AUDIO CAPTIONING BASED ON TRANSFORMER AND PRE-TRAINING FOR 2020 DCASE AUDIO CAPTIONING CHALLENGE

Technical Report

Yusong Wu[1], Kun Chen[1], Ziyue Wang[2], Xuan Zhang[2], Fudong Nian[3], Shengchen Li[1], Xi Shao[2]

[1] Beijing University of Posts and Telecommunications, Beijing, China,
{wuyusong, cksy1118, shengchen.li}@bupt.edu.cn
[2] Nanjing University of Posts and Telecommunications, Nanjing, China,
{1218012222, 1218012223, shaoxi}@njupt.edu.cn
[3] Anhui University, Anhui, China,
nianfudong@ahu.edu.cn

ABSTRACT

This report proposes an automated audio captioning model for the 2020 DCASE audio captioning challenge. In this challenge, a model is required to be trained from scratch to generate natural language descriptions of a given audio signal. However, as limited data available and restrictions on using pre-trained models trained by external data, training directly from scratch can result in poor performance where acoustic events and language are poorly modeled. For better acoustic event and language modeling, a sequence-to-sequence model is proposed which consists of a CNN encoder and a Transformer decoder. In the proposed model, the encoder and word embedding are firstly pre-trained. Regulations and data augmentations are applied during training, while fine-tuning is applied after training. Experiments show that the proposed model can achieve a SPIDEr score of 0.227 on audio captioning performance.

Index Terms— audio captioning, acoustic event detection, deep learning,

1. INTRODUCTION

Automated audio captioning is an multimodal translation task where the model outputs a textual description given an audio signal. Besides modeling acoustic scenes and events, audio captioning models also need to model natural language and other information in audio including spatiotemporal relationships of sources (e.g. turned on the gas on a gas canister and **then** switched the ignition key), foreground versus background discrimination (e.g. machine running and people talking in the **back**), concepts (e.g. "**muffled** sound"), and physical properties of objects and environment (e.g. "**hard** surface, **wooden** door") [1]. Audio captioning has received more attention recently [1–6]. In Detection and Classification of Acoustic Scenes and Events (DCASE) 2020, an audio captioning challenge is announced. In DCASE 2020 audio captioning challenge, Clotho dataset [1] is used which contains 4981 audio clips in 15-30 seconds each with 5 captions in 8-20 English words. Among the data in the dataset, 60% are used as development (training) set and 20% are used as evaluation set while the last 20% remains as test set. Audio captioning model is required to build without using any additional annotation or external pre-trained models, and submissions are ranked based on SPIDEr [7] score.

Similar to image captioning, audio captioning requires to extract feature representations of input space and map into natural language space. This requires effective feature extraction and language modeling. As limited data availbility for audio captioning tasks comparing to classification, direct training from scratch in an end-to-end manner may not enough to train an effective feature extractor. Thus, as image captioning [8–12], audio captioning models [3] may use a pre-trained feature extractor which usually is a Convolutional Neural Network (CNN) based network trained on large-scale classification task such as AudioSet [13].

In DCASE 2020 audio captioning challenge, external data and pre-trained models are not allowed. Effectively training a feature extractor only using a small amount of data and annotation provided by the dataset is important. In this report, a sequence-to-sequence model is proposed to be trained using only caption annotation. The proposed model consists of a 10-layer CNN [14] encoder and a Transformer [15] decoder for feature extraction and natural language generation. To achieve better feature extraction and language modeling, pre-training is applied on the CNN encoder and word embedding. To further improve performance and prevent over-fitting, label smoothing and data augmentation are applied during training, while a fine-tuning with small learning rate with parameters in the encoder frozen is applied after training is finished.

Experiment results show that the proposed method outperforms the previous baseline model and reached a SPIDEr score of 0.227 on audio captioning.

2. PROPOSED MODEL

A sequence-to-sequence model is proposed for audio captioning which consists of an encoder to extract feature sequence from input log Mel-spectrogram and a decoder which maps the feature sequence to output sentence. The encoder of the model is firstly pre-trained, and the trained encoder parameters are loaded and freezed in training initialization. Last, fine-tuning is applied by using a small learning rate. The diagram of the proposed model is shown in Fig. 1.

2.1. Pre-training

The CNN encoder in the model plays an important rule in extracting features of the input audio. However, direct training may not

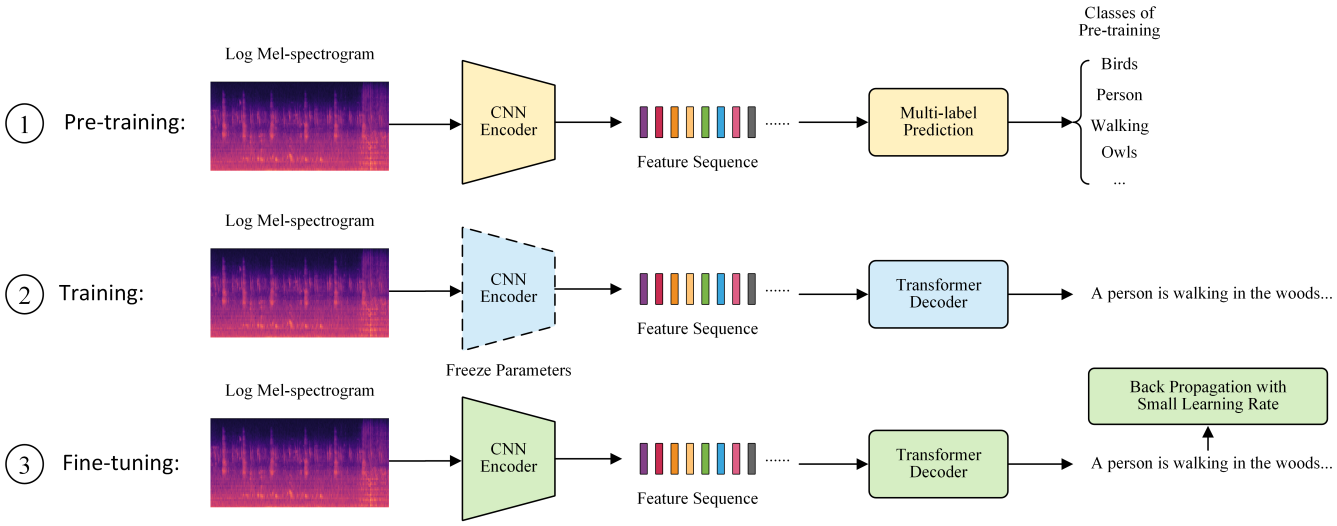

Figure 1: The overview diagram of the proposed model. The encoder extracts a sequence of feature vectors of the input log Mel-spectrogram, and the decoder generates each word while attending to the feature sequence. The encoder is firstly pre-trained by a multi-label prediction task. Fine-tuning is applied after training. The CNN encoder showed on the diagram remain same architecture during pre-training, training and fine-tuning.

be sufficient to train the encoder which makes decoder hard to optimize. Thus, to mroe effectively optimize the decoder during training, pre-training is applied here before training. Here, the CNN encoder is pre-trained by converting the audio captioning task into a multi-label classification task where 300 classes are used. The classes used in encoder pre-training is obtained by selecting a subset of the word in caption vocabulary. First, 20 words with the highest frequency and the words that have no more than 2 letters are excluded from the vocabulary which mainly contains meaningless words such as article words. Then, words in caption vocabulary are transformed into its original form by finding the stems of words such as "-ing", "-ly", "-d", "-s", etc, while the word frequency of the transformed words is added to the frequency of its original form. Last, 300 words with the highest frequency remain in vocabulary is selected as classes for pre-training.

In encoder pre-training, all 5 captions of each audio are combined to form one training label. Words in each caption are also transformed into its original form using the same rules above. The label of each audio is a multi-hot vector where each index of the word occurs in captions equal to 1.

The word embedding in the decoder is also pre-trained to improve language modeling performance. The word embedding is pre-trained using Word2Vec model [16] via python package `genism` [17]. Each caption sentence in the training set is used to form a training corpus.

2.2. Encoder

A 10-layer CNN [14] (CNN10) is used as the encoder in the proposed model. The CNN used here is for feature extraction of input spectrogram. As demonstrated to be successful in audio pattern recognition [14], CNN10 in [14] is adapted and used. The reason for choosing a relatively simple CNN rather than a deeper CNN such as VGG [18] or ResNet [19] is mainly to prevent over-fitting.

The CNN10 consists of four convolution blocks each having two 3×3 convolution layers with ReLU activation function and batch normalization, with 2×2 average pooling layer between the blocks. The number of channels of the convolution blocks are 64, 128, 256, 512, respectively. The output of the CNN is a 512 channel feature sequence down-sampled 16 times both in the time dimension and frequency dimension. Then, each feature vector in the frequency dimension is averaged. Last, 2 layers of fully-connected neural networks are used to map the dimension of each vector in feature sequence into the number of hidden dimensions used in the decoder for attention computation.

2.3. Decoder

The decoder used in the proposed model is a standard transformer [15] consist of multi-head self-attention on text sequence and multi-head encoder-decoder attention on extracted feature sequence. The reason for choosing the Transformer model as the decoder is because of its state-of-the-art performance on natural language processing and the non-recurrence computing of its structure which helps prevent gradient vanishing or exploding. The decoder uses a 2-layer Transformer with a hidden dimension of 192 and 4 heads.

2.4. Regulations and Data Augmentations

To improve performance and avoid over-fitting, regulations and data augmentations are applied. Label smoothing [20] is applied as regulation to improve the generalization of the model by introducing penalty to over-confident prediction. In label smoothing, the one-hot word prediction label is replaced by mixing of original distribution $q(k|x) = \delta_{k,y}$ with a uniform distribution $u(k) = 1/K$ where K is the number of word classes, such that

$$q'(k) = (1 - \epsilon)\delta_{k,y} + \frac{\epsilon}{K}.$$

Table 1: Scores of each metric for model performance on evaluation data.

Methods	$BLEU_1$	$BLEU_2$	$BLEU_3$	$BLEU_4$	$ROUGE_L$	METEOR	CIDEr	SPICE	**SPIDEr**
Baseline [1]	0.389	0.136	0.055	0.015	0.262	0.084	0.074	0.033	0.054
Proposed model	0.534	0.343	0.230	0.151	0.356	0.160	0.346	0.108	**0.227**

SpecAugment [21] is applied as data augmentation to increase the effective size of existing training data as demonstrated to be effective in performance improvement in automatic speech recognition models [21]. In SpecAugment, frequency masks and time masks are randomly applied onto the log Mel- spectrogram before input to the CNN encoder.

2.5. Fine-tuning

In the training process, only the parameters of the decoder is trained while the parameters of the encoder is freezed. After the training is finished, fine-tuning is applied which was found to improve the model performance. The model which yields the highest performance in evaluation is selected and continue training for 20 epochs with a small learning rate of 10^{-4}.

3. EXPERIMENT SETUP

In training, batch size of 16 is used with a learning rate of 3×10^{-4} and a $l2$ regularization applied to all trainable parameters with factor $\lambda = 10^{-6}$. Label smoothing [20] is set with $\epsilon = 0.1$ and SpecAugment [21] is set with 2 frequency masks and 2 time masks in parameter $W = 40, T = 30$ with a probability of 0.2. Dropout in $p = 0.2$ is applied to CNN encoder and Transformer decoder.

In the training process, each audio is combined with each one of five caption annotations and used as a sample. In the evaluation, each audio is used as one sample and all five captions are used as reference for metric computation. The log Mel-spectrogram input is obtained by first getting the 64 Mel-band Mel-spectrogram of the audio, then converting the amplitude into a decibel scale. In the inference stage, a beam search with a beam size of 3 is implemented to achieve better decoding performance.

The CNN encoder is pre-trained for 60 epochs before training, and the Word2Vec model is trained 1000 epochs with random parameter initialization. The proposed model is trained 100 epochs before the model with the highest performance is selected for fine-tuning. The selection of the model is based on SPIDEr [7] score of the evaluation performance. As the challenge allows to submit up to 4 results, 4 models that has the highest SPIDEr score is selected for result submission.

4. RESULTS

The performance of the proposed model is shown in Tab. 1 comparing with a baseline model[1] on the same dataset [1] where 9 metrics are shown. Among the metrics used, $BLEU_n$ [22] measures a modified n-gram precision. $ROUGE_L$ [23] measures a score based on the longest common subsequence. METEOR [24] measures a harmonic mean of weighted unigram precision and recall. CIDEr [25] measures a weighted cosine similarity of n-grams. SPICE [26] measures the F-score of semantic propositions extracted from caption and reference. SPIDEr [7] is the arithmetic mean between the SPICE score and the CIDEr score. The submitted results are ranked by the SPIDEr score. In all metrics used, higher scores indicate better performance. It can be seen that the proposed model reached a SPIDEr score of 0.227 and outperforms the baseline model on each metric in a large margin.

[1]The performance the baseline model is presented according to `http://dcase.community/challenge2020/task-automatic-audio-captioning#results-for-the-development-dataset`.

5. REFERENCES

[1] 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.

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

[3] C. D. Kim, B. Kim, H. Lee, and G. Kim, "Audiocaps: Generating captions for audios in the wild," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 119–132.

[4] 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.

[5] S. Ikawa and K. Kashino, "Neural audio captioning based on conditional sequence-to-sequence model," in *2019 DCASE Workshop*, 2019.

[6] X. Xu, H. Dinkel, M. Wu, and K. Yu, "What does a car-ssette tape tell?" *arXiv preprint arXiv:1905.13448*, 2019.

[7] 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.

[8] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3156–3164.

[9] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.

[10] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual

recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625–2634.

[11] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4651–4659.

[12] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6077–6086.

[13] 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 *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[14] 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," *arXiv preprint arXiv:1912.10211*, 2019.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[16] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *ICLR Workshop Papers*, 2013.

[17] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Valletta, Malta: ELRA, May 2010, pp. 45–50, http://is.muni.cz/publication/884893/en.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

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

[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," *Proc. Interspeech 2019*, pp. 2613–2617, 2019.

[22] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[23] C.-Y. Lin, "Rouge: a package for automatic evaluation of summaries," in *Workshop on Text Summarization Branches Out, Post-Conference Workshop of ACL 2004, Barcelona, Spain*, July 2004. [Online]. Available: https://www.microsoft.com/en-us/research/publication/rouge-a-package-for-automatic-evaluation-of-summaries/

[24] S. Banerjee and A. Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," in *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005, pp. 65–72.

[25] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4566–4575.

[26] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Spice: Semantic propositional image caption evaluation," in *European Conference on Computer Vision*. Springer, 2016, pp. 382–398.

[27] M. Z. Hossain, F. Sohel, M. F. Shiratuddin, and H. Laga, "A comprehensive survey of deep learning for image captioning," *ACM Computing Surveys (CSUR)*, vol. 51, no. 6, pp. 1–36, 2019.

[28] S. Lipping, K. Drossos, and T. Virtanen, "Crowdsourcing a dataset of audio captions," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop (DCASE)*, Nov. 2019. [Online]. Available: https://arxiv.org/abs/1907.09238