ENSEMBLE SYSTEMS WITH CONTRASTIVE LANGUAGE-AUDIO PRETRAINING AND ATTENTION-BASED AUDIO FEATURES FOR AUDIO CAPTIONING AND RETRIEVAL

Technical Report

Feiyang Xiao[1]*, Qiaoxi Zhu*[2]*, Haiyan Lan*[1]*, Wenwu Wang*[3]*, and Jian Guan*[1*]

[1]Group of Intelligent Signal Processing (GISP), College of Computer Science and Technology, Harbin Engineering University, Harbin, China
[2]Centre for Audio, Acoustic and Vibration (CAAV), University of Technology Sydney, Ultimo, Australia
[3]Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guildford, UK

ABSTRACT

This technical report describes our submission on Task 6 (automated audio captioning and language-based audio retrieval) of the Detection and Classification of Acoustic Scenes and Events (DCASE) 2023 Challenge. The proposed systems in this submission are based on a contrastive language-audio pretraining strategy and the attention-based audio feature representation. Experiments show that our systems can achieve a SPIDE$_\tau$-FL score of 28.32 on automated audio captioning and an mAP score of 31.18 on language-based audio retrieval.

***Index Terms*—** Automated audio captioning, language-based audio retrieval, contrastive learning for audio pretraining.

1. INTRODUCTION

In the Detection and Classification of Acoustic Scenes and Events 2023 challenge (DCASE2023 Challenge), Task 6 includes two down-stream tasks [1], i.e., automated audio captioning [2] (Task 6A) and language-based audio retrieval [3] (Task 6B). Automated audio captioning is a cross-modal task that aims to describe the content information of the input audio signal with text, i.e., caption. In contrast, language-based audio retrieval is a multi-modal task that retrieves the most matched audio clips within a database from a given caption.

This technical report presents our DCASE 2023 Challenge Task 6A and 6B submissions. This report introduces our submission of automated audio captioning and language-based audio retrieval systems, mainly based on the contrastive language-audio pretraining strategy (CLAP) [4] and the attention audio feature representation [5, 6].

2. LANGUAGE-BASED AUDIO RETRIEVAL SYSTEM

Language-based audio retrieval methods compare the matching degree between a text query (caption) and audio signals to select the most matched audio signal. Existing methods usually employ contrastive learning (CLAP) to train the retrieval methods [4, 7, 8]. Inspired by this CLAP-based methodology, our submission is developed based on CVSSP's audio retrieval work [7] and our previous work [8]. The proposed retrieval method includes an audio encoder, a text encoder and two projector modules.

The audio encoder extracts the audio feature vector from the audio signal to represent the content information of the audio signal. The text encoder extracts the text feature vector from the caption to reflect the global content information of the caption. Projector modules map the audio feature vector and the text feature vector into the audio embedding and text embedding, respectively, where the dimension of the audio embedding and the text embedding is the same for the alignment between audio and caption content.

2.1. Audio Encoder

Our submitted retrieval systems apply either of the two choices below for the audio encoder structure.

1. **CNN14**: Employ the CNN14 module in pretrained audio neural networks (PANNs) [9] as the audio encoder.
2. **CNN14-Attention**: Combine the CNN14 module and the self-attention module [10] as the audio encoder, following our recent work [6].

In both choices, the mean pooling is used on the extracted audio feature to obtain the output audio feature vector.

2.2. Text Encoder

Our submitted retrieval systems apply either of two choices below for the text encoder structure.

1. **BERT**: Employ the language model BERT [11] as the text encoder.
2. **RoBERTa**: Employ the language model RoBERTa [12] as the text encoder.

In both choices of the text encoder, the caption is mapped into text feature vectors by the text encoder. We select the text feature vector corresponding to the classification token, i.e., "[CLS]", as the output text feature vector of the text encoder.

2.3. Projectors and Contrastive Loss

Our submitted retrieval systems have two projector modules, i.e., audio projector and text projector. The projector modules consist of multi-layer perception (MLP), but the audio and the text projector have different parameters. These projector modules result in

*Corresponding author.

This work was partly supported by the Natural Science Foundation of Heilongjiang Province with Grant No. YQ2020F010 and LH2022F010, and the GHfund with Grant No. 202302026860.

audio embedding and text embedding, respectively. The output dimension of the audio and text embedding should be the same for the contrastive learning loss function. We optimise our submission retrieval systems using the contrastive learning loss function, i.e., NT-Xent loss [13], which realising the contrastive language-audio pretraining (CLAP) system for this multi-modal task.

3. AUTOMATED AUDIO CAPTIONING SYSTEM

Automated audio captioning aims to model the audio feature and predict the caption from the learned audio feature. Existing methods usually employ the encoder-decoder structure. An audio encoder extracts the audio feature, and a text decoder decodes the extracted feature and predicts the caption words [2,5,14,15]. Our submission also uses the encoder-decoder structure, where the audio encoder is initialized by the pre-trained parameters from the CLAP processing in our language-based audio retrieval systems.

3.1. Audio Encoder from CLAP

In our submitted captioning system, the audio encoder includes the PANNs module, i.e., CNN14, but employs the pre-trained parameters of CNN14 in the beforementioned retrieval system for the parameter initialization. Here, we have four choices of the audio encoder structure as follows.

1. **CNN10**: Just employ the pretrained parameters of CNN14 in the retrieval system to initialize a CNN10 PANNs module.
2. **CNN10-GAT**: Employ the pretrained parameters of CNN14 in the retrieval system to initialize a CNN10 PANNs module and a graph attention module in GraphAC [5].
3. **CNN10-keywords**: A CNN10 audio encoder with the keywords prediction in [16].
4. **CNN10-GAT-keywords**: A CNN10-GAT audio encoder with the keywords prediction in [16].

3.2. Text Decoder

In our submitted captioning system, the text decoder uses the cross-attention mechanism to incorporate the audio feature and text information and predict the words in the caption. The text decoder consists of a two-layer Transformer decoder module. Here, a Word2Vec language model [17] is used to convert the words into text embeddings. Then, the Transformer decoder module incorporates the audio feature from the audio encoder and the text embeddings to predict the caption.

3.3. Loss Function

Our submitted captioning system employs the cross-entropy function with the label smoothing [18] as the loss function for training processing.

4. EXPERIMENTS

4.1. Dataset

We use three different datasets to conduct our experiments, WavText5K [19], AudioCaps [20] and Clotho [3].

The audio retrieval system is pretrained on WavText5K and AudioCaps, and then fine-tuned on the Clotho dataset. The audio captioning system first loads the CNN14 module pretrained in the retrieval system. Then the audio captioning system is pretrained on the AudioCaps dataset and fine-tuned on the Clotho dataset.

We noted that some caption does not clearly represent the content of the audio signal in Clotho. Therefore, we employed the GPT-3.5 API [21] to judge the accuracy degree of the caption by comparing it with the audio file name tag, and revised the caption that was judged as an inaccurate caption by GPT-3.5. The revised dataset is called "Revised-Clotho" in this report, and the Revised-Clotho dataset is another choice for fine-tuning the retrieval method.

4.2. Language-base Audio Retrieval

When pretraining on the WavText5K, the batch size of the retrieval method is set as 32. When pretraining on AudioCaps, the batch size is set as 64. During the fine-tuning stage on Clotho or Revised-Clotho, the batch size is set as 32. In both the pretraining and the fine-tuning, the Adam optimizer [22] is used to optimise the retrieval method.

We have eight retrieval systems as follows:

1. CNN14-BERT-Clotho: This system includes a CNN14 audio encoder and a BERT text encoder and is fine-tuned on the Clotho dataset.
2. CNN14-BERT-Revised-Clotho: This system includes a CNN14 audio encoder and a BERT text encoder and is fine-tuned on the Revised-Clotho dataset.
3. CNN14-RoBERTa-Clotho: This system includes a CNN14 audio encoder and a RoBERTa text encoder and is fine-tuned on the Clotho dataset.
4. CNN14-RoBERTa-Revised-Clotho: This system includes a CNN14 audio encoder and a RoBERTa text encoder and is fine-tuned on the Revised-Clotho dataset.
5. CNN14-Attention-BERT-Clotho: This system includes a CNN14-Attention audio encoder and a BERT text encoder and is fine-tuned on the Clotho dataset.
6. CNN14-Attention-BERT-Revised-Clotho: This system includes a CNN14-Attention audio encoder and a BERT text encoder and is fine-tuned on the Revised-Clotho dataset.
7. CNN14-Attention-RoBERTa-Clotho: This system includes a CNN14-Attention audio encoder and a BERT text encoder and is fine-tuned on the Clotho dataset.
8. CNN14-Attention-RoBERTa-Revised-Clotho: This system includes a CNN14-Attention audio encoder and a BERT text encoder and is fine-tuned on the Revised-Clotho dataset.

Based on the above eight systems, we build 4 ensemble systems as our submission for Task 6B:

1. **Submission 1**: The ensemble system of the above eight systems, and each system has the same weight.
2. **Submission 2**: The ensemble system of the above No. 3-8 systems, and each system has the same weight.
3. **Submission 3**: The ensemble system of the above eight systems, and each system has a different weight.
4. **Submission 4**: The ensemble system of the above No. 3-8 systems, and each system has a different weight.

Table 1: Performance of our retrieval systems.

Method	R1	R5	R10	R50	mAP10
CNN14-BERT-Clotho	17.67	42.09	56.46	85.91	28.29
CNN14-BERT-Revised-Clotho	17.44	42.81	56.67	84.63	28.24
CNN14-RoBERTa-Clotho	17.82	42.97	57.65	**87.29**	28.87
CNN14-RoBERTa-Revised-Clotho	18.33	43.44	57.17	86.56	28.97
CNN14-Attention-BERT-Clotho	18.03	42.62	56.11	84.27	28.55
CNN14-Attention-BERT-Revised-Clotho	18.03	42.07	55.83	83.96	28.49
CNN14-Attention-RoBERTa-Clotho	**18.56**	**44.19**	57.32	85.07	**29.37**
CNN14-Attention-RoBERTa-Revised-Clotho	17.95	43.85	**57.95**	84.71	29.03
Submission 1	19.37	45.82	58.95	86.60	30.54
Submission 2	19.83	46.20	59.22	86.74	30.93
Submission 3	19.69	46.07	59.43	86.70	30.93
Submission 4	**20.00**	**46.37**	**59.85**	**86.99**	**31.18**

Table 2: Performance results of our captioning systems.

Method	METEOR	CIDE$_r$	SPICE	SPIDE$_r$	SPIDE$_r$-FL
CNN10-Transformer	18.10	42.23	12.52	27.37	26.88
CNN10-keywords-Transformer	17.92	41.28	12.54	26.91	26.30
CNN10-GAT-Transformer (Submission 1)	18.16	43.81	12.58	28.19	27.48
CNN10-GAT-keywords-Transformer (Submission 2)	18.06	42.61	12.39	27.50	26.67
Submission 3	**18.44**	**45.04**	**12.93**	**28.98**	**28.32**
Submission 4	18.36	44.25	12.75	28.50	27.87

4.3. Automated Audio Captioning

When pretrained on AudioCaps, the batch size is set as 64. When fine-tuned on Clotho, the batch size is set as 32. The AdamW optimizer is used for optimization in both the pretraining and fine-tuning of the audio captioning system.

We have four captioning systems as follows:

1. CNN10-Transformer: Include a CNN10 audio encoder and a Transformer decoder.
2. CNN10-GAT-Transformer: Include a CNN10-GAT audio encoder and a Transformer decoder.
3. CNN10-keywords-Transformer: Include a CNN10-keywords audio encoder and a Transformer decoder.
4. CNN10-GAT-keywords-Transformer: Include a CNN10-GAT-keywords audio encoder and a Transformer decoder.

Then, based on the above captioning systems and our previous work in [23], we have four ensemble systems for submission as follows:

1. **Submission 1**: The CNN10-GAT-Transformer captioning system.
2. **Submission 2**: The CNN10-GAT-keywords-Transformer captioning system.
3. **Submission 3**: The ensemble system of the above four captioning systems, and each system has a different weight.
4. **Submission 4**: The ensemble system of the CNN10-GAT-Transformer and CNN10-GAT-keywords-Transformer captioning systems.

5. RESULTS

5.1. Results for Language-base Audio Retrieval

The performance of our eight retrieval systems and four submitted ensemble systems are shown in Table 1. In this table, R1, R5, R10 and R50 are the recall metric of the top-1, top-5, top-10, and top-50 retrieved audio signals. The mAP10 is the mean average precision of the top-10 retrieved audio signals. Table 1 shows that submission 4 performs best in all metrics. The CNN14-Attention-RoBERTa-Clotho retrieval system achieves the best mAP performance in the eight single systems.

5.2. Results for Automated Audio Captioning

The performance of our four captioning systems and our submissions are shown in Table 2. Here, machine translation metrics (i.e., METEOR) and captioning metrics (CIDE$_r$, SPICE, SPIDE$_r$ and SPIDE$_r$-FL) are adopted for performance evaluation. Table 2 shows that submission 3 performs best in all metrics.

6. CONCLUSION

This report introduces our submissions for DCASE language-based audio retrieval and automated audio captioning tasks. We use the contrastive language-audio pretraining strategy for language-based audio retrieval to build our retrieval systems. For automated audio captioning, we load the parameters of the audio encoder in our retrieval systems for the initialization and use pretraining and fine-tuning strategies to obtain our captioning systems. Both retrieval and captioning submissions include ensemble systems to improve performance.

7. REFERENCES

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

[2] K. Drossos, S. Adavanne, and T. Virtanen, "Automated audio captioning with recurrent neural networks," in *Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, Oct. 2017.

[3] H. Xie, S. Lipping, and T. Virtanen, "Language-based audio retrieval task in DCASE 2022 challenge," in *Proc. of the Detection and Classification of Acoustic Scenes and Events (DCASE) Workshop*, 2022, pp. 216–220.

[4] B. Elizalde, S. Deshmukh, M. A. Ismail, and H. Wang, "CLAP: Learning audio concepts from natural language supervision," *arXiv preprint arXiv:2206.04769*, 2022.

[5] F. Xiao, J. Guan, Q. Zhu, and W. Wang, "Graph attention for automated audio captioning," *IEEE Signal Processing Letters*, vol. 30, pp. 413–417, 2023.

[6] X. Feiyang, Z. Qiaoxi, G. Jian, and W. Wenwu, "Enhancing audio retrieval with attention-based encoder for audio feature representation," in *Proc. of European Signal Processing Conference (EUSIPCO)*. IEEE, 2023 (Submitted).

[7] X. Mei, X. Liu, H. Liu, J. Sun, M. D. Plumbley, and W. Wang, "Language-based audio retrieval with pre-trained models," Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge, Tech. Rep., July 2022.

[8] F. Xiao, J. Guan, H. Lan, Q. Zhu, and W. Wang, "Language-based audio retrieval with pretrained CNN and graph attention," Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge, Tech. Rep., July 2022.

[9] 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.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. of Advances in Neural Information Processing Systems (NIPS)*, vol. 30. Curran Associates, Inc., 2017.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional Transformers for language understanding," in *Proc. of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, 2019, pp. 4171–4186.

[12] 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," *arXiv preprint arXiv:1907.11692*, 2019.

[13] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. of International Conference on Machine Learning (ICML)*. PMLR, 2020, pp. 1597–1607.

[14] F. Xiao, J. Guan, H. Lan, Q. Zhu, and W. Wang, "Local information assisted attention-free decoder for audio captioning," *IEEE Signal Processing Letters*, vol. 29, pp. 1604–1608, 2022.

[15] X. Mei, X. Liu, M. D. Plumbley, and W. Wang, "Automated audio captioning: An overview of recent progress and new challenges," *EURASIP journal on audio, speech, and music processing*, vol. 2022, no. 1, pp. 1–18, 2022.

[16] X. Mei, X. Liu, H. Liu, J. Sun, M. D. Plumbley, and W. Wang, "Automated audio captioning with keywords guidance," Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge, Tech. Rep., July 2022.

[17] G. C. Tomas Mikolov, Kai Chen, "Efficient estimation of word representations in vector space," in *Proc. of International Conference on Learning Representations (ICLR)*, 2013.

[18] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. of the Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016, pp. 2818–2826.

[19] S. Deshmukh, B. Elizalde, and H. Wang, "Audio retrieval with WavText5K and CLAP training," *arXiv preprint arXiv:2209.14275*, 2022.

[20] C. D. Kim, B. Kim, H. Lee, and G. Kim, "AudioCaps: Generating captions for audios in the wild," in *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019, pp. 119–132.

[21] OpenAI, "Models referred to as GPT 3.5," https://platform.openai.com/docs/model-index-for-researchers, 2023, [API documentation of OpenAI].

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

[23] F. Xiao, J. Guan, H. Lan, Q. Zhu, and W. Wang, "Ensemble learning for audio captioning with graph audio feature representation," Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge, Tech. Rep., July 2022.