Automated audio captioning by fine-tuning bart with audioset tags

Félix Gontier, Romain Serizel, Christophe Cerisara

AUTOMATED AUDIO CAPTIONING BY FINE-TUNING BART WITH AUDIOSET TAGS

Félix Gontier, Romain Serizel, Christophe Cerisara

Université de Lorraine, CNRS, Inria, Loria, F-54000, France.
felix.gontier@inria.fr, {romain.serizel,christophe.cerisara}@loria.fr

ABSTRACT

Automated audio captioning is the multimodal task of describing environmental audio recordings with fluent natural language. Most current methods utilize pre-trained analysis models to extract relevant semantic content from the audio input. However, prior information on language modeling is rarely introduced, and corresponding architectures are limited in capacity due to data scarcity. In this paper, we present a method leveraging the linguistic information contained in BART, a large-scale conditional language model with general purpose pre-training. The caption generation is conditioned on sequences of textual AudioSet tags. This input is enriched with temporally aligned audio embeddings that allows the model to improve the sound event recognition. The full BART architecture is fine-tuned with few additional parameters. Experimental results demonstrate that, beyond the scaling properties of the architecture, language-only pre-training improves the text quality in the multimodal setting of audio captioning. The best model achieves state-of-the-art performance on AudioCaps with 46.5 SPIDEr.

Index Terms— Audio captioning, language models, transfer learning, BART, audio tagging

1. INTRODUCTION

The task of automated audio captioning [1] aims at improving the description of environmental sounds through the production of textual descriptions of input audio. This field of research has seen recent growth in interest within the audio community, with a recurring dedicated task introduced in 2020 to the DCASE challenge[1].

Audio captioning methods typically rely on sequence-to-sequence approaches, that encode audio features and produce sentences through a separate decoder [2]. With increasing focus on the format and vocabulary of captions [3, 4], recent advances have been achieved by encoding textual inputs in addition to audio representations. In particular, keyword prediction [5, 6] or similar captions retrieval [7] have been investigated as supplementary guidance material for captioning systems. Any such supplementary information must be inferred directly from the audio signal.

In terms of audio features, pre-trained embeddings such as VGGish [8] are commonly utilized. These embeddings are highly informative compared to other representations (eg. Mel spectrograms), which reduces the model capacity necessary to extract relevant semantic content as a result. In previous studies, however, text-based conditioning inputs are produced by a dedicated module trained on the captioning dataset. The small amounts of available data and the diversity of sound objects heavily limit the capabilities of such architectures. This often leads to poor accuracy in guidance inputs, and thus a lower semantic correctness of the resulting captions. For instance, Koizumi et al. [7] find that a model conditioned on ground truth similar captions in the dataset reaches near-human performances, whereas learned audio-based similar caption retrieval leads to significantly worse results.

Beyond textual input extraction, the language generation module in captioning systems is often trained from scratch. Thus, the model must learn to reproduce a fluent language structure with diverse vocabulary in addition to conveying semantic information from the input audio. Concurrently, many pre-trained models have been proposed in the natural language processing (NLP) community that efficiently model the syntax of natural language for representation learning [9] or generation [10]. Nevertheless, directly applying language models to multi-modal tasks such as captioning is not straightforward. Koizumi et al. [7] integrated a frozen GPT-2 [10] instance as the main language modeling part in their captioning system, and obtained results on par with the previous state of the art with fewer trainable parameters.

In this paper, we investigate scaling audio captioning architectures to the capacity of large-scale language models by utilizing audio and language pre-training. To do so, we present a novel method that adapts a transformer encoder-decoder with the BART general purpose pre-training [11] to produce captions by attending to both audio and text embeddings. This setting is illustrated in Figure 1. Thus, contrary to Koizumi et al. [7] no additional module combining audio and text information is learned from scratch. Furthermore, instead of learning guidance textual guidance inputs on the captioning dataset, we condition generation on AudioSet tags [12] obtained from a pre-trained model, YAMNet [13].

Specifically, the contributions of the present work are as follow:

- We propose a multi-modal conditioning scheme based on aligned temporal sequences of text and audio embeddings obtained from pre-trained models. The combination method relies on very few additional trainable parameters.
- We demonstrate that fine-tuning BART on these inputs results in high audio captioning performance, outperforming previous state-of-the-art systems.
- Through complementary experiments, we show the scaling properties of the BART architecture, as well as the potential of pre-training in tackling smaller captioning datasets.

To encourage the use of the proposed method in future work, the code for all presented experiments is made available[2].

This work was funded under the ANR project LEAUDS (Grant No. ANR-18-CE23-0020).

[1] https://dcase.community

[2] https://github.com/felixgontier/dcase2021aac

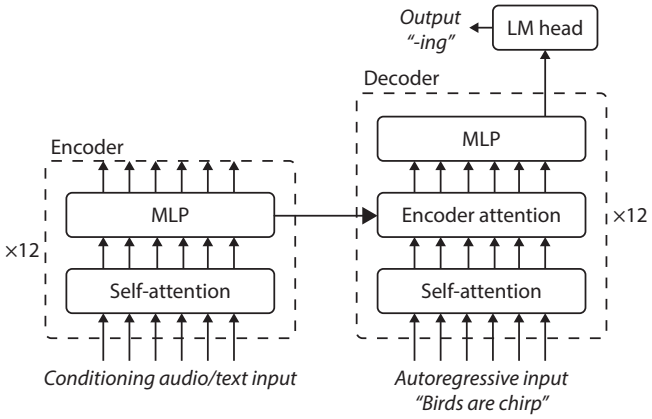

Figure 1: BART transformer architecture (residual connections and layer normalization omitted for clarity) and proposed setting in the captioning application.

2. METHODS

2.1. The BART model

The architecture associated with BART is a standard sequence-to-sequence transformer [14]. It is composed of a bi-directional encoder and an autoregressive decoder as illustrated in Figure 1. Encoder transformer blocks contain a multi-head self-attention layer followed by a multi-layer perceptron (MLP), as well as residual connections and layer normalization after each transformation. Decoder layers are further conditioned on the encoder output through an additional multi-head cross-attention layer after self-attention. Lastly, a dense layer outputs logits across all tokens in the data vocabulary. The main BART model comprises 12 layers in both the encoder and decoder with and internal dimension of 1024 in all hidden layers. The language tokenizer uses byte-pair encoding with a vocabulary of 50265 tokens. As a result, the architecture totals about 400 million parameters.

2.2. Text conditioning

In accordance with previous studies, we condition captioning with inputs describing the semantic content in text form. To do so, we propose to infer AudioSet tags from the audio input using YAMNet [13]. The YAMNet model achieves high tagging accuracy, and operates on 1 s audio frames. Contrary to other textual conditioning (e.g. keyword prediction), this results in a temporal sequence of identified sound objects in audio extracts. Such sequential detail is often found in ground truth captions, although at a coarser scale.

To obtain the conditioning input, YAMNet is applied to 1 s non-overlapping frames $\mathbf{x}_i$ of the audio input. This process is shown in Figure 2. Instead of selecting the AudioSet tag as the maximum of YAMNet logits, we sample tags from the output distribution at each iteration. The empirical motivation of this design is to increase the robustness of the model to YAMNet prediction errors, by randomly introducing incorrect yet plausible tags to the conditioning input. AudioSet tags are then utilized in their textual form (eg. *Chirp, tweet*). After application of the BART tokenizer, each tag typically results in a sub-sequence of one to six tokens. Conditioning sequences are produced by concatenating all sub-sequences with separator tokens. In our experiments, the choice of token (resp. ".", "and", "then", "<mask>") did not have significant impact on the initial loss, convergence rate, or final captioning performance. Thus, all discussed models are trained with '. ' as the separator token. The pre-trained BART embedding layer then encodes tokens into real-valued vectors $\mathbf{y}_{i,k}$ of dimension 1024, where k denotes the token position within the i^{th} tag sub-sequence. Lastly, positional embeddings $\mathbf{e}_t$ are further added to $\mathbf{y}_{i,k}$ to disambiguate the sequential order of inputs in the model, where t is the token position in the model input sequence independently from the audio frame i.

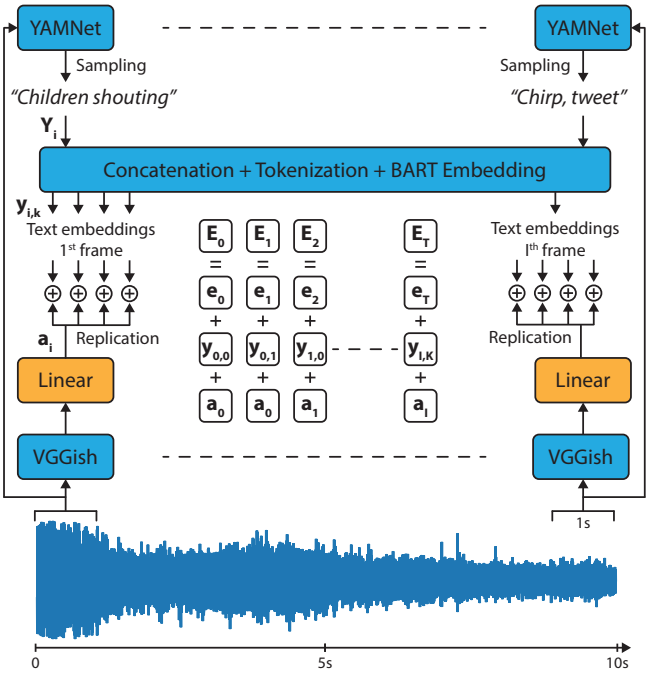

Figure 2: Conditioning input construction from pre-trained audio embeddings and textual tags. Blue and orange shading indicate frozen and learned modules respectively. Audio, textual, and positional embeddings (resp. $\mathbf{a_i}$, $\mathbf{y_{i,k}}$, and $\mathbf{e_t}$) are added to produce the input $\mathbf{E_t}$.

2.3. Audio conditioning

Although conditioning the generation on text only is a very close setting to the pre-training objective of BART, YAMNet predictions may be erroneous for part of the audio extract. In order to help the model select correct semantic content from given tags, audio embeddings are added to the encoder input. Embeddings from the YAMNet model likely contain the information as the conditioning tags. Thus, we explore deep embeddings from the penultimate layer of two other tagging models: VGGish [8] and PANNs [15], specifically the *Wavegram-Logmel-CNN* variant.

VGGish is able to provide 128-dimensional embedding vectors $\mathbf{a}_i$ for 1 s audio frames. This matches the granularity of YAMNet predictions, and allows for the alignment of textual and audio sequences to condition the caption generation. Figure 2 illustrates the corresponding conditioning process. VGGish embeddings are replicated over all tokens of the corresponding YAMNet tag. Following Huang et al. [16], the embeddings are directly added to their text and positional equivalents (resp. $\mathbf{y}_{i,k}$ and $\mathbf{e}_t$). As the audio em-

bedding size is different from the internal dimension of BART, we introduce a trainable dense layer to perform the adaptation.

Alternatively, the PANNs model infers a single embedding vector of dimension 2048 for 10 s of audio. Compared to VGGish, the lower temporal detail in PANNs embeddings is compensated by greater semantic content, which translates to higher performance on AudioSet tagging. Conditioning on PANNs is also straightforward: embeddings are replicated over encoder input timesteps corresponding to 10 consecutive tags, mapped to 1024-dimensional vectors by a dense layer, then added to text and positional embeddings.

3. EXPERIMENTAL SETUP

3.1. Dataset

All the experiments are conducted on the AudioCaps dataset [17]. AudioCaps comprises training, validation, and evaluation splits of about 49000, 485 and 955 audio extracts. The dataset is a subset of AudioSet [12], thus most audio examples have a duration of 10 s. Training examples are associated with a single annotated caption, whereas validation and test splits contain 5 captions per audio file.

3.2. Evaluation metrics

We evaluate the quality of the generated captions on standard captioning metrics. We report BLEU-1 to BLEU-4, METEOR, ROUGE-L, and CIDEr [18], which are all based on n-gram matching. In addition, SPICE [19] is computed as an evaluation of the semantic quality of the generated captions. The overall performance is given by SPIDEr [20], the average of CIDEr and SPICE, which is the main metric of the DCASE challenge task 6 on captioning.

3.3. Baselines

We compare our approach against two baselines in the literature. First, the `TopDown-AlignedAtt` model in the AudioCaps dataset paper [17] achieves the best reported performance according to the SPIDEr metric. Secondly, the system presented by Koizumi et al. [7] is, to our knowledge, the first to include a pre-trained language model (frozen GPT-2) in an audio captioning framework. Lastly, the current state-of-the-art system on AudioCaps is described in [6].

3.4. Training procedure

Model parameters are trained to minimize the categorical cross-entropy loss over the 50265 classes in the BART tokenizer. Optimization is performed using the AdamW [21] optimizer with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and with a learning rate of 10^{-5}. In our experiments, we observed stable convergence without major overfitting. All models are trained for 4 epochs, or about 24000 iterations with a batch size of 8 examples. Validation is carried out after every 1000 iterations, and we retain the model checkpoint with the lowest validation loss.

3.5. Caption generation procedure

In order to preserve deterministic inference at evaluation, conditioning tags are selected as the maximum value over YAMNet logits as opposed to the sampling scheme applied during training. Captions are generated autoregressively via beam search, with a beam size of 4 and no constraint or penalization on the total caption length.

4. EXPERIMENTS

4.1. Conditioning evaluation

We investigate combinations of audio and textual encoder inputs, including text-only and audio-only captioning guidance. The model for each conditioning setting is trained 3 times with different random seeds. Table 1 details the main experimental results, with means and standard deviations of metrics reported over the 3 instances. The proposed approach is compared to baselines as well as human performance, which refers to the cross-validation of reference captions in the evaluation set [17]. This anchor reflects discrepancies in content and syntax among ground truth captions, and constitutes a reasonable upper bound on caption quality.

First, the model conditioned on PANNs embeddings, which only contain one vector for AudioCaps examples, fails to generate well-structured captions. VGGish embeddings perform significantly better, hinting that providing the model with information on the sequence of sound events is critical in audio-only conditioning designs. However, the variance in performance between training instances is very high compared to other settings.

Conditioning solely on YAMNet tags further improves both the fluency and faithfulness of captions. Tags directly provide vocabulary guidance to the model, whereas relevant terms must be inferred from audio embeddings. In addition, the model only operates on language in this case, thus the task setting is close to that of the BART pre-training scheme.

The relative increase in performance for settings combining text and audio conditioning suggests that the information in both inputs is complementary. Contrary to audio-only experiments, PANNs performs better than VGGish when paired with YAMNet tags. Because sequential detail is already contained in the input tokens, the temporal granularity of audio embeddings is less important than their semantic content. Empirical analysis reveals a higher - although weak - correlation between YAMNet tagging accuracy and SPICE scores in the text-only model compared to that combining tags and PANNs embeddings. This behavior may indicate that audio embeddings mitigate the appearance rate of incorrectly identified sound objects in produced captions.

4.2. System performance

The proposed approach achieves similar results to Eren et al. [6] on reported metrics, and outperforms other baselines. It is on par or better than human performance according to BLEU-1, BLEU-2, BLEU-3, and ROUGE-L. However, these metrics only evaluate matching n-grams, and do not correlate well with human evaluations of quality [22, 23]. On more advanced metrics for syntactic fluency and semantic correctness, respectively CIDEr and SPICE, the best model is below human performance by a large margin. Still, the low remaining gap in terms of SPICE score confirms that the high accuracy of YAMNet source recognition is well conveyed to output captions.

In the following subsections, we present complementary experiments on the properties of BART pre-training for audio captioning. All experiments retain the best conditioning setting of YAMNet tags combined with PANNs embeddings.

4.3. Interest of language pre-training

Because of the multi-modal nature of inputs in the captioning task, it is relevant to assess whether the system performance can be linked

Table 1: Evaluation of the proposed approach on AudioCaps. The displayed scores are means and standard deviations over three instances. The highest value for each metric is shown in bold.

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr	SPICE	SPIDEr
TopDown-AlignedAtt [17]	61.4	44.6	31.7	21.9	20.3	45.0	59.3	14.4	36.9
Koizumi et al. [7]	63.8	45.8	31.8	20.4	19.9	43.4	50.3	13.9	32.1
Eren et al. [6]	71.1	49.3	37.6	23.2	28.7	58.7	75.0	-	-
BART + PANNs	47.2 (1.4)	27.9 (0.6)	15.6 (0.9)	7.8 (0.9)	11.2 (0.6)	32.2 (0.8)	6.5 (2.5)	6.1 (0.9)	6.3 (1.7)
BART + VGGish	57.6 (2.9)	39.3 (3.2)	26.2 (2.9)	17.0 (2.6)	17.6 (1.8)	39.8 (1.9)	37.6 (8.0)	11.9 (1.3)	24.7 (4.6)
BART + YAMNet	61.1 (0.6)	43.3 (0.5)	30.5 (0.5)	20.8 (0.3)	19.8 (0.2)	43.6 (0.2)	54.7 (0.6)	14.1 (0.2)	34.4 (0.4)
BART + YAMNet + VGGish	65.8 (0.5)	48.7 (0.4)	35.4 (0.4)	25.3 (0.5)	21.9 (0.1)	46.5 (0.1)	63.9 (1.0)	15.9 (0.3)	39.9 (0.7)
BART + YAMNet + PANNs	**69.9 (0.5)**	**52.3 (0.7)**	**38.0 (0.8)**	26.6 (0.9)	24.1 (0.3)	49.3 (0.4)	75.3 (0.9)	17.6 (0.3)	46.5 (0.6)
Human	65.4	48.9	37.3	**29.1**	**28.8**	**49.6**	**91.3**	**21.6**	**56.5**

Table 2: Performance metrics for complementary experiments.

Variant	CIDEr	SPICE	SPIDEr
Reference	75.3	17.6	46.5
No BART pre-training	71.0	16.7	43.8
Frozen decoder param.	68.5	16.6	42.5
BART-XSum	71.8	17.3	44.5
BART-CNN	72.2	17.7	44.2
BART-CNN, frozen decoder param.	70.4	15.6	43.0
BART-base	73.1	16.8	45.0

to BART pre-training as opposed to its high-capacity architecture. We do so by evaluating a model without BART pre-training, i.e. randomly initialized. Discrepancies with the reference setting in Table 2 demonstrate that while catastrophic forgetting, i.e. BART overfitting to the target task and forgetting its pre-trained generic information about language, may occur due to multi-modal inputs, some information is retained from the denoising pre-training. Nonetheless, the large-scale transformer architecture is for a large part responsible for improvements over systems in the literature.

In a subsequent experiment, we freeze parameters of self-attention and MLP blocks in BART decoder layers. These parameters are expected to hold most of the knowledge on language modeling. Encoder parameters, as well as decoder cross-attention and layer normalization weights, remain freely fine-tuned. The number of trainable parameters is about 250 million, reduced from 408 million in the full model. This variant achieves higher SPIDEr than both baselines, which suggests that BART language modeling parameters can already produce high quality captions.

4.4. Pre-training task

Within the proposed conditioning setting, the captioning task solved by BART is related to summarization: the model is given about 10 often recurring AudioSet tags, whereas most captions in the dataset describe one to three sequential events. The authors of BART demonstrated the potential of its pre-training scheme when applied to the CNN/DM [24] and XSum [25] summarization datasets, with model parameters made available to the community.

We investigate the effect of summarization fine-tuning on our captioning method, by replacing regular BART parameters in the proposed model with BART-CNN and BART-XSum checkpoints at initialization. These setups result in slightly lower performance than the reference method in Table 2. However, we observed a significant decrease in the initial training loss in both cases. In addition, freezing decoder parameters (see Section 4.3) in the BART-CNN model produces better syntax compared to the equivalent setting with standard BART initialization, according to CIDEr. This indicates that the language modeling learned for summarization is better suited to captioning than that of denoising. Thus, using the BART-CNN decoder as the starting point may be preferable on smaller captioning datasets, if overfitting prevents training the full BART architecture.

4.5. Model capacity

We investigate the impact of model capacity on the quality of the generated captions. To do so, we replace the standard BART architecture with the BART-base variant provided by the authors. BART-base undergoes the same pre-training scheme with half as many encoder and decoder layers (6 instead of 12) as well as a internal dimension of 768 (from 1024). These modifications reduce the number of trainable parameters from 408 million to about 140 million.

Interestingly, the large reduction in model capacity does not translate to similarly important decrease in performance (see Table 2). We hypothesize that, even though the language modeling capabilities of BART-base are inferior to those of the standard BART, the highly formatted nature of captions requires less knowledge to model than general text in other tasks. This conjecture is in part supported by the fair syntactic quality of captions produced by smaller architectures in the literature. As a byproduct, it is also unlikely that further increasing model capacity from that of the standard BART architecture would yield appreciably higher performances.

5. CONCLUSION

In this paper, we presented an audio captioning scheme by fine-tuning BART with combined audio and textual conditioning. Our results demonstrate that transfer learning can be applied to scale captioning architectures to the size of state of the art NLP models, in spite of the limited data availability. Using pre-trained architectures to retrieve both audio and language guidance material removes the need for dedicated modules, and enables semantic conditioning with high accuracy and controlled generalization properties. We find that the proposed model can be scaled down or partly frozen with limited decreases in performance, hence diminishing the trade-off between high quality caption production and model capacity.

This study highlights interesting avenues for future research. In particular, upcoming work will explore methods to better utilize the knowledge of encoders with text pre-training in multi-modal downstream tasks. Determining the optimal granularity of temporal detail to reduce information redundancy in guidance inputs, or developing adaptive audio segmentation matching separate sound events, will also be investigated in future work.

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)*, 2017, pp. 374–378.

[2] K. Nguyen, K. Drossos, and T. Virtanen, "Temporal subsampling of audio feature sequences for automated audio captioning," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*, November 2020, pp. 110–114.

[3] D. Takeuchi, Y. Koizumi, Y. Ohishi, N. Harada, and K. Kashino, "Effects of word-frequency based pre- and post-processings for audio captioning," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*, November 2020, pp. 190–194.

[4] E. Çakır, K. Drossos, and T. Virtanen, "Multi-task regularization based on infrequent classes for audio captioning," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2020 Workshop (DCASE2020)*, November 2020, pp. 6–10.

[5] Y. Koizumi, R. Masumura, K. Nishida, M. Yasuda, and S. Saito, "A Transformer-Based Audio Captioning Model with Keyword Estimation," in *Proc. Interspeech 2020*, 2020, pp. 1977–1981.

[6] A. O. Eren and M. Sert, "Audio captioning based on combined audio and semantic embeddings," in *2020 IEEE International Symposium on Multimedia (ISM)*, 2020, pp. 41–48.

[7] Y. Koizumi, Y. Ohishi, D. Niizumi, D. Takeuchi, and M. Yasuda. (2020) Audio captioning using pre-trained large-scale language model guided by audio-based similar caption retrieval. [Online]. Available: https://arxiv.org/abs/2012.07331

[8] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, "Cnn architectures for large-scale audio classification," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 131–135.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 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)*, June 2019, pp. 4171–4186.

[10] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019.

[11] 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," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, July 2020, pp. 7871–7880.

[12] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audioset: An ontology and human-labeled dataset for audio events," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 776–780.

[13] M. Plakal and D. Ellis. Yamnet. [Online]. Available: https://github.com/tensorflow/models/tree/master/research/audioset/yamnet

[14] 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*, vol. 30, 2017.

[15] 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.

[16] W.-C. Huang, C.-H. Wu, S.-B. Luo, K.-Y. Chen, H.-M. Wang, and T. Toda, "Speech recognition by simply fine-tuning bert," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 7343–7347.

[17] 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)*, June 2019, pp. 119–132.

[18] 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 (CVPR)*, June 2015.

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

[20] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy, "Improved image captioning via policy gradient optimization of spider," in *2017 IEEE International Conference on Computer Vision (ICCV)*, oct 2017, pp. 873–881.

[21] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations (ICLR)*, 2019.

[22] C.-W. Liu, R. Lowe, I. Serban, M. Noseworthy, L. Charlin, and J. Pineau, "How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Nov. 2016, pp. 2122–2132.

[23] J. Novikova, O. Dušek, A. Cercas Curry, and V. Rieser, "Why we need new evaluation metrics for NLG," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Sept. 2017, pp. 2241–2252.

[24] R. Nallapati, B. Zhou, C. dos Santos, Ç. Gulçehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond," in *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, Aug. 2016, pp. 280–290.

[25] S. Narayan, S. B. Cohen, and M. Lapata, "Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Oct.-Nov. 2018, pp. 1797–1807.