A HYBRID ASR MODEL APPROACH ON WEAKLY LABELED SCENE CLASSIFICATION

Technical Report

Heinrich Dinkel[1], *Yanmin Qian*[1], *Kai Yu*[1]

[1] Shanghai Jiao Tong University, Computer Science Dept., Shanghai, China,
{heinrich.dinkel}@gmail.com {yanminqian, kai.yu}@sjtu.edu.cn

ABSTRACT

This paper presents our submission to task 4 of the DCASE 2018 challenge. Our approach focuses on refining the training labels by using a HMM-GMM to obtain a frame-wise alignment from the clip-wise labels. Then we train a convolutional recurrent neural network (CRNN), as well as a single gated recurrent neural network on those labels in standard cross-entropy fashion. Our approach utilizes a "blank" state which is treated as a junk collector for all uninteresting events. Moreover, Gaussian posterior filtering is introduced in order to enhance the connectivity between segments. Compared to the baseline result, the proposed framework significantly enhances the models capability to detect short, impulsively occurring events such as speech, dog, dishes and alarm. Our best submission on the test set is a CRNN model with Gaussian posterior filtering, resulting in a 19.37 % macro average, as well as 24.41 % micro average F-score.

Index Terms— Deep Learning, Weakly labeled scene event detection and classification, HMM-GMM

1. INTRODUCTION

Since the introduction of smartphones into an era of ubiquitous computing, copious amounts of multimedia data is generated daily on a variety of internet platforms. In order to process this data, e.g., searching for specific audio clips, user defined tags can hint the contents of an audio clip. However, searching through content by user defined tags have their fair share of downsides. For once, the tags are not necessarily truthfully annotated, leading to unwanted query results. Another problem are the rather inaccurate labels themselves. They do not provide specific information when a certain tag appeared, rather only provide information that it appeared. In order to alleviate this problem, scene event detection (SED) within the machine learning community aimed to model the problem supervised, e.g., rely on large amount's of precisely segmented clip annotations, including onset and offset timestamps. However, obtaining this data is labor intensive and therefore costly. Weakly labeled audio event detection aims to automate this process, requiring clip level (weakly) data and producing precise event segmentations. This paper describes the SJTU approach on weakly labeled SED in domestic scenarios.

This paper is structured as follows. Section 2 first introduces goal of the DCASE 2018 weakly labeled task. Then in Section 3 we define our proposed model structure for this challenge. Consequently in Section 4 the experimental setup is provided and also the results are shown. Lastly in Section 5 a short conclusion of this work is given.

2. TASK DESCRIPTION AND DATA

The fourth task within the DCASE2018 challenge focuses on weakly labeled SED and classification within domestic environments. Weakly labeled scene classification can be considered as a segmentation task, where an audio clip of known events needs to be divided into its segments. The tasks difficulty lies in its inherent inability to be trained strictly supervised, since labels are given per clip, not per segment. Thus, a successful model needs to overcome the following problems:

- Sequence to segment prediction. The goal is to predict a coherent segement of a single event from clip labels.
- Overlapping events. At any specific point in time N events of interest could occur simultaneously.

The sequence to segment prediction problem makes it especially hard to train a sufficiently robust classifier in sparse data scenarios. Thus, our work focues on alleviating the sequence to segment prediction problem partially, by estimating per frame labels for the training phase.

2.1. Evaluation criterion

The evaluation scheme in this challenge uses an event-based macro average F-score. Event based scoring considers a predicted event as being correct, only if its temporal position overlaps with the same labeled event in the ground truth annotation. Since this metric is strict considering its temporal positioning, a collar is allowed for both on and offset timestamps. On top of that, a tolerance can also be added in order to lessen the strictness of this criterion. In this challenge, the collar is fixed at 200ms and a maximum tolerance of 20% of the ground truth event length is set.

2.2. Data

For this challenge, the dataset [1] is split into four distinct parts:

1. Weakly labeled training data, containing soft annotations (each utterance has at least one label)
2. Indomain data, containing no annotations but its events are from the same domain as the training data ones.
3. Outdomain data, containing no annotations and its events are guaranteed not to contain any challenge relevant events. This work neglects this data subset.
4. Development test data, containing hard annotations (label and time)

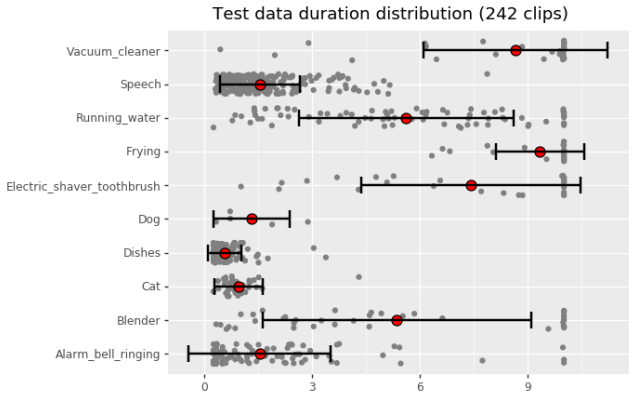

Figure 1: Duration distribution for each category within the test set. The red dot denotes a mean of a class and the error bar its standard deviation.

Each subset contains clips of at most 10 seconds length. The specific data subset lengths and number of clips can be seen in Table 1. The provided data contains $\mathcal{L}$ ={Speech, Alarm, Blender, Cat, Dishes, Dog, Electric-shaver-toothbrush, Frying, Running-water, Vacuum cleaner}, $|\mathcal{L}| = 10$ labels.

Datasubset	clips	Events	Length
Train	1578	10	4.3 h
InDomain	14412	10	40 h
OutDomain	39999	?	110 h
Test	288 (242)	10	0.8 h

Table 1: Challenge data subset length and clip counts. The size of the testset was adjusted during the competition.

It should be noted that during the challenge, it was discovered that **46** clips within the training set overlap with clips in the test set. Those were later removed, however in this work these clips are utilized to visualize our HMM-GMM alignment in Figure 2.

Another important aspect of the data is to analyze its duration. Test data event duration can be classified into short and long events, as seen in Figure 1. Speech, Dog, Cat, Dishes and Alarm bell ringing can be considered as short events, while Vacuum-cleaner, Electric-shaver-toothbrush, Blender and Frying are considered as long events. More importantly, within the training set, only the "Speech" class does not appear by itself, rather only in conjunction with another event.

3. FRAMEWORK

Due to the dataset being relatively small when compared to last years DCASE2017 dataset (4h vs. 40h), our initial belief is that unsupervised methods would lead to a better result. Our proposed framework follows a modified HMM-GMM model to estimate frame-level labels for a classifier to train on. The framework consists of three stages:

Stage 1 Estimate a frame-level alignment for each utterance using a modified HMM-GMM approac (see Figure 2).

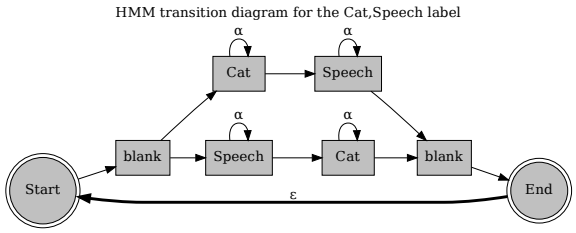

Figure 2: The proposed single state HMM-GMM model for the Cat, Speech label. HMM observation probabilities are not drawn. Each HMM represents a single event. End to start loops are applied on utterance level.

Stage 2 Train a classifier (usually neural network) on the estimated frame alignments.

Stage 3 Using the classifier from Stage 2, predict labels for the indomain data. Then use the indomain data as training data and the predictions from Stage 1 for cross validation.

Since our approach directly estimates hard-labels in Stage 1, the models trained during Stage 2 and Stage 3 do often perform equally well.

The neural network classifiers use cross-entropy to distingush between the estimated hard-labels as their training criterion.

In this initial work, the possibility of events overlapping is completely neglected. Our intention is to estimate a reliable frame-level alignment using a HMM-GMM model, similar to traditional ASR approaches. HMM-GMM ASR models use a so called Bakis model, which defines a directed left to right graph. However, for the use in weakly-labeled scene detection, we modified the transition scheme to allow clip-level loops (see Figure 2), in order to be able to detect reoccurring patterns e.g., $O = \{\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_1, \mathcal{L}_2\}$, as well as all possible permutations $\pi(O)$. While $\pi(O)$ could easily get uncomputeable, this dataset only has at most three events simultaneously occuring, making this approach feasible.

Each event is modeled by a single-state left-to-right HMM model. Additionally, a special state 'blank' is added as a default junk state for labels not being within the label-set $\mathcal{L}$ (see Figure 2). This extends our labelset to be $\mathcal{L}^* = \{\mathcal{L} \bigcup \mathcal{L}_{blank}\}$, similar to the key word spotting (KWS) task, this state collects all unwanted events.

3.1. Post-processing

The baseline approach uses a median filter method for post processing the labels. This method effectively removes short, noisy outputs.

However, median filtering does not connect adjacent output segments. This severely affects short, sporadic appearing events, such as Alarms, Cats, Dogs and Dishes.

In this work we suggest Gaussian posterior filtering (Equation (1)) as an alternative. Our intuition is that a high posterior value for a specific event should indicate that successive frames are also likely to be of the same event class. For each posterior probability (softmax) over a class's output, we apply a 1-dimensional Gaussian filter with a kernel size of $\sigma = 11$ frames.

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} \quad (1)$$

After filtering, the class with the highest score is picked as the representative one. It should be noted that this method neglects the overlapping nature of the task.

4. EXPERIMENTS

Experiments were conducted using Kaldi [2] for HMM-GMM modeling and Pytorch [3] as our deep learning framework. All experiments during stage 2 training split the data randomly into train/cv chunks with a probability of 0.8/0.2.

4.1. Data preprocessing

The DCASE 2018 dataset is a subset of the previously introduced Audioset dataset [4], which sampled audio from Youtube for scene event detection.

The given raw-waveform data is firstly preprocessed by extracting log mel spectrogram features (further denoted as logmel). For each 40 ms window a 64 dimensional logmel representation is extracted every 20 ms. These logmel features are normalized during training to be zero mean and unit variance. One of the main challenges with this data is its large variability. By randomly listening to the training dataset we came across the problem of varying loudness. Some labels are significantly louder than others, e.g., cats are generally quite silent, while alarm bells and blenders overtone other events. In order to partially circumvent loudness and a large variety within a specific event class, we adapted to not only normalize the extracted logmel features, but also normalize the raw waveform mean to be zero. The logmel + logmel-norm features represent a frame-wise concatenation of non-normalized and normalized waveform extracted logmel features. A comparison between normalized and unnormalized raw waveform alignments can be seen in Figure 3.

4.2. HMM training

The HMM-GMM is trained on standard 13-dimensional MFCC features with additional delta and acceleration coefficients. Each HMM state contains a GMM with 512 mixtures. During training we follow a standard ASR procedure by starting with $11 = |\mathcal{L}^*|$ Gaussians and increase the number of Gaussians by a linear factor after each iteration. The whole procedure is ran for 40 iterations, at which the alignment is obtained by the one-pass viterbi algorithm.

Figure 3 shows that the produced HMM-GMM alignment seems to be effective for speech events, but largely fails for cats. Moreover, we perceive the waveform normalized samples as being slightly in favor of non-normalized ones. Thus, our HMM-GMM was trained on normalized waveform MFCC features.

4.2.1. Classifiers

Having estimated the frame-wise labels by using an HMM-GMM alignment, a neural network classifier is used to obtain frame-wise posteriors.

For this task we submit three models (see Table 2). The convolutional neural network (CRNN) model follows closely the same structure as the baseline one [1]. Moreover, the gradient recurrent unit (GRU) model has a bidirectional two layer structure and has

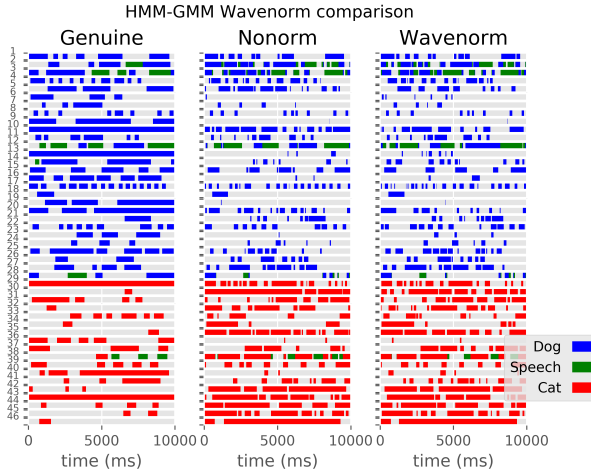

Figure 3: Alignments for normalized and unnormalized raw waveforms for the 46 overlapping training/test sentences. Note that overlapping segments are neglected.

Submission	Model	Feature	Stage	Post
Task4_1	GRU	logmel+logmelnorm	2	Median
Task4_2	CRNN	logmel+logmelnorm	3	Median
Task4_3	CRNN	logmel	3	Gauss
Task4_4	CRNN	logmel	3	Median

Table 2: Proposed model submissions. All models use cross-entropy as their training and evaluation criterion, therefore only output a single label. Stage 2 and 3 refers to the training on only training data and training + indomain data respectively.

128 input and 256 output nodes. Neural network training uses adam optimization [5] with a starting learning rate of 0.004. A patience of 15 epochs was implemented, basing its decision on the cross-validated cross-entropy loss.

4.3. Results

The results are provided as a summary of macro and micro average (Table 4), as well as per event macro-average F-scores (Table 3). The baseline approach is a CRNN model, trained solely on weak labels [1].

As we can see from the Event-wise results (Figure 4), all of our proposed models are more robust compared to the baseline. Specifically regarding short, impulsive events such as Speech, Dog and Dishes. However, all models struggle in detecting the Cat event. We think this behavior is due to the few Cat events encompass a large acoustic variety e.g., purring and meowing and other "Cat" sounds.

5. CONCLUSION

In this paper an ASR based HMM-GMM approach was proposed in order to overcome partially the segmentation problem of this task. HMM-GMM alignments on the training data were used as labels for later CRNN and GRU training. Gaussian posterior filtering was utilized in order to connect short disjoint segments with each other.

Model	Alarm_bell	Blender	Cat	Dishes	Dog	Electric shaver	Frying	Running water	Speech	Vacuum cleaner
Baseline	3.9	15.4	0	0	0	**32.4**	**31**	11.4	0	**46.5**
Task4_1	14	17.2	0	2.7	0	7.5	7.4	**26.7**	29.2	28.8
Task4_2	23.8	15.5	0	8.4	15.4	4	0	11.8	37.7	20
Task4_3	**22.2**	**19.3**	0	**18.3**	**37.5**	8.5	3.8	14.1	**43.7**	26.4
Task4_4	21.6	21.3	0	5.0	6.0	9.7	7.0	12.9	35.5	30.3

Table 3: Per event results of our submission

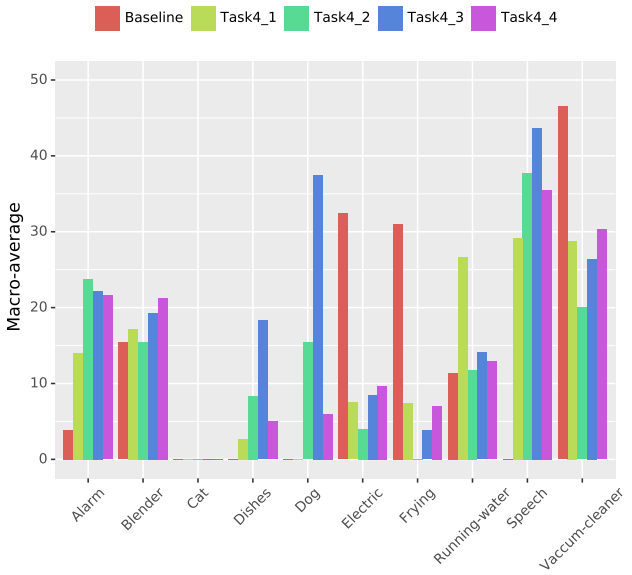

Figure 4: Per event results in terms of macro average

Submission	Micro-Avg	Macro-Avg
Baseline	7.9	14.04
Task4_1	19.21	13.36
Task4_2	20.5	13.66
Task4_3	**24.41**	**19.37**
Task4_4	18.28	14.93

Table 4: Micro and macro average results on the test set

Results show the superiority of the approach compared to the baseline specifically when detecting speech. Out best submission result improves 5% in absolute over the baseline, scoring at 19.37% F-score.

In our future work we would like to improve the postprocessing of our method, strictly enforcing segmented outputs, rather than frame-wise posteriors.

6. ACKNOWLEDGMENT

This work has been supported by the National Key Research and Development Program of China (Grant No.2017YFB1002102), the China NSFC project (No. 61603252 and No. U1736202). Experiments have been carried out on the PI supercomputer at Shanghai Jiao Tong University.

7. REFERENCES

[1] R. Serizel, N. Turpault, H. Eghbal-Zadeh, and A. Parag Shah, "Large-Scale Weakly Labeled Semi-Supervised Sound Event Detection in Domestic Environments," July 2018, submitted to DCASE2018 Workshop. [Online]. Available: https://hal.inria.fr/hal-01850270

[2] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2011.

[3] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[4] J. F. Gemmeke, D. P. W. 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 *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[5] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *International Conference on Learning Representations*, pp. 1–13, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980