Leveraging Pre-trained BERT for Audio Captioning

Xubo Liu[1], Xinhao Mei[1], Qiushi Huang[2], Jianyuan Sun[1], Jinzheng Zhao[1], Haohe Liu[1], Mark D. Plumbley[1], Volkan Kılıç[3], Wenwu Wang[1]

[1]Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, UK
[2]Department of Computer Science, University of Surrey, UK
[3]Department of Electrical and Electronics Engineering, Izmir Katip Celebi University, Turkey

***Abstract*—Audio captioning aims at using language to describe the content of an audio clip. Existing audio captioning systems are generally based on an encoder-decoder architecture, in which acoustic information is extracted by an audio encoder and then a language decoder is used to generate the captions. Training an audio captioning system often encounters the problem of data scarcity. Transferring knowledge from pre-trained audio models such as Pre-trained Audio Neural Networks (PANNs) have recently emerged as a useful method to mitigate this issue. However, there is less attention on exploiting pre-trained language models for the decoder, compared with the encoder. BERT is a pre-trained language model that has been extensively used in natural language processing tasks. Nevertheless, the potential of using BERT as the language decoder for audio captioning has not been investigated. In this study, we demonstrate the efficacy of the pre-trained BERT model for audio captioning. Specifically, we apply PANNs as the encoder and initialize the decoder from the publicly available pre-trained BERT models. We conduct an empirical study on the use of these BERT models for the decoder in the audio captioning model. Our models achieve competitive results with the existing audio captioning methods on the AudioCaps dataset.**

***Index Terms*—audio captioning, language models, BERT, Pre-trained Audio Neural Networks (PANNs), deep learning**

I. Introduction

Audio captioning is the task of generating a text description for an audio clip, which has various potential applications. For example, audio captioning can be used to generate text descriptions of sounds to help the hearing impaired in understanding an acoustic environment. Audio captioning has attracted increasing interest in the fields of acoustic signal processing and natural language processing (NLP).

Existing audio captioning systems are mostly based on an encoder-decoder architecture [1]–[5], in which acoustic information is extracted by an audio encoder, and then a language decoder is used to generate text descriptions. Training of an audio captioning system often encounters the problem of data scarcity. AudioCaps [6] is the largest public dataset for audio captioning research, however, it only has about 50k audio clips with one reference caption. Compared with the popular image captioning datasets such as MS COCO (∼123k images) [7] and Conceptual Captions (∼3.3M images) [8], the scale of the existing audio captioning dataset is much smaller. This can limit the performance of an audio captioning model in generating consistent natural language description.

To address the data scarcity issue of audio captioning, transferring knowledge from pre-trained audio models has been widely investigated. Xu et al. [9] propose an approach that uses transfer learning to exploit local and global information from audio tagging and acoustic scene classification, respectively. Pre-trained Audio Neural Networks (PANNs) [10] are the models pre-trained on AudioSet [11], which have achieved great success as the encoder [4], [5], [12]–[14] in the audio captioning system. Nevertheless, compared with the audio encoder, there is less attention on exploiting pre-trained NLP models for the language decoder in the audio captioning model.

Koizumi et al. [15] used a frozen Generative Pre-Training model (GPT-2) [16] with the retrieval of similar captions in the dataset. This method generates accurate results using ground-truth similar captions, whereas using the retrieved captions leads to degraded performance. Gontier et al. [17] proposed an approach for audio captioning by fine-tuning Bidirectional and Auto-Regressive Transformers (BART) [18] with AudioSet [11] tags as text conditions in the BART encoder, which achieved the state-of-the-art result on AudioCaps [6]. However, the performance of this method is highly dependent on the audio tagging model. In addition, the adaptation of BART (12 layers in both the encoder and decoder) results in a large number of training parameters (∼400 million).

BERT [19], which stands for Bidirectional Encoder Representations from Transformers, is an NLP model pre-trained on large-scale text datasets, which has been extensively used as strong baselines on many natural language understanding (NLU) benchmarks [20]. Recently, BERT has been exploited as the decoder in sequence-to-sequence models and has achieved state-of-the-art results on several Natural Language Generation (NLG) tasks such as Machine Translation and Text Summarization [21]. Weck et al. [22] used BERT embeddings for the decoder in the audio captioning model. However, using only word embedding layers may not fully utilize the linguistic knowledge of the pre-trained BERT model. In summary, the potential of using BERT for audio captioning has not been well studied in the literature.

In this paper, we investigate the exploitation of pre-trained BERT models for the decoder in the audio captioning model. We propose an encoder-decoder model in which PANNs are used as the audio encoder, and the pre-trained BERT is used in the decoder. To bridge the language decoder and the audio encoder, we add cross-attention layers with randomly initialized weights in the decoder, but retain the pre-trained weights from BERT models for other layers in the decoder.

TABLE I
CONFIGURATIONS OF BERT MODELS USED IN THIS WORK.

Type	Model	Layer	Head	Hidden
Compact BERT	BERT_tiny	2	2	128
	BERT_mini	4	4	256
	BERT_medium	6	8	512
BERT	BERT_base	12	12	768
RoBERTa	RoBERTa_base	12	12	768

In this way, the knowledge gained from the pre-trained BERT model can be transferred to the audio captioning decoder. We conduct an empirical study for the utility of various pre-trained BERT model such as BERT [19], Compact BERT [23] and RoBERTa [24] on the AudioCaps dataset. The experimental results demonstrate the efficacy of the pre-trained BERT models for audio captioning. Our proposed models achieve competitive results, as compared with existing audio captioning methods.

The remainder of this paper is organized as follows. The next section introduces our proposed method. Section III presents experimental setup. Section IV shows experimental results on the AudioCaps datasets. Conclusions are given in Section V.

II. PROPOSED METHOD

Our proposed audio captioning model is composed of PANNs based encoder and BERT based decoder. In this section, we first introduce the pre-trained BERT model, as depicted in Fig. 1. Then, we describe the audio encoder of our model, PANNs. Lastly, we discuss the BERT based language decoder in the audio captioning model. Fig. 2 visualizes the overall architecture of our proposed model.

A. Pre-trained BERT models

BERT [19] is based on a number of Transformer encoder blocks, where each block contains a multi-head bidirectional self-attention layer followed by a feed-forward layer. Each encoder block is equipped with residual connections and layer normalization. BERT is pre-trained on BooksCorpus [25] and English Wikipedia using Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) tasks. MLM aims to predict the masked tokens in the input sentence, while NSP aims to predict whether the input two sentences are paired. Pre-training on large datasets using these two tasks offers BERT the capabilities to capture linguistic information such as syntactic and semantic content.

In this work, we investigate three types of publicly available pre-trained BERT models: BERT [19], Compact BERT [23], and RoBERTa [24]. Compact BERT is a compressed version of BERT by knowledge distillation, with a smaller architecture. RoBERTa is built on BERT and uses different pre-training strategies, which shows better performance than BERT. The details of these BERT models are described in Table I.

B. PANNs encoder

PANNs [10] demonstrated powerful capabilities in extracting features of audio signals for audio recognition tasks such as audio tagging. In this work, we use the CNN10 model in PANNs as the audio encoder. The CNN10 consists of four convolutional blocks, each with two convolutional layers with a kernel size of 3×3. Batch normalization and ReLU are used after each convolutional layer. The number of channels per convolutional block is 64, 128, 256 and 512. An average pooling layer with kernel size 2×2 is applied for down-sampling. Global average pooling is applied along the frequency axis after the last convolutional block, followed by two fully-connected layers to align the dimension of the output with the hidden dimension D of the decoder. The CNN10 encoder takes the log mel-spectrogram of an audio clip as the input and outputs the features $I \in \mathbb{R}^{T \times D}$, where T and D represent the number of time frames and the dimension of the spectral feature at each time frame, respectively.

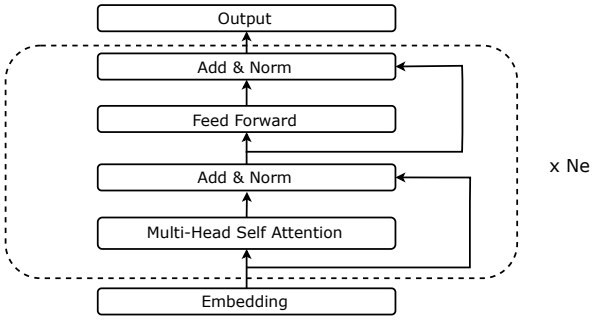

Fig. 1. The structure of BERT, where N_e is the number of encoder blocks.

C. BERT decoder

To use BERT as the language decoder in the audio captioning model, we make two adjustments. First, we modify the bidirectional self-attention used in the original BERT model, which considers both past and future context, to unidirectional self-attention by exploiting only the past contexts. This is because the bidirectional structure does not fit the language decoder. Second, the cross-attention layers are added after the self-attention layers to bridge the audio encoder and the language decoder. The cross-attention layer has two inputs, the encoder output $I \in \mathbb{R}^{T \times D}$ and the current state of the decoder $H \in \mathbb{R}^{N \times D}$, where N is the number of tokens already decoded, and D is the decoder hidden dimension. The cross-attention is calculated as:

$$\text{CrossAttn}(H, I) = \text{Attn}(H, I, I), \quad (1)$$

$$\text{Attn}(Q, K, V) = \text{Softmax}\left(\frac{(W^q Q)(W^k K)^T}{\sqrt{d}}\right) W^v V, \quad (2)$$

where W^q, W^k, W^v are three learnable matrices and d is a scaling factor. After that, we apply the add & norm operation, which contains a residual connection and layer normalization and can be written as:

$$\text{LayerNorm}(\text{CrossAttn}(H, I) + H). \quad (3)$$

The cross-attention layers are added with randomly initialized weights, while the other layers retain the pre-trained weights from BERT to transfer the NLP knowledge from BERT.

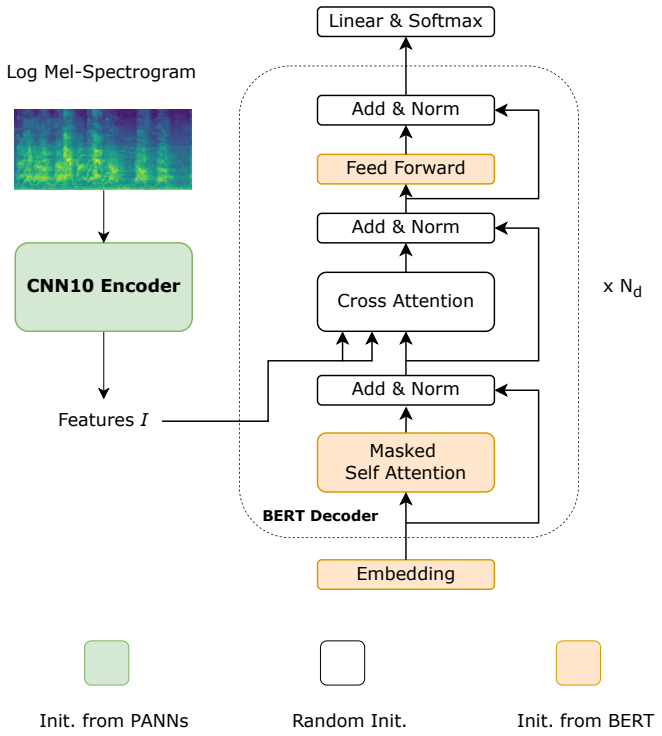

Fig. 2. The architecture of the proposed model using PANNs (CNN10) encoder and BERT decoder. The green, white, and orange blocks represent that the weights are initialized with parameters learned from PANNs, randomly initialized, and initialized from BERT, respectively. Here, N_d is the number of BERT decoder blocks.

III. Experiments

A. Dataset

AudioCaps [6] is the largest public audio captioning dataset with around 50k audio clips sourced from AudioSet [6]. AudioCaps is divided into three splits: training, validation and test. Each audio clip in the training set contains one human-annotated caption, while each clip in the validation and test set has five captions. Since some audio clips are now missing from YouTube, all our experiments are conducted on the version we downloaded, which contains 49 274 audio clips in the training set, 494 clips in the validation set, 957 clips in the test set.

B. Audio processing

We use the original sampling rate of 32 000 Hz to load audio data and the mel-spectrogram as the input to our model. Specifically, a 64-dimensional log mel-spectrogram is calculated using the short-time Fourier transform with a Hanning window of 1024 samples, and a hop size of 512 samples. SpecAugment [26] is used for data augmentation.

C. Text processing

We converted all captions in the AudioCaps dataset to lower case and removed punctuation. Two special tokens "<soc>" and "<eoc>" are added to the start and end of each caption. We tokenize our text corpus using the WordPiece [27] to match the BERT pre-trained vocabulary (∼30k tokens).

D. Training procedure

We trained the proposed model using Adam [28] optimizer with a batch size of 32. Warm-up is used in the first 5 epochs to increase the learning rate to the initial learning rate. The learning rate is then decreased to 1/10 of itself every 10 epochs. Dropout with a rate of 0.2 is applied in the BERT decoder to mitigate the over-fitting problem. To stabilize the training, we share the weights between the input embedding layer and the output token classification layer in the BERT decoder. We train the model for 30 epochs on the AudioCaps training set, with an initial learning rate of 5×10^{-5} for BERT_base and RoBERTa_base (as introduced in Table I) and 5×10^{-4} for other BERT configurations. Validation is carried out after every training epoch, and we save the model with the best performance on the validation set. For each experiment, we repeat three times and report their average performance.

E. Evaluation

During the inference stage, the mel-spectrogram along with the token "<soc>" are fed into the encoder and decoder separately to generate the first token. Then, the following tokens are predicted based on the previously generated tokens until the token "<eoc>" or the maximum length (50 tokens in our experiments) is reached. The beam search strategy [29] with a beam width up to 5 is used to generate captions.

We evaluate the performance of the proposed model using the same metrics adopted in DCASE 2021 Challenge on Task 6: "Automated Audio Captioning", including machine translation metrics: BLEU_n [30], METEOR [31], ROUGE_L [32] and captioning metrics: CIDEr [33], SPICE [34], SPIDEr [35]. BLEU_n measures the precision of n-gram inside the generated text. METEOR is a harmonic mean of precision and recall based on word-to-word matches. ROUGE_L calculates F-measures based on the longest common sub-sequence. CIDEr considers the cosine similarity between term frequency inverse document frequency (TF-IDF) of the n-gram. SPICE extracts captions into scenes graphs and calculates F-score based on them. SPICE score ensures captions are semantically faithful to the audio clip, while the CIDEr score ensures captions are syntactically fluent. SPIDEr is the mean score of CIDEr and SPICE.

IV. Results

A. Comparison with baseline methods

We compare our proposed approach with five baseline methods, namely, the TopDown-AlignedAtt [6] model, the CNN10-AT model [9] which uses pre-trained Audio Tagging model as the encoder, the Audio Captioning Transformer (ACT) [2], which is the first convolution-free architecture, the model in [15] that uses frozen GPT-2 and audio-based similar caption retrieval, and finally the current state-of-the-art model [17] on AudioCaps based on BART and AudioSet tags.

We report the performance of the first four baseline methods in the upper part of Table II. It can be observed that the final method [17] cannot generalize well to other datasets. Since its performance improvement depends on the AudioSet tags used as word hinters in caption annotation stage of AudioCaps, so

TABLE II

MODEL PERFORMANCE ON THE AUDIOCAPS DATASET. UPPER: PERFORMANCE OF EXISTING AUDIO CAPTIONING METHODS (BASELINE). BOTTOM: PERFORMANCE OF OUR PROPOSED PANNS (CNN10) ENCODER BERT DECODER MODEL. THE DISPLAYED SCORES ARE MEANS AND STANDARD DEVIATIONS OVER THREE EXPERIMENTS. THE HIGHEST VALUE FOR EACH METRIC IS SHOWN IN BOLD.

Model	$BLEU_1$	$BLEU_2$	$BLEU_3$	$BLEU_4$	$ROUGE_L$	METEOR	CIDEr	SPICE	SPIDEr
TopDown-AlignedAtt [6]	61.4	44.6	31.7	21.9	45.0	20.3	59.3	14.4	36.9
CNN10-AT [9]	65.5	47.6	33.5	23.1	46.7	22.9	66.0	16.8	41.4
ACT_small [2]	64.3	48.3	35.2	24.9	46.9	21.8	66.9	16.0	41.5
ACT_medium [2]	65.3	49.5	**36.3**	**25.9**	47.1	22.2	66.3	16.3	41.3
ACT_large [2]	64.7	48.8	35.6	25.2	46.8	22.2	**67.9**	16.0	**42.0**
GPT-2 + similar captions [15]	63.8	45.8	31.8	20.4	43.4	19.9	50.3	13.9	32.1
CNN10 + BERT_tiny	66.0 (0.7)	49.1 (0.3)	35.2 (0.3)	24.5 (0.2)	47.0 (0.5)	22.4 (0.2)	63.1 (0.9)	16.2 (0.3)	39.6 (0.5)
CNN10 + BERT_mini	**67.1 (0.9)**	49.8 (0.6)	35.8 (0.3)	25.1 (0.1)	**48.0 (0.7)**	**23.2 (0.4)**	66.7 (0.6)	**17.2 (0.1)**	41.9 (0.3)
CNN10 + BERT_medium	**67.1 (0.3)**	**50.1 (0.2)**	**36.3 (0.2)**	25.5 (0.2)	47.9 (0.4)	23.1 (0.4)	65.4 (1.2)	16.8 (0.5)	41.1 (0.6)
CNN10 + BERT_base	66.0 (0.4)	48.6 (0.3)	34.4 (0.4)	23.7 (0.5)	47.0 (0.2)	22.9 (0.1)	63.4 (1.3)	16.5 (0.1)	40.0 (0.6)
CNN10 + RoBERTa_base	66.1 (0.3)	48.6 (0.2)	34.4 (0.2)	23.7 (0.3)	46.9 (0.3)	22.3 (0.1)	63.7 (1.6)	16.1 (0.2)	39.9 (0.8)

TABLE III

THE PERFORMANCE OF THE STATE-OF-THE-ART SYSTEM (BART + AUDIOSET TAGS) AND HUMAN-GENERATED CAPTIONS (HUMAN).

Model	$BLEU_1$	$BLEU_2$	$BLEU_3$	$BLEU_4$	$ROUGE_L$	METEOR	CIDEr	SPICE	SPIDEr
BART + AudioSet tags [17]	69.9 (0.5)	52.3 (0.7)	38.0 (0.8)	26.6 (0.9)	49.3 (0.4)	24.1 (0.3)	75.3 (0.9)	17.6 (0.3)	46.5 (0.6)
Human [6]	65.4	48.9	37.3	29.1	49.6	28.8	91.3	21.6	56.5

TABLE IV

PERFORMANCE METRICS FOR THE ABLATION STUDY (RANDOMLY INITIALIZED BERT DECODER). THE VALUES IN THE METRICS WHERE RANDOMLY INITIALIZED BERT OUTPERFORMS THE PRE-TRAINED BERT ARE IN BOLD.

Decoder (Random)	$BLEU_1$	$BLEU_2$	$BLEU_3$	$BLEU_4$	$ROUGE_L$	METEOR	CIDEr	SPICE	SPIDEr
BERT_tiny	65.8 (0.7)	48.9 (0.2)	35.0 (0.4)	24.2 (0.4)	47.0 (0.3)	22.1 (0.2)	62.2 (1.2)	**16.3 (0.1)**	39.2 (0.7)
BERT_mini	66.4 (0.6)	49.2 (0.6)	35.5 (0.5)	**25.2 (0.5)**	47.8 (0.3)	23.2 (0.2)	65.3 (0.7)	16.8 (0.2)	41.0 (0.3)
BERT_medium	66.7 (0.5)	49.1 (0.6)	35.4 (0.6)	24.7 (0.6)	47.5 (0.3)	**23.2 (0.3)**	65.4 (1.0)	16.7 (0.1)	41.0 (0.5)
BERT_base	64.0 (0.2)	46.5 (0.3)	32.4 (0.3)	21.8 (0.1)	45.9 (0.3)	22.0 (0.3)	61.0 (1.1)	16.0 (0.0)	38.5 (0.6)
RoBERTa_base	64.9 (0.5)	47.6 (0.7)	33.5 (0.6)	22.9 (0.4)	46.1 (0.2)	22.0 (0.1)	62.0 (1.3)	**16.2 (0.4)**	39.1 (0.6)

we separately report its performance in Table III for reference. In addition, the performance of human-generated captions described in [6] is given in Table III.

B. *Efficacy of BERT decoder*

We report the performance of our proposed model in the bottom part of Table II. Experimental results demonstrate that Compact BERT achieves the best result, especially BERT_mini and BERT_medium. We found that although BERT and RoBERTa are more powerful pre-trained NLP models, they do not outperform Compact BERT. We empirically found that the degradation of BERT and RoBERTa is due to their large architecture, which potentially leads to over-fitting on this task. Compared with the baseline models, our models perform better on the machine translation related metrics. Specifically, BERT_mini achieved the highest scores in $BLEU_1$, METEOR and CIDEr, while BERT_medium performs the best in terms of $BLEU_1$, $BLEU_2$, $BLEU_3$, and $ROUGE_L$ metrics. This indicates that our models have a better ability in generating accurate words and fluent language descriptions than baseline models. In summary, our models show competitive performance as compared to the existing audio captioning models.

To further show the efficacy of the BERT decoders for audio captioning, we conducted an ablation study with randomly initialized weights of the BERT decoder. Note, that there is no structural difference between a BERT decoder and a standard transformer decoder. Experimental results are reported in Table IV, as for all BERT architectures, the pre-trained BERT decoders outperform the randomly initialized BERT decoders on most metrics. This shows that the knowledge from the pre-trained BERT model is helpful for audio captioning.

V. CONCLUSION

We have presented an encoder-decoder based audio captioning model by using pre-trained BERT models as language decoder and PANNs as the audio encoder. To bridge the language decoder and the audio encoder, the cross-attention layers are added with randomly initialized weights in the BERT decoder, while the other layers retain the pre-trained weights from BERT models. We conducted an empirical study on the utility of the pre-trained BERT models with a different scale on the AudioCaps dataset. The experimental results demonstrate the efficacy of the BERT model for audio captioning. Our proposed models show competitive results as compared to the existing audio captioning methods. In future work, we will investigate the usage of NLP models for audio captioning from other aspects, such as text data augmentation.

ACKNOWLEDGMENT

This work is supported by a Newton Institutional Links Award from the British Council and the Scientific and Technological Research Council of Turkey (TUBITAK), titled "Automated Captioning of Image and Audio for Visually and Hearing

Impaired" (Grant numbers 623805725 and 120N995), a grant EP/T019751/1 from the Engineering and Physical Sciences Research Council (EPSRC), and a Research Scholarship from the China Scholarship Council. For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

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)*, pp. 374–378.

[2] X. Mei, X. Liu, Q. Huang, M. D. Plumbley, and W. Wang, "Audio captioning Transformer," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, 2021.

[3] 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.

[4] X. Liu, Q. Huang, X. Mei, T. Ko, H. L. Tang, M. D. Plumbley, and W. Wang, "CL4AC: A contrastive loss for audio captioning," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, 2021.

[5] W. Yuan, Q. Han, D. Liu, X. Li, and Z. Yang, "The DCASE 2021 challenge task 6 system: Automated audio captioning with weakly supervised pre-traing and word selection methods," DCASE2021 Challenge, Tech. Rep., July 2021.

[6] 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.

[7] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *European Conference on Computer Vision*. Springer, 2014, pp. 740–755.

[8] P. Sharma, N. Ding, S. Goodman, and R. Soricut, "Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 2556–2565.

[9] X. Xu, H. Dinkel, M. Wu, Z. Xie, and K. Yu, "Investigating local and global information for automated audio captioning with transfer learning," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 905–909.

[10] 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.

[11] 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)*, pp. 776–780.

[12] X. Mei, X. Liu, J. Sun, M. D. Plumbley, and W. Wang, "Diverse audio captioning via adversarial training," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

[13] A. Koh, X. Fuzhao, and C. E. Siong, "Automated audio captioning using transfer learning and reconstruction latent space similarity regularization," in *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7722–7726.

[14] 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.

[15] 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:2012.07331*, 2020.

[16] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[17] F. Gontier, R. Serizel, and C. Cerisara, "Automated audio captioning by fine-tuning BART with AudioSet tags," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, 2021.

[18] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv:1910.13461*, 2019.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *arXiv:1810.04805*, 2018.

[20] A. Williams, N. Nangia, and S. R. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," *arXiv:1704.05426*, 2017.

[21] S. Rothe, S. Narayan, and A. Severyn, "Leveraging pre-trained checkpoints for sequence generation tasks," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 264–280, 2020.

[22] B. Weck, X. Favory, K. Drossos, and X. Serra, "Evaluating off-the-shelf machine listening and natural language models for automated audio captioning," *arXiv:2110.07410*, 2021.

[23] I. Turc, M.-W. Chang, K. Lee, and K. Toutanova, "Well-read students learn better: On the importance of pre-training compact models," *arXiv:1908.08962*, 2019.

[24] 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:1907.11692*, 2019.

[25] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler, "Aligning books and movies: Towards story-like visual explanations by watching movies and reading books," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 19–27.

[26] 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," *arXiv:1904.08779*, 2019.

[27] 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:1609.08144*, 2016.

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

[29] C. Tillmann and H. Ney, "Word reordering and a dynamic programming beam search algorithm for statistical machine translation," *Computational linguistics*, vol. 29, no. 1, pp. 97–133, 2003.

[30] 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 of the Association for Computational Linguistics*, 2002, pp. 311–318.

[31] A. Lavie and A. Agarwal, "METEOR: An automatic metric for mt evaluation with high levels of correlation with human judgments," in *Proceedings of the Second Workshop on Statistical Machine Translation*, 2007, pp. 228–231.

[32] L. C. Rouge, "A package for automatic evaluation of summaries," in *Proceedings of Workshop on Text Summarization of Association for Computational Linguistics (ACL), Spain*, 2004.

[33] 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.

[34] 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.

[35] 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.