DistilALHuBERT: A Distilled Parameter Sharing Audio Representation Model

Haoyu Wang
Department of Electronic Engineering, Tsinghua University, China
w-hy21@mails.tsinghua.edu.cn

Siyuan Wang
Department of Electronic Engineering, Tsinghua University, China
wsy21@mails.tsinghua.edu.cn

Yaguang Gong
TAL Education, China
gongyaguang@tal.com

Wei-Qiang Zhang*
Department of Electronic Engineering, Tsinghua University, China
wqzhang@tsinghua.edu.cn

ABSTRACT

Self-supervised pre-trained audio representation models such as Wav2vec or HuBERT have brought notable improvements to many downstream audio-related tasks, but the huge number of parameters of these pre-trained models sets a barrier to their application on memory-constrained edge devices. Recursive Transformers, represented by Albert, have proven that parameter sharing through transformer layers can obviously reduce the size of pre-trained models while maintaining most of the performance. In this paper, we propose DistilALHuBERT, a lightweight recursive transformer audio representation model distilled from Hubert. Evaluation results on the S3PRL benchmark show that DistilALHuBERT can significantly outperform the DistilHuBERT model with the same number of parameters. Our code and models are available at https://github.com/backspacetg/distilAlhubert.

CCS CONCEPTS

• **Computing methodologies**; • **Speech recognition; Unsupervised learning; Transfer learning**;

KEYWORDS

Knowledge distillation, model compression, representation learning

ACM Reference Format:
Haoyu Wang, Siyuan Wang, Yaguang Gong, and Wei-Qiang Zhang. 2023. DistilALHuBERT: A Distilled Parameter Sharing Audio Representation Model. In *2023 6th International Conference on Signal Processing and Machine Learning (SPML) (SPML 2023), July 14–16, 2023, Tianjin, China.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3614008.3614015

1 INTRODUCTION

Self-supervised pre-trained audio representation method has become one of the most attractive superstars in the speech domain. Massive amounts of unlabeled data and the huge capacities of the deep transformer models give these pre-trained models the ability to transform the raw audio into neural-network-friendly representations. From speech recognition to speaker verification, pre-trained models are continuously breaking records in many downstream tasks [9, 24, 26, 27].

However, pre-trained models, represented by Wav2vec 2.0 [2], HuBERT [11], and WavLM [4], typically have hundreds of millions of parameters, making them difficult to be applied on memory-constrained devices such as laptops and smartphones. Many lightweight speech tasks, such as speaker verification and keyword spotting, are usually performed on edge devices, which means that they cannot easily benefit from these large pre-trained models. As a result, some model compression methods have been applied to the pre-trained speech representation models to reduce the number of parameters.

Knowledge distillation is an efficient method to transfer the representation ability from a large pre-trained model to a compact student. DistilHuBERT uses a 2-layer student model to learn the representations of a 12-layer HuBERT-base model and appreciably reduces the model size [3]. Since representations from lower or deeper layers of the pre-trained model may be suitable for different tasks, DistilHuBERT designs a multi-task distillation method where the hidden states of the student model are projected by a group of linear layers and mimic the hidden states of different layers. Inspired by FitNets [20], FitHuBERT finds that thin and tall models are better choices than the wide and short ones with the same number of parameters [14]. Their student model has superior representation ability and achieves improved performance with fewer parameters than the DistilHuBERT model. In knowledge distillation, student models are usually less complex and more prone to underfitting in complex tasks [10, 23]. On the other hand, parameter sharing based on the recursive transformer provides a common solution to take full advantage of a small-sized model. By repeatedly using the output of a transformer layer as its input, recursive transformers can achieve similar performance to a non-recursive model with much fewer parameters. The recursive transformer structure is first proposed in the Universal Transformer [7], but is mainly used to solve the sequence modelling tasks such as algorithmic or language understanding. The Universal Transformer can also be used in speech tasks. The Universal Speech Transformer [28] inherits the recursive transformer structure, but removes the depth/position

*Corresponding author

SPML 2023, July 14–16, 2023, Tianjin, China

ACM ISBN 979-8-4007-0757-5/23/07.
https://doi.org/10.1145/3614008.3614015

embeddings and the partial updating of hidden states in the original Universal Transformer structure as they are not beneficial for automatic speech recognition.

ALBERT is a representative example of using the recursive transformer as a parameter-reduction method for unsupervised pre-trained models [13]. ALBERT notably reduces the number of parameters through cross-layer parameter sharing and outperforms the BERT model on the GLUE, RACE, and SQuAD benchmarks. Audio ALBERT is an extension of ALBERT for speech tasks [5]. inspired by Mockingjay, Audio ALBERT takes a masked spectrogram as the input. It also reduces over 90% of the parameters while achieving comparable performance with the massive Mockingjay model. W2V2-light also uses cross-layer parameter sharing to build a lightweight audio representation model but uses contrastive predictive coding loss following the Wav2vec 2.0 model [12]. Experiments on the Librispeech dataset show that W2V2-light has a comparable performance to the Wav2vec 2.0 base model on ASR tasks. Recently, MiniALBERT combines parameter sharing and knowledge distillation and obtains a compressed BERT model. Their MiniALBERT model achieves satisfactory performance on the GLUE benchmark and multiple biomedical Named Entity Recognition (NER) tasks, proving the feasibility of combining of parameter sharing and knowledge distillation on the compression of NLP pre-trained models. We believe that this approach is also likely to yield good results in pre-trained audio representation models.

In this paper, we propose DistilALHuBERT, a lightweight recursive audio representation model distilled from the HuBERT model. We design a feature alignment strategy for knowledge distillation, and demonstrate that the size of the transformer encoder rarely has a significant impact on the performance of the downstream tasks when the equivalent number of layers remains unchanged. Evaluations on the SUPERB benchmark show that our model can outperform the DistilHuBERT model with the same amount of parameters.

2 METHODS

2.1 The HuBERT Model

In this paper, our focus is mainly on the HubERT model, but our approach can be easily extended to other transformer-based pre-training models. HuBERT is a successful audio representation model in recent years. As the name suggests, hidden units (Hu) is very important for HuBERT. Before pre-training, the unlabeled data is clustered by Kmeans into hundreds of classes, and these discrete clustered classes are used as the hidden units of an unlabeled speech. The model is directly trained to predict the hidden units. Compared to the previous pre-training methods of audio representation models, the explicit introduction of these discrete clustered classes enables HuBERT to learn higher-level representations, which is beneficial for downstream tasks. The hidden-unit-based pre-training method has become the foundation of many advanced audio representation models (e.g. WavLM, Speech T5 [1], CTC BERT [8], and Speech LM [25]), and we also believe that our compression method for Hubert can also be applied to other pre-trained models.

The HuBERT model consists of a CNN audio feature extractor and an encoder network composed of transformer layers. A transformer layer is a stack of a multi-head attention block and

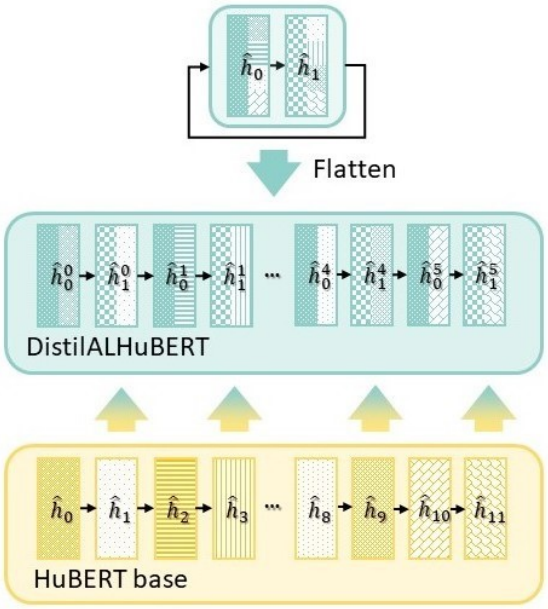

Figure 1: An overview of our method. With parameter sharing, a recursive transformer with G=2 can be extended to 12 layers. The prosodic, semantic and content-related features contained in different teacher layers are all used in the distillation, which is conducive to fully exploiting the parameters of the student model.

a feed-forward block. For each transformer layer i, let fi be the transformer layer, the output hi is computed as follows:

$$h_i = f_i\,(h_i - 1) \tag{1}$$

where $hi-1$ is the output from the previous layer or the output of the CNN feature extractor when i=1.

2.2 Cross-layer Parameter Sharing

Cross-layer parameter sharing is applied by repeatedly using the output of the network as its input. In DistilAL- HuBERT, instead of looping in a single layer, the hidden states pass through the entire transformer network and are reused as the input. For each layer i, let f_i be the ith layer, h_j, the output of f_i at jth loop, can be computed as follows:

$$h_i^j = f_i\left(h_{i-1}^j\right) \tag{2}$$

where $\mathrm{h}_{i-1}^{\mathrm{j}}$ is the output from the previous layer or the output of the previous loop[1] when $i = 1$.

2.3 Feature Alignment Distillation

We use the HuBERT-base model as the teacher and use mean square error (MSE) loss to mimic its hidden states. Different layers of the pre-trained model are applicable to different tasks (e.g., features from the bottom layers are usually related to speaker-related tasks,

[1] h_0 is the output of CNN feature extractor,

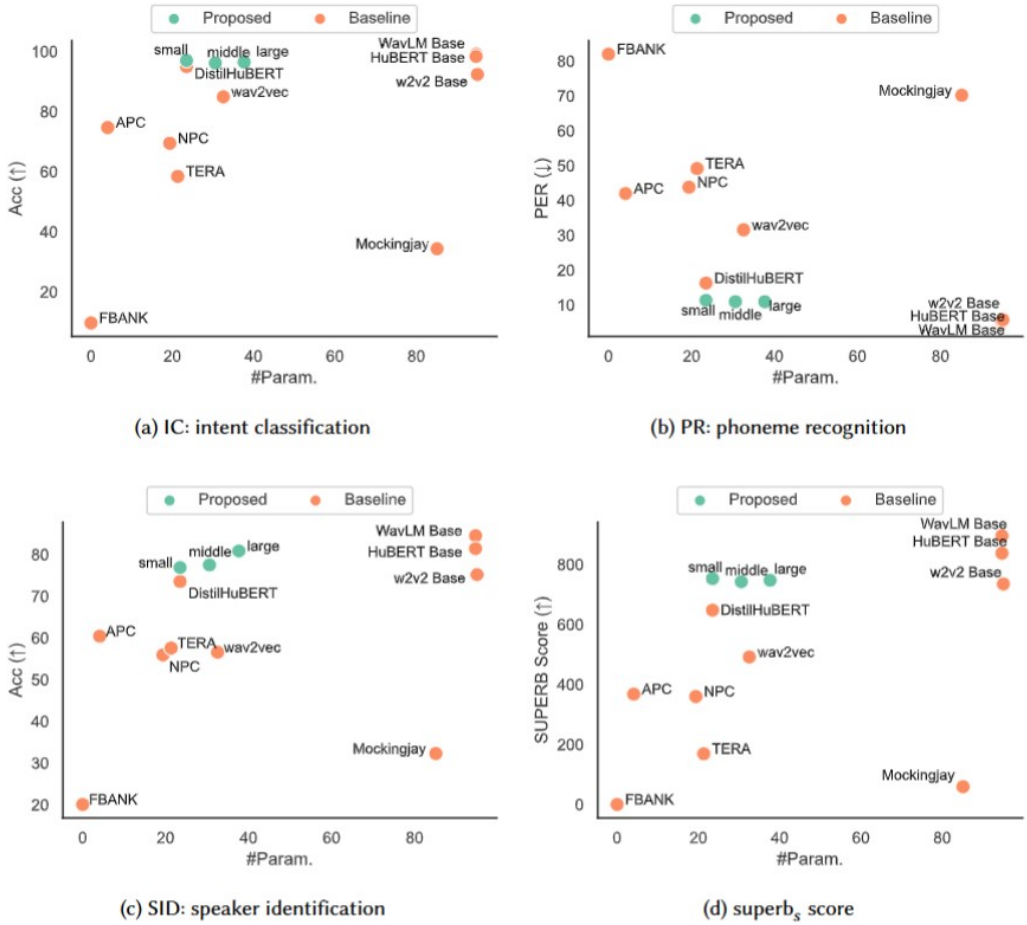

Figure 2: The relationship between the performances and the number of parameters. (a), (b), (c): Results on IC, PR and ASV tasks. (d): the overall superb*s* score. Results of APC [6], NPC [15], TERA [16], Mockingjay [17], Wav2vec [21] are from the SUPERB Benchmark.

while those from the top layers are generally applicable to content-related tasks such as speech recognition [3]). For better utilization of the hidden states from all the teacher layers, we apply a feature alignment distillation strategy (FAD). Inspired by MiniALBERT [18], we flatten the recursive transformer in DistilALHuBERT and let the output of certain layers from certain loops be aligned to the hidden states from the corresponding teacher layers. Formally speaking, let H be the hidden states of the teacher model and $\hat{H}$ be that of the student model, the distillation loss is computed as follows:

$$l_{\text{distil}}\left(\hat{H}, H\right) = \sum_{(i,j)\in S} l_{\text{MSE}}\left(\hat{h}_i^j,\ h_{Gj+i}\right) \tag{3}$$

Each tuple (i, j) in S indicates that the output of layer i at loop j is selected to compute the MSE loss, G is the total time of loops, and $\hat{h}$, h are the hidden states from a particular layer of the student and teacher model, respectively. Fig 1 shows an overview of our method with $G = 2$. Our multitask distillation method allows the learning of different types of features. The student may extract speaker-related features at the early stage of the loops and content-related features at the late stage of the loops, which can better adapt to various downstream tasks, especially in content-related tasks such as ASR.

3 EXPERIMENTS

3.1 Experimental Setup

Model. The basic model structure is similar to the Hubert-base model and consists of a 6-layer CNN feature extractor and a parameter-sharing transformer encoder. We design 3 types of transformer encoders, each with a different number of parameters. The large one has 4 transformer layers, and the middle/small one has 3/2 transformer layers. We let the loop times G be inversely proportional to the number of parameters. Specifically, for a model with N transformer layers, it is looped for G times so that $N \cdot G$ is a constant. In our experiments, the large, middle, and small models are looped 3, 4, and 6 times, respectively.

Training. The distillation is performed on 1000 hours of the LibriSpeech English dataset [19]. For each model type, we select 6

Table 1: Results on SUPERB of the baselines, teacher models, the DistilHuBERT model, and the proposed model. The bolder text is for the best results among the 3 proposed models. 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%). The superb*s* score can be considered as an average of all these task-specific scores.

Method	PR PER↓	KS Acc↑	IC Acc↑	SID Acc↑	ER Acc↑	ASR WER↓	QbE MTWV↑	SF F1↑/CER↓	ASV EER↓	SD DER↓	Superb_s↑
Baselines											
FBANK	82.01	41.38	9.65	20.06	48.24	23.18	0.58	69.64/52.94	9.56	10.05	0
Mockingjay [17]	70.19	83.67	34.33	32.29	50.28	22.82	0.07	61.59/58.89	11.66	10.54	59.23
TERA [16]	49.17	89.48	58.42	57.57	56.27	18.17	0.13	67.5/54.17	15.89	9.96	169.22
NPC [15]	43.81	88.96	69.44	55.92	59.08	20.2	2.46	72.79/48.44	9.4	9.34	360.23
APC [6]	41.98	91.01	74.69	60.42	59.33	21.28	3.1	70.46/50.59	8.56	10.53	368.09
Wav2vec [21]	31.58	95.59	84.92	56.56	59.79	15.86	4.85	76.37/43.71	7.99	9.9	491.59
Wav2vec 2.0 Base [2]	5.74	96.23	92.35	75.18	63.43	6.43	2.33	88.3/24.77	6.02	6.08	735
WavLM Base [4]	4.84	96.79	98.63	84.51	65.94	6.21	8.7	89.38/22.86	4.69	4.55	895.995
Teacher											
Hubert Base	5.41	96.3	98.34	81.42	64.92	6.42	7.36	88.53/25.2	5.11	5.88	838.07
DistilHuBERT Baseline											
DistilHuBERT	16.27	95.98	94.99	73.54	63.02	13.37	5.11	82.57/35.59	8.55	6.19	647.88
Proposed											
small	11.34	96.10	**96.99**	76.88	63.40	11.23	**6.80**	**85.04/30.80**	**5.46**	6.53	**753.15**
middle	10.94	**96.33**	96.15	77.51	64.62	**10.74**	6.38	84.16/30.33	6.10	6.69	742.62
large	**10.93**	**96.33**	96.41	**80.85**	**64.58**	10.77	6.25	84.93/30.54	6.16	**6.51**	747.41

i, j pairs to ensure that $\{Gi + j : (i,\ j \in S)\} = \{1, 3, 5, 7, 9, 11\}$. The batch size is set to 6 audios and the total number of update steps is 200k. We use a learning rate of 2.0e-4 and a warm-up strategy that linearly increases the learning rate to the set value in the first 14k steps. The distillation takes about 30 hours on an RTX 3090 GPU.

3.2 SUPERB Benchmark

SUPERB (Speech processing universal PERformance Benchmark) is a benchmark for evaluating the performance of pre-trained speech models [22]. 10 speech tasks from different domains are provided to test the quality of the features extracted from the pre-trained models, where these pre-trained models are not updated during fine-tuning. 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 (ASV), speaker diarization (SD), intent classification (IC), slot filling (SF), and emotion recognition (ER). During fine-tuning, we notice that some of the downstream tasks tend to overfit under the predefined hyperparameters using our distillation student model, so we adjust the learning rate and add some SpecAugment in these tasks. Specifically, we set the learning rate to 5.0e-5 and add SpecAugment in the SID task, and decrease the learning rate to 5.0e-4 and 5.0e-5 for the SF and IC tasks, respectively. For the rest of the tasks, we follow all the predefined training parameters. The hidden states of the last layer and the last loop are chosen as the features for the downstream tasks.

4 RESULTS

Table 1 shows the evaluation results of our proposed models on the SUPERB benchmark. The experiments show that the parameter-sharing-based DistilALHuBERT models outperform the DistilHuBERT model on most of the tasks. The comparison between the DistilALHuBERT-small model and the DistilHuBERT model demonstrates the effectiveness of parameter sharing more clearly. These two models have the same amount of parameters and are both distilled from the Hubert-Base model, but by extending the transformer encoder to 12 layers through parameter sharing, the former achieves a significant improvement in performance, especially on content-related tasks such as PR, ASR, and SF.

We also use the superb*s* score to obtain an overall evaluation of all the tasks. The superb*s* score is an average of the linear transformations of all the task-specific scores, and the scale intervals of these transformations are determined by the SOTA model on the benchmark and a predefined FBANK baseline. At the time of writing, the SOTA model is the WavLM-Large model, and we calculate all the SUPERB scores according to its performance[2]. Formally speaking, superb*s* score is defined as

$$superb_s = \frac{1}{T}\sum_{t\in T}\frac{1000}{s_t\,(sota) - s_t\,(fbank)}\left(s_t\,(u) - s_t\,(fbank)\right) \quad (4)$$

where $s_t(u)$ is the metric of task t and model u, superb(fbank) $\equiv 0$, superb(sota) $\equiv 1000$.

The superb*s* score also shows that the proposed parameter-sharing-based DistilALHuBERT model significantly outperforms

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

Table 2: Results of the Kruskal-Wallis test on the SUPERB tasks. We divide the test sets into 10 subsets (5 for the ER task), then evaluate the models and calculate the H-statistics and p-values. Model size is considered to have a significant effect on a task when p<0.05.

Tasks	H	p
PR: phoneme recognition	6.3245	0.0423
KS: keyword spotting	0.1711	0.9180
IC: intent classification	2.0916	0.3514
SID: speaker identification	19.5287	0.0006
ER: emotion recognition	1.6800	0.4317
ASR: automatic speech recognition	1.9000	0.3866
QbE: query-by-example spoken term detection	0.4955	0.7806
SF: slot filling	0.4709	0.7902
ASV: automatic speaker verification	14.3380	0.0007
SD: speaker diarization	2.9445	0.2294

the baseline DistilHuBERT model. On the other hand, there is no obvious difference between the three DistilALHuBERT models with different sizes of transformer encoders, either in the overall superb*s* metrics or in specific SUPERB tasks. Since the size of the evaluation dataset of some tasks is small and the differences in these tasks could be due to the randomness, we perform the Kruskal-Wallis test on all these tasks. We divide the test sets into 10 subsets (except for the emotion recognition task where cross-validation is required for evaluation. We simply use these cross-validation results) and calculate the H-statistic and p-value using the results from these subsets. Our null hypothesis is that the size of the transformer encoder has no effect on the test results. Table 2 shows that the size of the model does not have a significant influence on the performance of most of the tasks, which demonstrates the effectiveness of parameter sharing. Even when most of the parameters are shared with other layers, the small model can still achieve comparable performance to the large model in these tasks. On the other hand, the size of the transformer encoder does affect the performance in the PR, SID and ASV tasks. Given that no such significance is observed for similar tasks such as ASR and SD, this difference may be due to the structure of the downstream model or specific fine-tuning parameters, rather than any substantial discrepancies in the extracted representations.

We also visualize the relationship between the superb*s* score and the number of parameters in Fig. 2, where being closer to the top left means a compressed audio representation model is better. It can be seen that the DistilALHuBERT small model has comparable performance to the Wav2vec 2.0 base model with over 60% fewer parameters, and has a 70% improvement on the superb*s* score with an even smaller size.

5 CONCLUSION

In this paper, we propose DistilALHuBERT, a compressed audio representation model. we use the recursive transformer to implement cross-layer parameter sharing and use a feature alignment distillation method to capture the representation ability of the fully parameterized Hubert-base model. Experiments on the SUPERB benchmark demonstrate the effectiveness of parameter sharing. By repeatedly processing the input features using the same transformer encoder, the proposed model achieves significant improvement in semantic tasks compared to the DistilHuBERT baseline with the same amount of parameters. We also perform a Kruskal-Wallis test to show the effect of the size of the transformer encoders. We proved that if the equivalent number of transformer layers remains unchanged, using transformer encoders of different sizes does not have a significant impact on the performance.

ACKNOWLEDGMENTS

This work was supported by the National Key R&D Program of China under Grant No. 2020AAA0104500, and the National Natural Science Foundation of China under Grant No. 62276153.

REFERENCES

[1] Junyi Ao, Rui Wang, Long Zhou, Chengyi Wang, Shuo Ren, Yu Wu, Shujie Liu, Tom Ko, Qing Li, Yu Zhang, *et al.* 2022. SpeechT5: Unified- Modal Encoder-Decoder Pre-Training for Spoken Language Processing. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 5723–5738.

[2] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. Advances in Neural Information Processing Systems 33 (2020), 12449–12460.

[3] Heng-Jui Chang, Shu-wen Yang, and Hung-yi Lee. 2022. DistilHuBERT: Speech representation learning by layer-wise distillation of hidden-unit BERT. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 7087–7091.

[4] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, and Furu Wei. 2021. WavLM: Large-Scale Self-Supervised Pre-training for Full Stack Speech Processing. (2021). arXiv:2110.13900 [cs.CL]

[5] Po-Han Chi, Pei-Hung Chung, Tsung-Han Wu, Chun-Cheng Hsieh, Yen-Hao Chen, Shang-Wen Li, and Hung-yi Lee. 2021. Audio ALBERT: A Lite BERT for Self-supervised Learning of Audio Representation. http://arxiv.org/abs/2005.08575 arXiv:2005.08575 [cs, eess].

[6] Yu-An Chung, Wei-Ning Hsu, Hao Tang, and James Glass. 2019. An unsupervised autoregressive model for speech representation learning. arXiv preprint arXiv: 1904.03240 (2019).

[7] Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Łukasz Kaiser. 2019. Universal Transformers. https://doi.org/10.48550/arXiv.1807.03819 arXiv:1807.03819 [cs, stat].

[8] Ruchao Fan, Yiming Wang, Yashesh Gaur, and Jinyu Li. 2022. CTCBERT: Advancing Hidden-unit BERT with CTC Objectives. arXiv preprint arXiv:2210.08603 (2022).

[9] Zhiyun Fan, Meng Li, Shiyu Zhou, and Bo Xu. 2021. Exploring wav2vec 2.0 on speaker verification and language identification. Technical Report arXiv: 2012.06185. arXiv. http://arxiv.org/abs/2012.06185 arXiv:2012.06185 [cs, eess] type: article.

[10] Sara Hooker, Aaron Courville, Gregory Clark, Yann Dauphin, and Andrea Frome. 2019. What do compressed deep neural networks forget? arXiv preprint arXiv: 1911.05248 (2019).

[11] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. IEEE/ACM Transactions on Audio, Speech, and Language Processing 29 (2021), 3451–3460.

[12] Dong-Hyun Kim, Jae-Hong Lee, Ji-Hwan Mo, and Joon-Hyuk Chang. 2022. W2V2-Light: A Lightweight Version of Wav2vec 2.0 for Automatic Speech Recognition. In Interspeech 2022. ISCA, 3038–3042. https://doi.org/10.21437/Interspeech.2022-10339

[13] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. http://arxiv.org/abs/1909.11942 arXiv: 1909.11942 [cs].

[14] Yeonghyeon Lee, Kangwook Jang, Jahyun Goo, Youngmoon Jung, and Hoirin Kim. 2022. FitHuBERT: Going Thinner and Deeper for Knowledge Distillation of Speech Self-Supervised Learning. arXiv preprint arXiv:2207.00555 (2022).

[15] Alexander H Liu, Yu-An Chung, and James Glass. 2020. Non-autoregressive predictive coding for learning speech representations from local dependencies. arXiv preprint arXiv:2011.00406 (2020).

[16] Andy T Liu, Shang-Wen Li, and Hung-yi Lee. 2021. Tera: Self-supervised learning of transformer encoder representation for speech. IEEE/ACM Transactions on Audio, Speech, and Language Processing 29 (2021), 2351–2366.

[17] Andy T Liu, Shu-wen Yang, Po-Han Chi, Po-chun Hsu, and Hung-yi Lee. 2020. Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders. In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 6419–6423.

[18] Mohammadmahdi Nouriborji, Omid Rohanian, Samaneh Kouchaki, and David A. Clifton. 2022. MiniALBERT: Model Distillation via Parameter-Efficient Recursive Transformers. http://arxiv.org/abs/2210.06425 arXiv:2210.06425 [cs].

[19] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 5206–5210.

[20] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. 2014. Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550 (2014).

[21] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised pre-training for speech recognition. arXiv preprint arXiv: 1904.05862 (2019).

[22] Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, *et al.* 2021. Superb: Speech processing universal performance benchmark. arXiv preprint arXiv:2105.01051 (2021).

[23] Xue Ying. 2019. An overview of overfitting and its solutions. In Journal of physics: Conference series, Vol. 1168. IOP Publishing, 022022.

[24] Yu Zhang, James Qin, Daniel S. Park, Wei Han, Chung-Cheng Chiu, Ruoming Pang, Quoc V. Le, and Yonghui Wu. 2020. Pushing the Limits of Semi-Supervised Learning for Automatic Speech Recognition. Technical Report arXiv:2010.10504. arXiv. http://arxiv.org/abs/2010.10504 arXiv:2010.10504 [cs, eess] version: 1 type: article.

[25] Ziqiang Zhang, Sanyuan Chen, Long Zhou, Yu Wu, Shuo Ren, Shujie Liu, Zhuoyuan Yao, Xun Gong, Lirong Dai, Jinyu Li, *et al.* 2022. Speechlm: Enhanced speech pre-training with unpaired textual data. arXiv preprint arXiv:2209.15329 (2022).

[26] Jinming Zhao, Ruichen Li, Qin Jin, Xinchao Wang, and Haizhou Li. 2022. Memobert: Pre-Training Model with Prompt-Based Learning for Multimodal Emotion Recognition. In ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 4703–4707. https://doi.org/10.1109/ICASSP43922.2022.9746910 ISSN: 2379-190X.

[27] Jing Zhao and Wei-Qiang Zhang. 2022. Improving automatic speech recognition performance for low-resource languages with self- supervised models. IEEE Journal of Selected Topics in Signal Processing 16 (2022), 1227–1241. Issue 6. https://doi.org/10.1109/JSTSP.2022. 3184480

[28] Yingzhu Zhao, Chongjia Ni, Cheung-Chi Leung, Shafiq Joty, Eng Siong Chng, and Bin Ma. 2020. Universal Speech Transformer. In Interspeech 2020. ISCA, 5021–5025. https://doi.org/10.21437/Interspeech.2020-1716