcal{L}_{pred1}$ loss curves of Data2vec 2.0 and MCR-Data2vec 2.0, where the total number of updates is 400k.*

on the 10th layer, whereas the patterns of MCR-Data2vec 2.0 were more evenly distributed across layers 5 to 10. Similarly, for the SF task, Data2vec 2.0 predominantly assigned weights to the 6th and 7th layers, while MCR-Data2vec 2.0 utilized information more uniformly across layers 5 to 8.

5.3. $\mathcal{L}_{pred1}$ Loss Curve

To check the effectiveness of the proposed regularization term, we analyzed the pre-training loss function L_{pred1} (in Eq (1)). L_{pred1} measures the distance between the student sub-model's prediction and the target representation of the teacher model. It is important to note that, as shown in Figure 1, both Data2vec 2.0 and MCR-Data2vec 2.0 used L_{pred1} during the pre-training stage. The loss curves for $\mathcal{L}_{pred1}$ are shown in Figure 3. The results showed that $\mathcal{L}_{pred1}$ of MCR-Data2vec 2.0 significantly decreased compared to that of Data2vec 2.0. This indicates that the proposed regularization improved the student model's ability to predict the target representation of the teacher model.

6. Conclusion

We proposed a novel pre-training method, MCR-Data2vec 2.0, to improve the model-level consistency of speech SSL. The proposed framework could reduce the gap between the pre-training and fine-tuning stages while improving the overall quality of the SSL model. From experimental results on the SUPERB benchmark, it is verified that MCR-Data2vec 2.0 achieved promising performance, outperforming recent SSL models.

7. Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2021-0-00456, Development of Ultra-high Speech Quality Technology for Remote Multi-speaker Conference System)

8. References

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

[2] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: a framework for self-supervised learning of speech representations," in *Proc. NIPS*, 2020.

[3] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, J. Wu, L. Zhou, S. Ren, Y. Qian, Y. Qian, J. Wu, M. Zeng, X. Yu, and F. Wei, "Wavlm: large-scale self-supervised pre-training for full stack speech processing," *JSTSP*, 2022.

[4] A. Baevski, W. Hsu, Q. Xu, A. Babu, J. Gu, and M. Aulim, "Data2vec: a general framework for self-supervised learning in speech, vision and language," in *Proc. ICML*, 2022, pp. 1298–1312.

[5] V. S. Lodagala, S. Ghosh, and S. Umesh, "Ccc-wav2vec 2.0: clustering aided cross contrastive self-supervised learning of speech representations," in *Proc. IEEE SLT*, 2022.

[6] E. Kharitonov, M. Rivière, G. Synnaeve, L. Wolf, P. Mazaré, M. Douze, and E. Dupoux, "Data augmenting contrastive learning of speech representations in the time domain," in *Proc. IEEE SLT*, 2021.

[7] A. Sriram, M. Auli, and A. Baevski, "Wav2vec-aug: improved self-supervised training with limited data," *arXiv preprint arXiv:2206.13654v1*, 2022.

[8] C. Gao, G. Cheng, and P. Zhang, "Multi-variant consistency based self-supervised learning for robust automatic speech recognition," *arXiv preprint arXiv:2112.12522v2*, 2022.

[9] Q. Zhu, L. Zhou, J. Zhang, S. Liu, Y. Hu, and L. Dai, "Robust data2vec: noise-robust speech representation learning for asr by combining regression and improved contrastive learning," in *Proc. ICASSP*, 2023.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, 2017, pp. 5998–6008.

[11] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *JMLR*, 2014.

[12] A. Fan *et al.*, "Reducing transformer depth on demand with structured dropout," in *Proc. ICLR*, 2020.

[13] G. Huang *et al.*, "Deep networks with stochastic depth," in *Proc. ECCV*, 2016.

[14] L. Tseng, Y. Fu, H. Chang, and H. Lee, "Mandarin-english codewwitching speech recognition with self-supervised speech representation models," in *Proc. AAAI SAS*, 2022.

[15] S. Lee, S. Kim, J. Lee, E. Song, M. Hwang, and S. Lee, "Hierspeech: bridging the gap between text and speech by hierarchical variational inference using self-supervised representations for speech synthesis," in *Proc. NIPS*, 2022.

[16] H. Siuzdak, P. Dura, P. Rijn, and N. Jacoby, "Wavthruvec: latent speech representation as intermediate features for neural speech synthesis," in *Proc. INTERSPEECH*, 2022.

[17] B. Thomas, S. Kessler, and S. Karout, "Efficient adapter transfer of self-supervised speech models for automatic speech recognition," in *Proc. ICASSP*, 2022.

[18] N. Pham, A. Waibel, and J. Niehues, "Adaptive multilingual speech recognition with pretrained models," in *Proc. INTERSPEECH*, 2022.

[19] R. Wang *et al.*, "Lighthubert: lightweight and configurable speech representation learning with once-for-all hidden-unit bert," in *Proc. INTERSPEECH*, 2022.

[20] X. Liang, L. Wu, J. Li, Y. Wang, Q. Meng, T. Qin, W. Chen, M. Zhang, and T. Liu, "R-drop: regularized dropout for neural networks," in *Proc. NIPS*, 2021.

[21] K. Zolna, D. Arpit, D. Suhubdy, and Y. Bengio, "Fraternal dropout," in *Proc. ICLR*, 2018.

[22] A. Tarvainen and H. Valpola, "Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. NIPS*, 2017.

[23] B. Zheng, L. Dong, S. Huang, W. Wang, Z. Chi, S. Singhal, W. Che, T. Liu, X. Song, and F. Wei, "Consistency regularization for cross-lingual fine-tuning," in *Proc. ACL*, 2021.

[24] B. Baevski, A. Babu, W. Hsu, and M. Auli, "Efficient self-supervised learning with contextualized target representations for vision, speech and language," *arXiv preprint arXiv:2212.07525v1*, 2022.

[25] S. Yang *et al.*, "Superb: speech processing universal performance benchmark," in *Proc. INTERSPEECH*, 2021, pp. 1194–1198.

[26] A. Tarvainen and H. Valpola, "Superb-sg: enhanced speech processing universal performance benchmark for semantic and generative capabilities," in *Proc. ACL*, 2022.

[27] J. Grill *et al.*, "Bootstrap your own latent: A new approach to self-supervised learning," in *Proc. NIPS*, 2020.

[28] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. CVPR*, 2022.

[29] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Proc. ICASSP*, 2015, pp. 5206–5210.

[30] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, "Fairseq: a fast, extensible toolkit for sequence modeling," in *Proc. NAACL*, 2019, p. 48–53.