SEMI-SUPERVISED ACOUSTIC SCENE CLASSIFICATION WITH TEST-TIME ADAPTATION

Wen Huang[1], Anbai Jiang[2], Bing Han[1], Pingyi Fan[2], Yanmin Qian[1]

[1]Auditory Cognition and Computational Acoustics Lab
Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China
[2]Department of Electronic Engineering, Tsinghua University, Beijing, China

ABSTRACT

This technical report presents our submission for the ICME 2024 Grand Challenge titled "Semi-supervised Acoustic Scene Classification under Domain Shift". Acoustic Scene Classification (ASC) plays a crucial role in audio signal processing, with applications ranging from urban soundscapes to smart homes. However, challenges like domain shift and scarce labeled data hinder its development, highlighting the need for semi-supervised learning strategies. Our submission outlines a semi-supervised ASC system that employs pretraining on available datasets, followed by finetuning through FixMatch and pseudo-labeling, and concludes with test-time adaptation. This approach seeks to effectively utilize unlabeled data and mitigate domain shift, ultimately enhancing the ASC system's performance.

Index Terms— acoustic scene classification, semi-supervised learning, domain shift

1. INTRODUCTION

Acoustic Scene Classification (ASC) is a pivotal task in the realm of audio signal processing, aiming to categorize audio recordings into predefined scenes based on their acoustic characteristics. This technology underpins numerous applications, from enhancing urban soundscapes to advancing smart home devices, making its development a focal point for researchers and technologists alike.

However, as ASC research advances, it confronts significant obstacles. Challenges such as domain shift significantly influence ASC, where discrepancies in acoustic properties between training and testing scenarios can degrade model performance. Additionally, the scarcity of labeled data presents a hurdle for supervised learning methods, pushing the need for semi-supervised techniques that tap into the wealth of unlabeled audio data.

These challenges also form the core of the "Semi-supervised Acoustic Scene Classification under Domain Shift" challenge [1]. Within this context, the challenge provides a development dataset from the CAS 2023 collection, featuring 4.8 hours of labeled and 19.3 hours of unlabeled data. Besides, a notable domain shift exists in the evaluation dataset, including recordings from cities not covered in the development data.

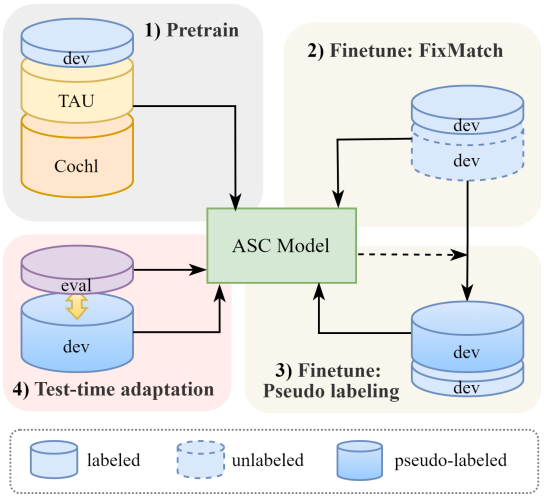

Fig. 1. The pipeline of our system

To navigate these obstacles, we propose a semi-supervised ASC system. Figure 1 illustrates our method, which unfolds in four steps. Initially, we pretrain the model using a variety of available datasets. Next, we finetune it on the challenge development dataset employing the FixMatch [2] strategy. As the model acquires knowledge and becomes accustomed to the development set, we generate pseudo labels for the remaining unlabeled data and further finetune the model using these labels. Lastly, in the testing phase, since the evaluation set cannot be used for training, we opt for a test-time adaptation [3] method to mitigate the domain shift between the development and evaluation sets. In the following sections, we will delve into the details of each step.

2. DATA PREPROCESSING AND AUGMENTATION

2.1. Datasets

In compliance with the challenge rules, apart from the ASC challenge development dataset [1], we utilize the TAU Urban Acoustic Scenes (UAS) 2020 Mobile development dataset [4] and the CochlScene dataset [5] for model pretraining. These are the only two additional datasets permitted for use in this challenge.

TAU UAS. The TAU Urban Acoustic Scenes 2020 Mobile dataset [4] features 64 hours of recordings from various European cities across ten acoustic scenes, captured simultaneously using four devices (A, B, C, and D). Additionally, it includes synthetic recordings from devices S1-S11, created by simulating audio from device A, a high-quality binaural recorder, to enhance the dataset's diversity.

CochlScene. The Cochl Acoustic Scene Dataset [5], also known as CochlScene, is an acoustic scene dataset with recordings entirely sourced from crowdsourcing participants in Korea. By selecting a subset pertinent to Acoustic Scene Classification (ASC) from the full collection, it has 76,115 ten-second audio files across 13 different acoustic scenes, contributed by 831 participants.

2.2. Feature extraction

In our data preprocessing pipeline, we standardize audio files to a sample rate of 44,100 Hz. The process involves generating spectrograms using a Hann window of 1024 with a 320 hop size and an FFT window of 2048. These spectrograms are then converted into log-Mel spectrograms with 64 Mel bins, ranging from 10 Hz to half the sample rate.

2.3. Data augmentation

For model training, we primarily employ three data augmentation techniques: SpecAugment [6], Mixup [7] and Freq-MixStyle [8, 9].

SpecAugment. SpecAugment [6] was initially crafted for speech data improvement, and can also enhances audio by applying frequency and time masking to log mel spectrograms. It randomly hides frequency bins and time segments, thereby increasing model robustness to frequency and temporal variations. This dual-masking approach effectively guards against audio distortions.

Mixup. Mixup [7] creates new dataset entries by blending the inputs and targets of two audio clips. Given two audio inputs x_1 and x_2 with their corresponding targets y_1 and y_2, the augmented input x and the target y are formed as $x = \lambda x_1 + (1 - \lambda)x_2$ and $y = \lambda y_1 + (1 - \lambda)y_2$, with λ being drawn from a Beta distribution. Typically, this technique is applied to the log mel spectrogram of the audio clips from one batch.

Freq-MixStyle. Freq-MixStyle (FMS) [8, 9] is an adaptation of the original MixStyle [10] concept but tailored for frequency. It first normalizes the frequency bands within a spectrogram, then reintroduces variability by denormalizing them using the combined frequency statistics from two different spectrograms. The application of FMS to any given batch occurs with a probability determined by the hyperparameter p_{FMS}, with mixing coefficients drawn from a Beta distribution shaped by α.

3. PRETRAINING ASC MODEL

3.1. Network architecture

Our ASC model employs the CNN10 configuration from PANNs [11], adapted for the audio tagging task. This architecture consists of 10 layers, including 4 convolutional blocks. Each block contains 2 convolutional layers with 3x3 kernels. Batch normalization is incorporated between convolutional layers to enhance training efficiency and stability, along with the ReLU activation function. For downsampling, average pooling with a 2x2 kernel size is applied within each convolutional block. The model consists of 6.037M parameters in total.

3.2. Training strategy

To train the ASC model, we use data from the challenge development set, TAU, and CochlScene. These datasets vary in both classes and quantities, requiring us to reorganize them. We combine identical classes from each dataset and introduce new ones, resulting in a total of 20 classes. To ensure each dataset contributes equally, we apply weighted sampling for data from the three datasets.

However, the number of audio clips still varies among different scene classes. To address this, we adopt a strategy that samples audio clips from all sound classes equally for each minibatch.

For additional robustness, our training includes data augmentations like SpecAugment, Mixup, and Freq-MixStyle, improving the model's performance across various acoustic scenes. The model is trained using binary cross entropy loss and optimized by Adam optimizer.

4. TWO-STAGE FINETUNING

After pretraining, we finetune the model on the challenge development dataset in two stages. In the first stage, we use FixMatch, a semi-supervised algorithm, to finetune the model with both labeled and unlabeled data. In the second stage, we generate pseudo labels for all unlabeled data using the stage 1 model. Then, we finetune the model further using either labeled data or data with these pseudo labels.

4.1. Stage 1: FixMatch

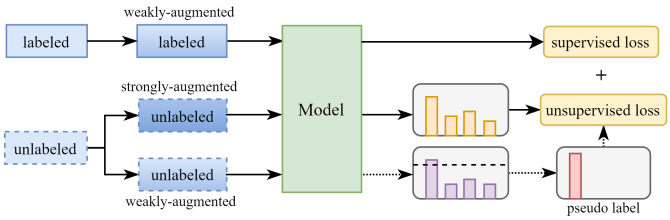

Fig. 2. Diagram of FixMatch

Figure 2 illustrates that during each training step of FixMatch [2], every batch contains a mix of labeled and unlabeled data, leading to the calculation of two types of cross-entropy loss: supervised L_s and unsupervised L_u. The supervised loss L_s calculates the standard cross-entropy on weakly augmented labeled data. For the unsupervised loss L_u, we first determine the model's predicted class from the weakly-augmented unlabeled data. After selecting a confidence threshold, we use the chosen predictions as pseudo labels to calculate the cross-entropy loss for the strongly-augmented unlabeled data. In our system, SpecAugment serves as the weak augmentation. For strong augmentation, we enhance SpecAugment with an additional technique, Freq-MixStyle.

4.2. Stage 2: Pseudo Labeling

Following stage 1 training, we posit that the model can accurately predict labels for unlabeled data. Hence, we use the stage 1 model to create pseudo labels for the remaining unlabeled training data and then proceed to finetune the model with this newly labeled data. During this stage, we employ the same strong augmentation used in stage 1 for all data.

5. TEST-TIME ADAPTATION

A test-time adaptation method [3] based on k-nearest neighbor (KNN) is adopted to bridge the gap between the development and the evaluation sets. The embeddings of all labeled samples of the development set are pre-extracted to form a memory bank for KNN. During inference, the embedding of each query sample is compared with the memory bank via cosine similarity, and the distances to top-k neighbors are utilized as the scoring coefficient. Specifically, let $\mathcal{M}_L$ denote the set of embeddings of all labeled samples in the development set. For each query embedding x_i, we search $\mathcal{M}_L$ for a subset of top-k neighbors $N_{\mathcal{M}_L}(x_i)$ by means of cosine similarity:

$$w_{ij} = \frac{x_i^T x_j}{\|x_i\|_2 \|x_j\|_2} \tag{1}$$

Then the final prediction of the model can be given by:

$$\eta(x_i) = \text{Softmax}\left(\sum_{x_j \in N_{\mathcal{M}_L}(x_i)} w_{ij} \mathbf{1}\{y_j\}\right) \tag{2}$$

where y_j is the label of x_j, and $\mathbf{1}\{y_j\}$ denote the one-hot vector of x_j. It is noted that the adopted method neither finetunes the model on the evaluation set, nor utilizes the statistics of the evaluation set, which is in compliance with the challenge rules.

6. CONCLUSION

In this technical report, we describe our submission to ICME 2024 Grand Challenge "Semi-supervised Acoustic Scene Classification under Domain Shift". To overcome the domain shift and label scarcity challeges, we develop a semi-supervised ASC system. Our methodology involved pretraining on various datasets, finetuning with FixMatch, generating pseudo labels for further refinement, and employing test-time adaptation to alleviate the domain shift for evaluation.

7. REFERENCES

[1] Jisheng Bai, Mou Wang, Haohe Liu, Han Yin, Yafei Jia, Siwei Huang, Yutong Du, Dongzhe Zhang, Mark D Plumbley, Dongyuan Shi, et al., "Description on ieee icme 2024 grand challenge: Semi-supervised acoustic scene classification under domain shift," *arXiv preprint arXiv:2402.02694*, 2024.

[2] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *Advances in neural information processing systems*, vol. 33, pp. 596–608, 2020.

[3] Yifan Zhang, Xue Wang, Kexin Jin, Kun Yuan, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan, "Adanpc: Exploring non-parametric classifier for test-time adaptation," in *International Conference on Machine Learning*. PMLR, 2023, pp. 41647–41676.

[4] Toni Heittola, Annamaria Mesaros, and Tuomas Virtanen, "Acoustic scene classification in dcase 2020 challenge: generalization across devices and low complexity solutions," *arXiv preprint arXiv:2005.14623*, 2020.

[5] Il-Young Jeong and Jeongsoo Park, "Cochlscene: Acquisition of acoustic scene data using crowdsourcing," in *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2022, pp. 17–21.

[6] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le, "Specaugment: A simple data augmentation method

for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.

[7] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[8] Florian Schmid, Shahed Masoudian, Khaled Koutini, and Gerhard Widmer, "Knowledge distillation from transformers for low-complexity acoustic scene classification.," in *DCASE*, 2022.

[9] Byeonggeun Kim, Seunghan Yang, Jangho Kim, Hyunsin Park, Juntae Lee, and Simyung Chang, "Domain generalization with relaxed instance frequency-wise normalization for multi-device acoustic scene classification," *arXiv preprint arXiv:2206.12513*, 2022.

[10] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang, "Domain generalization with mixstyle," *arXiv preprint arXiv:2104.02008*, 2021.

[11] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.