DCASE2024 TASK1 SUBMISSION: DATA-EFFICIENT ACOUSTIC SCENE CLASSIFICATION WITH SELF-SUPERVISED TEACHERS

Technical Report

Yiqiang Cai[1]*, Minyu Lin*[1]*, Shengchen Li*[1]*, Xi Shao*[2]

[1] Xi'an Jiaotong-Liverpool University, School of Advanced Technology, Suzhou, China,
{yiqiang.cai21, minyu.lin20}@student.xjtlu.edu.cn, shengchen.li@xjtlu.edu.cn
[2] Nanjing University of Posts and Telecommunications,
College of Telecommunications and Information Engineering, Nanjing, China,
shaoxi@njupt.edu.cn

ABSTRACT

The task 1 of the DCASE Challenge 2024 focuses on developing low-complexity acoustic scene classification (ASC) systems with limited labeled training data. This technical report details the systems we submitted. We firstly use self-supervised learning (SSL) techniques to pre-train large teacher models on AudioSet. The self-supervised teachers are then fine-tuned on ASC dataset using weight-freezing strategies. Knowledge distillation is employed to transfer the self-supervised knowledge to a low-complexity ASC model. The student model, TF-SepNet-64, is designed to meet the upper complexity limit of the challenge requirements. To mitigate the device shift problem, we used Freq-MixStyle and device impulse response augmentation. In experiments, our best system, trained on 5 given subsets, achieves an average accuracy of 56.6%[1].

Index Terms— Acoustic Scene Classification, data efficiency, low complexity, self-supervised learning, knowledge distillation

1. INTRODUCTION

Acoustic Scene Classification (ASC) [1] is a fundamental task in the field of audio signal processing, aiming to classify audio recordings into predefined scene categories such as streets, parks, or airports. Over the years, the development of ASC has significantly progressed through the annual Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge. The early challenges focused on fundamental classification tasks, but recent editions have included more sophisticated scenarios such as mismatched conditions and low-complexity models [2], promoting robustness and real-world applicability.

Traditionally, ASC models heavily rely on supervised learning approaches [3, 4, 5], which necessitate large amounts of labeled data to achieve high performance. Obtaining such labeled datasets is resource-intensive, both in terms of time and human effort. In task 1 of the DCASE Challenge 2024, participants are required to develop low-complexity acoustic scene classification (ASC) systems with limited training data [6]. Specifically, 5 training subsets are provided, containing 5%, 10%, 25%, 50%, and 100% of the original training set's size. The submitted systems, trained on these 5 subsets, are evaluated by the average accuracy.

[1]Source code: `https://github.com/yqcai888/easy_dcase_task1`

Knowledge distillation has been demonstrated to be effective for the ASC task in previous years [4, 7]. In this report, we explore a data-efficient approach to ASC by incorporating self-supervised learning (SSL) teachers. Specially, we introduce BEATs [8], an audio SSL model, as the teacher models. The teachers are self-supervised pre-trained on AudioSet [9] and then fine-tuned on ASC dataset using different weight-freezing strategies. Remarkably, we find that even pure SSL models with several epochs of fine-tuning can achieve over 50% accuracy with the 5% training subset. For the student model, we continue with the TF-SepNet [10], which achieved second-top ranking in the last year. We propose TF-SepNet-64 by adjusting several components of TF-SepNet to maximize the model parameters and computational complexity. The experiments show that the self-supervised teacher ensembles significantly improve the classification accuracy of student model.

2. DATA PREPROCESSING AND AUGMENTATION

2.1. Dataset

The TAU Urban Acoustic Scene 2022 Mobile development dataset [2] consists of recordings captured using mobile devices in urban environments. The dataset includes 230,350 audio clips, each with a duration of 1 seconds and a hard label of an acoustic scene. There are totally 10 different acoustic scene categories including airport, bus, metro, metro station, park, public square, street pedestrian, street traffic, tram, and urban park. The recordings were captured across several cities around the world and using a wide range of mobile devices. The dataset for the task1 of DCASE2024 Challenge has exactly the same content as the TAU Urban Acoustic Scenes 2022 Mobile development dataset, but the training sets of different sizes are provided. These train subsets contain approximately 5%, 10%, 25%, 50%, and 100% of the audio snippets in the train split provided in previous years. Participants are required to develop ASC systems on specified data subsets.

2.2. Feature Extraction

For TF-SepNet-64 [10], we generally follow the baseline settings [11] for feature extraction. The audio recordings are firstly resampled to 32 kHz. Time-frequency representations are then extracted using a 4096-point FFT with a window size of 96 ms and a hop size of 16 ms. The primary difference in our approach is the appli-

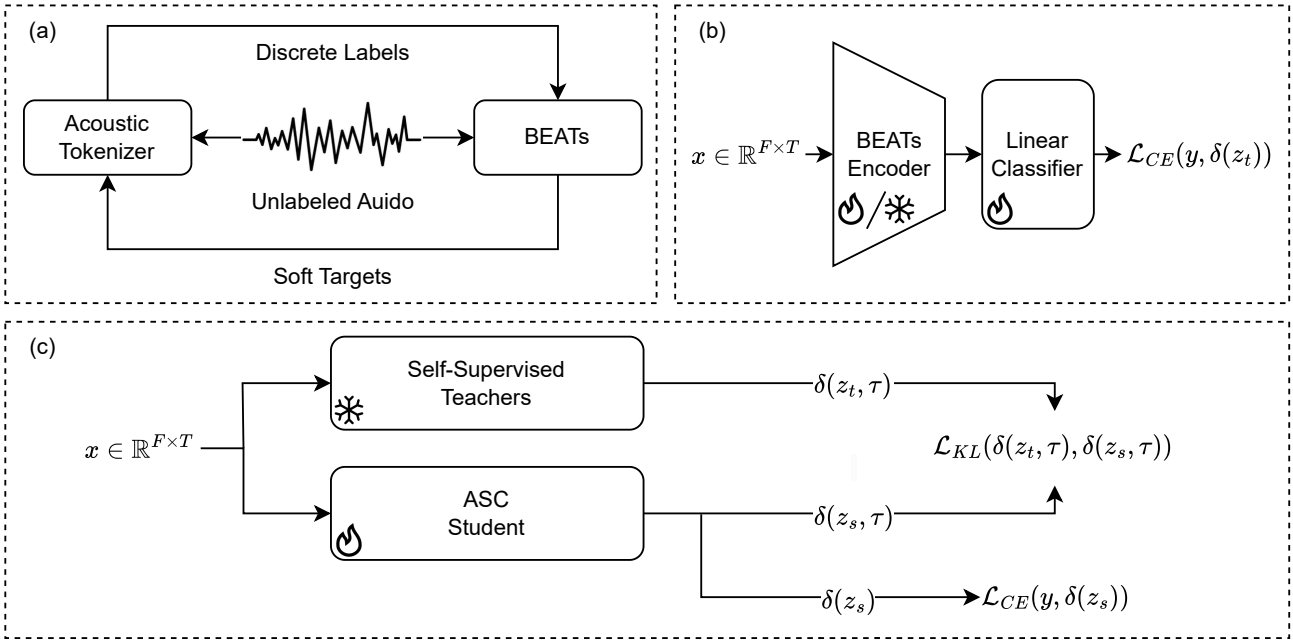

Figure 1: Knowledge distillation with self-supervised teachers. **(a)** Self-supervised pre-training teachers on AudioSet. **(b)** Fine-tuning teachers on ASC dataset. **(c)** Distilling knowledge from self-supervised teachers to low-complexity ASC student. **Snowflake** icon indicates that the parameters of the corresponding part are frozen, while **flame** icon indicates the opposite.

cation of a Mel-scaled filter bank with a large number of frequency bins, 512, to convert the spectrograms into mel spectrograms, which leads to a slight improvement on the classification accuracy. The final input size for TF-SepNet is (512, 64).

As for BEATs [8], we use the default configuration in the original work. Each raw waveform is resampled to 16 kHz and extract 128-dimensional Mel-filter bank features using a 25 ms Povey window with a 10 ms shift. The features are normalized according to the mean and standard deviation of AudioSet [9]. Each acoustic feature is then divided into 16 × 16 patches and flattened into a sequence of patches to serve as input for the BEATs.

2.3. Data Augmentations

Data augmentation is a crucial technique in ASC tasks, especially when the labeled data is limited. In our approach, we use a combination of Soft Mixup, Freq-MixStyle [7], and Device Impulse Response (DIR) augmentation [12] to enhance the diversity and quality of our training data. All augmentations are implemented to be plug-and-play during training.

- **Soft Mixup** is the adjusted version of Mixup [13] for mixing both the ground truth labels and the soft labels of teachers. Mixup generates a new training sample by linearly interpolating two random examples and their corresponding labels. Specially, given two examples $(x_i, y_i, \tilde{y}_i)$ and $(x_j, y_j, \tilde{y}_j)$, where x is the input feature, y is the ground truth label, and $\tilde{y}$ is the teacher logits, Soft Mixup generates a new sample as follows:

$$x_{new} = \lambda x_i + (1 - \lambda) x_j \quad (1)$$

$$y_{new} = \lambda y_i + (1 - \lambda) y_j \quad (2)$$

$$\tilde{y}_{new} = \lambda \tilde{y}_i + (1 - \lambda) \tilde{y}_j \quad (3)$$

- **Freq-MixStyle** is the frequency-wise version of MixStyle [14]. It normalizes the frequency bands instead of channels, which has been demonstrated effective for mitigating the device shift problem in ASC tasks.
- **Device Impulse Response (DIR) Augmentation** aims to simulate recordings from one device to other devices. The raw waveform is convolved with a device impulse response randomly selected from the MicIRP[2]. The probability of application is controlled by a hyperparameter p_{dir}.

3. KNOWLEDGE DISTILLATION WITH SELF-SUPERVISED TEACHERS

Knowledge distillation is a technique used to transfer knowledge from a large, complex model (teacher) to a smaller, more efficient model (student) [15]. In this work, we extend the concept of knowledge distillation by employing self-supervised learning (SSL) to pre-train the teacher models, thus reducing the dependency on labeled data required for ASC tasks. The framework of knowledge distillation with self-supervised teachers is shown in Figure 1.

3.1. Self-Supervised Teachers: BEATs

Self-supervised learning enables models to learn representations from unlabeled data by solving pretext tasks. In this work, we use the BEATs (Bidirectional Encoder Representations from Audio Transformers) [8] as our self-supervised teachers. BEATs is an SSL framework specifically designed for audio tasks. As shown in Figure 1 (a), we follow the default configuration in [8] to SSL pre-train the teacher models on the allowed large external dataset, Au-

[2]https://micirp.blogspot.com/

System	5%	10%	25%	50%	100%	Avg.
Baseline	42.4	45.3	50.3	53.2	57.0	49.6
BEATs(SSL)*	50.7	51.8	54.1	54.9	55.8	53.5
BEATs(SSL)	52.8	54.5	58.1	59.5	61.2	57.2
BEATs(SSL+SL)	**54.3**	**56.6**	**59.7**	**60.4**	**62.1**	58.6
5 Ensemble	**55.7**	57.7	61.2	62.5	**64.3**	60.3
10 Ensemble	55.6	**57.9**	**61.4**	**62.6**	64.2	60.3

Table 1: Accuracy of fine-tuned BEATs on the test set of TAU Urban Acoustic Scene 2022 Mobile development dataset [2]. **SSL** indicates the BEATs are self-supervised pre-trained on AudioSet. **SL** denotes the BEATs are additionally supervised fine-tuned on AudioSet. * represents the encoder of BEATs is frozen during the fine-tuning on ASC dataset. Top-1 accuracy is presented.

dioSet [9]. As shown in Figure 1 (b), the SSL pre-trained BEATs are then fine-tuned on the labeled ASC dataset, with the encoder either frozen or unfrozen. If the encoder is frozen, only the linear classifier is trained. Additionally, we also test the SSL+SL pre-trained BEATs, where the SSL pre-trained BEATs are further supervised fine-tuned on the AudioSet before fine-tuned on the ASC dataset.

As shown is Table 1, the SSL pre-trained BEATs with frozen encoder achieve more than 50% accuracy on 5% training subset, which outperforms the fully supervised baseline system by 8.3% in accuracy. The SSL pre-trained BEATs without freezing naturally get a higher accuracy and the SSL+SL pre-trained BEATs obtain the best performance. We also test the ensemble models followed the same configurations in [8] and get remarkable improvements.

3.2. Student: TF-SepNet-64

The Time-Frequency Separate Network (TF-SepNet) [10] is a deep CNN architecture designed specifically for low-complexity ASC tasks, achieving second place in last year's competition. TF-SepNet processes features separately along the time and frequency dimensions using one-dimensional (1D) kernels, which reduces computational costs. The separate kernels also provide a larger effective receptive field (ERF), allowing the model to capture more time-frequency features.

To optimize model complexity, we have made several adjustments for TF-SepNet-64, as illustrated in Table 3.2. First, the number of base channels is set to 64. Second, all Adaptive Residual Normalization layers [5] are replaced with Residual Normalization layers [3] to reduce the number of model parameters. Third, a Max-pooling layer is added before the last TF-SepConvs block to further reduce the feature size. In the finish, the total parameter number of TF-SepNet-64 is 126,858. For an input feature size of (512, 64), the maximum number of MACs per inference is 29.4196 MMACs.

3.3. Knowledge Distillation

We adopt the widely used knowledge distillation framework in previous years' challenges [4, 7], which focuses on directly mimicking the final predictions of the teacher model. As illustrated in Figure 1 (c), the knowledge transfer involves two main steps.

The input feature is a log-mel spectrogram $x \in \mathbb{R}^{F \times T}$. For the teacher path, once the self-supervised teachers are fine-tuned, as shown in Figure 1 (b), the predictions on a specified training subset are computed, serving as the teacher logits in the knowledge distillation process. For the student path, the ASC student is trained

Output Shape	Architecture	k	s	p
$1, F, T$	Input	-	-	-
$C/2, F/2, T/2$	ConvBnRelu	3	2	1
$2C, F/4, T/4$	ConvBnRelu, g=$C/2$	3	2	1
$C, F/4, T/4$	TF-SepCovs $\times 2$	-	-	-
$C, F/8, T/8$	MaxPool	2	2	0
$1.5C, F/8, T/8$	TF-SepCovs $\times 2$	-	-	-
$1.5C, F/16, T/16$	MaxPool	2	2	0
$2C, F/16, T/16$	TF-SepCovs $\times 2$	-	-	-
$2C, F/32, T/32$	MaxPool	2	2	0
$2.5C, F/32, T/32$	TF-SepCovs $\times 3$	-	-	-
$10, F/32, T/32$	Conv	1	1	0
$10, 1, 1$	Avgpool	-	-	-

Table 2: Architecture of the adjusted TF-SepNet-64 [10]. C, F, and T respectively represent channels, frequency bins, and time clips of feature maps. k, s, p and g separately denote kernel size, stride, padding and group. The number of base channels is set to 64. The number of parameters is 126,858 and the computational overheads per inference is 29.4196 MMACs.

on the specified training subset using a combination of the ground truth labels and the soft targets provided by the teacher model. Give a vector of logits z as the outputs of the last classification layer of a model, the soft targets are the probabilities that the input belongs to the classes and can be estimated by a softmax function $\delta(\cdot)$ as

$$\delta(z_i, \tau) = \frac{\exp(z_i/\tau)}{\sum_j \exp(z_j/\tau)} \tag{4}$$

where z_i is the logit for the i-th class, and a temperature factor τ is introduced to control the importance of each soft target. The training objective of student model is to minimize the divergence between the student's predictions and the soft targets from the teacher, as well as to correctly classify the labeled data. The overall loss function for the student can be formulated as

$$\mathcal{L} = \lambda \mathcal{L}_{CE}(y, \delta(z_s)) + (1-\lambda)\tau^2 \mathcal{L}_{KL}(\delta(z_t, \tau), \delta(z_s, \tau)) \tag{5}$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss between the ground truth labels and the student's predictions, and $\mathcal{L}_{KL}$ is the Kullback-Leibler divergence between the soft targets from the teacher and the student's predictions. λ is a hyperparameter to balance the weight between label and distillation loss.

4. TRAINING SETUP

We train TF-SepNet-64 for 150 epoch using Adam optimizer with different initial learning rate for 5 subsets, 0.06 for split5, 0.05 for split50 and 0.04 for all other splits. Stochastic Gradient Descent with Warm Restarts (SGDR) [16] is applied with T_0 =10 and T_{mult} = 2, where the learning rate is periodically reset to initial value and then decayed with cosine annealing. The batch size is set to 512. α of Mixup [13] is set to 0.3. α and p of Freq-MixStyle [7] are respectively set to 0.4 and 0.8. p_{dir} of DIR augmentation [12] is set to 0.4. We fix $\lambda = 0.02$ and $\tau = 2$ for the knowledge distillation as in [4]. After training, Post-Training Static Quantization is implemented through the Intel Neural Compressor[3] to quantize the weights of model into INT8 data type.

[3]https://intel.github.io/neural-compressor

System	t	5%	10%	25%	50%	100%	Avg.
lr	-	0.006	0.004	0.004	0.005	0.004	-
Vanilla	0	44.7	50.0	54.9	58.7	62.3	54.1
S1	1	47.4	52.1	57.5	**61.1**	62.4	56.1
S2	3	**49.0**	52.3	**57.9**	60.7	**62.9**	**56.6**
S3	12	47.9	**52.3**	57.5	60.1	62.8	56.1

Table 3: Accuracy of submitted systems on the test set of TAU Urban Acoustic Scene 2022 Mobile development dataset [2]. S1-S3 indicates the submitted system 1 to system 3. The difference between submitted systems is the number of ensemble teachers, denoted as t. Different initial learning rates lr are applied for 5 subsets. Top-1 and quantized accuracy is presented.

5. SUBMISSION

The submission systems are presented in Table 3. We submitted a total of three systems, each with the same complexity. The systems differs in the number ensemble teachers t, used in the knowledge distillation process. "Vanilla" is a vanilla TF-SepNet-64 used for comparison, where knowledge distillation is not applied during training. Nevertheless, it still outperforms the baseline by 4.5% in average accuracy. For system 1, the teacher logits come from a single SSL+SL pre-trained BEATs. The teacher logits of system 2 consist of an ensemble of a SSL*, a SSL and a SSL+SL pre-trained BEATs. The teacher logits of system 3 include an ensemble of a SSL*, a SSL and 10 SSL+SL pre-trained BEATs. System 2 achieves the best performance, with an average accuracy of 56.6%.

6. CONCLUSION

In this report, we introduce self-supervised learning (SSL) techniques to address the challenge of data-efficient low-complexity acoustic scene classification (ASC). We pre-train BEATs on AudioSet as self-supervised teachers and then transfer knowledge to the low-complexity student, TF-SepNet-64, through a knowledge distillation framework. The experimental results demonstrate the effectiveness of self-supervised teachers in reducing the dependency on labeled data, providing a pathway for developing robust and efficient ASC systems with limited labeled data.

7. ACKNOWLEDGEMENT

This project is supported by the Gusu Innovation and Entrepreneurship Leading Talents Programme (No: ZXL2022472).

8. REFERENCES

[1] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, "Acoustic scene classification: Classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 16–34, 2015.

[2] T. Heittola, A. Mesaros, and T. Virtanen, "Acoustic scene classification in dcase 2020 challenge: generalization across devices and low complexity solutions," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*, 2020, pp. 56–60.

[3] B. Kim, S. Yang, J. Kim, and S. Chang, "QTI submission to DCASE 2021: Residual normalization for device-imbalanced acoustic scene classification with efficient design," DCASE2021 Challenge, Tech. Rep., June 2021.

[4] F. Schmid, T. Morocutti, S. Masoudian, K. Koutini, and G. Widmer, "CP-JKU submission to dcase23: Efficient acoustic scene classification with cp-mobile," DCASE2023 Challenge, Tech. Rep., May 2023.

[5] Y. Cai, M. Lin, C. Zhu, S. Li, and X. Shao, "Dcase2023 task1 submission: Device simulation and time-frequency separable convolution for acoustic scene classification," DCASE2023 Challenge, Tech. Rep., May 2023.

[6] F. Schmid, P. Primus, T. Heittola, A. Mesaros, I. Martín-Morató, K. Koutini, and G. Widmer, "Data-efficient low-complexity acoustic scene classification in the dcase 2024 challenge," *arXiv preprint arXiv:2405.10018*, 2024.

[7] F. Schmid, S. Masoudian, K. Koutini, and G. Widmer, "CP-JKU submission to dcase22: Distilling knowledge for low-complexity convolutional neural networks from a patchout audio transformer," DCASE2022 Challenge, Tech. Rep., June 2022.

[8] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, W. Che, X. Yu, and F. Wei, "BEATs: Audio pre-training with acoustic tokenizers," in *Proceedings of the 40th International Conference on Machine Learning (ICML)*. PMLR, 23–29 Jul 2023, pp. 5178–5193.

[9] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[10] Y. Cai, P. Zhang, and S. Li, "Tf-sepnet: An efficient 1d kernel design in cnns for low-complexity acoustic scene classification," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 821–825.

[11] F. Schmid, T. Morocutti, S. Masoudian, K. Koutini, and G. Widmer, "Distilling the knowledge of transformers and CNNs with CP-mobile," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2023 Workshop (DCASE2023)*, 2023, pp. 161–165.

[12] T. Morocutti, F. Schmid, K. Koutini, and G. Widmer, "Device-robust acoustic scene classification via impulse response augmentation," in *2023 31st European Signal Processing Conference (EUSIPCO)*. IEEE, 2023, pp. 176–180.

[13] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[14] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain generalization with mixstyle," in *International Conference on Learning Representations (ICLR)*, 2021.

[15] 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.

[16] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.