Temporal Knowledge Distillation for On-device Audio Classification

Kwanghee Choi[1*] Martin Kersner[1*] Jacob Morton[1*] Buru Chang[1] *Equal contributions.

{kwanghee.choi,jake.m,buru.chang}@hpcnt.com [1]Hyperconnect, Republic of Korea

HYPERCONNECT

Summary

- Our method can distill the temporal knowledge from attention weights of large transformer-based teacher models to on-device student models of various architectures.
- We design the attention distillation loss for the on-device models by attaching a simple attention layer only at training time to align the teacher and the student attention weights.
- We conduct experiments on a real-world AED dataset (FSD50K) and a noisy KWS dataset (background noise injected to Google Speech Commands v2) to show the effectiveness of our method.

Motivation

Why knowledge distillation (KD)? Compared to large models, improving the performance of on-device models is challenging due to the restricted computing resources in the mobile environment. Many studies leverage KD to alleviate this problem by transferring the knowledge from large models to on-device models.

Limitation of traditional KD Several studies focus on the knowledge embedded in logits produced by the classification layer. However, temporal information, which is known to be beneficial in audio tasks, cannot be easily distilled when it is compressed into logits.

Limitation of transformer-based KD With the success of the transformer, recent studies have focused on distilling the knowledge from self-attention maps, preserving the temporal information. However, those methods are limited to transformer-based architectures only, alienating other on-device-friendly architectures such as convolutional neural networks or recurrent neural networks.

Our goal

We aim to incorporate the temporal knowledge embedded in attention weights of large transformer-based models into on-device models with various types of architectures.

Teacher model We employ the XLSR-wav2vec 2.0 as our teacher model, which is a large-scale transformer-based ASR model with state-of-the-art performance on multilingual ASR.

Student model We consider the following on-device audio classification models: a simple RNN-based model (**LSTM-P**), a CNN-based model (**TC-ResNet**), a model that uses both CNN and RNN (**CRNN**), a model including an attention mechanism (**Att-RNN**), and a multi-head variant of Att-RNN (**MHAtt-RNN**).

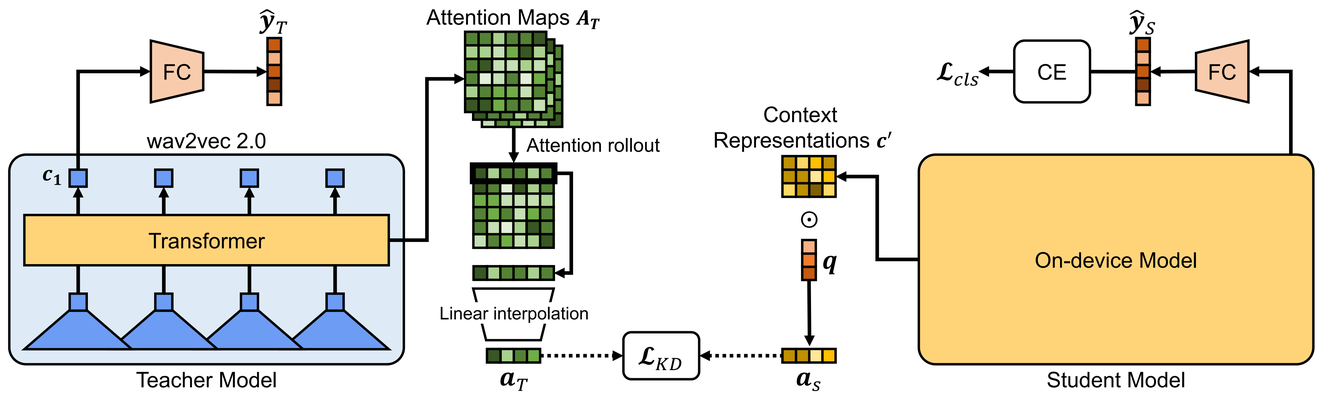

Figure 1:Illustration of our proposed method.

Our method: Temporal KD

Training the teacher model To perform audio classification, we attach a fully-connected (FC) layer to the first feature output $\mathbf{c}_1$ of the teacher model, similar to the fine-tuning of language models.

Extracting the teacher attention maps To extract the temporal knowledge from the teacher model, we leverage self-attention maps $\mathbf{A}_T$ from multiple transformer layers. The attention rollout technique is applied to result in a single unified attention map.

Extracting the student attention maps We integrate the attention mechanism from every student model except Att-RNN and MHAtt-RNN, only in training time. Medium context representation projection is used for the query $\mathbf{q}$.

Minimizing the distance between attention maps We shrink the teacher attention map by linear interpolation to match the dimension of student attention map. Then, the distance between the two maps are minimized via the KL divergence: $\mathcal{L}_{\text{KL}} = D_{\text{KL}}(\mathbf{a}_S|\mathbf{a}_T)$.

Final loss Classification loss $\mathcal{L}_{\text{CLS}}$ and the KD loss $\mathcal{L}_{\text{KL}}$ is jointly optimized via $\lambda\mathcal{L}_{\text{KL}} + (1-\lambda)\mathcal{L}_{\text{CLS}}$, with a trade-off hyperparameter λ.

Results and Analysis

Model	wav2vec 2.0	LSTM-P	TC-ResNet	CRNN	Att-RNN	MHAtt-RNN
w/o KD	0.5498	0.1141	0.1814	0.2789	0.2856	0.2647
w/ KD	N/A	**0.1300**	**0.1951**	**0.3053**	**0.3471**	**0.3317**

Table 1:mAP performance comparison on the FSD50K dataset, a real-world multi-label audio event detection (AED) dataset, with and without applying our KD loss.

L	Model	wav2vec 2.0	LSTM-P	TC-ResNet	CRNN	Att-RNN	MHAtt-RNN
2s	w/o KD	90.59	88.73	87.77	89.96	89.88	89.75
	w/ KD	N/A	**89.31**	**88.08**	**90.06**	<u>91.67</u>	<u>91.75</u>
4s	w/o KD	91.22	85.19	87.60	89.69	90.65	91.19
	w/ KD	N/A	**89.08**	**88.33**	**90.21**	<u>91.98</u>	<u>92.12</u>
6s	w/o KD	90.93	45.27	86.00	88.58	90.88	90.58
	w/ KD	N/A	**85.58**	**86.85**	**89.88**	<u>91.19</u>	<u>91.67</u>
8s	w/o KD	90.95	78.44	77.81	88.94	88.81	88.33
	w/ KD	N/A	**82.19**	**85.79**	**89.79**	<u>90.98</u>	<u>91.79</u>

Table 2:Test accuracy (%) performance comparison on the Noisy Speech Commands v2 dataset. Best accuracies are in bold, and the performance of the student models that outperform the teacher model is underlined. The dataset is constructed by inserting the existing keyword spotting (KWS) dataset, Speech Commands v2, to the background speech noise of varying length L.

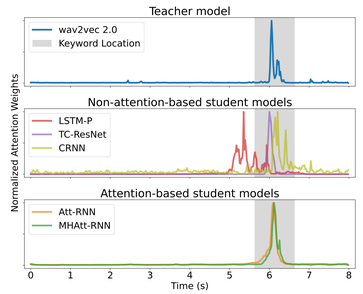

Figure 2:Visualization of attention weights extracted from multiple models. We input an arbitrary sample from the Noisy Speech Commands v2 dataset with 8 seconds noise. We plot the location of the one second keyword to all the plots. Even though the teacher model is trained only with the classification label, attention weights focuses on the keyword location. Also, all the on-device models attend at similar positions, indicating that attention weights are accurately aligned.