UNSUPERVISED ANOMALOUS DETECTION BASED ON UNSUPERVISED PRETRAINED MODELS

Technical Report

Zhiqiang Lv[1]*, Bing Han*[2]*, Zhengyang Chen*[2]*, Yanmin Qian*[2]*, Jiawei Ding*[1]*, Jia Liu*[1]

[1] Huakong AI Plus Company Limited, Beijing, China
[2] Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China
lvzhiqiang@aithu.com, {hanbing97, zhengyang.chen, yanminqian}@sjtu.edu.cn,
{dingjiawei, liujia}@aithu.com

ABSTRACT

Unsupervised pretrained models have been widely applied in lots of scenarios successfully. DCASE 2023 challenge Task2 is about first-shot unsupervised anomalous sound detection. To solve this problem, we tried to use several unsupervised pretrained models trained on thousands hours of speech. By fine-tuning pretrained big models with datasets of DCASE 2023 challenge Task2, we found that pretrained models outperformed small models trained from scratch. Our best pretrained model achieve hmean of 66.56% on the development dataset, which is much better than the auto-encoder baseline.

***Index Terms*—** Unsupervised pretrained models, first-shot, unsupervised anomalous sound detection

1. INTRODUCTION

The task of anomalous sound detection (ASD) involves determining whether the sound produced by a specific machine is classified as normal or anomalous. The task 2 of DCASE 2023 [1] Challenge focuses on identifying abnormal states of the target machine through the analysis of sounding data. In contrast to acoustic scene classification, this task is unsupervised learning scenario because the training data only consists solely of samples from the normal-state class. However, the goal is to determine whether a test sample belongs to another class known as the anomaly class, which encompasses various anomalous situations. In practical scenarios, changes in a machine's operational states or environmental noise can lead to domain shifts. Participants are also required to use domain generalization techniques to address domain shifts, where the distributions of the training and test data are different.

This year, the key challenge of this task is the "first shot" problem. In practical scenarios, it is difficult for us to collect data and train models on a new machine, or the number of machines is very small. So the main differences in the task of this year are that:

- The development dataset and evaluation dataset don't have the same machine type.
- There is only one section for each machine type.

To solve this problem, we try to find the audio encoders with generalization ability to avoid overfitting on a small amount of training data. The pretrained models on large-scale audio data meets our needs. In recent years, pre-trained models have emerged as the dominant approach for achieving state-of-the-art performance in various natural language processing (NLP) tasks. Building upon the groundbreaking achievements of models like BERT [2] and GPT [3], researchers in the speech community have introduced several innovative approaches, such as wav2vec 2.0 [4], HuBERT [5], Unispeech [6] and WavLM [7], which harness large-scale unlabeled data. These methods have yielded impressive results in automatic speech recognition (ASR) tasks, capitalizing on the power of pre-training and demonstrating the potential of leveraging vast amounts of unlabeled data in the speech domain.

Inspired by the excellent performance of the pre trained model in various generalization tasks [8, 9], we adopt several pretrained models to anomalous sound detection task for "first shot" generalization performance. In order to solve the problem of poor data diversity of a single machine, we also use "speed perturb" to augment the data which is firstly proposed in automatic speech recognition [10]. Finally, in order to obtain a more stable system, we also use transformer pooling method for subsystems fusion. In summary:

- We utilize several unsupervised pre-trained models for general performance.
- In addition, we propose data augmentation method named "speed perturb" to simulate different operation status of machine.
- we use transformer pooling methods to ensemble several subsystems.

In Section 2, we provide an overview of the unsupervised pre-trained models utilized in our systems. Following that, in Section 3, we present a detailed description of our developed system, outlining each subsystem it comprises. For each subsystem, we elaborate on its training process and the methods employed to update its hyperparameters. Moving forward, in Section 4, we showcase our detection results. Finally, in Section 5, we draw conclusions based on our report.

2. PRE-TRAINED MODELS

In this section, we give a brief introduction of four unsupervised pretrained models we used in our systems.

2.1. Wav2Vec 2.0

Wav2vec 2.0 [4] is a continuation of the wav2vec [11] series. It replace the original architecture's convolutional context network with multi-layers transformer based encoder. While wav2vec 2.0 incorporates discrete speech units and a quantization module similar to the vq-wav2vec [12] model, it reverts to the original contrastive objective used in the first version of wav2vec instead of adopting BERT's masked language modeling objective. It's noted that we use the scale-up XLS-R version, which uses 300M parameters and is trained on half a million hours of speech in 128 different languages.

2.2. UniSpeech

UniSpeech [6] presents a multi-task model that integrates a self-supervised learning objective, similar to wav2vec 2.0, with a supervised ASR objective using Connectionist temporal classification. This combined approach enables enhanced alignment between discrete speech units and the phonetic structure of the audio, resulting in improved performance in multi-lingual speech recognition and audio domain transfer tasks.

2.3. HuBERT

HuBERT [5] utilizes the architecture of wav2vec 2.0 while substituting the contrastive objective with BERT's original masked language modeling objective. To achieve this, the model undergoes a pre-training process that involves two steps. In the clustering step, short segments of speech are assigned pseudo-labels, and in the prediction step, the model is trained to predict these pseudo-labels at randomly-masked positions within the original audio sequence. This approach enables the utilization of BERT's objective within the wav2vec 2.0 architecture.

2.4. WavLM

WavLM [7] models follow the HuBERT framework while focusing on data-augmentation during the pre-training stage to improve speaker representation learning and speaker-related downstream tasks. The WavLM model is especially efficient for downstream tasks, it is currently leading the SUPERB leaderboard [13], a performance benchmark for re-using speech representations in a variety of tasks such as automatic speech recognition, phoneme recognition, speaker identification, emotion recognition.

3. APPROACHES

3.1. Speed Perturbation

Speed perturbation data augmentation is a technique commonly used in the field of automatic speech recognition (ASR) [10] and speaker verification (SV) [14] to improve the robustness and generalization of systems. Inspired by these, we apply speed perturbation method for augmenting the different running conditions of machines.

During the speed perturbation process, the original speech signal is modified by stretching or compressing its duration while maintaining the original pitch. This can be achieved by resampling the signal at a different rate or adjusting the playback speed. By altering the speed, the operation status of the machines are modified, resulting in a diverse set of augmented data.

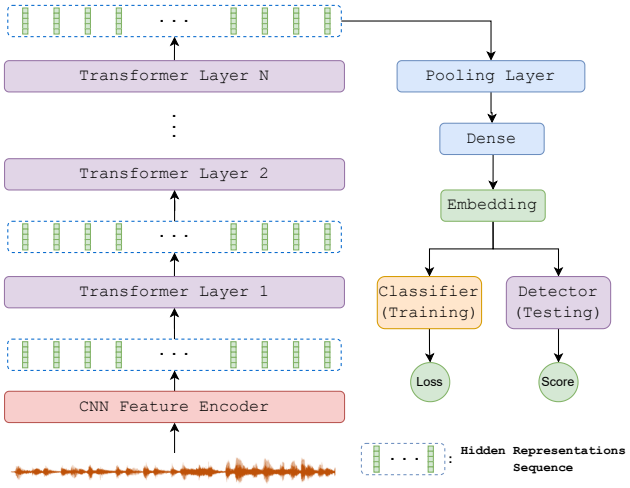

Figure 1: Overview of the pretrained models based classification system. Rawwav audio is encoded into sequence representation by transformer layers. Then, a pooling layer here to aggregate the temporal information into segment-level audio embeddings. A classfier is applied for finetuning the pretrained models on machine dataset. After training, a outlier detector is used to provide the anomaly score based on embedding distribution.

3.2. Classification with pre-trained models

The general idea of classification with pretrained models is finetuning the pretrained models with classification objective function to extract the embeddings of the samples by classifying labels extracted from the metadata.

The overview is shown in Figure 1 and it can be divided into three stages. First, we finetune the pretrained models on training data. The input feature of pretrained model is waveform. After several convolution and transformer layers, the wavform are encoded into a sequence representation. Then, the sequence will be aggregated by a pooling layer for chunk-level audio embedding. In our system, the network is optimized to predict the attributes ID from meta data using arcmargin softmax loss [15] as Equation 1. Compare with tradition softmax loss, it can explicitly enforce the similarity for intra-class samples and the diversity for inter-class samples.

$$L_{AAM} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s(cos(\theta_{y_i,i}+m))}}{Z} \quad (1)$$

where $Z = e^{s(cos(\theta_{y_i,i}+m))} + \sum_{j=1,j\neq i}^{c} e^{s(cos(\theta_{j,i}))}$, $\theta_{j,i}$ is the angle between the column vector $\mathbf{W_j}$ and embedding $\mathbf{x_i}$, where both $\mathbf{W_j}$ and $\mathbf{x_i}$ are normalized. s is a scaling factor and m is a hyperparameter to control the margin.

Second, after the pretrained models achieve the coverage in training data, we use pretrained network to extract the embeddings of training set to get the distribution of normal machine data. These data can be used to train the outlier detector. For the outlier detection algorithm, we tried several well known algorithms such as k-NN [16], LOF [17], cosine distance and Mahalanobis distance. This task is "first shot" challenge, and we cannot tune the hyperparameters on the machines of evaluation. So we choose KNN as the only outlier detector in our systems.

Table 1: Results of different unsupervised pretrained models on anomalous sound detection.The **ck** denotes different checkpoints. TF-Pool means using transformer pooling.

System Index	Pretrained Models	All Hmean	**Machines**						
			Bearing	Fan	GearBox	Slider	ToyCar	ToyTrain	Valve
①	HuBERT	63.41	71.29	59.45	67.61	80.82	56.09	54.79	61.33
②	HuBERT (TF-Pool)	63.66	71.19	62.29	69.7	77.99	59.52	53.02	58.71
③	Wav2Vec (ck1)	66.49	65.61	70.63	82.67	82.35	55.73	52.32	68.59
④	Wav2Vec (ck2)	66.56	62.62	66.66	77.77	83.96	58.92	55.92	68.61
⑤	Wav2Vec (ck3)	65.18	70.65	67.57	76.17	88.62	55.88	48.20	64.98
⑥	Wav2Vec (ck4)	65.16	65.00	63.88	74.12	84.42	57.82	54.45	64.96
⑦	Wav2Vec (TF-Pool)	65.88	64.02	64.77	71.31	83.92	60.01	56.53	67.08
⑧	WavLM	65.49	71.70	55.70	74.65	82.88	55.20	61.43	66.07
⑨	WavLM (TF-Pool)	64.64	71.14	55.87	75.89	86.42	57.53	58.75	58.19
⑩	UniSpeech	65.08	74.74	56.92	73.49	80.87	57.26	54.40	67.64
⑪	UniSpeech (TF-Pool)	65.39	74.90	57.39	69.67	85.02	57.36	56.96	65.93
	Fusion Systems								
① + ③ + ⑧ + ⑩	Submission 1	68.01							
③ + ④ + ⑤ + ⑥	Submission 2	67.55							
③ + ④ + ⑤ + ⑥ + ⑧ + ⑩	Submission 3	68.59							
② + ⑦ + ⑨ + ⑪	Submission 4	67.28							

Finally, for testing set, we extract the corresponding embeddings with a sliding window and then compute the anomaly score based on the pretrained outlier detector before.

3.2.1. Training Configuration

For the detailed training configuration, we adopt the AdamW as the optimizer to optimize the whole network. In order to prevent overfitting on training data, we use a relatively small learning rate of 5e-4. The weight decay is set to 1e-4. The whole training process will last 10k steps and we choose the best one on the validation set. Besides, to construct the training batch effectively, we randomly sampled 2s from each recording in the training process.

3.3. Transformer Pooling

We found that in a recording, the effective signal does not exist continuously. Therefore, it is very important for the model to learn to discover effective information on its own. As mentioned in the last section, we chunk the recoding into some short segments in the training process. For each segment, we can extract one embedding representation. It is necessary to gather the embeddings from the same recording into one embedding. Based on the above considerations, we decided to use a transformer layer to fuse multiple embeddings into one. The input dimension of the transformer layer corresponds to the embedding dimension and we set the hidden dimension in the transformer layer to 2048. Same to the training objective in Figure 1, a classification loss is used to optimize the transformer pooling layer.

3.4. Ensemble

To obtain more stable results for submission, we use score-level fusion to ensemble several subsystems. The score fusion can be divided into three steps:

- for each subsystem, we compute the anomalous score by backbone detector algorithms.
- In order to balance the various systems, we apply the min-max normalization based on the scores distribution of each subsystem.

$$S_{norm} = \frac{S_{norm}}{S_{max} - S_{min}} \tag{2}$$

- Finally, we use weighted sum to ensemble several subsystems.

$$S_{fusion} = \sum_i w_i S_i \tag{3}$$

where w_i and S_i are the weight and score of subsystem i.

4. RESULTS

We list all the pre-trained model results in Table 1 and we fuse part of the systems to get our final 4 submission systems following the steps in section 3.4. From the results, we find that the Wav2Vec model performs the best. But the gap between different pre-trained models is not that large. Interestingly, the different pre-trained models complement each other and the fusion operation can bring further improvement.

5. CONCLUSION

In this task, to tackle the "first shot" problem, we apply several wav2vec-style unsupervised pretrained models which are trained on large scale speech data to anomalous detection for generalization performance. Comparing with autoencoder based baseline system, it can achieve excellent results on validation set. In addition, we propose speed perturbation on this task for augmenting data with simulated different operation status. In the last, we ensemble our subsystems with score-level fusion, and we choose four fusion results with different weights as our final submission.

6. REFERENCES

[1] K. Dohi, K. Imoto, N. Harada, D. Niizumi, Y. Koizumi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, and Y. Kawaguchi, "Description and discussion on dcase 2023 challenge task 2: First-shot unsupervised anomalous sound detection for machine condition monitoring," *arXiv preprint arXiv:2305.07828*, 2023.

[2] 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.

[3] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[4] 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.

[5] 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.

[6] C. Wang, Y. Wu, Y. Qian, K. Kumatani, S. Liu, F. Wei, M. Zeng, and X. Huang, "Unispeech: Unified speech representation learning with labeled and unlabeled data," in *International Conference on Machine Learning*. PMLR, 2021, pp. 10 937–10 947.

[7] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[8] Z. Chen, S. Chen, Y. Wu, Y. Qian, C. Wang, S. Liu, Y. Qian, and M. Zeng, "Large-scale self-supervised speech representation learning for automatic speaker verification," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 6147–6151.

[9] L. Xu, L. Wang, S. Bi, H. Liu, and J. Wang, "Semi-supervised sound event detection with pre-trained model," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[10] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Sixteenth annual conference of the international speech communication association*, 2015.

[11] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," *arXiv preprint arXiv:1904.05862*, 2019.

[12] A. Baevski, S. Schneider, and M. Auli, "vq-wav2vec: Self-supervised learning of discrete speech representations," *arXiv preprint arXiv:1910.05453*, 2019.

[13] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin, *et al.*, "Superb: Speech processing universal performance benchmark," *arXiv preprint arXiv:2105.01051*, 2021.

[14] Z. Chen, B. Han, X. Xiang, H. Huang, B. Liu, and Y. Qian, "Build a sre challenge system: Lessons from voxsrc 2022 and cnsrc 2022," *arXiv preprint arXiv:2211.00815*, 2022.

[15] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.

[16] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient algorithms for mining outliers from large data sets," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 427–438.

[17] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 93–104.