Towards Data Efficient And Robust Speech Representation Model Distillation

Pheobe Sun[*]
University College Dublin

Ruibo Shi
JP Morgan Chase & Co.

Ahmad Emani
JP Morgan Chase & Co.

Sean Moran
JP Morgan Chase & Co.

Abstract

While knowledge distillation has been proven effective in learning student models of smaller size on various tasks, a large amount of distillation training data is required to keep the performance of the student model competitive to the teacher model. Our research aims to further improve the efficiency in task-agnostic speech representation model pre-training. By perturbing the training data distribution, we distil a more robust task-agnostic speech representation model with a lower training data requirement. By learning representations from both a) the teacher model, which is trained via self-supervised learning (SSL) and b) the known effective hand-crafted features, we effectively regularize and compensate the representation loss due to the distillation process. Our proposed methods are evaluated on a number of downstream tasks and are shown to be effective in certain aspects, which prompts future research that builds on our work to develop efficient task-agnostic speech representation model distillation approaches.

1 Introduction

Task-agnostic speech representation learning has gained much attention lately due the promise of learning effective speech and audio representations from vast amounts of unlabeled data. It is common that high-performing downstream speech tasks highly depend on the choice of hand-crafted audio features. For example, while Mel Frequency Cepstrum (MFCC) and Log-mel Filterbanks (FBANK) are typically used in automatic speech recognition [1] [2], Constant Q Cepstral Coefficients (CQCC) has shown to be effective in Anti-spoofing Speaker Verification tasks [3]. Though Deep Neural Networks (DNNs) can extract powerful embeddings, when trained discriminatively in a supervised setup, these latent representations are usually not general-purpose. Therefore, learning a general-purpose speech representation without the need for labels promises great model re-usability and a light-weight adaptation process for downstream tasks in terms of task-specific model complexity and data requirements.

Currently, self-supervised learning (SSL) is the key approach to pre-train such general-purpose speech representation models [4], following the success of learning visual representations [5] and language representations [6]. However, the success of SSL relies on the availability of large amounts of training data and the abundance of computational resources that can support complex and deep models such as HuBERT [7], Wav2vec 2.0 [8], data2vec [9], and WavLM [10]. The computational resource requirement of large pre-trained models make it cumbersome and less accessible for the larger community to fine-tune and perform inference using these models. This motivates the research in model compression and acceleration [11], especially for large deep models trained using SSL.

[*] Work done while the author was an intern at JP Morgan Chase & Co.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

Knowledge distillation (KD) is a popular technique for compressing deep models. 'Knowledge' can be regarded in three ways [12]: response-based (*e.g.* logits), feature-based (*e.g.* representation), and relation-based knowledge. Knowledge transfer is achieved using a teacher-student architecture. Distilling speech representation requires feature-based knowledge, of which a lot are from the intermediate layers [13]–[15]. DistilHuBERT [16] is a lightweight model that maintains a competitive performance with only approximately 25% of the trainable parameter of HuBERT. However, the distilled model experiences a loss in speech representation as seen in the decreased performance in several downstream tasks, especially the speaker verification task. The same amount of data (960 hours) was used in the distillation process as what was used to pre-train the original deep HuBERT model. Such requirement on the data size makes the knowledge distillation process less useful to transfer knowledge in a low-resource scenario. With that, we are interested in looking for ways to increase the data efficiency as well as to improve the speech representation learnt by the lightweight model.

To improve the data efficiency and model robustness, we pre-process the student data input with a probabilistic combination of distortions thereby forcing the student model to learn effective and salient representations. Furthermore, to compensate the potential loss of speech representations that are not explicitly learnt in the teacher model which is trained with SSL, we propose a knowledge injection network inspired by [17], [18]. The goal is to see whether leveraging the advantages of both SSL and the known effective hand-crafted feature would help the model to learn representations more efficiently. We found both approaches have shown positive effect on the learnt features by the lightweight student model. In the following sections, we introduce and present the methodology, training setup and experiment results in detail.

2 Methodology

2.1 Architecture

We adopt a multi-task learning setup to inject the knowledge of multiple known effective hand-crafted features while distilling from a task agnostic teacher model that is trained with SSL. Figure 1 shows the mechanism of the multi-task design.

The model distillation process is depicted in blue in Figure 1. We distill the student model from the pre-trained HuBERT-base model [7] using a teacher-student framework similar to DistilHuBERT [16] where the output of the student model learns to reconstruct the target hidden layers from the teacher model. Multiple layers were chosen as learning targets to facilitate learning a variety of speech representations, since different middle layers were found to be correlated with different aspects of speech [13] such as speaker, phonetic, and semantic information. The target layers (the 4th, 8th, and 12th transformer layers) were selected based on previous layer-wise analysis [13] and empirical experiments [16].

To strengthen the representation with known hand-crafted features that were proven effective, we assign additional tasks for the student model to reconstruct the selected hand-crafted features. Specifically, the additional tasks are introduced by attaching simple workers to the desired intermediate layer in the student network, each worker is parameterised by two dense layers and corresponds to one specific hand-crafted feature. The architecture is described by the yellow part in Figure 1. We chose five hand-crafted features including mel-frequency cepstral coefficients (MFCC), filter banks (FBANKS), log power spectrum (LPS), Gammatone, and prosody (see Appendix B for more details) inspired by previous studies [17], [18].

2.2 Loss Functions

The objective function consists of two parts, one evaluates the errors in model distillation $\mathcal{L}_{distil}$ and another stands for model's ability to reconstruct the target hand-crafted features $\mathcal{L}_{feature}$ (see the blue and the yellow boxes in Figure 1). $\mathcal{L}_{distil}$ is the sum of the distillation errors for each target layer, namely $\mathcal{L}^{4}_{distil}$, $\mathcal{L}^{8}_{distil}$, and $\mathcal{L}^{12}_{distil}$. The model distillation evaluation takes both L1 $\mathcal{L}_{l1}$ and cosine similarity $\mathcal{L}_{cos}$ losses for better performance based on empirical results [16] (see Appendix A for more details); the second part of the objective function averages the mean square errors (MSE) across all extra heads that predict different desired hand-crafted features. We used the weight combination

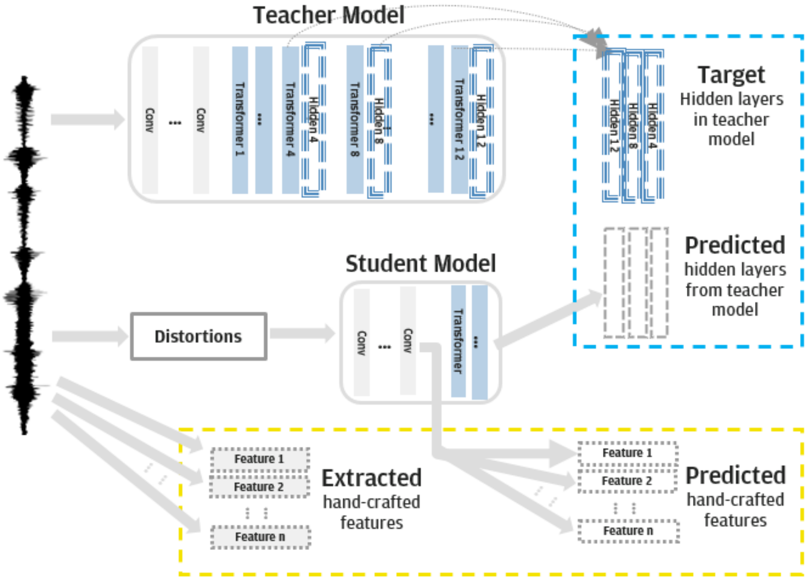

Figure 1: Mechanism of distillation using distorted input and knowledge injection

β_1 = 0.9 and β_2 = 0.1 in our experimental setup when we attached the knowledge injection network to the student network (see Tab 4).

$$
\begin{aligned}
\mathcal{L}_{total} &= \beta_1 \mathcal{L}_{distil} + \beta_2 \mathcal{L}_{feature} && (1)\\
&= \beta_1 (\mathcal{L}_{l1} + \mathcal{L}_{cos}) + \beta_2 \mathcal{L}_{feature} && (2)
\end{aligned}
$$

2.3 Data distortion

We fed the distorted data to the student network and pass the original data to the teacher model (see Fig 1). This is to encourage the student network to learn a more robust speech representation. A variety of distortions were applied with an assigned probability (see Appendix C).

3 Experiment

We used S3PRL [2] and its distiller upstream framework to train our lightweight models. 100hr-clean LibriSpeech (pre-prepared, see Appendix E for more details) was used to pre-rain our models, as oppose to 960-hour as used in [16]. Two baseline models were distilled from HuBERT using LibriSpeech 100hr-clean and LibriSpeech 960hr training dataset. The models were trained and tested on a number of downstream tasks for fixed 200k iterations with a batch size of 8.

To evaluate the speech representation of the pre-trained task-agnostic models, we chose four downstream tasks from SUPERB [19] across three task categories: phoneme recognition (PR) from content-related tasks, intent classification (IC) from semantic-related tasks, and speaker verification (SV) from speaker-related tasks. For the PR task, we also tested the task performance using the LibriSpeech test-other in addition to the default LibriSpeech test-clean.

4 Result

To evaluate the effectiveness of our proposed approach, we benchmarked the quality of speech representation against the model pre-trained with 100 hours of LibriSpeech (see model 2 in Tab 4).

[2] `https://github.com/s3prl/s3prl`

	Pre-trained Model			PR test-clean	PR test-other	IC	SV
No.	Dataset	Distortion	Knowledge Injec	PER ↓	PER ↓	ACC ↑	EER ↓
1	LS-960hr*	-	-	15.27	28.81	93.78	6.15
2	LS-100hr-clean	-	-	18.50	37.18	91.48	7.24
3	LS-100hr-clean	✓	-	**17.40**	**32.47**	94.86	6.80
4	LS-100hr-clean	✓	✓	17.45	32.89	**95.04**	**6.73**

Table 1: Comparison of four downstream task performance of four pre-trained models. All models were distilled using the LibriSpeech (LS) dataset. Model 1-2 are the baselines; 3-4 are distilled using our proposed approach. The 1st model distilled using 960 hr data is taken from the public checkpoint. 'Test-clean' and 'test-other' refer to two different test datasets in LibriSpeech. Phoneme error rate (PER) is used to evaluate the PR task, accuracy (ACC) for IC, and equal error rate (EER) for SV.

We found a significant performance improvement in the model pre-trained using our approach. We also include the performance of the model pre-trained with 960 hr LibriSpeech to show the performance gap (model 1 in Tab 4). We confirm that the size of pre-train data has a dominant effect on model performance (see model 1 & 2 in Tab 4). Surprisingly, our models pre-trained with 100 hr data even outperform the model pre-trained with 960 hr data in the intent classification task. The paragraphs below will analyse the effect of each treatment separately.

4.1 Effect of distortion

Both model 3 & 4 trained using distorted data outperformed the benchmark model 2 trained with original data (see Tab 4). This finding indicates that the application of distortion to the student network is a promising technique to further increase the data usage efficiency for pre-training. The significant PER drop in the PR task using the test-other dataset indicates that the model has learnt a more robust representation for the PR task.

4.2 Effect of knowledge injection

In addition to distorting the training data, we also added extra heads for the student network to learn to reconstruct selected hand-crafted features. When five additional heads were added to the student network, we saw a further improvement in the intent classification and speaker verification downstream tests albeit the gaps were small. However, the performances in phoneme recognition tasks were slightly worsened.

5 Conclusion

In this paper, we proposed a method that uses distorted data and a knowledge injection network to enhance the efficiency of data usage when pre-training a lightweight speech representation model. The quality of speech representation is evaluated using four different downstream tasks relating to content, semantic, and speaker. We found that distorting the pre-training data feeding into the student network is an impactful means to improve the learnt speech representation in the teacher-student distillation setup when only 100 hr training data was available. Although the data distortion process requires extra computational resources, no extra parameter is introduced to the student network. Thus, using distorted data for model distillation can be regarded as an efficient way to pre-train a robust lightweight speech representation model, especially in the scenario with limited training data. Our experimental results also showed a further improvement in the semantic and speaker-related speech representation as exemplified by the intent classification and speaker verification task performances when we attached a knowledge injection network to the student network, although the impact is much smaller than that of distortion. The slight improvement in the representation learnt came with a trade-off as extra trainable parameters (about 5%) were introduced to the pre-training stage by the knowledge injection network. The size of the model does not change as the knowledge injection network will be taken away in the fine-tuning or inference stages. It is still hard to evaluate the usefulness of using a knowledge injection network along with the knowledge distillation process. Future experiments can use more hand-crafted features, and fine-tune the hyper-parameters including the weights of different losses as well as the duration spans that produce the target hand-crafted audio features.

References

[1] D. Amodei, S. Ananthanarayanan, R. Anubhai, *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ser. ICML'16, New York, NY, USA: JMLR.org, 2016, pp. 173–182.

[2] P. Bell, J. Fainberg, O. Klejch, J. Li, S. Renals, and P. Swietojanski, "Adaptation algorithms for neural network-based speech recognition: An overview," *IEEE Open Journal of Signal Processing*, vol. 2, pp. 33–66, 2021. DOI: 10.1109/OJSP.2020.3045349.

[3] J. Yang and R. K. Das, "Improving anti-spoofing with octave spectrum and short-term spectral statistics information," *Applied Acoustics*, vol. 157, p. 107 017, 2020, ISSN: 0003-682X. DOI: https://doi.org/10.1016/j.apacoust.2019.107017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0003682X19302397.

[4] S. Liu, A. Mallol-Ragolta, E. Parada-Cabeleiro, *et al.*, "Audio self-supervised learning: A survey," *arXiv preprint arXiv:2203.01205*, 2022.

[5] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *European conference on computer vision*, Springer, 2016, pp. 69–84.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[7] 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.

[8] 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.

[9] 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," *arXiv preprint arXiv:2202.03555*, 2022.

[10] S. Chen, C. Wang, Z. Chen, *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[11] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "Model compression and acceleration for deep neural networks: The principles, progress, and challenges," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 126–136, 2018.

[12] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[13] A. Pasad, J.-C. Chou, and K. Livescu, "Layer-wise analysis of a self-supervised speech representation model," in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, IEEE, 2021, pp. 914–921.

[14] A. Baevski, W.-N. Hsu, A. Conneau, and M. Auli, "Unsupervised speech recognition," *Advances in Neural Information Processing Systems*, vol. 34, pp. 27 826–27 839, 2021.

[15] X. Chang, T. Maekaku, P. Guo, *et al.*, "An exploration of self-supervised pretrained representations for end-to-end speech recognition," in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, IEEE, 2021, pp. 228–235.

[16] 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.

[17] S. Pascual, M. Ravanelli, J. Serra, A. Bonafonte, and Y. Bengio, "Learning problem-agnostic speech representations from multiple self-supervised tasks," *arXiv preprint arXiv:1904.03416*, 2019.

[18] M. Ravanelli, J. Zhong, S. Pascual, *et al.*, "Multi-task self-supervised learning for robust speech recognition," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 6989–6993.

[19] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, *et al.*, "Superb: Speech processing universal performance benchmark," *arXiv preprint arXiv:2105.01051*, 2021.

[20] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[21] 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.

A Loss function

The total loss takes the sum of $\mathcal{L}_{distil}$ and $\mathcal{L}_{feature}$, where $\mathcal{L}_{distil}$ is defined as:

$$\mathcal{L}_{distil} = \sum_{t=1}^{T}[\frac{1}{D}\|h_t^{(l)} - \hat{h}_t^{(l)}\|_1 - log\sigma(cos(h_t^{(l)}, \hat{h}_t^{(l)}))] \tag{3}$$

As defined in [20], the L_{distil} loss is calculated at each t^{th} time step taking into considerations of all the target $l^(th)$ hidden layer h_t^l. h_t^l denotes the target hidden layer in the teacher network and $\hat{h}_t^l$ denotes the predicted counterpart by the student network. $\mathcal{D}$ denotes the number of dimensions of the feature vectors. σ denotes the sigmoid activation o f the cosine similarity. The goal for the distillation process is to minimise the L_1 distance whilst maximising the cosine similarity.

B Hand-crafted features

We used hand-crafted features including mel-frequency cepstral coefficients (MFCC), filter banks (FBANKS), log power spectrum (LPS), Gammatone, and prosodic features based on the ablation studies in [17] and [18]. The prosodic features include the fundamental frequency, voiced and unvoiced ratio, zero-crossing rate, and energy [17].

The ground truth values of the hand-crafted feature are calculated directly from the input audio segments with a span of 25 ms and the hop size of 20 ms (window size = 400, hop size = 320) using Librosa and the Gammatone toolkit [3]. First and second order of the feature value were also calculated (see more details in Tab B). Each feature carries the same weight and is z-normalised in the loss calculation.

Feature	Parameters
MFCC	order = 13
FBanks	n_filters = 40
Gammatone	n_channels = 40
LPS	n_fft = 2048

Table 2: Distortions applied and the corresponding probability

C Distortions

We applied a variety of distortions on each speech segment with an assigned probability (see Tab C).

Distortion type	Probability
Additive noise	0.4
Clipping	0.2
Chopping	0.2
Downsample	0.25
Band drop	0.35
Reverb	0.5

Table 3: Distortions applied and the corresponding probability

D Training resources

The models were pre-trained and fine-tuned on one 24GB A10 GPU with 32 CPUs. Due to computation limit, each experiment only ran once.

[3] `https://github.com/detly/gammatone`

E Dataset

The training portion of the LibriSpeech corpus comes with three sizes: 100, 360 and 500 hours [21]. Both 100-hr and 360-hr training sets come from the same pool – the ‘clean’ pool. In each training set, the total duration of speech from a single speaker is limited to 25 minutes. The ‘clean’ pool contains the speech read by speakers with lower word error rate. To the contrary, the ‘other’ pool which counts for 500 hours, are recordings from the speakers with higher word error rate. We used the 100-hr LibriSpeech corpus in our experiment as the training dataset is gender- and speaker-balanced.