

# Efficient Pruning for Large-Scale Seq2Seq Speech Models without Back-Propagation

Tianteng Gu<sup>1,2</sup>, Bei Liu<sup>1</sup>, Yanmin Qian<sup>1\*</sup>

<sup>1</sup>*Auditory Cognition and Computational Acoustics Lab, MoE Key Lab of Artificial Intelligence, AI Institute  
Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China*

<sup>2</sup>*SJTU Paris Elite Institute of Technology  
{995999277, beiliu, yanminqian}@sjtu.edu.cn*

**Abstract**—Large-scale Seq2Seq speech models like Whisper excel in speech recognition but are limited by their high computational demands, making them difficult to be deployed on resource-constrained devices. This paper introduces a novel and efficient pruning method for compressing these models without retraining and back-propagation, focusing on models with encoder-decoder architectures. We adapt layer-wise pruning to large speech models and introduce a mixed sparsity allocation strategy that uses only forward propagation. This approach effectively reduces model size while maintaining high performance. Evaluated on the Whisper-large-v3 across various datasets, our method can almost maintain Whisper’s performance and robustness with about 60% reduction in parameters. It could also be combined with other model compression methods such as distillation to further reducing model size.

**Index Terms**—Model Compression, Pruning, Mixed Sparsity, Large Speech Model, Encoder-Decoder Architecture

## I. INTRODUCTION

Large-scale sequence-to-sequence speech recognition models, such as Whisper [1] and SenseVoice [2], have demonstrated exceptional versatility and robustness, effectively handling complex inputs across various accents, dialects, and noisy environments. Additionally, several recent works [3]–[5] have successfully integrated large language models into speech recognition systems, yielding impressive results. A common feature of these models is their massive decoders, which possess a large number of parameters and are responsible for transcribing the speech information extracted by the encoder into text. These decoders excel in accurately interpreting context, handling diverse linguistic structures, and generating coherent transcriptions, even in challenging acoustic conditions. However, despite their impressive performance, the substantial memory and computational demands of these models limit their deployment on resource-constrained devices.

Pruning is an effective method for model compression, as it reduces model size and inference computation by eliminating redundant parameters. However, current methods lack applicability to large-scale sequence-to-sequence speech recognition models with encoder-decoder architectures. Previous works [6]–[14] have primarily focused on medium-sized self-supervised speech representation learning models [15]–[17], typically around 100 million parameters, which usually consist of only an encoder. These methods are challenging to directly

apply to large sequence-to-sequence speech recognition models with over 1 billion parameters, as they either require re-training on the original dataset or consume significant memory to compute Hessian matrices.

Motivated by this, we focus on developing a method for pruning large-scale sequence-to-sequence speech models with an encoder-decoder architecture in an efficient way that does not require back-propagation re-training. The key contributions of this paper are as follows:

- 1) We extend and improve layer-level pruning to large-scale encoder-decoder architecture speech models, offering an efficient pruning framework that requires minimal calibration data while maintaining model robustness and generality.
- 2) We introduce a mixed sparsity allocation strategy that relies solely on forward propagation, eliminating the need for retraining through back-propagation. This strategy determines an optimal allocation strategy based on the impact of pruning on each weight and flexibly retains important parameters, minimizing the impact of pruning on performance.

We evaluated our proposed method on Whisper-large-v3 and various publicly available speech recognition datasets: LibriSpeech [18], Common Voice [19], Tedlium [20], VoxPopuli [21] and FLEURS [22]. The experimental results show that our method can almost maintain Whisper’s performance and robustness while reducing about 60% of the parameters. This process only requires a small calibration dataset and runs for six hours on one single GPU, without the need for retraining on a massive dataset.

## II. BACKGROUND

### A. Whisper

Whisper-large is a large-scale pre-trained speech model with 1.5 billion of parameters known for its robust performance in speech recognition. It is trained on an extensive dataset comprising hundreds of thousands of hours of diverse audio, including various accents, dialects, and noisy environments, which enhances its generalization capabilities. The model employs an encoder-decoder architecture, where the encoder processes the input audio and the decoder generates the corresponding transcription. This structure, combined with extensive pre-training, endows Whisper with remarkable versatility and robustness, enabling accurate performance across a wide range of real-world scenarios.

\*Corresponding Author

## B. Layer-wise Pruning

Layer-wise pruning [23] is a model compression technique suitable for large-scale models where each layer of the neural network is pruned sequentially, utilizing a calibration dataset to guide the process ensuring minimal accuracy loss. Methods like SparseGPT [24] and Wanda [25] exemplify this approach. In this paper, we use SparseGPT as the foundational pruning framework where the task of pruning an entire model is broken down into a series of steps focused on minimizing the reconstruction error of individual weight matrices:

$$\operatorname{argmin}_{\hat{\mathbf{w}}} \sum_i^N \|\mathbf{w}\mathbf{X}_i - \hat{\mathbf{w}}\mathbf{X}_i\|_2^2 \text{ s.t. } \text{sparsity}(\hat{\mathbf{w}}) = s. \quad (1)$$

Here, the weight matrix is parameterized by  $\mathbf{w}$  and  $\hat{\mathbf{w}}$  denotes the pruned weight matrix,  $\mathbf{X}$  is the calibration dataset containing  $N$  samples and  $s$  is the target sparsity.

The calibration dataset is passed through each layer in the same order as data is propagated during inference. The process can be described by the following formula:

$$\mathbf{X}^{(l)} = f_{\text{pruned}}^{(l-1)}(\mathbf{X}^{(l-1)}) \quad (2)$$

where  $\mathbf{X}^{(l)}$  represents the calibration dataset for layer  $l$ , and  $f_{\text{pruned}}^{(l-1)}$  is the function representing the pruned layer  $l-1$ , applied to the calibration datasets of the previous layer  $\mathbf{X}^{(l-1)}$ .

## III. METHODS

### A. Calibration-aware Layer-wise Pruning

In this section, we extend layer-wise pruning for large-scale speech model with encoder-decoder architecture. We propose a novel calibration-aware strategy for decoder pruning by generating the more accurate calibration datasets. Taking Whisper as an illustration, we use text transcribed by Whisper itself and the hidden states output by the dense encoder as calibration datasets for pruning the decoder. These two improvements are illustrated in Figure 1.

Directly applying layer-wise pruning can lead to significant performance degradation. We attribute this to the heterogeneity of the encoder-decoder architecture and the transcription format used in large speech models. However, this issue can be mitigated by modifying the calibration datasets used for pruning the decoder. Contrary to the typical data flow in the model, we prune the decoder first, followed by the encoder.

Pruning the decoder is slightly more complex because it includes a cross-attention layer that processes two modalities: text and audio. Therefore, we need to prepare separate calibration datasets for each modality. For the text modality calibration dataset used for pruning self-attention, cross-attention and feed-forward layers, using open-source datasets' text labels can confuse the model due to differences in transcription formats, resulting in degraded outputs. Whisper's encoding format preserves rich information such as punctuation and case sensitivity, whereas most open-source datasets normalize text, retaining only uppercase letters and spaces, as Figure 1 shows. Therefore, we use texts transcribed by Whisper itself as the text calibration set for the decoder. This strategy maximizes the

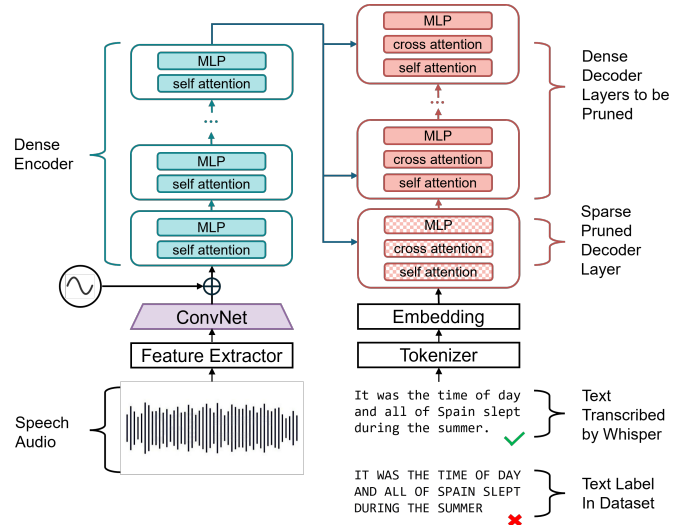


Fig. 1. The process of pruning the decoder and generating the calibration dataset for Whisper.

retention of Whisper's capabilities while ensuring high-quality and diverse transcriptions. For the audio modality calibration dataset for pruning cross-attention, using the output of pruned layers as the next layer's calibration dataset can introduce cumulative errors, especially at high sparsity levels. Errors introduced by pruning accumulate across layers and significantly impact final performance, particularly given Whisper-large's 32-layer encoder. Additionally, due to the nature of audio data, even a few seconds of audio can generate a large number of frames, further increasing errors in the pruned encoder's output. Therefore, for the audio modality calibration dataset, we use the output of the dense encoder to avoid misleading the assessment of parameter importance and degrading model performance. This rationale underlies our decision to prune the decoder before the encoder.

After pruning the decoder, we apply standard layer-wise pruning to the encoder of the pre-trained model, resulting in a sparse version of Whisper.

### B. Mixed Sparsity Allocation with Forward Propagation Only

Like other transformer based models [26], in large speech models, important parameters are not uniformly distributed across weight matrices; they tend to concentrate in specific weights. Inspired by this observation, we propose a mixed sparsity allocation method which assigns lower sparsity levels to weight matrices containing more important parameters, thereby maximizing the retention of crucial parameters in the model. This method relies solely on the model's forward propagation, eliminating the need for gradients and Hessian matrices required by previous methods, and enables efficient identification of an optimal sparsity allocation from a vast search space in a short time.

We use the reconstruction error between the pruned weight matrices and the original weight matrices, which is positively correlated with the sparsity, as an indicator of the deviation caused by pruning on the output of the weight matrix and

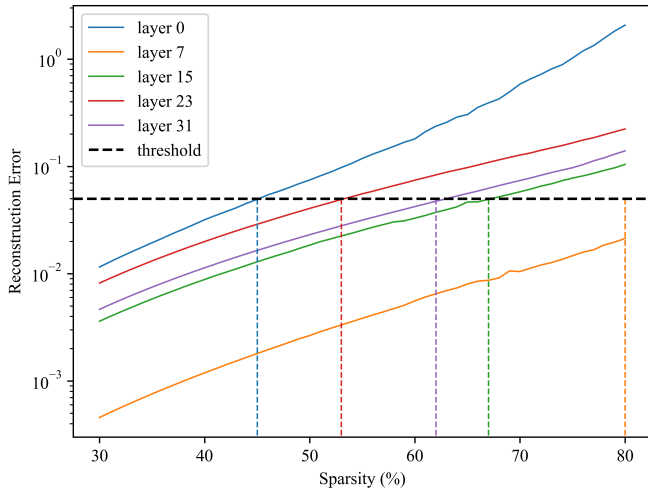


Fig. 2. The reconstruction error with respect to different sparsity of the fully connected layer 2 in certain layers of Whisper-large-v3’s decoder. The maximal sparsity that ensures the reconstruction error is below the threshold is set as the target sparsity.

---

**Algorithm 1** Proposed mixed sparsity allocation method.

---

**Inputs:** candidate sparsity set  $\mathbf{S}$ , calibration dataset  $\mathbf{X}$ , Whisper model  $f(\mathbf{X}; \mathbf{W})$ , dictionary of thresholds for reconstruction error  $\mathbf{T}$

**for** each candidate sparsity  $s$  in  $\mathbf{S}$  **do**  
  **for** each weight matrix  $\mathbf{w}$  in  $\mathbf{W}$  **do**  
    Prune  $\mathbf{w}$  to  $s$  and record the reconstruction error $[\mathbf{w}][s]$   
  **end for**  
**end for**  
**for** each weight matrix  $\mathbf{w}$  in  $\mathbf{W}$  **do**  
   $\mathbf{C}[\mathbf{w}] = \max \{s \in \mathbf{S} \mid \text{reconstruction error}[\mathbf{w}][s] < \mathbf{T}[\mathbf{w}]\}$   
**end for**  
**Return:** Mixed sparsity configuration  $\mathbf{C}$

---

can be obtained during forward propagation. By controlling the sparsity, we aim to keep the reconstruction error of each weight matrix below a reasonable threshold, reducing the difference between the outputs of the pruned and original weight matrices.

As illustrated in Figure 2, we first record the reconstruction error of each weight matrix under all candidate sparsity conditions. Then, we set a threshold for the reconstruction error of each weight matrix and select the maximal sparsity in the set of candidate sparsity  $\mathbf{S}$  that satisfies. To reduce the complexity of selecting the threshold, we categorize the model’s weight matrices into groups, with each group sharing a common threshold. In this work, the thresholds are manually set as hyper-parameters to ensure the model’s overall sparsity meets the required constraints. The overall calculation process is presented in Algorithm 1.

#### IV. EXPERIMENT

##### A. Experimental Setup

**Models.** In this work, we use Whisper-large-v3 to evaluate our proposed methods. Only its encoder and decoder are targeted for pruning, as they constitute the majority of the model’s

size. Additionally, we prune Distil-Whisper-large-v3’s encoder to demonstrate that our method can be effectively combined with knowledge distillation.

**Data.** We evaluated the pruned models across various test sets to demonstrate that our method preserves Whisper’s robustness. For English speech recognition, we selected the LibriSpeech test clean, LibriSpeech test other, TED-LIUM 3, Common Voice 5.1, VoxPopuli.en, and Fleurs.en datasets. To assess multilingual speech recognition, we used French, German, Spanish, and Italian test sets from the Multilingual LibriSpeech [27] dataset.

**Pruning.** The calibration dataset consists of 2,048 randomly sampled entries from the LibriSpeech train.500 dataset. We divide Whisper’s weight matrices into eight groups: (self\_attn.k/q/v\_proj), (self\_attn.out\_proj), (fc1), (fc2) of all encoder layers and (self/encoder\_attn.k/q/v\_proj), (self/encoder\_attn.out\_proj), (fc1), (fc2) of all decoder layers. The thresholds of reconstruction error for 60% total sparsity is (0.5, 0.05, 3, 0.05, 0.5, 0.05, 3, 0.05). The candidate sparsity levels form an arithmetic sequence ranging from 0.3 to 0.8 with a common difference of 0.01. It takes 5.5 hours on a single RTX 4090 GPU to obtain the reconstruction error for each sparsity level and another 0.5 hours for pruning.

**Baseline.** The standard magnitude pruning method is chosen as the baseline because it selects parameters for pruning solely based on their absolute values, without the need for a calibration dataset or retraining.

##### B. Main Result

We evaluate the automatic speech recognition performance of the pruned model on the test sets of LibriSpeech, Common Voice, Tedlium, VoxPopuli, and FLEURS to demonstrate that our method maintains the model’s performance and robustness across various domains.

As shown in Table I, our method delivers much more accurate models than magnitude pruning and SparseGPT on every dataset. It can reduce 50% of parameters in Whisper-large-v3 almost with neglected degradation, and the parameter reduction can further increase to 60% with less than relative 10% accuracy degradation compared to the dense model. In contrast, WER will degrade more than relative 50% for the SparseGPT with 60% sparsity.

Compared to Distil-Whisper: distil-large-v3 with 756M parameters, the sparse Whisper-large-v3 pruned by our method with 60% sparsity and only 657M parameters exhibits better performance across all datasets. Moreover, our method does not require extensive computational resources for retraining in Distil-Whisper, making it more efficient.

##### C. Combined with distillation

We also pruned the encoder of Distil-Whisper: distil-large-v3, as it shares the same encoder architecture with Whisper-large-v3, and the results are illustrated in the bottom part of Table I. The performance of the pruned distilled model closely matches that of the original, demonstrating that our

TABLE I  
ENGLISH AUTOMATIC SPEECH RECOGNITION PERFORMANCE ON VARIOUS DATASETS MEASURED BY WORD ERROR RATE (%)

| Method                                 | Params | Sparsity | Librispeech Clean | Librispeech Other | Tedlium | Common Voice | VoxPopuli | Fleurs | Avg    |
|----------------------------------------|--------|----------|-------------------|-------------------|---------|--------------|-----------|--------|--------|
| <b>Whisper-large-v3</b>                |        |          |                   |                   |         |              |           |        |        |
| Dense                                  | 1550 M | 0%       | 2.02              | 3.91              | 3.89    | 8.56         | 7.32      | 4.03   | 4.96   |
| Magnitude                              | 807 M  | 50%      | 6.22              | 20.02             | 9.18    | 44.41        | 26.62     | 16.11  | 20.43  |
| Ours                                   | 807 M  | 50%      | 2.01              | 4.07              | 3.42    | 9.27         | 7.85      | 4.22   | 5.14   |
| SparseGPT                              | 657 M  | 60%      | 3.47              | 6.85              | 4.47    | 12.23        | 9.67      | 7.55   | 7.37   |
| Ours                                   | 657 M  | 60%      | 2.12              | 4.24              | 3.51    | 9.81         | 7.94      | 4.77   | 5.39   |
| SparseGPT                              | 507 M  | 70%      | 6.88              | 11.12             | 10.66   | 25.49        | 16.19     | 20.11  | 15.08  |
| Ours                                   | 507 M  | 70%      | 2.73              | 6.57              | 4.23    | 20.25        | 9.06      | 7.82   | 8.44   |
| <b>Distil-Whisper: distil-large-v3</b> |        |          |                   |                   |         |              |           |        |        |
| Dense                                  | 756 M  | 0%       | 2.51              | 5.17              | 3.86    | 10.85        | 8.23      | 5.81   | 6.07   |
| Magnitude                              | 394 M  | 50%      | 204.52            | 214.98            | 171.04  | 367.06       | 193.62    | 290.37 | 240.27 |
| Ours                                   | 394 M  | 50%      | 2.54              | 5.47              | 3.92    | 11.87        | 8.41      | 6.22   | 6.41   |

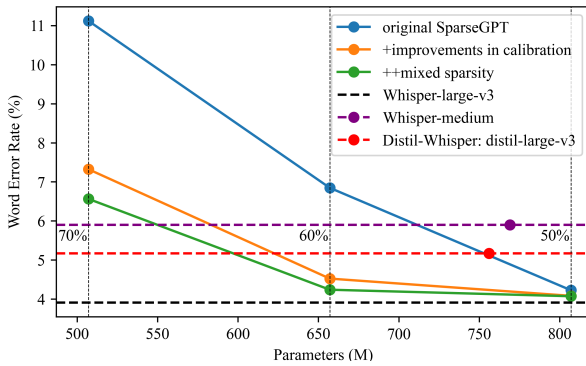


Fig. 3. Ablation study across different sparsity levels

method can be effectively combined with knowledge distillation to achieve a higher compression ratio. Notably, the pruned DistilWhisper with 394 million parameters at 50% sparsity significantly outperforms the pruned Whisper-large-v3 with 507 million parameters at 70% sparsity, despite having fewer parameters. This highlights that the combination of pruning and distillation can achieve a higher lossless compression ratio while maintaining better performance.

#### D. Comparison across Different Sparsity Levels

In Figure 3, we present an ablation study to assess the effectiveness of our proposed methods across different sparsity levels: 50%, 60%, and 70%. The performance is quantified by the word error rate (WER) on the Librispeech test-other dataset. Our results indicate that the methods we propose are effective across all sparsity levels, with more pronounced improvements at higher sparsity. Specifically, incorporating Whisper’s own transcriptions as the calibration dataset significantly leveraging the full encoder output for calibration enhance the pruned model’s performance across all levels of sparsity. Furthermore, our innovative mixed sparsity allocation method provides an additional performance boost, underscoring the robustness and versatility of our approach. These improvements make the pruned Whisper-Large-v3 model with 60% sparsity perform nearly on par with the original model, while significantly surpassing Whisper-Medium and DistilWhisper in performance

with fewer parameters, offering more options for practical deployment.

#### E. Multilingual Speech Recognition Performance

TABLE II  
MULTILINGUAL AUTOMATIC SPEECH RECOGNITION PERFORMANCE ON MULTILINGUAL LIBRISPEECH MEASURED BY WORD ERROR RATE (%)

| Method    | Params | Sparsity | fr    | de    | es    | it    |
|-----------|--------|----------|-------|-------|-------|-------|
| Dense     | 1550 M | 0%       | 4.52  | 4.29  | 2.81  | 8.48  |
| Magnitude | 807 M  | 50%      | 39.52 | 44.66 | 19.37 | 34.45 |
| SparseGPT | 807 M  | 50%      | 5.71  | 5.63  | 3.44  | 10.25 |
| Ours      | 807 M  | 50%      | 5.48  | 5.41  | 3.09  | 9.53  |

Although we used only English data as the calibration dataset for pruning, the pruned model still performed remarkably well when recognizing the multilingual datasets, as shown in Table II. Compared to the original dense Whisper-large-v3, the pruned model using the proposed method with 50% sparsity can keep the WER degradation around relative 10% cross the multilingual corpus without retraining, which is much smaller compared to the baseline Magnitude and SparseGPT approaches. This demonstrates that our new method is more effective on preserving the model’s multilingual capabilities.

#### V. CONCLUSION

In this work, we proposed an efficient pruning method for large-scale Seq2Seq speech models like Whisper, significantly reducing model size while maintaining performance. Our approach requires minimal calibration data and no retraining, while preserving the model robustness across multiple languages and domains. This method enhances the feasibility of deploying advanced speech models on resource-constrained devices and could be combined with knowledge distillation.

#### VI. ACKNOWLEDGEMENT

This work was supported in part by China NSFC projects under Grants 62122050 and 62071288, in part by Shanghai Municipal Science and Technology Commission Project under Grant 2021SHZDZX0102.

## REFERENCES

- [1] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," 2022. [Online]. Available: <https://arxiv.org/abs/2212.04356>
- [2] K. An, Q. Chen, C. Deng, Z. Du, C. Gao, Z. Gao, Y. Gu, T. He, H. Hu, K. Hu, S. Ji, Y. Li, Z. Li, H. Lu, H. Luo, X. Lv, B. Ma, Z. Ma, C. Ni, C. Song, J. Shi, X. Shi, H. Wang, W. Wang, Y. Wang, Z. Xiao, Z. Yan, Y. Yang, B. Zhang, Q. Zhang, S. Zhang, N. Zhao, and S. Zheng, "Funaudiollm: Voice understanding and generation foundation models for natural interaction between humans and llms," *CoRR*, vol. abs/2407.04051, 2024. [Online]. Available: <https://doi.org/10.48550/arXiv.2407.04051>
- [3] Y. Bai, J. Chen, J. Chen, W. Chen, Z. Chen, C. Ding, L. Dong, Q. Dong, Y. Du, K. Gao, L. Gao, Y. Guo, M. Han, T. Han, W. Hu, X. Hu, Y. Hu, D. Hua, L. Huang, M. Huang, Y. Huang, J. Jin, F. Kong, Z. Lan, T. Li, X. Li, Z. Li, Z. Lin, R. Liu, S. Liu, L. Lu, Y. Lu, J. Ma, S. Ma, Y. Pei, C. Shen, T. Tan, X. Tian, M. Tu, B. Wang, H. Wang, Y. Wang, Y. Wang, H. Xia, R. Xia, S. Xie, H. Xu, M. Yang, B. Zhang, J. Zhang, W. Zhang, Y. Zhang, Y. Zhang, Y. Zheng, and M. Zou, "Seed-asr: Understanding diverse speech and contexts with llm-based speech recognition," *CoRR*, vol. abs/2407.04675, 2024.
- [4] S. Ling, Y. Hu, S. Qian, G. Ye, Y. Qian, Y. Gong, E. Lin, and M. Zeng, "Adapting large language model with speech for fully formatted end-to-end speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2024, Seoul, Republic of Korea, April 14-19, 2024*. IEEE, 2024, pp. 11046–11050.
- [5] E. Lakomkin, C. Wu, Y. Fathullah, K. Kalinli, M. L. Seltzer, and C. Fuegen, "End-to-end speech recognition contextualization with large language models," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2024, Seoul, Republic of Korea, April 14-19, 2024*. IEEE, 2024, pp. 12406–12410.
- [6] 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," in *NeurIPS*, 2021.
- [7] 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 *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.
- [8] Y. Peng, Y. Sudo, S. Muhammad, and S. Watanabe, "DPHuBERT: Joint Distillation and Pruning of Self-Supervised Speech Models," in *Proc. INTERSPEECH 2023*, 2023, pp. 62–66.
- [9] H. Wang, S. Wang, W.-Q. Zhang, S. Hongbin, and Y. Wan, "Task-Agnostic Structured Pruning of Speech Representation Models," in *Proc. INTERSPEECH 2023*, 2023, pp. 231–235.
- [10] H. Jiang, L. L. Zhang, Y. Li, Y. Wu, S. Cao, T. Cao, Y. Yang, J. Li, M. Yang, and L. Qiu, "Accurate and Structured Pruning for Efficient Automatic Speech Recognition," in *Proc. INTERSPEECH 2023*, 2023, pp. 4104–4108.
- [11] 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)*, 2022, pp. 7087–7091.
- [12] Y. Lee, K. Jang, J. Goo, Y. Jung, and H. R. Kim, "FitHuBERT: Going Thinner and Deeper for Knowledge Distillation of Speech Self-Supervised Models," in *Proc. Interspeech 2022*, 2022, pp. 3588–3592.
- [13] H. Shao, W. Wang, B. Liu, X. Gong, H. Wang, and Y. Qian, "Whisper-kdq: A lightweight whisper via guided knowledge distillation and quantization for efficient ASR," *arXiv preprint arXiv:2305.10788*, 2023.
- [14] T. Gu, B. Liu, H. Shao, and Y. Qian, "Sparsewav: Fast and accurate one-shot unstructured pruning for large speech foundation models," in *Interspeech 2024*, 2024, pp. 4498–4502.
- [15] 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*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 12449–12460.
- [16] 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.
- [17] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, J. Wu, L. Zhou, S. Ren, Y. Qian, Y. Qian, J. Wu, M. Zeng, X. Yu, and F. Wei, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.
- [18] 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)*, 2015, pp. 5206–5210.
- [19] R. Ardlila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, Eds. Marseille, France: European Language Resources Association, May 2020, pp. 4218–4222. [Online]. Available: <https://aclanthology.org/2020.lrec-1.520>
- [20] F. Hernandez, V. Nguyen, S. Ghannay, N. A. Tomashenko, and Y. Estève, "TED-LIUM 3: Twice as much data and corpus repartition for experiments on speaker adaptation," in *Speech and Computer - 20th International Conference, SPECOM 2018, Leipzig, Germany, September 18-22, 2018, Proceedings*, ser. Lecture Notes in Computer Science, A. Karpov, O. Jokisch, and R. Potapova, Eds., vol. 11096. Springer, 2018, pp. 198–208. [Online]. Available: [https://doi.org/10.1007/978-3-319-99579-3\\_21](https://doi.org/10.1007/978-3-319-99579-3_21)
- [21] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, "VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Online: Association for Computational Linguistics, Aug. 2021, pp. 993–1003. [Online]. Available: <https://aclanthology.org/2021.acl-long.80>
- [22] A. Conneau, M. Ma, S. Khanuja, Y. Zhang, V. Axelrod, S. Dalmia, J. Riesa, C. Rivera, and A. Bapna, "FLEURS: few-shot learning evaluation of universal representations of speech," in *IEEE Spoken Language Technology Workshop, SLT 2022, Doha, Qatar, January 9-12, 2023*. IEEE, 2022, pp. 798–805. [Online]. Available: <https://doi.org/10.1109/SLT54892.2023.10023141>
- [23] X. Dong, S. Chen, and S. J. Pan, "Learning to prune deep neural networks via layer-wise optimal brain surgeon," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 4857–4867.
- [24] E. Frantar and D. Alistarh, "SparseGPT: Massive language models can be accurately pruned in one-shot," *arXiv preprint arXiv:2301.00774*, 2023.
- [25] M. Sun, Z. Liu, A. Bair, and J. Z. Kolter, "A simple and effective pruning approach for large language models," *CoRR*, vol. abs/2306.11695, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2306.11695>
- [26] H. Shao, B. Liu, and Y. Qian, "One-shot sensitivity-aware mixed sparsity pruning for large language models," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 11296–11300.
- [27] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, "MLS: A large-scale multilingual dataset for speech research," in *21st Annual Conference of the International Speech Communication Association, Interspeech 2020, Virtual Event, Shanghai, China, October 25-29, 2020*, H. Meng, B. Xu, and T. F. Zheng, Eds. ISCA, 2020, pp. 2757–2761.