AUTO-ENCODER AND METRIC-LEARNING FOR ANOMALOUS SOUND DETECTION TASK

Technical Report

Qingkai WEI, Yanfang LIU

Beijing Kuaiyu Electronic Ltd.
Beijing, PRC
wqk@kuaiyu.com

ABSTRACT

DCASE 2020 task 2 aim at the problem of anomalous sound detection, to judge whether the target machine is in normal status by the sound it emitted [1]. The challenge of this task is to detect anomalous status while only sound of normal status is provided. With only samples of normal status, supervised learning which is usually used in sound event detection cannot be applied then. The given baseline use auto-encoder with log-mel-spectrogram as input and to reconstruct it, error of reconstruction as the anomalous score. Based on the idea of baseline, we tuned the parameters of auto-encoder net structure, tried variant auto-encoder and convolutional auto-encoder. The results show that only tuning parameters of auto-encoder shows 0.05 improvement of AUC for part of the machine types. In addition, we applied metric learning, which is usually used in face recognition, in this task to extract feature vector. Then local outlier factor is used to get the anomalous score. The results on validation dataset shows a larger improvement, increasing about 0.1 of pAUC for four types of machine.

Index Terms— auto-encoder, convolution, variant auto-encoder, metric learning

1. INTRODUCTION

Automatically detection of machine anomalous status is an essential technology, which can help factories detect the failure will occur or just occur in time. Together with other monitored parameters, sounds emitted by machine may be useful detection of machine anomaly by observing.

For the working machine, anomaly could not be allowed to occur frequently and usually last a very short time. This lead to the main challenge for machine anomaly monitoring by sound, samples of unknown anomalous sounds are really rare.

DCASE 2020 task 2 aim at this challenge, to detect unknown anomalous sounds under the condition that only normal sound samples have been provided as training data. So traditional supervised learning for sound event detection such as baby cry or alarm cannot work in this task then. Data set used for this task is carefully designed with six different machine type as pump, valve, fan, slider, ToyCar, ToyConveyer, details are described in [2, 3, 4]. The detection effect is evaluated with the area under the receiver operating characteristic (ROC) curve (AUC) and the partial-AUC (pAUC). The pAUC is an AUC calculated from a portion of the ROC curve over the pre-specified range of interest. [2]

The given baseline use auto-encoder with log-mel-spectrogram as input and to reconstruct it, error of reconstruction as the anomalous score. Based on the baseline, we tuned the parameters of auto-encoder net structure, tried variant auto-encoder and convolutional auto-encoder. The results show that only tuning parameters of auto-encoder shows improvement of AUC. More details will be demonstrated in Sec. 2.

In addition, we applied metric learning, which is usually used in face recognition, in this task. It shows a larger improvement. Details will be shown in Sec. 3.

2. AUTO-ENCODER METHODS

The given baseline is auto-encoder (AE) with fully connected layers, with anomalous score calculated with the reconstruction error of input feature. The training data is log-mel-spectrogram of original waveform with window of frame 64 ms (50 % hop size), mel-band energies (128 bands), and five frames concatenated as one input feature.

For the normal sounds, AE is trained to minimize the reconstruction error, which will lead to small anomaly scores. While for unknown anomalous sounds not used in training, AE cannot reconstruct those samples well and will give a larger reconstruction error.

There are three straightforward ideas: tune parameters of AE, try variant AE (VAE), try convolutional AE with 2-dimensionnal log-mel-spectrogram as input feature.

After lots of test, VAE and convolutional AE performs worse than baseline with AUC about 0.6 and pAUC about 0.5. Therefore, details about VAE and convolutional AE are not demonstrated here in this report.

For AE with fully connected layers, parameters such as input feature size, number of frames, size of bottleneck layer and other training parameters can be tuned. After comparison, one setup shows best performance among al, which is little better than baseline. The details of AE are shown in the Table and parameters are as below:

a. Concatenated frames: 5;
b. Mel band: 128;
c. Parameters: 2,710,992;
d. Bottle neck size: 16;
e. Optimizer: Adam;
f. Epochs: 100.

1	Input size: 128 * 5
2	Dense 1024, BN, ReLU
3	Dense 512, BN, ReLU
4	Dense 256, BN, ReLU
5	Dense 128, BN, ReLU
6	Dense 16, BN, ReLU
7	Dense 128, BN, ReLU
8	Dense 256, BN, ReLU
9	Dense 512, BN, ReLU
10	Dense 1024, BN, ReLU
11	Output size: 128 * 5

The performance of tuned AE is slightly better than the given baseline, for some machine type like ToyConveyor, valve, the improvement could be about 0.05 for AUC.

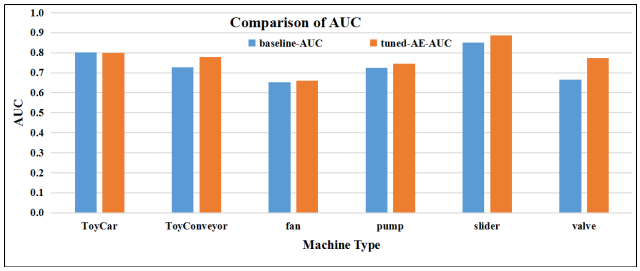

Figure 1: AUC comparison of baseline and tuned AE.

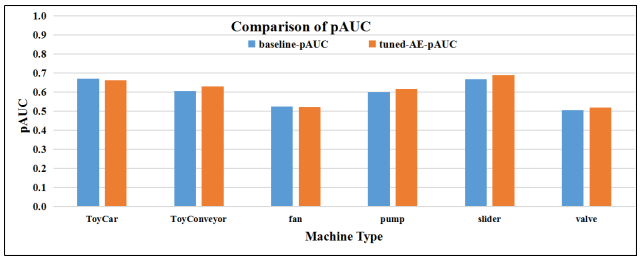

Figure 2: pAUC comparison of baseline and tuned AE.

3. METRIC LEARNING, L2-SOFTMAX

Our test results with AE do not show much improvement for the AUC of anomaly detection. Then, we try to apply some new methods may be useful.

Metric learning, aims to measure the similarity among samples while using an optimal distance metric for learning tasks. The main target of metric learning is to maximize the inter-class variations and minimize the intra-class variations. [5] Metric learning is effective in face recognition task, to use metric classify if the unseen face is similar with one in train dataset. L2-softmax method is to add a L2 feature normalization for the feature extraction layer (feature just before the output layer) [6]. L2 feature normalization could help all the samples in same class have similar features.

$$\begin{aligned} \text{minimize} \quad & -\frac{1}{M}\sum_{i=1}^{M}\log\frac{e^{W_{y_i}^T f(\mathbf{x}_i)+b_{y_i}}}{\sum_{j=1}^{C} e^{W_j^T f(\mathbf{x}_i)+b_j}} \\ \text{subject to} \quad & \|f(\mathbf{x}_i)\|_2 = \alpha, \ \forall i = 1, 2, ...M, \end{aligned}$$

For this task, MobileNetV2 [8] is used as a basic feature extractor. The last layer before output layer is used as the feature extractor, whose dimension is 1280. Then a L2-normalization layer is used for each 1280-dimensional feature. Finally, a 6 classes classifier layer with softmax is used as an output layer.

1	LogMelSpectrogram 64*313*1
2	Resize 224*224*3
3	MobileNetV2 without output layer
4	L2-normalize layer, 1280
5	Dense 6, softmax

We use data of all the six machine types to train the classifier. And the feature extractor will be used for the normal and abnormal samples to get the 1280-dimensional feature. To attain a anomalous score for a sample, the local outlier factor (LOF) will be applied on the feature to give a score for each feature.

In anomaly detection, the local outlier factor (LOF) is an algorithm proposed by Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng and Jörg Sander in 2000 for finding anomalous data points by measuring the local deviation of a given data point with respect to its neighbours. The Local Outlier Factor (LOF) algorithm is an unsupervised anomaly detection method which computes the local density deviation of a given data point with respect to its neighbors. [7]

For four machine types, the performance on pAUC of L2-softmax is much better than baseline, with an improvement about 0.1.

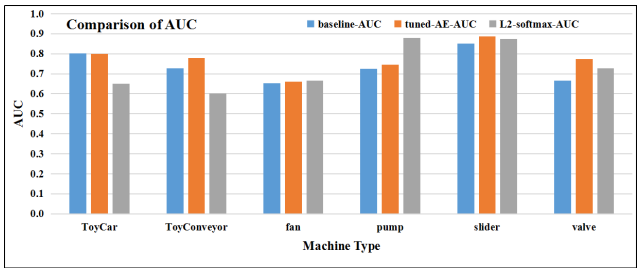

Figure 3: AUC comparison of baseline, tuned AE and L2-softmax.

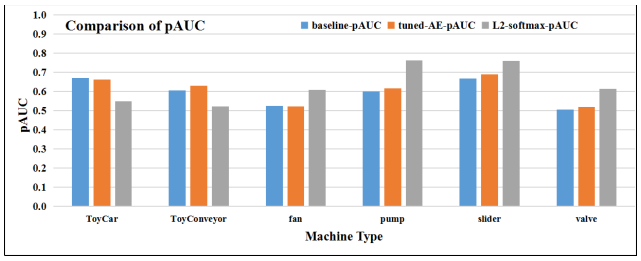

Figure 4: pAUC comparison of baseline, tuned AE and L2-softmax.

4. SUMMARY AND CONCLUSIONS

In our experiment, AE and L2-softmax shows little improvement for the anomaly detection task. However, the AUC is still about 0.8, which may not be enough for real application now. We looking forward for effective methods for anomaly detection problems.

For this task, we submit four anomalous score predictions:
System 1: L2-softmax predictions, model trained with evaluation dataset;
System 2: L2-softmax predictions, model trained with both dev and evaluation dataset;
System 3: tuned AE predictions, model trained with evaluation dataset;
System 4: tuned AE predictions, model trained with both dev and evaluation dataset.

5. REFERENCES

[1] http://dcase.community/workshop2020/.
[2] http://dcase.community/challenge2020/task-unsupervised-detection-of-anomalous-sounds
[3] Yuma Koizumi, Shoichiro Saito, Hisashi Uematsu, Noboru Harada, and Keisuke Imoto. ToyADMOS: a dataset of miniature-machine operating sounds for anomalous sound detection. In Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 308–312. November 2019.
[4] Harsh Purohit, Ryo Tanabe, Takeshi Ichige, Takashi Endo, Yuki Nikaido, Kaori Suefusa, and Yohei Kawaguchi. MIMII Dataset: sound dataset for malfunctioning industrial machine investigation and inspection. In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019), 209–213. November 2019.
[5] Kulis B. Metric learning: A survey[J]. Foundations and trends in machine learning, 2012, 5(4): 287-364.
[6] Ranjan R, Castillo C D, Chellappa R. L2-constrained softmax loss for discriminative face verification[J]. arXiv preprint arXiv:1703.09507, 2017.
[7] Breunig, M. M.; Kriegel, H.-P.; Ng, R. T.; Sander, J. (2000). LOF: Identifying Density-based Local Outliers (PDF). Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. SIGMOD. pp. 93–104.
[8] Sandler M, Howard A, Zhu M, et al. Mobilenetv2: Inverted residuals and linear bottlenecks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 4510-4520.