INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Task-Agnostic Structured Pruning of Speech Representation Models

Haoyu Wang[1], *Siyuan Wang*[1], *Wei-Qiang Zhang*[1]*, *Hongbin Suo*[2], *Yulong Wan*[2]

[1] Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
[2] Data & AI Engineering System, OPPO, Beijing 100026, China

w-hy21@mails.tsinghua.edu.cn, wq-zhang@tsinghua.edu.cn

Abstract

Self-supervised pre-trained models such as Wav2vec2, Hubert, and WavLM have been shown to significantly improve many speech tasks. However, their large memory and strong computational requirements hinder their industrial applicability. Structured pruning is a hardware-friendly model compression technique but usually results in a larger loss of accuracy. In this paper, we propose a fine-grained attention head pruning method to compensate for the performance degradation. In addition, we also introduce the straight through estimator into the L_0 regularization to further accelerate the pruned model. Experiments on the SUPERB benchmark show that our model can achieve comparable performance to the dense model in multiple tasks and outperforms the Wav2vec 2.0 base model on average, with 72% fewer parameters and 2 times faster inference speed.

Index Terms: Model pruning, knowledge distillation, model compression, representation learning

1. Introduction

Recently, self-supervised pre-training has become one of the most attractive topics in the speech domain [1, 2]. With this method, a large amount of unlabeled data can be used to train a deep model to extract high-level representations from raw audio, which can bring significant improvement to many downstream tasks.

While pre-trained models provide a tremendous performance improvement, they also require large amount of memory and computing power. Large self-supervised pre-trained speech models such as Wav2vec2 [3], Hubert [4], and WavLM [5] typically have hundreds of millions of parameters, making them unsuitable for use on consumer products such as laptops and smartphones. This is an obstacle to the application of these models in many real-world scenarios. As a result, model compression has become a major concern for these large self-supervised models.

Knowledge distillation usually uses a teacher model to guide a smaller student model, and the structure of the student model must be carefully designed to achieve better performance. DistilHubert [6] distills a 12-layer Hubert-based model to obtain a 2-layer student model and significantly reduces the model size. FitHubert [7], which is inspired by FitNets [8], designs a thin but deep student network to provide better representation ability.

Model pruning attempts to discard the unimportant weights and obtain a subnetwork from the pre-trained model. In unstructured pruning, these discarded weights are randomly distributed in the matrices; in structured pruning, network units such as attention heads or feed-forward layers are removed entirely. Structurally pruned models do not require specially designed hardware for acceleration, which may be more appropriate for consumer devices. LightHubert treats model pruning as a neural architecture search problem and significantly reduces the performance degradation, but the search process still requires some time-consuming manual selections [9]. Peng et al. propose a more flexible method by applying the L_0-regularization-based pruning method [10] to the Wav2vec 2.0 model, but their method is task-specific and comes at some additional cost when applied to downstream tasks [11].

We attempt to use a similar L_0-regularization-based method to obtain a task-agnostic compressed model. However, learning the pruning masks using L_0 regularization on unsupervised pre-training tasks such as contrastive predictive coding [12] requires large computational resources. The combination of distillation and pruning is a promising solution [13, 14]. The representation provided by the pre-trained model not only reduces the training effort of the downstream models, but also provides task-independent information for model pruning.

Compared to existing unstructured pruning methods of the pre-trained speech models [15, 16], structure pruning usually suffers from a larger performance degradation [17]. The crux of this problem is that using structure rather than individual weights as the basic unit of pruning reduces the degree of freedom, resulting in the removal of some important weights. To compensate for the performance degradation, we introduce a fine-grained attention head pruning method that prunes each attention head separately. To promote the pruning of coarse-grained structures and further speed up the pruned model, we also introduce the straight through estimator (STE) [18] into the mutil-scale structured pruning method [13] based on L_0 regularization.

Experiments on the SUPERB benchmark show the generalization ability of the proposed model on different downstream tasks. With the help of the pre-trained teacher, the proposed model is task-agnostic and can be directly fine-tuned to many downstream tasks. Further contrast experiments demonstrate the effectiveness of fine-grained attention head pruning and STE. Our model outperforms the distilled baselines, and achieves comparable results to the teacher model on multiple tasks, with 72% fewer parameters and 2 times faster in speed.

2. Backgrounds

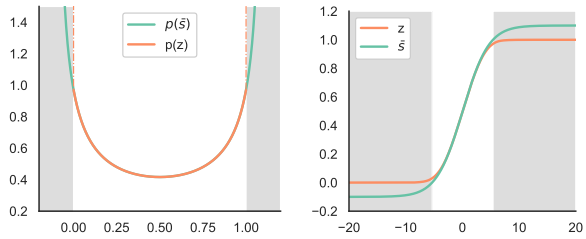
2.1. Pre-trained Speech Representation Models

Our experiment is mainly performed on WavLM [5], but the method can be easily extended to Wav2vec 2.0 [3], data2vec

* Corresponding author

This work was supported by the National Natural Science Foundation of China under Grant No. 62276153.

10.21437/Interspeech.2023-1442



(a) (b)

Figure 1: ***(a)*** *the possibility distribution of z and $\bar{s}$.* ***(b)*** *z and $\bar{s}$ as a function of logα, averaged on 500 samples. z can be exactly 0 or 1 or any value in between. In the shadow region, $\partial z / \partial \bar{s} = 0$.*

[19], Hubert [4], and other models with similar transformer-based structures.

WavLM is a set of state-of-the-art self-supervised pre-trained models. During pre-training, offline clustered units are used as the training target and the models learn to represent the continuous inputs by some discrete hidden units. WavLM also introduces masked speech denoising and gated relative position bias to improve the performance.

2.2. Pruning Based on the L_0 Regularization

Pruning based on L_0 regularization is one of the mask learning methods. In some pruning methods, parameters are discarded according to some artificially set criteria, such as the magnitude of weights or gradients. On the other hand, mask learning methods tend to consider pruning as an optimization problem [10]. As the name implies, L_0-regularization-based pruning adds a mask to the parameters (or parameter groups) and uses the L_0 norm of these pruning masks as a regularization term of the loss function. For example, in our experiments, the training objective is:

$$\mathcal{R}(\theta, \pi) = E_{z \sim q(\pi)}[\frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(f_s(x_i, \widetilde{\theta}), f_t(x_i)) + \lambda ||\widetilde{\theta}||_0], \tag{1}$$

where f_s and f_t are the student and teacher models for knowledge distillation, x_i is the ith input data, θ is the parameter set of the student model, $z \in \{0, 1\}$ is the pruning mask set, $\widetilde{\theta} = \theta \odot z$ is the parameter set after masking. The discrete random variable z follows a Bernoulli distribution $q(\pi)$.

However, this objective function cannot be optimized by gradient descent methods because the process of sampling z for $q(\pi)$ is not differentiable. Louizos et al. introduce a reparameterization trick to deal with this problem [10]. After the reparameterization, z becomes a continuous variable, determined by a learnable parameter α and an additional random variable u that "collects" the randomness from z. Formally speaking, z is computed by:

$$\begin{aligned} u &\sim U(0, 1), s = \text{sigmoid}(\frac{1}{\beta} \text{log}(\frac{u}{1-u}) + \text{log}\alpha) \\ \bar{s} &= s(\zeta - \gamma) + \gamma, z = \text{hardtanh}(\bar{s}), \end{aligned} \tag{2}$$

where u is sampled from a uniform distribution $U(0, 1)$, $\zeta = 1.1, \gamma = -0.1$ are 2 constants to scale s to a larger interval and make sure z can be exactly 0 or 1. β controls the temperature, and α is the learnable parameter.

Figure 1a shows the probability distribution of z and $\bar{s}$, while figure 1b shows their values as functions of logα. We can see that the reparameterization trick turns the discrete masks z into continuous variables while still allowing them to be exactly 0 or 1.

2.3. Multi-scale Structured Pruning

The L_0 regularization does not limit the grain of the pruning. If z masks some structure, L_0 regularization can be used for structured pruning. The grain can be as large as an entire layer or as small as a certain dimension of a weight matrix. Recently, Xia et al. introduce a multi-scale pruning method that removes fine-grained and coarse-grained structures in parallel to promote the removal of large structures and achieve further speedup [13]. We introduced this method to increase the possibility of removing coarse-grained structures to compensate for the potential negative effects of our fine-grained attention head pruning method on the inference speed of the model.

3. Methods

3.1. Fine-grained Attention Head Pruning

In previous works [11, 13], the attention heads are used as the smallest units for pruning. This may reduce the degree of freedom of pruning and lead to more performance degradation. To make structure pruning more flexible, we propose a fine-grained attention method that separately prunes each dimension of matrices in the attention layer based on the multi-scale structured pruning method of Xia et.al [13]. Formally speaking, a transformer block is masked as follows:

$$\begin{aligned} f_{\text{MHA}}(X) &= z_{\text{MHA}} \cdot \text{concat}(f_{\text{ATT}}(X)) \\ f_{\text{ATT}}(X) &= S_c \cdot (XW_V^i) \cdot \text{diag}(z_{vo}^i) \\ S_c &= \text{softmax}((XW_Q^i) \cdot \text{diag}(z_{qk}^i) \cdot (XW_K^i)^T) \\ f_{\text{FFN}}(X) &= z_{\text{FFN}} \cdot \text{gelu}(XW_{\text{U}}) \cdot \text{diag}(z_{\text{int}}) \cdot W_{\text{D}}, \end{aligned} \tag{3}$$

where X is the input data, W_Q^i, W_K^i, W_V^i, W_O is the query, key, value and output matrices, respectively. z_{MHA}, z_{qk}^i , z_{vo}^i, z_{FFN}, z_{int} denote the pruning mask for multi-head attention layers, attention matrices, feed-forward layers, and intermediate dimensions. We omit the scale factors in $f_{\text{ATT}}(X)$ for clarity, and please note that W_O should also be pruned according to z_{vo}^i. For $W_Q, W_V \in \mathbb{R}^{d_{\text{hidden}} \times d_{\text{head}}}$, z_{qk}^i and z_{vo}^i will have d_{head} variables.

3.2. Optimizing Pruning Masks with STE

Although the reparameterization trick makes z differentiable, the introduction of hardtanh in Eq. 2 creates a new obstacle to optimization. As shown in Figure 1b, when logα takes a value in the shaded region, the presence of hardtanh makes $\partial z / \partial s = 0$, and the learnable parameter α cannot be updated. That is to say, the model decides to keep a structure when z is 1, but it cannot evaluate that decision.

This problem becomes more obvious for multi-scale structured pruning. Figure 3a shows that the mean value of z_{FFN} does not change during training, which makes multi-scale pruning ineffective. The reason may be that in the early stages of training, pruning the entire FFN layer can lead to a huge performance degradation, so α may be optimized to a large positive value, and difficult to update in the remaining training steps.

The failure to cut the coarse-scale structures will cause the sparse weight of the pruning model to be too dispersed, resulting in lower acceleration ratio. To address this problem, We apply the straight through estimator [18] to make sure that the

gradient can pass through the hardtanh function in Eq. 2. Since the gradients from STE are not the gradients for the loss function, optimizing in this direction may not lead to the most accurate student and may cause instability near some local minima [20]. For the stability of training, we define the gradient of STE such that:

$$\frac{\partial \mathcal{L}}{\partial \bar{s}} = \begin{cases} 1, \text{if } \frac{\partial \mathcal{L}}{\partial z} >= 1; \\ -1, \text{if } \frac{\partial \mathcal{L}}{\partial z} < -1; \\ \frac{\partial \mathcal{L}}{\partial z}, \text{otherwise.} \end{cases} \tag{4}$$

3.3. Training Objective

Hidden states of different layers contain different types of information [6, 21]. Therefore, we follow Xia et al. [13] to use learnable multi-task knowledge distillation to learn the representation of different layers. We also follow Wang et al. to change the 2nd term on the r.h.s of eq. 1 into a Lagrangian term to better control the sparsity [22]. Our training objective is as follows:

$$\mathcal{L} = \frac{1}{N} \sum_{i=0}^{N} \sum_{(j,k) \in D} \mathcal{L}_{\text{MSE}}(h_i^j, \hat{h}_i^k) + \lambda_1(\hat{p} - p) + \lambda_2(\hat{p} - p)^2, \tag{5}$$

where $\hat{p}$ is the approximate model sparsity, p is the target sparsity. λ_1 and λ_2 are learnable parameters for the Lagrangian regularization. D is the teacher-student layer pairing relation learned during training [13], for sample i, h_i^j and $\hat{h}_i^k$ are the output of layer j/k of the student and teacher models, respectively.

4. Experiments

4.1. SUPERB

SUPERB (Speech processing Universal PERformance Benchmark) is a benchmark for evaluating the performance of speech pre-training models [23]. SUPERB provides 10 predefined speech tasks from different perspectives where the pre-trained models are used as upstream feature extractors. These tasks include phoneme recognition (PR), automatic speech recognition (ASR), keyword spotting (KS), query-by-example spoken term detection (QbE), speaker identification (SID), automatic speaker verification (SV), speaker diarization (SD), intent classification (IC), slot filling (SF), and emotion recognition (ER).

4.2. Pruning setup

Model. Our model is initialized from the WavLM base model, which consists of a 7-layer CNN feature extractor and a 12-layer transformer encoder. For the matrices in Eq. 3, $W_Q^i, W_K^i, W_V^i \in \mathbb{R}^{768\times64}$, $W_O \in \mathbb{R}^{768\times768}$, $W_U \in \mathbb{R}^{768\times3072}$, and $W_D \in \mathbb{R}^{3072\times768}$. For each transformer block, we have 12 attention heads, leading to $12 * 64 = 768$ elements in z_{qk} and z_{vo}. We also have 3072 elements in z_{int} for each dimension in the FFN layer, and 1 element in z_{MHA} and z_{FNN} to mask the entire layer. The target pruning sparsity is set to 80%. The teacher model of knowledge distillation is also the WavLM base model.

Data. We use the 960 hours Librispeech [24] corpus for pruning. For SUPERB tasks, we use the dataset according to the official guidelines[1].

[1]https://github.com/s3prl/s3prl/

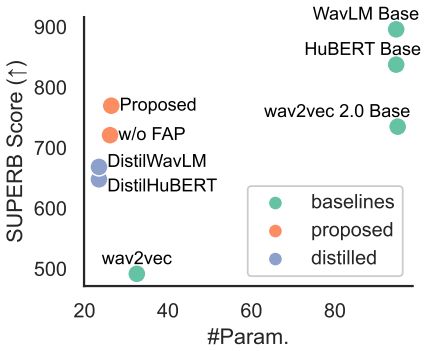


Figure 2: *The relationship between the SUPERB score and the number of parameters.*

Pruning. Pruning is performed on an RTX 3090 GPU for 200k steps and takes about 36 hours. Our training hyperparameters are chosen according to DistilHuBERT [6] and Xia et al. [13]. The learning rate increases linearly to 2.0e-4 in the first 7% steps and decreases linearly to 0 in the remaining steps, and the target sparsity increases linearly to 80% in the first 7% steps and remains constant for the rest.

5. Results

Table 1 shows the evaluation results on the SUPERB downstream tasks. Our model has comparable performance to the teacher model in KS, IC, ER, SV, and SD tasks, demonstrating the effectiveness of our approach. The performance degradation occurred mainly in PR, ASR, and SF tasks. These tasks require more complex content-related information, which is more likely to be lost during pruning. Using the same WavLM base teacher model, our method outperforms the distilled models in most tasks, especially in content-related tasks such as ASR, showing that our model better preserves the performance of the teacher model.

In addition to the task-specific metrics, we also use the SUPERB score (superb_s) to provide an overall evaluation. The SUPERB score is an average of the linear transformations of all the task-specific metrics, and is determined by the SOTA model on the benchmark and a predefined FBANK baseline. At the time of writing, the SOTA model is WavLM-Large[2]. Formally speaking, the SUPERB score is defined as:

$$\text{superb}_s = \frac{1}{T} \sum_{t \in T} \frac{1000}{m_t^{\text{sota}} - m_t^{\text{fbank}}} (m_t^u - m_t^{\text{fbank}}), \tag{6}$$

where m_t^u is the metric of task t and model u, $m_t^{\text{sota}} \equiv 1000$, $m_t^{\text{fbank}} \equiv 0$.

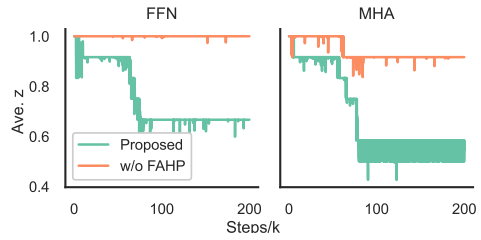
Figure 2 shows the relationship between the SUPERB score and the number of parameters. Our model significantly outperforms the distillation models with similar number of parameters, and even has superior performance to the Wav2vec 2.0 base model. These results show that the proposed method achieves a better balance between performance and the number of parameters compared to the distillation-based method.

We also compare our method with the previous pruning method which directly removes the attention heads (w/o FAHP in Table 1). Again, the improvement is mainly reflected in tasks

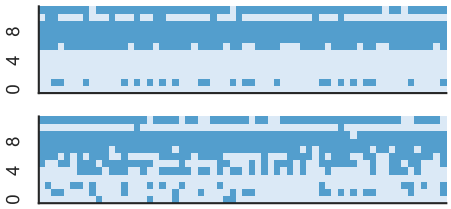
[2]The performance of the WavLM-Large model can be found at https://superbbenchmark.org/leaderboard.

Table 1: *Results on SUPERB of the proposed model, and other baselines. The performances are evaluated by Phoneme Error Rate (PER%), Accuracy (Acc%), Word Error Rate (WER%), Maximum Term Weighted Value(MTWV), F1 Score (F1%), Concept Error Rate (CER%), Equal Error Rate (EER%), and Diarization Error Rate (DER%). DistilWavLM is our reproduction of DistilHubert with the teacher changed to WavLM base; FAHP is the abbreviation for the proposed Fine-grained Attention Head Pruning method.*

Method	**KS**	**IC**	**PR**	**ASR**	**ER**	**QbE**	**SF**	**SID**	**SV**	**SD**	**Superb$_s$**↑
	Acc↑	Acc↑	PER↓	WER↓	Acc↑	MTWV↑	F1↑/CER↓	Acc↑	EER↓	DER↓	
Baselines											
wav2vec [25]	95.59	84.92	31.58	15.86	59.79	4.85	76.37/43.71	56.56	7.99	9.90	491.59
w2v2 Base	96.23	92.35	5.74	6.43	63.43	2.33	88.3/24.77	75.18	6.02	6.08	735.00
HuBERT Base	96.30	98.34	5.41	6.42	64.92	7.36	88.53/25.2	81.42	5.11	5.88	837.63
WavLM Base	96.79	98.63	4.84	6.21	65.94	8.70	89.38/22.86	84.51	4.69	4.55	895.99
Distilled Models											
DistilHuBERT	95.98	94.99	16.27	13.37	63.02	5.11	82.57/35.39	73.54	8.55	6.19	647.88
DistilWavLM	96.40	96.39	14.18	13.24	63.69	7.07	85.27/31.80	71.00	8.87	7.2	668.39
Ours											
Proposed	96.57	98.08	9.09	10.61	63.61	7.40	87.14/27.13	74.56	6.17	6.11	**769.62**
w/o FAHP	96.14	98.05	10.51	11.83	63.78	5.19	85.57/30.91	70.03	6.12	7.18	721.79



(a) *The average value of z_{FFN} and z_{MHA}.*



(b) *Remaining (blue) parameters in W_V^0 for 12 layers.*

Figure 3: *The effectiveness of STE*

such as ASR, suggesting that fine-grained attention head pruning can help compensate for the loss of complex information in structured pruning.

Figure 3a shows the average of the pruning masks z_{FFN} and z_{MHA} during pruning. By introducing STE, the pruning masks of coarse-grained structures change more frequently and eventually drop to lower values, which proves the effectiveness of STE. Figure 3b shows the distribution of the remaining weights of each layer after pruning. Since the coarse-grained structures can be entirely removed, the remaining parameters tend to be concentrated, leading to further acceleration.

In addition, the remaining weight is concentrated at the top of the network. Since content-related information is more prominent in the features of the top layers, this distribution of remaining weights may be one of the reasons for the network's improvement in content-related tasks.

We also measure the inference time of the 2 models above. Table 2 shows the speed effect of STE. It can be seen that the concentrated weight distribution brought by STE significantly improves the inference speed of the model. With STE, the pruned model is 1.4 times faster with a similar number of parameters.

Table 2: *Inference time measured on a RTX3090 GPU, by extracting features of librispeech dev-clean set and are averaged on 5 runs.*

Method	**#Params**	**Infer. time**
	Millions	Seconds
WavLM base	94.70	91.87(1.0x)
Proposed	26.57	**46.08(1.99x)**
w/o STE	26.37	67.78(1.35x)

Table 3: *Influence of STE on accuracy. ASR, IC, ER, SID are representative of SUPERB content, paralinguistic, speaker, and semantic tasks.*

Methods	**ASR**	**IC**	**ER**	**SID**
	WER↓	Acc↑	Acc↑	Acc↑
proposed	**10.29**	**98.08**	63.61	74.56
w/o STE	10.61	97.07	**64.17**	**74.65**

Furthermore, we show the effect of STE on accuracy. Among these 4 tasks, STE brings improvement in ASR and IC, while causing degradation in ER and SID, but both the positive and negative influence are not significant. The degradation in ER and SID may be due to the parameters removed from the lower layers that are related to speaker or emotion information.

6. Conclusion

In this paper, we present a task-agnostic structured pruning method of pre-trained speech representation models. By using fine-grained attention head pruning, we retain the ability to represent content-level information and reduce the performance degradation caused by structured pruning. We introduce STE to multi-scale structured pruning to further accelerate the model. Our experiments prove that the proposed model reduces 72% of the parameters while having comparable performance to the dense model in multiple tasks, and outperforms the Wav2vec2 base model in average performance.

7. References

[1] A. Mohamed, H.-y. Lee, L. Borgholt, J. D. Havtorn, J. Edin, C. Igel, K. Kirchhoff, S.-W. Li, K. Livescu, L. Maaløe, T. N. Sainath, and S. Watanabe, "Self-supervised speech representation learning: A review," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1179–1210, Oct. 2022, conference Name: IEEE Journal of Selected Topics in Signal Processing. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9893562

[2] J. Zhao and W.-Q. Zhang, "Improving Automatic Speech Recognition Performance for Low-Resource Languages With Self-Supervised Models," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1227–1241, Oct. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9801640/

[3] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 449–12 460, 2020. [Online]. Available: https://dl.acm.org/doi/abs/10.5555/3495724.3496768

[4] 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. [Online]. Available: https://dl.acm.org/doi/abs/10.1109/TASLP.2021.3122291

[5] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022. [Online]. Available: https://x-lance.sjtu.edu.cn/en/papers/2022/zyc97-jstsp22.pdf

[6] H.-J. Chang, S.-w. Yang, and H.-y. Lee, "DistilHuBERT: Speech representation learning by layer-wise distillation of hidden-unit bert," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7087–7091. [Online]. Available: https://ieeexplore.ieee.org/document/9747490/

[7] Y. Lee, K. JANG, J. Goo, Y. Jung, and H.-R. Kim, "FitHuBERT: Going thinner and deeper for knowledge distillation of speech self-supervised learning," in *23rd Annual Conference of the International Speech Communication Association, INTERSPEECH 2022*. ISCA, 2022, pp. 3588–3592. [Online]. Available: https://www.isca-speech.org/archive//pdfs/interspeech_2022/lee22p_interspeech.pdf

[8] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," *arXiv preprint arXiv:1412.6550*, 2014. [Online]. Available: https://arxiv.org/abs/1412.6550

[9] R. Wang, Q. Bai, J. Ao, L. Zhou, Z. Xiong, Z. Wei, Y. Zhang, T. Ko, and H. Li, "LightHuBERT: Lightweight and Configurable Speech Representation Learning with Once-for-All Hidden-Unit BERT," in *Interspeech 2022*. ISCA, Sep. 2022, pp. 1686–1690. [Online]. Available: https://www.isca-speech.org/archive/interspeech_2022/wang22t_interspeech.html

[10] C. Louizos, M. Welling, and D. Kingma, "Learning sparse neural networks through l0 regularization." in *Sith International Conference on Learning Representations, 2018*, 2018. [Online]. Available: https://openreview.net/pdf?id=H1Y8hhg0b

[11] Y. Peng, K. Kim, F. Wu, P. Sridhar, and S. Watanabe, "Structured pruning of self-supervised pre-trained models for speech recognition and understanding," in *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Jun. 2023, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10095780

[12] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018. [Online]. Available: https://arxiv.org/abs/1807.03748

[13] M. Xia, Z. Zhong, and D. Chen, "Structured pruning learns compact and accurate models," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 1513–1528. [Online]. Available: https://aclanthology.org/2022.acl-long.107

[14] V. Sanh, T. Wolf, and A. Rush, "Movement pruning: Adaptive sparsity by fine-tuning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 20 378–20 389, 2020. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/eae15aabaa768ae4a5993a8a4f4fa6e4-Paper.pdf

[15] M. Yang, A. Tjandra, C. Liu, D. Zhang, D. Le, and O. Kalinli, "Learning ASR pathways: A sparse multilingual ASR model," in *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Jun. 2023, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10094300

[16] C.-I. J. Lai, Y. Zhang, A. H. Liu, S. Chang, Y.-L. Liao, Y.-S. Chuang, K. Qian, S. Khurana, D. Cox, and J. Glass, "PARP: Prune, adjust and re-prune for self-supervised speech recognition," Oct. 2021, arXiv:2106.05933 [cs, eess]. [Online]. Available: http://arxiv.org/abs/2106.05933

[17] Z. Liu, M. Sun, T. Zhou, G. Huang, and T. Darrell, "Rethinking the value of network pruning," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/pdf?id=rJlnB3C5Ym

[18] Y. Bengio, N. Léonard, and A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation," *arXiv preprint arXiv:1308.3432*, 2013. [Online]. Available: https://arxiv.org/abs/1308.3432

[19] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "Data2vec: A general framework for self-supervised learning in speech, vision and language," in *International Conference on Machine Learning*. PMLR, 2022, pp. 1298–1312. [Online]. Available: https://proceedings.mlr.press/v162/baevski22a/baevski22a.pdf

[20] P. Yin, J. Lyu, S. Zhang, S. J. Osher, Y. Qi, and J. Xin, "Understanding straight-through estimator in training activation quantized neural nets," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=Skh4jRcKQ

[21] L. Chen, M. Asgari, and H. H. Dodge, "Optimize Wav2vec2s architecture for small training set through analyzing its pre-trained models attention pattern," in *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2022, pp. 7112–7116. [Online]. Available: https://ieeexplore.ieee.org/document/9747831

[22] Z. Wang, J. Wohlwend, and T. Lei, "Structured pruning of large language models," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 6151–6162. [Online]. Available: https://aclanthology.org/2020.emnlp-main.496.pdf

[23] S. wen Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin, T.-H. Huang, W.-C. Tseng, K. tik Lee, D.-R. Liu, Z. Huang, S. Dong, S.-W. Li, S. Watanabe, A. Mohamed, and H. yi Lee, "SUPERB: Speech Processing Universal PERformance Benchmark," in *Proc. Interspeech 2021*, 2021, pp. 1194–1198. [Online]. Available: https://www.isca-speech.org/archive/interspeech_2021/yang21c_interspeech.html

[24] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210. [Online]. Available: https://www.danielpovey.com/files/2015_icassp_librispeech.pdf

[25] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "Wav2vec: Unsupervised Pre-Training for Speech Recognition," in *Proc. Interspeech 2019*, 2019, pp. 3465–3469. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-1873