# CAU SUBMISSION TO DCASE 2022 TASK6B: LANGUAGE-BASED AUDIO RETRIEVAL USING TRANSFER LEARNING

Technical Report

*Jiwon Park*[1], *Chaewon Hwang*[1], *Il-Youp Kwak*[1], *Changwon Lim*[1]

[1]Chung-Ang University, Department of Applied Statistics, Seoul, South Korea,
{jiwon3401, ladyikol , ikwak2, clim}@cau.ac.kr

## ABSTRACT

This report proposes a language-based audio retrieval model for the 2022 DCASE audio retrieval challenge. In this challenge, to make use of the learned feature from AudioSet data, we utilized CNN10 network pre-trained on AudioSet data. With the transfer learning, our proposed model took 10-layers CNN and adding GRU after CNN Module. We used pre-trained Word2Vec as text encoder[1]. Experiments show that the proposed model achieved mAP score of 0.091 and showed better performance compared to baseline mAP score of 0.067.

***Index Terms—*** audio retrieval, transfer learning, deep learning

## 1. INTRODUCTION

This technical report was written to describe the model of Audio Retrieval addressed from task 6 subtask B of the DCASE 2022 challenge[2]. Purpose of this task is to get 10 audio files from a specified dataset for each text query and sort them depending on how well they fit the query. One example of audio retrieval is if a text query “sound of waves” is given, then calculating relevance scores of audio samples to the given query will be done, and then audio sample will be sorted in descending order by their relevance scores. This results in sorting 10 audio files based on scores. The goal of the model is to embed audio features by selecting an encoder that can be embed them well.

Dataset we utilized for this subtask is Clotho v2[3], also used in DCASE 2021 Challenge task6, but has been repurposed for language-based retrieval. The Clotho v2 dataset contains audio samples ranging in length from 15 to 30 seconds, each with five captions ranging in length from eight to twenty words. There are 6974 audio samples, and each sample has 5 captions. With the transfer learning, our proposed model in audio encoder took 10-layers CNN(CNN10) trained on AudioSet[4]. We used pre-trained Word2Vec as text encoder[1].

## 2. PROPOSED METHOD

### 2.1. System Overview

The Figure 1 shows the proposed system overview.

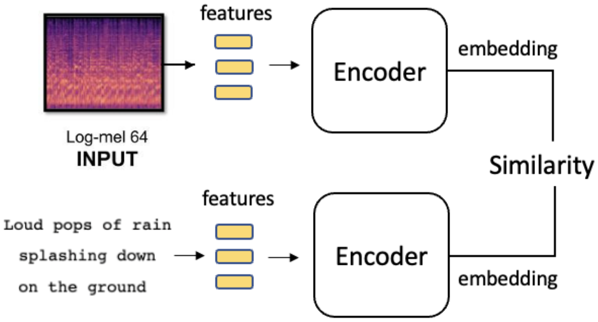

Figure 1: Model Architecture

### 2.2. Preprocessing

The feature extension used in the model extracted the input audio log mel spectrogram using the librosa package used in the baseline model. The log-mel spectrogram converts waveform to a log magnitude mel-frequency spectrogram. We applied the sampling rate of data with 44.1 kHz, duration of each window to analyze with 0.04, advance between successive analysis windows with 0.02 and number of Mel bands equals to 64.

### 2.3. Pretrained Audio Neural Networks using AudioSet

Pretrained audio neural networks (PANNs) are proposed by Kong et al. (2020), and are trained on the large-scale AudioSet dataset, which contains over 5,000 hours of audio recordings with 527 sound classifications. These PANNs are transferred to variety of audio-related tasks. AudioSet released embedding features of audio clips extracted from a pretrained convolutional neural network. We took the 10-layer CNNs consist of 4 convolutional layers as our model.

### 2.4. Proposed model

#### *2.4.1. Audio Encoder*

For feature extraction, we used CNN10 and GRU as audio encoder. We take CNN10, which is a set of pre-trained networks learnt from PANNs to AudioSet[4], as our audio encoder. We freeze the weights learned from pretrained networks and employed them to our model. After CNN10 architecture, GRU is added. Table 1 shows CNN10 architecture we used. The numbers after “@” refer to the number of feature maps in this layer.

Table 1: CNN10 architecture

| CNN10 |
|---|
| Log-mel spectrogram<br>64 mel bins |
| (3 × 3 @64<br>BN,ReLU)×2 |
| Pooling 2 × 2 |
| (3 × 3 @128<br>BN,ReLU)×2 |
| Pooling 2 × 2 |
| (3 × 3 @256<br>BN,ReLU)×2 |
| Pooling 2 × 2 |
| (3 × 3 @512<br>BN,ReLU)×2 |
| Pooling 2 × 2 |
| FC 512, ReLU |
| FC 527, Sigmoid |

#### 2.4.2. *Text Encoder*

The pre-trained model word2vec is used as word embedding as we load prior embedding weights[1]. The word2vec tool takes a text corpus as input and produces the word vectors as output.

#### 2.4.3. *Embedding Similarity*

The Triplet Loss reduces the distance between an anchor and a positive with the same identity while increasing the distance between an anchor and a negative with a different identity[5]. It is a method of comparing the similarity of the n-dimension embedding with L2 norm. We compute similarity score of audio embedding and text embedding and get the triplet margin ranking loss for each anchor audio and text pair.

## 3. EXPERIMENTS

### 3.1. Experimental Setups

In the training process, epoch 150 and batch size of 32 is used with a learning rate of $10^{-3}$. The optimizer we used is Adam[6] and learning rate of the optimizer is 1e-3. All trainable parameters are applied with this setting, and the experiment was conducted by changing the learning rate schedule. Learning rate scheduler we use is ReduceLROnPlateau, Cosine Annealing and ExponentialLR[7]. The audio feature extracted from log-mel spectrogram is obtained by sampling rate 44.1kHz and 16kHz, and performance of using 16kHz is significantly lower, so we choose 44.1kHZ. Since we can submit up to four submissions, we select the model based on the mAP score.

### 3.2. Experimental Results

In the Audio Encoder, the score was the highest when using ReduceLRonPlateau Scheduler on CNN10. Changing the learning rate scheduler when performing hyperparameter tuning affects the score. If the validation loss is no longer reduced during training, ReduceLROnPlateau reduces the learning rate to (existing learning rate * factor) we set.

Table 2: Score for model performance on evaluation data

| **Model** | R@1 | R@5 | **mAP** |
|---|---|---|---|
| Baseline Model | 0.032 | 0.109 | 0.068 |
| CNN10 + GRU | 0.032 | 0.109 | 0.066 |
| CNN10 + GRU<br>(Transfer-learning<br>with Cosine Schedular) | 0.035 | 0.127 | 0.07 |
| CNN10 + GRU<br>(Transfer-learning<br>with ExponentialLR Schedular) | 0.041 | 0.141 | 0.086 |
| CNN10 + GRU<br>(Transfer-learning<br>with ReduceLROnPlateau Schedular) | 0.048 | 0.148 | 0.091 |

## 4. CONCLUSION

When embedding audio features using the CNN10 PANNs pretrained model, we see that the features of audio features are extracted well. With the transfer learning network using CNN10 achieved mAP score of 0.091 which is better than baseline.

## 5. REFERENCES

[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *arXiv preprint arXiv:1301.3781*, 2013.

[2] http://dcase.community/challenge2022/task-language-based-audio-retrieval.

[3] K. Drossos, S. Lipping, and T. Virtanen, “Clotho: An audio captioning dataset,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 736–740.

[4] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. 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.

[5] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, “Signature verification using a” siamese” time delay neural network,” *Advances in neural information processing systems*, vol. 6, 1993.

[6] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[7] Z. Li and S. Arora, “An exponential learning rate schedule for deep learning,” *arXiv preprint arXiv:1910.07454*, 2019.