Distilling Efficient Audio Models using Data Pruning with CLAP

Alexander Werning[1], Reinhold Haeb-Umbach[1]
[1] *Department of Communications Engineering, Paderborn University, 33098 Paderborn, Germany,*
email: {werning,haeb}@nt.upb.de

Abstract
Running state-of-the-art large-scale audio models on edge devices is often infeasible due to their limited storage and computing resources. It is therefore necessary to compress and tune the models for the specific target task and hardware. This is commonly achieved by distilling the audio model, the teacher, to a small target model, the student. However, this approach can be improved by prepending a dataset pruning stage and training the teacher on the pruned data set only, which contains examples relevant to the target task. Recently, CLAP models have emerged that embed audio and text examples in a common embedding space. We use the audio embeddings of the CLAP model for the above pruning stage, which is realized using a domain classifier. After knowledge distillation, the student is eventually fine-tuned on some data from the target domain. The CLAP architecture combines text and audio embedding spaces, which allows to search for data given only a textual description, such as a class label. We show how this can help data pruning.

Introduction
The capabilities of foundation models scale with their size but, unfortunately, so do the computational requirements needed for inference [1, 2]. This poses problems for applications on edge devices, e.g., on distributed sensor nodes or other mobile hardware with restricted power and computational resources. Our goal is to be able to deploy efficient models to these resource-constrained devices in order to process data where it is collected. To obtain these efficient models, we wish to make use of preexisting large models as they are known to be very powerful.

The application domain considered here is audio tagging, i.e., recognizing the sound class in an audio clip. Recent work showed that knowledge distillation can be applied even across model architectures to obtain efficient audio tagging models [3], resulting in efficient student models distilled from large-scale, usually transformer-based teacher models.

In prior work we could show that for a specific downstream application, selecting a small subset of data combined with knowledge distillation produces better compressed models [4]. While that work showed the principle feasibility, using the well-known PaSST model as the teacher [5], we here go a step further and employ an audio-text bimodal model which allows to select the subset by using only a textual description of the sounds to be recognized in the target task. We further thoroughly assess its performance both on an audio tagging task and in an audio-text retrieval task.

Contrastive Language-Audio Pre-trained (CLAP) models are trained on paired text-audio data in a way that allows to project data from the two modalities into a common embedding space [6]. The architecture can be separated into an audio encoder $\mathbf{E}_\mathrm{A}$ and a text encoder $\mathbf{E}_\mathrm{T}$, which map audio $\mathrm{A_i}$ and text $\mathrm{T_i}$ to a common embedding space, respectively. The audio and text embeddings of a trained CLAP model enable zero-shot classification. At test time, classes can be defined using textual descriptions and classification is performed by computing the distances between an audio embedding and all embeddings computed from the textual class descriptions.

When a system based on a CLAP model is deployed, we expect to classify many audio samples into classes which are defined using text and encoded once using the text encoder. Instead of compressing the full multi-modal model [7], it is thus sufficient to have an efficient audio encoder. Note that while the training or distillation of a CLAP model requires paired audio-text data, the compression of just the CLAP audio encoder into, for example, an efficient CNN architecture [8] does not.

Domain adaptation is the problem of applying a model that was trained on a source dataset to a target dataset from a different domain [9]. Here, we consider the special case where the target domain is a subset of the source domain. For this reason, we call the process a specialization, transforming a generalist to a specialist system.

In order to obtain a specialized model, we perform a pruning of the training data. While dataset pruning is often used to remove noisy or difficult data [10, 11], here the purpose of pruning is to removed data not related to the target domain. The underlying hypothesis is that fewer and better data should improve the resulting models on tasks in the target domain [12].
Pre-trained audio embedding models allow to find relevant in-domain data [4], this is also the case for multimodal models which can pair audio data with semantically similar text data [13, 10].
We propose to prune a source dataset such that is similar to a target dataset. The data is selected via similarity between audio embeddings to text embeddings of the target classes. A new model is then distilled on the pruned data set, learning to estimate CLAP embeddings. This distilled model is then fine-tuned on the target dataset to solve a classification task.

Method

The proposed method consists of three steps: data pruning, knowledge distillation, and fine-tuning. The first step determines which data from a source dataset is relevant for a task defined by a target dataset. This data is then used to train a student model on outputs of a pre-trained teacher. The student is then adapted to the label set of the target dataset in the fine-tuning stage. This framework is based on previous work [4].

Data Pruning

The audio data from the source dataset that is most similar to the textual class descriptions is selected using a domain classifier as follows. Each class label is transformed to a text prompt $\mathrm{P_i}$ and encoded using the text encoder of the CLAP teacher model $\mathbf{E}_\mathrm{T}(\mathrm{P_i})$ into an embedding that we call class prototype. The audio data from the source dataset is also embedded using the respective audio encoder. We select data with embeddings $\mathbf{E}_\mathrm{A}(\mathrm{A_i})$ closest to the embeddings of the prompts $\mathbf{E}_\mathrm{T}(\mathrm{P_i})$.

The proportion p of the data that should be selected is set as a hyperparameter. The data is balanced by target class by choosing the closest $\frac{p}{C}$ audio examples per class prototype, where C is the number of classes in the target dataset.

Knowledge Distillation

Knowledge distillation is the process of training a neural network to predict the output of another, trained neural network [14]. We train an efficient student model on the output $\boldsymbol{y}$ of the teacher CLAP audio encoder, to obtain an efficient CLAP audio encoder.

Following previous work, we adopt a cosine loss to train a student model on the output $\boldsymbol{y}$ of a CLAP model audio encoder [8]:

$$\mathcal{L}_\mathrm{cos} = -\frac{\hat{\boldsymbol{y}}^\top \boldsymbol{y}}{\|\hat{\boldsymbol{y}}\|\|\boldsymbol{y}\|}. \tag{1}$$

The cosine loss is a natural choice for CLAP models, since the pre-training objective aligns paired data based on the cosine distance.

Fine-Tuning

The student model is fine-tuned on labeled target data to adapt it to the target domain. The last (linear) layer of the model is modified to match the number of classes in the target domain. The parameters of this layer are randomly initialized. A cross-entropy loss ($\mathcal{L}_{CE}$) is used for the training.

Experiments

We present the main experiment showing the novel zero-shot data pruning, compression and fine-tuning method. Following this are several ablations with comparisons to previous work. The details for the setup are given with a description of the teacher model, student architecture and the datasets used.

Teacher model

We employ a pre-trained CLAP model, consisting of an audio and a text encoder, from a recent publication[1] [15]. The embedding space has 512 dimensions. The audio encoder architecture has 31 M trained parameters.

Student architecture

Following previous work [4], we choose the MobileNetV3 [16] architecture for the student model. The size of the architecture is determined by a width multiplier hyperparameter. Throughout the experiments, a value of 2.0 is used, and the number of output dimensions is set to 512 to match the teacher embeddings. This configuration has 16.7 M learnable parameters, which is about half the size of the teacher architecture.

Datasets

The proposed method expects a source dataset from which data is selected. Here, the AudioSet [17] dataset is used for this purpose. We evaluate the method by applying it to the ESC-50 dataset [18] as the target dataset. This dataset consists of recordings curated from Freesound and is labelled using 50 classes of environmental sound types. ESC-50 comes with splits into five folds, for which cross-validation is recommended.

Data Pruning and Compression via CLAP embeddings

Previous work used a different type of embedding for the knowledge distillation. We compare CLAP embeddings to embeddings from a PaSST [5] model ensemble as the distillation target. The first ESC-50 fold is used for validation and held out during data pruning and fine-tuning. The domain classifier and the fine-tuning require labeled training data, for which the remaining data folds are used. The protocol for the fine-tuning follows previous work [4].

The experimental results are shown in Table 1. PaSST-audio uses PaSST audio embeddings and the target audio data for pruning. The same holds for the CLAP-audio pruning. Only CLAP-text pruning uses the proposed method of embedding textual class descriptions without target audio data from the target domain.

Table 1: Validation accuracy (and standard deviation on three fine-tuning runs) on held-out ESC-50 fold 1 for different teacher models. The compressed models are adapted by fine-tuning on labeled audio data.

Pruning	PaSST	CLAP
All Source Data	93.4 (1.0)	93.8 (0.6)
PaSST-audio [4]	95.3 (0.3)	95.3 (0.7)
CLAP-audio	94.6 (0.6)	**95.9** (0.8)
CLAP-text	**96.1** (0.5)	94.5 (0.3)
No Source Data	49.5	

The accuracy improves if the source data is pruned compared to no pruning or not using source data at all. Combining a pruning based on text and distillation with PaSST embeddings gives the best accuracy. The differ-

[1] https://huggingface.co/lukewys/laion_clap/blob/main/630k-audioset-best.pt

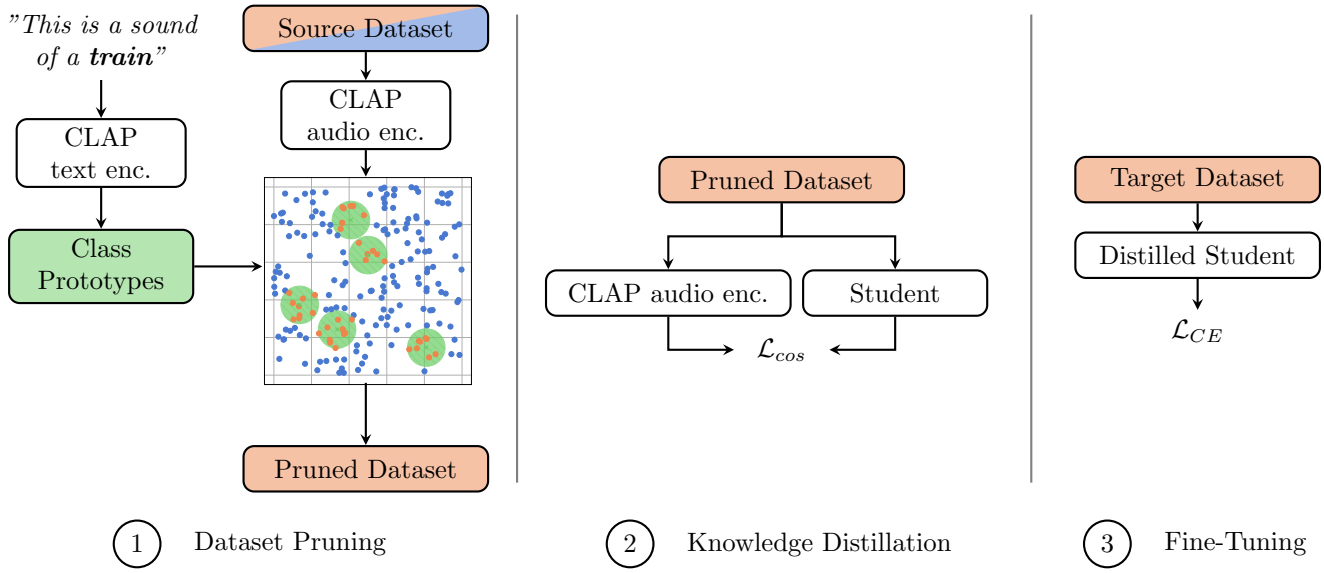

Figure 1: Overview of the proposed method with zero-shot data pruning, knowledge distillation and fine-tuning to the target domain.

ence to the other configurations is small, especially when the standard deviation between runs is considered. Importantly, the proposed method is shown to result in similar or possibly better accuracy, although no target audio data was used for the pruning. Note that the data pruning threshold used for all pruning approaches was tuned for the PaSST embeddings.

Balancing Text-based Pruning

As described above, our data selection using the CLAP text encoder balanced the data by target dataset class. We found that simply using the proportion p of audio embeddings closest to any of the C class prototypes produces highly imbalanced class distributions. Instead, we attempt to balance the data by selecting an equal proportion $\frac{p}{C}$ of closest data per target class as described above. In Table 2, we show how balancing the data is necessary for the data selection to be useful.

Note that this balancing still produces an unbalanced selected dataset. The closest class prototype for a selected audio example is not necessarily the one which got it selected. If the audio embedding density around a class prototype is higher, there is a chance "close" embeddings will not be selected or, if they are, be selected due to a different class prototype.

Table 2: Impact of class-balancing in data pruning stage on accuracy after fine-tuning on ESC-50.

Pruning	Accuracy
All data	93.8 (0.6)
unbalanced CLAP-text	92.2 (0.9)
balanced CLAP-text	**94.5** (0.3)

Table 3: Retrieval measured using mean average precision over the top ten ranked results (mAP@10) for audio to text (A→T) and text to audio (T→A).

Data Pruning	AudioCaps		Clotho	
	T→A	A→T	T→A	A→T
All Source Data	44.4	28.8	21.2	11.2
CLAP-audio	35.6	19.0	21.7	11.2
CLAP-text	38.4	22.7	21.6	10.9
CLAP-teacher	49.2	33.3	24.7	14.5

Retrieval Evaluation on Compressed Models

Audio-to-caption and caption-to-audio retrieval performance are common evaluation metrics for CLAP models. We investigate how much of the (general) capability of the teacher model is preserved in the compressed models. Two datasets are commonly used for retrieval evaluation. The Clotho dataset [19] pairs audio examples with five different captions. The data is sourced from Freesound. Audio tagged as music, sound effects or speech is omitted. Common tags of audio in the dataset include water, nature, birds, and rain. Second, we use the AudioCaps dataset [20], a collection of audio clips sourced from AudioSet with one associated caption each. Audio tagged as music or instruments is not included and the data is sampled to balance classes. The top tags for audio in the dataset include male and female speech as well as different engine sounds.

The data pruning results in worse retrieval only for AudioCaps, the Clotho results are largely equivalent. We hypothesize that this stems from the data composition of these specific datasets in relation to ESC-50, which was the target of the data pruning. For exam-

ple, human speech is not part of ESC-50 and also removed from Clotho, whereas it is common in AudioCaps. The amount of audio with human speech in the pruned dataset is very low, as was shown in previous work [4].

Conclusions

This paper presents a method to compress a CLAP audio encoder model with respect to a target domain. It does so by pruning irrelevant data from a general audio dataset. This pruning is based on textual descriptions of the target classes. We show that this zero-shot data pruning approach results in a performance comparable to a previous supervised approach, which has access to in-domain audio data. In future work, the fine-tuning step of the proposed method could be replaced by a zero-shot classifier based on text, similar to the data pruning. This approach would then be fully zero-shot, as no audio data from the target domain is needed.

Acknowledgement

The project was funded by the Federal Ministry of Education and Research (BMBF) under grant no. 01IS22094E WEST-AI.
The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time through the John von Neumann Institute for Computing (NIC) on the GCS Supercomputer JUWELS[21] at Jülich Supercomputing Centre (JSC).
The authors gratefully acknowledge the funding of this project by computing time provided by the Paderborn Center for Parallel Computing (PC2).

References

[1] Kaplan, J., McCandlish, S., Henighan, T., *et al.*, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[2] Cherti, M., Beaumont, R., Wightman, R., *et al.*, "Reproducible Scaling Laws for Contrastive Language-Image Learning," in *CVPR*. IEEE, June 2023, pp. 2818–2829.

[3] Schmid, F., Koutini, K., and Widmer, G., "Efficient Large-Scale Audio Tagging Via Transformer-to-CNN Knowledge Distillation," in *ICASSP*. IEEE, 2023, pp. 1–5.

[4] Werning, A. and Haeb-Umbach, R., "Target-specific dataset pruning for compression of audio tagging models," in *EUSIPCO*. IEEE, 2024.

[5] Koutini, K., Schlüter, J., Eghbal-zadeh, H., *et al.*, "Efficient Training of Audio Transformers with Patchout," in *Interspeech*. ISCA, 2022, pp. 2753–2757.

[6] Elizalde, B., Deshmukh, S., Al Ismail, M., *et al.*, "Clap learning audio concepts from natural language supervision," in *ICASSP*. IEEE, 2023, pp. 1–5.

[7] Wu, K., Peng, H., Zhou, Z., *et al.*, "Tinyclip: Clip distillation via affinity mimicking and weight inheritance," in *CVPR*. IEEE, 2023, pp. 21 970–21 980.

[8] Paissan, F. and Farella, E., "tinyclap: Distilling constrastive language-audio pretrained models," in *Interspeech*, 2024, pp. 1685–1689.

[9] Li, J., Zhu, L., and Du, Z., *Unsupervised Domain Adaptation - Recent Advances and Future Perspectives*. Springer, 2024. [Online]. Available: https://doi.org/10.1007/978-981-97-1025-6

[10] Yang, S., Ye, P., Ouyang, W., *et al.*, "A clip-powered framework for robust and generalizable data selection," *arXiv preprint arXiv:2410.11215*, 2024.

[11] Paul, M., Ganguli, S., and Dziugaite, G. K., "Deep Learning on a Data Diet: Finding Important Examples Early in Training," in *NeurIPS*, vol. 34. Curran Associates, Inc., 2021, pp. 20 596–20 607.

[12] Sorscher, B., Geirhos, R., Shekhar, S., *et al.*, "Beyond neural scaling laws: beating power law scaling via data pruning," in *NeurIPS*, vol. 35, 2022, pp. 19 523–19 536.

[13] Wang, Z., Subakan, C., Subramani, K., *et al.*, "Unsupervised improvement of audio-text cross-modal representations," in *WASPAA*. IEEE, 2023, pp. 1–5.

[14] Hinton, G., Vinyals, O., and Dean, J., "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[15] Wu, Y., Chen, K., Zhang, T., *et al.*, "Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation," in *ICASSP*. IEEE, 2023, pp. 1–5.

[16] Howard, A., Sandler, M., Chu, G., *et al.*, "Searching for mobilenetv3," in *CVPR*. IEEE, 2019, pp. 1314–1324.

[17] Gemmeke, J. F., Ellis, D. P., Freedman, D., *et al.*, "Audio set: An ontology and human-labeled dataset for audio events," in *ICASSP*. IEEE, 2017, pp. 776–780.

[18] Piczak, K. J., "ESC: Dataset for Environmental Sound Classification," in *Proceedings of the 23rd Annual ACM Conference on Multimedia*. ACM Press, 2015, pp. 1015–1018.

[19] Drossos, K., Lipping, S., and Virtanen, T., "Clotho: An audio captioning dataset," in *ICASSP*. IEEE, 2020, pp. 736–740.

[20] Kim, C. D., Kim, B., Lee, H., *et al.*, "Audiocaps: Generating captions for audios in the wild," in *NAACL: Human Language Technologies, Volume 1*, 2019, pp. 119–132.

[21] Jülich Supercomputing Centre, , "JUWELS Cluster and Booster: Exascale Pathfinder with Modular Supercomputing Architecture at Juelich Supercomputing Centre," *Journal of large-scale research facilities*, vol. 7, no. A138, 2021.