AUTOMATED AUDIO CAPTIONING WITH MULTI-TASK LEARNING

Technical Report

Zhongjie Ye[1]*,Yuexian Zou*[1,*]*, Fan Cui*[2]*, Yujun Wang*[2]

[1] ADSPLAB, School of ECE, Peking University, Shenzhen, China
[2] Xiaomi Corporation, Beijing, China
zhongjieye@stu.pku.edu.cn, zouyx@pku.edu.cn, cuifan@xiaomi.com, wangyujun@xiaomi.com

ABSTRACT

This technical report describes an automated audio captioning (AAC) model for the Detection and Classification of Acoustic Scenes and Events (DCASE) 2022 Task 6A Challenge. Our model consists of a convolution neural network (CNN) encoder and a single layer (LSTM) decoder with a temporal attention module. In order to enhance the representation in the domain dataset, we use the ResNet38 pretrained on the AudioSet dataset as our audio encoder and finetune it with keywords of nouns and verbs as labels which are extracted from the captions. For training the whole caption model, we first train the model with the standard cross entropy loss, and fine-tune it with reinforcement learning to directly optimize the CIDEr score. Experimental results show that our single model can achieve a SPIDEr score of 31.7 on the evaluation spilt.

Index Terms— Audio caption, pre-training, keyword classification, multi-task learning

1. INTRODUCTION

Automated audio captioning (AAC) is a new and challenging task that involves different modalities. It could be described as generating a textual description (i.e. caption) given an audio signal, where the caption should be as close as possible to a human-assigned one [1]. Since the automated audio captioning task was held in Detection and Classification of Acoustic Scenes and Events (DCASE) 2020 and 2021 challenges, it has attached more attention recently [2, 3, 4, 5].

Most existing audio captioning models are based on encoder-decoder frameworks, which usually consists of an audio encoder to extract acoustic information and a language decoder to generate the sentence with the extracted features. CNN-RNN [2, 4] and CNN-Transformer [3, 5] based audio captioning model are widely adopted by most methods. Recently, Transformer-only audio captioning model is proposed and shows competitive performance [6]. Because of the small-scale dataset, many methods adopt transfer learning like initializing the audio encoder with the PANNs [7], pre-training the encoder with extracted keywords from the captions [2, 8] and so on. In addition, Yuan *et al.* uses extra audio clips downloaded from the website to create large-scale audio-caption pairs. Its results show that the caption model can get better performance with sufficient training samples.

Inspired by the previous works, we extend our proposed method (*i.e.* MAAC) which presents a great performance in the audio captioning task. Specifically, we use the pre-trained ResNet38 [7] as our audio encoder then finetune it with the keywords extracted from the captions, and a decoder with a multi-modal attention module. For training the whole caption model, firstly we freeze weights of the audio encoder and train the language decoder with standard cross entropy loss and tagging loss. Then the pre-trained caption model is finetuned by optimizing CIDEr-D via a reinforcement learning method (*i.e.* SCST [9]).

The organization of the paper is as follows. Section 2 introduces our proposed method. The experimental setup and the results are shown in Section3 Finally, the conclusion is presented in Section 4.

2. PROPOSED METHOD

In this section, we will introduce the architecture and training details of our proposed method. Firstly we use the MAAC as our baseline model which is proposed in DCASE 2021 challenge. Then we will introduce the three training stages of the proposed method.

MAAC consists of an audio encoder and a language decoder with the temporal attention mechanism. Specifically, we use the ResNet38 which is pretrained on AudioSet [10] and adopt a hierarchy structure to combine the multi-scale features. The details of the architecture of our audio encoder are shown in Table 2. Then we will introduce our three training stages as follows.

The first stage is training the audio encoder with keywords (*i.e.* nouns and verbs) that are extracted from the 5 captions of each audio clip to form the training labels. The audio encoder consists of six convolutional blocks. We utilize the hierarchy structure to refine the model and use f_1, f_2 and f_3 to represent the output of FC_1, FC_2 and FC_3 respectively, and $\hat{y}$ represents the output of CLS and GAP means global average pooling. Then we use f_1, f_2 and f_3 to obtain the predictions $\hat{y} \in \mathbb{R}^N$ where N is the number of keywords.

$$\hat{y} = \sigma(Linear(concat(f_1, f_2, f_3))) \tag{1}$$

$$\mathcal{L}_{tag} = \mathcal{L}_{bce}(y, \hat{y}) = -\sum_{i=1}^{N} y(i) log\ \hat{y}(i) \tag{2}$$

where the ground truth $y \in \mathbb{R}^N$, σ means sigmoid activation function, $\hat{y}$ is the output of the CLS. Standard binary cross entropy loss is used as the loss function, which is defined as the negative log likelihood of the expected keyword y_i given transcription $\hat{y}_i$ at the position i.

The second training stage is that we freeze the main blocks of the pretrained audio encoder and just train the language decoder

*Yuexian Zou is the corresponding author

Table 1: Performances of different architectures on the Clotho evaluation splits. †denotes our own implementation. B-n, M, R-L, C, S, Sr denote BLEU-n, METEOR, ROUGE-L, CIDEr, SPICE, and SPIDEr, respectively. For all metrics, higher values indicate better performance.

	CE Optimization							RL Optimization						
Model	B-1	B-4	M	R-L	C	S	Sr	B-1	B-4	M	R-L	C	S	Sr
Baseline	55.5	15.6	16.4	36.4	35.8	10.9	23.3	-	-	-	-	-	-	-
submission1	57.8	16.4	**17.9**	38.0	42.8	12.3	27.5	64.7	19.4	18.5	41.4	50.3	**13.2**	31.7
submission2	**58.1**	**17.6**	17.8	**38.2**	**44.5**	**12.7**	**28.6**	64.5	18.3	18.6	40.8	49.5	13.1	31.3
submission3	57.5	16.5	17.6	37.7	41.9	12.5	27.2	64.6	18.6	18.6	40.9	49.7	11.9	30.8
submission4	-	-	-	-	-	-	-	**66.3**	**19.5**	**18.9**	**41.6**	**52.0**	12.6	**32.3**

Table 2: The architecture of the audio encoder. GAP means the global average pooling layer. Linear(128, 2048) means that the input dimension of the fully-connected layer is 128 and the output dimension is 2048. We take FC_1 as an example that the input features firstly go through the global average pooling layer, and then are passed into a fully-connected layer with ReLU activation function.

X	log mel spectrogram
Conv_1	(Conv 3 × 3 @ 64, BN, ReLU) ×2 Pooling 2 × 2
Conv_2	(BasicB @ 64) ×3 Pooling 2 × 2
Conv_3	(BasicB @ 128) ×4 Pooling 2 × 2
Conv_4	(BasicB @ 256) ×6 Pooling 2 × 2
Conv_5	(BasicB @ 512) ×3 Pooling 2 × 2
Conv_6	(Conv 3 × 3 @ 2048, BN, ReLU) ×2 Pooling 2 × 2
$\mathbf{FC_1}$	$GAP, Linear(128, 2048), ReLU$
$\mathbf{FC_2}$	$GAP, Linear(256, 2048), ReLU$
$\mathbf{FC_3}$	$GAP, Linear(2048, 2048), ReLU$
CLS	$Linear(6144, 300), Sigmoid$

with cross entropy loss and tagging loss:

$$L_{CE} = -\frac{1}{L}\sum_{l=1}^{L} logp(y_l|\boldsymbol{\mathcal{V}}, \boldsymbol{y_{l-1}})$$
$$L_{total} = L_{tag} + L_{CE} \quad (3)$$

where $\boldsymbol{\mathcal{V}}$ denotes the acoustic features from the audio encoder and $p(y_l|\boldsymbol{\mathcal{V}}, \boldsymbol{y_{l-1}})$ is the posterior probability of the word over the vocabulary.

The final stage is that we conduct reinforcement learning to directly optimize the evaluation metric (*i.e.* CIDEr) after the caption model is trained by CE and tagging loss.

3. EXPERIMENT

Experiment setups: We train and evaluate our proposed method on the Clotho v2 [11] dataset which contains a total of 5,929 audio clips labeled with 5 captions. The Clotho v2 consists of 3,839 training audio clips, 1,045 validation audio clips, and 1,045 evaluation audio clips. In the training stage, we combine the training and validation sets for training the whole captioning model and evaluate the model on the evaluation set, We convert all tokens of sentences to lower-cases and remove all punctuation marks resulting in 4368 words including special tokens "BOS", "EOS", and "PAD". The architecture settings are the same to MAAC [2].

Training details: In the phase of pre-training the audio encoder, it is trained for 40 epochs by Adam optimizer with the learning rate of 5×10^{-4}. Then we freeze the weights of the audio encoder and just train the language decoder for 30 epochs with the learning rate of 3×10^{-4}. Finally, we optimize CIDEr-D score with SCST [9] for 35 epochs with an initial learning rate of 5×10^{-5}. In the inference stage, we adopt beam search with a beam size of 4.

Experimental results: The experimental results of the submissions are shown in Table 1. The details of the submission methods are following:

- submission1. MAAC is trained by the standard cross entropy loss and fine-tuned by reinforcement learning.
- submission2. MAAC is trained by the standard cross entropy combined with the tagging loss and fine-tuned by reinforcement learning.
- submission3. CNN-LSTM with a temporal attention mechanism is trained by the standard cross entropy loss and fine-tuned by reinforcement learning.
- submission4. Ensemble of four RL fine-tuned CNN-LSTM models.

4. RESULTS

The experimental results are presented in Table 1. We can see that MAAC trained by standard cross entropy and tagging loss outperforms other methods, producing SPIDEr as high as 28.6. In addition, MAAC can reach the highest SPIDEr score at 31.7 after the reinforcement learning optimization.

5. CONCLUSION

The technical report describes our proposed method submitted to DCASE2022 challenge Task 6A. Our proposed method can get SPIDEr score of 28.6 by standard cross entropy and tagging loss. In addition, MAAC can get the best performance that is fine-tuned by reinforcement learning optimization.

6. REFERENCES

[1] K. Drossos, S. Adavanne, and T. Virtanen, "Automated audio captioning with recurrent neural networks," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2017, pp. 374–378.

[2] Z. Ye, H. Wang, D. Yang, and Y. Zou, "Improving the performance of automated audio captioning via integrating the acoustic and semantic information," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2021 Workshop (DCASE2021)*, Barcelona, Spain, November 2021, pp. 40–44.

[3] X. Mei, Q. Huang, X. Liu, G. Chen, J. Wu, Y. Wu, J. Zhao, S. Li, T. Ko, H. L. Tang, X. Shao, M. D. Plumbley, and W. Wang, "An encoder-decoder based audio captioning system with transfer and reinforcement learning for DCASE challenge 2021 task 6," DCASE2021 Challenge, Tech. Rep., July 2021.

[4] X. Xu, Z. Xie, M. Wu, and K. Yu, "The SJTU system for DCASE2021 challenge task 6: Audio captioning based on encoder pre-training and reinforcement learning," DCASE2021 Challenge, Tech. Rep., July 2021.

[5] Y. Koizumi, R. Masumura, K. Nishida, M. Yasuda, and S. Saito, "A transformer-based audio captioning model with keyword estimation," *Proc. Interspeech*, pp. 1977–1981, 2020.

[6] X. Mei, X. Liu, Q. Huang, M. D. Plumbley, and W. Wang, "Audio captioning transformer," in *Proceedings of the 6th Detection and Classification of Acoustic Scenes and Events 2021 Workshop (DCASE2021)*, Barcelona, Spain, November 2021, pp. 211–215.

[7] 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.

[8] M. Wu, H. Dinkel, and K. Yu, "Audio caption: Listen and tell," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 830–834.

[9] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7008–7024.

[10] J. F. Gemmeke, D. P. 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 *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 776–780.

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