AUTOMATED AUDIO CAPTIONING WITH KEYWORDS GUIDANCE

Technical Report

Xinhao Mei, Xubo Liu, Haohe Liu, Jianyuan Sun, Mark D. Plumbley, Wenwu Wang

Centre for Vision, Speech, and Signal Processing (CVSSP),
University of Surrey, UK
{x.mei, xubo.liu, haohe.liu, jianyuan.sun, m.plumbley, w.wang}@surrey.ac.uk

ABSTRACT

This technical report describes an automated audio captioning system we submitted to Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge 2022 Task 6a. The proposed system is built on an encoder-decoder architecture we submitted to DCASE 2021 Challenge Task 6 last year, where the encoder is a pre-trained 10-layer convolutional neural network and the decoder is a Transformer network. In this new submission, we investigate the use of keywords estimated from input audio clips to guide the caption generation process. The results show that keywords guidance can improve the system performance especially when the pre-trained encoder is frozen, and can also reduce the variance of the results when the model is trained with different seeds. The overall system consists of a pre-trained keywords estimation model and a CNN-Transformer audio captioning model. The captioning model is first trained via the cross-entropy loss and then fine-tuned with reinforcement learning to optimize the evaluation metric CIDEr. The proposed system significantly improves the scores of all the evaluation metrics as compared to the baseline system.

***Index Terms*—** Audio captioning, audio tagging, cross-modal task, reinforcement learning

1. INTRODUCTION

Automated audio captioning (AAC) [1, 2], a cross-modal translation task aiming at describing the overall content of an audio clip using natural language (sentences), has attracted increasing attention recently. This task has the potential to be applied in a variety of applications such as multimedia data retrieval, hearing-impaired assistance, and human-computer interaction. With AAC being held as a challenge task in DCASE 2020 and 2021 [3], numerous methods have been proposed and the performance of the audio captioning systems has been significantly improved [4, 5, 6, 7, 8, 9].

This technical report describes an automated audio captioning system submitted to DCASE Challenge 2022 Task 6a. The proposed system is built on a CNN-Transformer model we submitted to DCASE Challenge 2021 [10]. In this model, a 10-layer pre-trained convolutional neural network (CNN) encoder is used to extract audio features from input audio clips and a Transformer decoder is used to generate captions based on the extracted audio features. Audio is a time-series signal with variable lengths, where sound events can occur over arbitrary time frames and are often overlapped. The overlapping sound events make it challenging to directly map an audio signal into natural language description. Furthermore, each sound event can be described using different words, making this task even more challenging. In our new submission, a module for keywords estimation is introduced to guide the caption generation process. Estimating keywords from audio clips is essentially an audio tagging task, therefore, it is relatively easier than directly translating audio clips into sentences. The keywords may convey the class information of the presented audio events and objects making sounds in the input audio clips, therefore, the decoder can generate more accurate captions while referring to the keywords.

The use of keywords to improve captioning systems has been investigated in the literature. Koizumi et al. [11] argued that a sound event can be described with different words, which could lead to a word-selection indeterminacy problem. They proposed to add a branch for keyword estimation after the audio encoder and use keywords to guide the caption generation. Ye et al. [6] adopted a similar method to estimate keywords, however, they claimed that their captioning model did not converge when combined with the keyword estimation branch. In addition, Han et al. [5] retrieve audio clips in the dataset that are similar to the input audio clip, and extract keywords from captions of these retrieved audio clips to enhance the decoder. Rather than adding a branch for keyword estimation after the audio encoder, we employ a separate audio tagging model to estimate keywords, and the estimated keywords are further concatenated with the extracted audio features from the encoder before being fed into the decoder to guide the caption generation. Our experimental results show that the keyword guidance can improve the system performance significantly when the pre-trained encoder is frozen. Furthermore, the incorporation of keywords reduces the variance of the models trained with difference seeds.

In conclusion, our submitted system consists of a pre-trained keyword estimation model and a standard CNN-Transformer captioning model. The captioning model is first trained via a cross-entropy loss and then fine-tuned via reinforcement learning. The submitted system significantly improves the scores of all the evaluation metrics, as compared to the baseline system. The remainder of this technical report is organized as follows. In Section 2, our method is introduced in detail. The experiments and results are discussed in Section 3. Finally, we conclude this work in Section 4.

2. SYSTEM DESCRIPTION

Fig. 1 shows the overview of our system. We first introduce the model architecture and then describe the rule-based keywords extraction pipeline in this section.

2.1. Model architecture

Transfer learning has been successfully applied to address the data scarcity problem and improve the system performance in the litera-

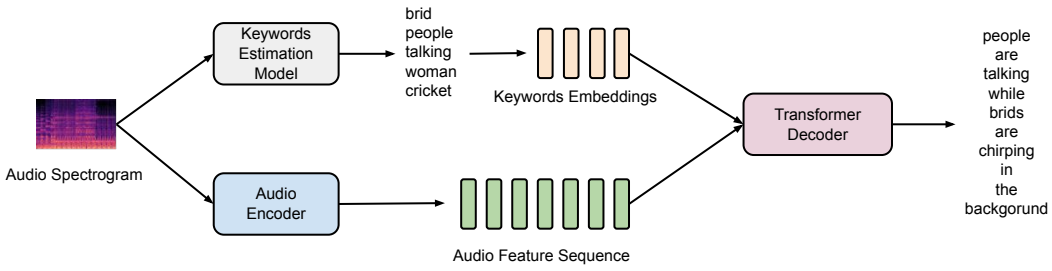

Figure 1: The diagram of the submitted system, which consists of a keywords estimation model and a CNN-Transformer captioning model.

ture [10]. Therefore, a 10-layer CNN from pre-trained audio neural networks (PANNs) [12] is employed for both the keyword estimation model and the audio encoder in the captioning model. The 10-layer CNN consists of 4 convolutional blocks and each block has 2 convolutional layers with a kernel size of 3×3. Batch normalization [13] and a ReLU [14] nonlinear layer are applied after each convolutional layer. The last convolutional block is followed by two linear layers to further increase the feature representation ability [12] and to adjust the dimension of the outputs.

The language decoder is a standard Transformer decoder [15] that is built on the self-attention mechanism. A word embedding layer is applied before the Transformer decoder to convert the words into vectors, which is initialized with the pre-trained Word2Vec model [16] and kept frozen during training. The Transformer decoder consists of two identical blocks, each of which has a masked self-attention layer, a cross-attention layer and a multi-layer perceptron (MLP) block. A linear layer with softmax activation is used after the Transformer decoder to output a word probability over the vocabulary.

Given an audio clip as input, the keywords estimation model is used to get the top-k keywords, and then the audio encoder is used to obtain the audio feature sequence. The top-k keywords are further transformed to keywords embeddings through a pre-trained word embedding layer, which is the same as that in the decoder. The keywords embeddings are then concatenated with the audio feature sequence, and fed into the decoder for word generation. Here, k is set to five. The keywords estimation model is first pre-trained as an audio tagging task and kept froze during the captioning training process. Therefore, the estimated keywords remain fixed for a particular audio clip throughout the whole training process for captioning.

2.2. Keyword extraction

Keywords may convey the class information of the presented audio events and objects making sounds in the input audio clips. Estimation of keywords can be considered as an audio tagging task. Here we apply a set of rules to the captions in the Clotho dataset to extract keywords.

Each audio clip in the Clotho dataset [17] has five human-annotated reference captions. For each audio clip, we first tokenize its caption into words, and then tag the part-of-speech (POS) of each word using the natural language toolkit (NLTK) [18]. If the word is a noun, we then convert it into its lemma, and if the lemma is in the vocabulary, we add the lemma into the list of potential keywords for that audio clip. After processing all five captions for that audio clip, the words occurring more than twice in the list of potential keywords are finally selected as the keywords. Using the preceding rules, we get a total of 456 keywords.

3. EXPERIMENTS

3.1. Dataset

All experiments are carried out on the official dataset of DCASE Challenge 2022 Task 6a, Clotho v2 [17]. The published development set includes three subsets, development-training, development-validation, and development-testing, each of which contains 3839, 1045 and 1045 audio clips, respectively. In the captioning task of the DCASE Challenge, all the data in the development set can be used for training the final submitted models. We randomly select 100 audio clips as validation set for model selection and use the remaining 5829 audio clips for training the final submitted models.

In this technical report, we present the results on the development-testing set, where the models are trained and validated on the development-training and development-validation sets, respectively.

3.2. Data pre-processing

The texts within the captions in the dataset are first converted to lower case with punctuation removed. The vocabulary has 4367 words including two special tokens "<sos>" and "<eos>" to indicate the start and the end of each caption. The sampling rate of the audio clips is 44.1 KHz. The acoustic features we used are the log mel-spectrograms, which are extracted with a 1024-points Hanning window with a hop size of 512-points.

3.3. Experimental setups

The keywords estimation model is trained for 20 epochs with a learning rate of 1×10^{-4}, and mean average precision (mAP) score on the validation set is used to select the final model to avoid overfitting. For the training of the captioning model, the batch size is set to 32. The training can be divided into two stages, i.e. a cross-entropy training stage and a reinforcement learning training stage. For the cross-entropy training, the model is trained for 30 epochs and the learning rate is linearly increased to 1×10^{-4} using warm-up in the first 5 epochs, and decreased by a factor of 10 every 10 epochs after the warm-up. The model achieving the highest SPIDEr [19] score on the validation set is then selected for fine-tuning using reinforcement learning to optimize CIDEr, where the self-critical sequence training (SCST) [20] method is employed and the model is fine-tuned for 100 epochs with a constant learning rate of 5×10^{-5}. In addition, SpecAugment [21] is applied throughout all the training

Table 1: Results on the Clotho development-testing set. B_1, B_4, RG, ME, CD, SP and SD denote $BLEU_1$, $BLEU_4$, $ROUGE_l$, METEOR, CIDEr, SPICE and SPIDEr, respectively. Proposed: a single CNN-Transformer model with keyword guidance.

	Cross-entropy training							**RL fine-tuning**						
Model	B_1	B_4	RG	ME	CD	SP	SD	B_1	B_4	RG	ME	CD	SP	SD
Baseline	55.5	15.6	36.4	16.4	35.8	10.9	23.3	-	-	-	-	-	-	-
Proposed	57.3	16.7	38.2	17.7	40.9	12.3	26.6	66.7	18.0	40.1	18.3	47.5	12.7	30.1

Table 2: Results of the ablation study of the effects of keywords on the CNN-Transformer captioning model. The models are trained with the cross-entropy loss without fine-tuning using reinforcement learning.

Encoder Frozen	Keywords	B_1	B_4	RG	ME	CD	SP	SD
Yes	No	$56.1_{\pm 0.08}$	$15.8_{\pm 0.37}$	$37.6_{\pm 0.26}$	$17.1_{\pm 0.17}$	$38.6_{\pm 0.76}$	$11.7_{\pm 0.12}$	$25.1_{\pm 0.45}$
No	No	$56.7_{\pm 0.81}$	$16.2_{\pm 0.21}$	$38.1_{\pm 0.47}$	$17.7_{\pm 0.05}$	$39.9_{\pm 0.62}$	$12.2_{\pm 0.17}$	$26.1_{\pm 0.39}$
Yes	Yes	$56.8_{\pm 0.21}$	$16.3_{\pm 0.22}$	$37.8_{\pm 0.25}$	$17.4_{\pm 0.04}$	$40.4_{\pm 0.40}$	$12.1_{\pm 0.08}$	$26.3_{\pm 0.24}$
No	Yes	$56.9_{\pm 0.29}$	$16.8_{\pm 0.12}$	$38.3_{\pm 0.08}$	$17.5_{\pm 0.14}$	$40.6_{\pm 0.29}$	$12.1_{\pm 0.12}$	$26.4_{\pm 0.17}$

stages, and label smoothing [22] is used in the cross-entropy training stage. A beam search with a beam size of 3 is used for decoding during inference.

Four submissions are allowed in this task. The details of our submissions are as follows:

- Submission 1: Single model without keywords guidance.
- Submission 2: Single model with keywords guidance.
- Submission 3: Ensemble of models without keywords guidance.
- Submission 4: Ensemble of models with keywords guidance.

3.4. Results

Table 1 shows the results of our proposed model (single model with keyword guidance) on the development-testing set. It can be seen clearly that the proposed model improves all the metrics especially after the optimization of CIDEr using reinforcement learning.

We further investigated the effects of the keywords on our model. We compared 4 training settings: (a) the CNN-Transformer model without using keywords estimation and the audio encoder is frozen during training, (b) the CNN-Transformer model without using keywords estimation and the audio encoder is fine-tuned during training, (c) the CNN-Transformer model with keywords estimation and the audio encoder is frozen during training, (d) the CNN-Transformer model with keywords and the audio encoder is fine-tuned during training. Since the captioning models generally exhibit high variance [23], models for each setting are trained with three different training seeds, and the mean and standard deviation of the metrics are reported in Table 2.

When keywords are not applied, the results obtained with models trained with both audio encoder frozen and fine-tuned show larger variance especially on the CIDEr metric, which indicates that the keywords guidance can reduce the variance of captioning models. When the audio encoder is frozen, the keywords guidance can significantly improve the system performance, and the frozen models even outperform the fine-tuned models trained without keywords guidance. With keywords guidance, the fine-tuned model slightly improve some metrics as compared with the frozen models.

4. CONCLUSION

This technical report briefly describes our system submitted to DCASE 2022 Task 6a. We introduce a keyword estimation model to improve the CNN-Transformer captioning model that we submitted last year. The results show that keywords guidance can improve the system performance especially when the encoder is frozen, and also reduce the variance of the results when the model is trained with different seeds.

5. ACKNOWLEDGMENT

This work is partly supported by a Newton Institutional Links Award from the British Council, titled "Automated Captioning of Image and Audio for Visually and Hearing Impaired" (Grant number 623805725) and a grant EP/T019751/1 from the Engineering and Physical Sciences Research Council (EPSRC). For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

6. REFERENCES

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

[2] X. Mei, X. Liu, M. D. Plumbley, and W. Wang, "Automated audio captioning: An overview of recent progress and new challenges," *arXiv preprint arXiv:2205.05949*, 2022.

[3] http://dcase.community/challenge2022/.

[4] Y. Koizumi, Y. Ohishi, D. Niizumi, D. Takeuchi, and M. Yasuda, "Audio captioning using pre-trained large-scale language model guided by audio-based similar caption retrieval," *arXiv preprint arXiv:2012.07331*, 2020.

[5] Q. Han, W. Yuan, D. Liu, X. Li, and Z. Yang, "Automated audio captioning with weakly supervised pre-training and word selection methods," in *Proceedings of the 6th Detection and Classification of Acoustic Scenes and Events 2021 Workshop (DCASE2021)*, Barcelona, Spain, November 2021, pp. 6–10.

[6] Z. Ye, H. Wang, D. Yang, and Y. Zou, "Improving the performance of automated audio captioning via integrating the acoustic and textual information," *DCASE2021 Challenge*, Tech. Rep., July 2021.

[7] X. Mei, X. Liu, J. Sun, M. D. Plumbley, and W. Wang, "Diverse audio captioning via adversarial training," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022.

[8] X. Liu, X. Mei, Q. Huang, J. Sun, J. Zhao, H. Liu, M. D. Plumbley, V. Kılıç, and W. Wang, "Leveraging pre-trained BERT for audio captioning," in *30th European Signal Processing Conferences (EUSIPCO)*, 2022.

[9] 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.

[10] X. Mei, Q. Huang, X. Liu, G. Chen, J. Wu, Y. Wu, J. Zhao, S. Li, T. Ko, H. L. Tang, M. D. Plumbley, and W. Wang, "An encoder-decoder based audio captioning system with transfer and reinforcement learning," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, 2021.

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

[12] 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.

[13] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*. PMLR, 2015, pp. 448–456.

[14] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 315–323.

[15] 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 *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

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

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

[18] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*. O'Reilly Media, Inc, 2009.

[19] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy, "Improved image captioning via policy gradient optimization of SPIDEr," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 873–881.

[20] 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.

[21] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," *Interspeech 2019*, Sep 2019. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2680

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

[23] W. Zhu, X. Wang, P. Narayana, K. Sone, S. Basu, and W. Y. Wang, "Towards understanding sample variance in visually grounded language generation: Evaluations and observations," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 8806–8811. [Online]. Available: https://aclanthology.org/2020.emnlp-main.708