DCASE 2024 CHALLENGE TASK 8 TECHNICAL REPORT

Technical Report

Minjun Chen[1], Yangyang Liu[1], Bo Peng[1], Jie Chen[1]
[1] Samsung Research China-Nanjing, Nanjing, China
{minjun.chen, yang17.liu, b.peng, ada.chen}@samsung.com

ABSTRACT

We describe our submitted systems for DCASE2024 Task 8 in this technical report: Language-based Audio Retrieval. Our proposed system focus on training audio encoder and text encoder combined to get expressive audio and text presentation, which helps distinguishing different audios and text more efficiently. We use pre-trained audio and text encoder of VAST, which were trained on a large multi-modality dataset VAST-27M. We train these encoders on several audio caption datasets, include AudioCaps, WavCaps, FSD50K, Laion630k, and ClothoV2 furtherly with three learning objectives, except the audio-text contractive objective, we also use audio-text match and masked language model objective to strengthen the training procedure. We use the mix-up as the data augment policy during pre-training. Our proposed systems achieve 0.37 mAP@10, and 0.244 R@1, with model ensemble, our systems achieve 0.406 mAP@10, and 0.278 R@1 on the ClothoV2 evaluation set.

Index Terms— Language-based audio retrieval, Audio caption, Multi-modality, BEATs, VAST, BERT

1. INTRODUCTION

In this technical report, we describe our submitted systems for the task 8 of the DCASE 2024 challenge: Language-based Audio Retrieval [1]. The target of this task is, retrieving audio signals using their sound content textual descriptions (i.e., audio captions). Currently, most of the methods for language-based audio retrieval (Text2Audio) focus on training an effective audio and text encoder to get expressive audio and text presentation, then calculate the similarity of the two presentations, and rank the retrieved items according to the similarity scores.

It is important to get an expressive presentation of audio and text description for these kinds of tasks, include Audio Caption, Audio2Text retrieval, and Text2Audio retrieval. It has been observed that pre-trained model on large datasets could transfer its knowledge to downstream tasks, and bring significant improvement of downstream tasks. Recently multi-modality models and datasets have shown impressive progress in vision, audio and text domains. For this task, we used the BEATs [2] as the audio encoder and BERT [3] as the sentence encoder, both were pre-trained on a large multi-modality dataset VAST-27 [4], combined with several multi-modality objectives, and used mix-up as data augment policy, trained the whole model on several audio caption datasets. Most of the datasets, we used to train the model further only have audio and caption pair information, without the related vision, subtitle and speech content information, so we just used the audio and text (caption) pair to train and fine-tune the model. We pre-trained the model on WavCaps [5], AudioCaps [6], FSD50K [7], and Laion630k [8] dataset first, and then fine-tuned on ClothoV2 [9] dataset further (We excluded all the audio clips overlapped with the ClothoV2 evaluation and test set during training and validation). We used model ensemble to improve the model performance on the final submissions.

This technical report is organized as follows: Section 2 describes the models and policies we used to train the T2A systems. In Section 3, we demonstrate the experimental results of our proposed scheme. Finally, we list the references in Section 4.

2. PROPOSED METHOD

2.1. Dataset

We trained and evaluated our model on the datasets provided by DCASE 2024 task8 and several external datasets.

• **Clothov2**: the Clotho v2 dataset consists of audio samples of 15 to 30 seconds duration, with each audio sample having five captions of eight to 20 words length. We use development-training and development-validation split for fine-tune and validation; we report the scores on development-testing split. The development-testing and evaluation set are excluded from below external datasets according the provide Freesound ids.

• **AudioCaps**: this is a large-scale dataset of about 46K audio clips. All audio clips labelled with human-written text pairs collected via crowdsourcing on the AudioSet [15] dataset. The whole dataset used as training set during our pre-train.

• **WavCaps**: The captions are generated by ChatGPT, and the audio clips are extracted from several sources, include Freesound, BBC Sound Effects, SoundBible, and AudioSet Strongly-labelled Subset. We used the whole dataset for pre-train.

• **FSD50K**: it contains over 51k audio clips (~100 hours) manually labeled using 200 classes drawn from the AudioSet Ontology. For each audio clip, the caption generated by prompting ChatGPT (GPT-4) with its sound event tags. We used the whole dataset as training set for our pre-train.

• **Laion630k**: LAION-Audio-630K is a large-scale audio-text dataset consisting of 633,526 pairs with the total duration of 4,325.39 hours. It contains audios of human activities, natural sounds and audio effects, consisting of eight data sources from publicly available websites. We downloaded parts of it, and

excluded the audios overlapped with WavCaps, leaving about 130000 audio clips.

• **DCASE 2024 LASS validation (synth) set**: This part of data include 1000 clips and each clips with 3 captions. This is the develop-validation subset of DCASE 2024 Task 9. We included this set in our training set.

2.2. Basic Framework

As shown in Figure 1, our system employs an end-to-end transformer architecture, comprising an audio encoder, and a text encoder. To take the advantage of multi-modality pre-training, we extracted the audio encoder (BEATs) and text encoder (BERT) from VAST, which pre-trained on a large multi-modality dataset VAST-27M [4], including vision, audio, subtitles and captions. We dropped the vision encoder of VAST, used multiple large audio captions datasets to continue training the model, and then fine-tuned it on Clothov2 dataset, multiple learning objectives are employed during the training procedure.

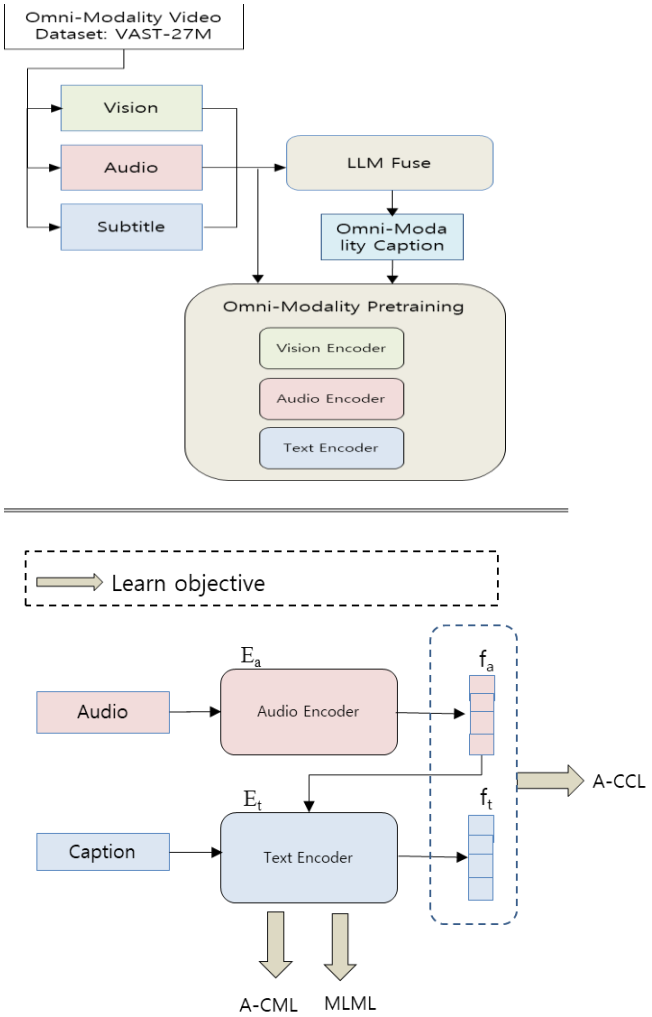

Figure1: The top part of the figure shows the VAST pre-training diagram. The bottom part shows the training framework of our model. The Audio Encoder and Text Encoder take the pre-trained weights of VAST model.

2.3. Learning Objectives

In addition to audio-caption contractive objective (A-CCL), we also employed audio-caption match objective (A-CML), and masked language model objective (MLML) to strengthen the training procedure of the model.

Audio-Caption Contrastive Loss (A-CCL). The audio presentation f_a and caption presentation f_t are obtained from the audio encoder E_a and text encoder E_t respectively. The f_a, f_t are then projected into a shared semantic space with same dimension as a and t. The contractive objective is expected to pull similar presentations closer and push different ones away. The audio-caption pairs from the same sample are view as similar, otherwise would be treat as different ones. The contractive loss could descript as below:

$$L_{A-CCL} = -\frac{1}{2}\sum_{i=1}^{B} log \frac{exp\left(\mathfrak{t} \cdot sim(a_i,\ t_j)\right)}{\sum_{j=1}^{B} exp\left(\mathfrak{t} \cdot sim(a_i, t_j)\right)} - \frac{1}{2}\sum_{i=1}^{B} log \frac{exp\left(\mathfrak{t} \cdot sim(a_j, t_i)\right)}{\sum_{j=1}^{B} exp\left(\mathfrak{t} \cdot sim(a_j,\ t_i)\right)} \quad (1)$$

Audio-Caption Match Loss (A-CML). This loss encourages the model to infer whether a pair of audio and caption matched or not. The E_t is feed with the caption again and also the audio presentation f_a as cross attentions. The output then feed into a MLP layers to make binary predictions P_{acm}. We follow the hard negative mining strategy [4] and [10] to create the informative negative pairs. The loss function is formulated as follows, where $y = 1$ if audio and caption are matched, and 0 otherwise:

$$L_{A-CML} = \mathbb{E}_{(a_i, t_i)\sim(\mathcal{V},\mathcal{C})}[y \log P_{acm} + (1-y)\log(1-P_{acm}) \quad (2)$$

Masked Language Model Loss (MLML). This objective encourages the model to predict the masked tokens of captions. We mask 60% of the tokens of the caption follow VAST settings. The cross attention layers of E_t are activated, and the f_a is feed as the encoder hidden conditions. We use BERT's vanilla prediction layer to generate the output follow [4]. The loss defined as follows, t_m denotes the masked tokens, and $t<m$ denotes the tokens before them:

$$L_{MLML} = \mathbb{E}_{(a_i, t_i)\sim(\mathcal{V},\mathcal{C})} \log P(t_m | \ t < m,\ a) \quad (3)$$

The overall loss is just a sum of the above three loss, all with the same weight:

$$L = L_{A-CCL} + L_{A-CML} + L_{MLML} \quad (4)$$

With these three learning objectives, we believe the model could learn more expressive presentations of audio and text.

3. EXPERIMENTS AND RESULT

3.1. Preprocessing

To deal with different durations of audio clips from different datasets, we padded all the audio clips to max 30s with zero, and the audio clips were segmented to 10s before feed into the model. All audio clips are resampled to 16k, and extracted 64-bin log-MEL spectrograms with 25ms frame length and 10ms frame hop. For audio captions, we transformed all characters to lowercase, and removed all the punctuations. We used WordPiece tokenizer [11] to tokenize the sentences and the length is limited to max 40 tokens, exceed tokens are dropped.

3.2. Data augmentation

For enriching the data diversity, we employed mix-up as the data augment policy. We mixed up part of the audio clips in one batch on their log-MEL spectrograms. For corresponding captions, we did mix-up on the word embedding space of BERT embedding layers. The mixed audio and text embedding are then paired and concatenated to make a bigger batch. We used mix-up only for pre-train, when fine-tune on Clothov2, the mix-up was not applied.

3.3. Training

We pre-trained the model on Clothov2, AudioCaps, WavCaps, FSD50K, Laion630k, and DCASE 2024 LASS validation (synth) set first. We used Pytorch Adamw optimizer for training 18 epochs, the learning rate is warmup for one epoch, after then is decayed from 1e-5 to 1e-7 using a cosine schedule. We applied 2e-5 as the weight decay for all linear layers. The mAP@10 on the Clothov2 development-validation was used the checkpoint monitor. After pre-train, the model was fine-tuned on Clothov2 further. The hyper-parameters were similar with pre-train, except the initial learning rate was set to small 2e-6, and the max training epoch was set to 15. The data mix-up policy was not applied here. We monitored the mAP@10 on the Clothov2 development-validation to select the best weights.

For improving the performance further, we trained multiple models for model ensemble. We believe ensemble models with different structures could improve the whole performance. Therefore, we trained several models using PaSST [12] (and its variants) as the audio encoder and used Roberta [13] as the sentence encoder following [14]. Although these models have lower mAP@10 scores on Clothov2 development-validation, the ensemble models could benefit from the complementary of different model structures and learned knowledge.

Because the ClothoV2 dataset is not large enough, to enrich the fine-tune dataset and avoid overfitting, we used k-fold cross validation policy to train more models for model ensemble. The development-training and development-validation sets are combined and split into 5 folds for training and validation (the development-testing set was not used for training and validation).

3.4. Results

Below we will report the metrics mAP@10, R@1, R@5, and R@10 on the development-testing set (ClothoV2 evaluation set) for different model structures and pre-train policies. The development-testing set and Task 8 evaluation set are excluded from the training and validation during the pre-train and fine-tune stages.

Table 1: Text2Audio performance on ClothoV2 test for different model configurations. PT means whether pre-train on the audio captions datasets listed in section 2.1. Y means pre-trained on these datasets, N means only fine-tuned on ClothoV2 dataset. The superscript (1, 2, 3, 4, 5, 6, and 7) denotes the system id. The beats and bert encoder in system 5 and 6 are also pre-trained on VAST-27M.

Model	PT	mAP@10	R@1	R@5	R@10
passt[1] + bert	N	0.27	0.165	0.41	0.557
passt[2] + bert	Y	0.332	0.22	0.497	0.644
passt_l[3] + roberta_l	Y	0.329	0.207	0.499	0.638
passt_s[4] + roberta_l	Y	0.344	0.221	0.505	0.653
beats[5] + bert	N	0.355	0.24	0.505	0.645
beats[6] + bert	Y	**0.37**	**0.244**	**0.534**	**0.662**
model[7] ensemble	_	**0.406**	**0.278**	**0.576**	**0.705**

As can be seen from Table 1, pre-training on more audio caption datasets gives better performance, these audio caption datasets provide a wide variety of audios and corpus, which could help model to capture the differences and correlations of audio and text expression. Our best single model (system 6) achieved 0.37 mAP@10. Our ensemble system 7, which is composition of 42 models trained with the six system configurations, achieved 0.406 mAP@10. These show the effectiveness of our training policies and methods. Finally, we submitted two systems for the task 8, single model system 6, and the ensemble system 7.

4. REFERENCES

[1] https://dcase.community/challenge2024/
[2] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, and F. Wei, "Beats: Audio pre-training with acoustic tokenizers," arXiv preprint arXiv:2212.09058, 2022.
[3] 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.

[4] S. Chen, H. Li, Q. Wang, Z. Zhao, M. Sun, X. Zhu, and J. Liu, "VAST: A Vision-Audio-Subtitle-Text Omni-Modality Foundation Model and Dataset," arXiv preprint arXiv:2305.18500, 2023.

[5] X. Mei, C. Meng, H. Liu, Q. Kong, T. Ko, C. Zhao, Mark D. Plumbley, Y. Zou, and W. Wang, "WavCaps: A ChatGPT-Assisted Weakly-Labelled Audio Captioning Dataset for Audio-Language Multimodal Research," arXiv preprint arXiv:2303.17395, 2023.

[6] C. Dongjoo Kim, B. Kim, H. Lee, and Gunhee Kim, "AudioCaps: Generating Captions for Audios in The Wild," in NAACL-HLT, 2019.

[7] E. Fonseca, X. Favory, J. Pons, F. Font, and Xavier Serra, "FSD50K: An Open Dataset of Human-Labeled Sound Events," arXiv preprint arXiv:2010.00475, 2020.

[8] Y. Wu, K. Chen, T. Zhang, Y. Hui, M. Nezhurina, T. Berg-Kirkpatrick, and Shlomo Dubnov, "LARGE-SCALE CONTRASTIVE LANGUAGE-AUDIO PRETRAINING WITH FEATURE FUSION AND KEYWORD-TO-CAPTION AUGMENTATION," arXiv preprint arXiv:2211.06687, 2024.

[9] K. Drossos, S. Lipping, and Tuomas Virtanen, "Clotho: An Audio Captioning Dataset," arXiv preprint arXiv:1910.09387, 2019.

[10] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," Advances in neural information processing systems, vol. 34, pp. 9694–9705, 2021.

[11] Y.Wu, M. Schuster, Z. Chen, Q. V. Le,M. Norouzi,W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey et al., "Google's neural machine translation system: Bridging the gap between human and machine translation," arXiv preprint arXiv:1609.08144, 2016.

[12] K. Koutini, J. Schl ̈uter, H. Eghbal-zadeh, and G.Widmer, "Efficient training of audio transformers with patchout," in 23rd Annual Conf. of the Int. Speech Communication Association, Interspeech, 2022.

[13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized BERT pretraining approach," CoRR, vol. abs/1907.11692, 2019.

[14] P. Primus, K. Koutini1, and Gerhard Widmer1, "CP-JKU'S SUBMISSION TO TASK 6b OF THE DCASE2023 CHALLENGE: AUDIO RETRIEVAL WITH PaSST AND GPT-AUGMENTED CAPTIONS," Technical Report for Challenge on Detection and Classifi-cation of Acoustic Scenes and Events 2023.

[15] 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 Int. Conf. Acoustic., Speech and Signal Process., ICASSP, 2017.