AUDIO-CHANGE CAPTIONING TO EXPLAIN MACHINE-SOUND ANOMALIES

Shunsuke Tsubaki[1] *Yohei Kawaguchi*[2]*, Tomoya Nishida*[2]*, Keisuke Imoto*[1]
Yuki Okamoto[3]*, Kota Dohi*[2]*, Takashi Endo*[2]

[1]Doshisha University, Faculty of Science and Engineering, Kyoto, Japan
[2]Hitachi, Ltd., Japan, [3]Graduate School of Information Science and Engineering,
Ritsumeikan University, Kyoto, Japan

ABSTRACT

This paper defines the new problem of "audio-change captioning," which describes what has changed between two audio samples. Conventional audio-captioning methods cannot be used to explain such change, and conventional image-change-captioning methods cannot explain the differences in audio samples. To address these issues, we propose a neural-network model for generating sentences that explain how a machine's normal and anomalous sounds changed in relation to each other. We also created a dataset called MIMII-Change by annotating pairs of normal and anomalous samples extracted from MIMII-DG for each type of sound in machine-operation sounds. The experimental results indicate that our model with spatial attention architecture is effective for stationary sounds because it is able to determine changes in global features, while our model with Transformer Encoder architecture is effective for periodic and sudden sounds because it is able to determine temporal dependencies.

Index Terms— Automated audio captioning, Natural language generation, Deep learning

1. INTRODUCTION

Automated audio captioning (AAC) [1] is one of the tasks that has received particular attention in the field of environmental sound analysis (ESA). The purpose of AAC is to automatically generate textual descriptions (captions) of an audio signal. By representing an audio signal with captions, the relationship between acoustic events and acoustic scenes in the audio signal and their respective states can be described. AAC is expected to have practical applications in a variety of areas, such as assisting the hearing-impaired to understand environmental sounds and analyzing sound in video-based security surveillance systems. It can also be used for other fields such as multimedia retrieval [2, 3]. The framework commonly used in AAC is the sequence-sequence encoder-decoder [4], and like many natural-language-processing tasks, Transformer [5] is the predominant model in AAC [6, 7, 8]. Several studies were conducted to improve the performance of caption generation by providing additional information beyond the encoded audio-embedding information to the text decoder [6, 9]. The utility of such semantic guidance has been explored in image and video captioning, achieving better performance [10, 11].

While the purpose of AAC is to describe a single sound, in real-world problem solving, it may be useful to compare two acoustic signals and describe the changes between them. The anomalous sound detection (ASD) [12] system for machine-operation sounds, only informs about the presence of anomalies without specifying what has changed and how. As a result, experts need to verify the detection results and perform additional tasks to determine if repairs are necessary and which components should be repaired. To simplify this process and reduce the workload for experts, we propose representing the differences between normal and anomalous sounds using linguistic information. This approach allows for an efficient analysis of anomalous machine operation sounds, enabling experts to identify the specific changes and alleviate their burden.

Hence, we define the task of describing the change between two audio signals as audio-change captioning, address the task of explaining anomalous sounds in machines, and introduce the task description and learning scheme. It should be noted that in this study, the objective is not to classify anomalous sounds as in traditional ASD, but rather to focus on expressing how they are anomalous.

Change captioning has already been studied in the image domain. It is used to describe what has changed between two image scenes (before/after) using natural language. Jhamtani and Berg-Kirkpatrick [13] used a pixel-difference-based approach to identify regions of change between before and after images. Because images are assumed aligned and that there is always a change between the two images, this approach cannot distinguish relevant changes from distractors, which is data disguised as change such as viewpoint changes. Therefore, to make it more useful for users, Park et al. [14] created a model that distinguishes between distractors, such as viewpoint change or lighting change, and semantically significant changes such as object movement or change. The model was made robust to distractors by using a dual-attention mechanism to identify regions of change between images. Thus, while change captioning has been studied in the image domain and various methods have been proposed, a pixel-difference-based approach, such as Jhamtani and Berg-Kirkpatrick's [13], is not considered effective for the audio domain, which is time-series information. This study is the first attempt at automated audio-change captioning.

We propose a neural-network model for generating change captions from two sounds. The aim is to generate a textual caption of the changes between the audio files and that is as close as possible to the change caption given by a human for the same audio file. As the suitable model architecture differs due to the sound-occurrence interval or section, we divided sound types into three categories in accordance with sound occurrence and used different architectures for our model. We used Transformer Encoder, which is effective in many AAC tasks, and spatial attention, which is also considered effective [14], as model architectures. For stationary sound changes, we employed spatial attention, while for periodic and non-periodic sounds, we employed Transformer Encoder. In addition to the metrics used in Detection and Classification of Acoustic Scenes and Events (DCASE) [15], i.e., BLEU [16],

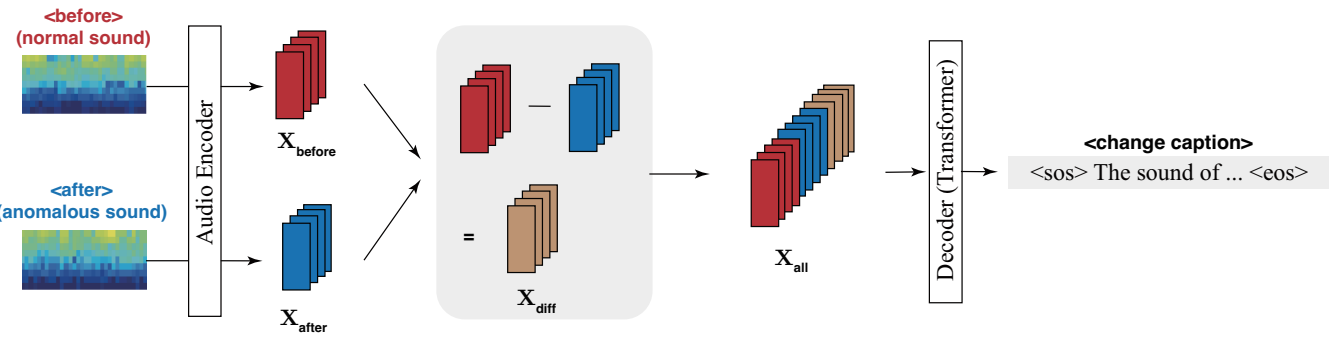

Figure 1: Our proposed audio-change captioning model to explain machine-sound anomalies

METEOR [17], CIDEr [18], and SPIDEr [19], we used Sentence-BERT [20, 21], which is used to evaluate sentence-meaning agreement. Since no suitable dataset for audio-change captioning existed, we created the malfunctioning industrial machine investigation and inspection (MIMII)-Change dataset, which is based on the malfunctioning industrial machine investigation and inspection for domain generalization (MIMII-DG) [22] which was created for anomalous sound detection (ASD) that contains both normal and anomalous sounds of five types of machine-operation sounds (bearing, fan, gearbox, slider, valve). MIMII-Change consists of pairs of normal and anomalous sounds, and each pair is annotated with the changes between these sounds.

2. TASK DESCRIPTION

We now introduce the proposed neural-network model for generating change captions from two sounds. Like many models for explanatory-sentence-generation tasks, it uses a network architecture that encodes variable-length inputs into a fixed-dimension vector and uses this representation to "decode" them into the desired output sentence. Thus, we propose to directly maximize the probability of a correct description given two sounds using the following formulation:

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \sum_{(A_{\text{before}}, A_{\text{after}}, W)} \log p(W | A_{\text{before}}, A_{\text{after}}; \theta), \quad (1)$$

where θ are the parameters of our model, A_{before} is an audio before changing, A_{after} is an audio after changing, and W is its correct description. Sentences are generally generated autoregressively from the left (i.e. first word) to the right (i.e. final word). That is, at time step t, the decoder predicts the posterior probability on the vocabulary given the encoded acoustic feature, the start token w_0, and the previously generated words w_1 to w_{n-1}. Thus, $p(W|A_{\text{before}}, A_{\text{after}})$ can be formulated as

$$\log p(W | A_{\text{before}}, A_{\text{after}}) = \sum_{n=0}^{N} \log p(w_n | A_{\text{before}}, A_{\text{after}}, w_0, ..., w_{n-1}), \quad (2)$$

where N is the length of sentence. Note that θ has been removed for convenience. The description-generation process ends when a stop token is generated or the maximum number of generation steps is reached.

3. PROPOSED MODEL

3.1. Training scheme

To analyze the content of a sound clip, it is important to obtain a valid feature representation of the sound clip. We first extract the spectrogram then obtain embedding vectors $X_{\text{all}} \in \mathbb{R}^{(T*2)\times D}$ by using the encoder. This procedure can be formulated as

$$X_{\text{before}}, X_{\text{after}} = \mathcal{E}(A_{\text{before}}, A_{\text{after}}), \quad (3)$$

where ($A_{\text{before}} \in \mathbb{R}^{T\times F}$, $A_{\text{after}} \in \mathbb{R}^{T\times F}$) are the log mel-spectrograms of "before" and "after" sounds, $X_{\text{before}} \in \mathbb{R}^{T\times D}$ and $X_{\text{after}} \in \mathbb{R}^{T\times D}$ are embedding vectors extracted by encoders $\mathcal{E}$, T is the number of time frames, F is the number of mel bins, and D is the dimension of the latent embedding.

We then subtract X_{before} from X_{after} to capture semantic differences in the embedding space. The resulting vector X_{diff} is concatenated with X_{before}. This procedure can be formulated as

$$X_{\text{diff}} = X_{\text{after}} - X_{\text{before}} \quad (4)$$

$$X_{\text{all}} = [X_{\text{before}} : X_{\text{after}} : X_{\text{diff}}], \quad (5)$$

where [:] indicates concatenation.

We used Transformer Encoder [5] and spatial attention [14, 23] as the audio encoders. Spatial attention consists of a two-layer convolutional neural network (CNN) and creates spatial-attention maps $a_{\text{before}}, a_{\text{after}} \in \mathbb{R}^{T\times F}$ a from A_{before}, A_{after}. Thus, spatial attention can localize the change areas between A_{before} and A_{after} and is valid for image-change captioning [14]. For more information on this model architecture, see Park et al.'s study [14].

The decoder predicts the entire caption using X_{all}. The <bos> and <eos> tokens are added before and after the original caption to indicate the beginning and end of the sentence, respectively. The decoder operates in a step-by-step auto-regressive decoding scheme: at the first time step, <bos> is sent to the decoder, then at each time step n, the decoder takes the output word w_{n-1} of the last time step and generates word w_n as the input word in the next time step until <eos>. Finally, the decoder generates a sentence $S = \{w_1, ..., w_N\}$, where w_n is a word and N is the number of words in the sentence. The entire model is trained end-to-end by cross entropy loss. We use a standard transformer [5] as a decoder, which consists of multi-head self-attention on the caption sequence and multi-head encoder-decoder attention on the extracted feature sequence. An overview of our proposed model is given in 1

3.2. Category division

Machine-operating sounds include a variety of sounds, such as regularly occurring and suddenly occurring sounds, and the suitable model architecture differs depending on the type of sound. Regarding changes in regularly occurring sounds, it is considered important to determine the changes in the global characteristics between the two sounds, whereas it is considered important to determine the temporal dependencies between the two sounds for changes in suddenly occurring sounds. Therefore, as proposed method, we divided the sound types into three categories in accordance with the interval of sound occurrence and interval between sound occurrences, and MIMII-Change was created so that a single pair of normal and anomalous sounds had three captions (see Section 4 for more details).

4. MIMII-CHANGE DATASET

Since there is no appropriate dataset to study audio-change captioning, we created MIMII-Change. All sounds are single channel, 10 s in duration, and down-sampled to 16 kHz. We utilized a test dataset from MIMII-DG, consisting of five types of machine sounds (each type consisting of 300 normal sounds and 300 anomalous sounds), and created pairs by assigning one anomalous sound to one normal sound. This resulted in a total of 1,500 pairs (300 pairs × 5 machine types).

Three annotators compared the normal and anomalous sounds of each pair and annotated the changes. The annotators were instructed to always use onomatopoeia when describing sound changes. This is because onomatopoeia, which is a character sequence for phonetically imitating a sound, are effective for describing diverse environmental sound features [24, 25]. Onomatopoeia can be used to describe detailed changes, such as changes in the pitch of a machine's operating sound. The annotator also created three captions for a pair in accordance with the three categories of "stationary sound changes," "periodic sound changes" and "non-periodic sound changes." This is because it is difficult to express all changes in a single sentence, and from a model-learning perspective, it is undesirable for sentences to be redundant. Each of the three categories is defined as follows: "stationary sound": a single sound that occurs continuously for more than about 5 s, "periodic sound": a sound that repeats (including intervals) for more than 5 s, and "non-periodic sound": a sound that occurs multiple times but has no periodicity or appears and disappears suddenly. To improve learning efficiency, annotators provided captions according to templates. Templates mean that, for example, a change in pitch is always described as, "The pitch of ... became higher/lower."

The 1.500 pairs were divided into two 75 and 25% segments, which we call development and evaluation, respectively. All words in the captions must be included in the development split, and there should be no words that are only included in the evaluation split. This prevents the presence of unused words in training (i.e. words that only appear in development) and unknown words in evaluation (i.e. words that do not appear in development). We also split the data so that the word-occurrence frequency in development is always greater than that in evaluation. The number of data items, words, and onomatopoeia after splitting of each category are as listed in Table 1.

Table 1: Number of words of each category

	onomatopoeia/other words	Total
stationary	146 / 68	214
periodic	756 / 107	863
non-periodic	1,155 / 105	1,260

Table 2: Experimental conditions

Optimizer	Adam [26]
Training epoch	100
Batch size	16
GPU	GeForce RTX 3060

5. EXPERIMENTS

5.1. Evaluation metrics

To evaluate audio-change captioning, we used the conventional rule-based evaluation metrics BLEU [16], METEOR [17], CIDEr [18], SPICE [27], and SPIDEr [19]. Most conventional rule-based metrics focus on *n*-gram or sub-sequence-based matching between candidate and reference captions. CIDEr and SPICE, proposed for image captioning, show better correlation with human judgment in the captioning task. However, they cannot evaluate the semantic similarity between sentences, and they have not yet been able to resemble human evaluation [21]. To address this issue, we used the model-based evaluation metric Sentence-BERT [20, 21]. Sentence-BERT can be used to obtain a fixed-length sentence-embedding vector for input captions. The sentence embeddings are then used to calculate similarities between candidate and reference captions by calculating their cosine similarities. We also used the phoneme error rate (PER) [28] to evaluate onomatopoeia correspondence. Since each onomatopoeia is tokenized, it is not possible to match onomatopoeia with similar constituent phonemes. For example, "gagaga" and "gaga" would be evaluated as completely different onomatopoeia. To address this issue, onomatopoeia were broken down into phonemes according to a previous study [29], and similarity was calculated between onomatopoeia in terms of the PER. The PER is the "edit distance" between two phoneme sequences, normalized by the length of target phonemes, and expressed using Eq. 6. Since the number of onomatopoeia appearing in different sentences may differ, we used the mean phoneme error rate (MPER). The MPER is the average of PER of all combinations of onomatopoeia in a sentence and expressed using Eq. 7, where N is the number of phonemes in a reference caption, M is the number of phonemes in candidate caption, R_n is the n-th onomatopoeia of a no reference caption, and C_m is the m-th onomatopoeia of a candidate caption.

$$\mathrm{L}(R_n, C_m) = \frac{\text{Repalcement Err.} + \text{Insertion Err.} + \text{Deletion Err.}}{\text{Number of Target Phonemes}} \tag{6}$$

$$\text{MPER} = \frac{\sum_{n=1}^{N}\sum_{m=1}^{M}\mathrm{L}(R_n, C_m)}{N * M} \tag{7}$$

Since the PER is calculated for all combinations of onomatopoeia, it is not possible to evaluate onomatopoeia order correspondence,

Table 3: Experimental results

model_type (#model_parameters)	BLEU_3	BLEU_4	METEOR	CIDEr	SPICE	SPIDEr	Sentence-BERT	MPER
Stationary								
TraEnc. (10.8M)	0.616	0.542	0.427	0.969	0.340	0.655	0.793	0.281
SpaAttn. (0.9M)	**0.669**	**0.601**	**0.441**	**1.086**	0.365	**0.726**	**0.796**	0.266
PANNs+TraEnc. (82.6M)	0.659	0.583	0.436	0.933	**0.381**	0.657	0.791	**0.251**
Periodic								
TraEnc. (10.8M)	**0.464**	**0.387**	0.390	**0.946**	**0.255**	**0.601**	0.725	**0.338**
SpaAttn. (0.9M)	0.426	0.354	**0.402**	0.881	0.249	0.565	**0.727**	0.380
PANNs+TraEnc. (82.6M)	0.383	0.306	0.369	0.729	0.213	0.471	0.689	0.362
Non-periodic								
TraEnc. (10.8M)	**0.413**	**0.339**	**0.427**	**1.864**	**0.373**	**1.118**	**0.728**	0.327
SpaAttn. (0.9M)	0.328	0.269	0.411	1.441	0.304	0.873	0.678	**0.321**
PANNs+TraEnc. (82.6M)	0.346	0.284	0.392	1.434	0.331	0.882	0.682	0.365

so the MPER is used only as a metric to measure onomatopoeia agreement in sentences. For example, the PER value of the candidate sentence "A changed to B" and the candidate sentence "B changed to A" would be the same with respect to the reference caption "Changed from A to B." Here, A and B are onomatopoeia.

5.2. Experimental setup

We used the 64-dimensional log mel-band energy as an acoustic feature, which is extracted on the basis of a 64-ms frame length with a 32-ms shift size. Other conditions are listed in Table 2. As this paper presents the first methodology for audio-change captioning, there are no previous results to compare the presented ones. For that reason, several model architectures are compared to investigate their effectiveness.

Transformer Encoder Transformer encoders can determine the temporal dependencies of each input sequence. Therefore, it is considered effective for periodic and non-periodic sound with short sound onset intervals. In this experiment, Transformer Encoder with three layers and four multi-head attention was used.

Spatial attention Spatial attention [23] is an architecture based on convolutional neural networks and it generates a spatial-attention map by using the inter-spatial relationship of features. Spatial attention differs from channel attention in that it focuses on where information is located and has been shown to be effective in locating points of change [14]. In this experiment, spatial attention consisting of a two-layer CNN was used. The spatial attention architecture is able to determine global features, which may be effective for stationary sound.

Acoustic feature extraction with pretrained audio neural networks (PANNs) The effectiveness of transfer learning of pretrained models has been shown in many audio-related tasks. To confirm the effectiveness of pre-trained models, we used PANNs [30], a pre-trained model for acoustic recognition, as a feature extractor. Specifically, we used a pre-trained 14-layer CNN (CNN14). Acoustic features are extracted from the spectrogram by using PANNs, the outputs X_{before} and X_{after} is subtracted, and X_{all}, calculated in the same manner as Eq. 5, is passed through an encoder.

5.3. Results

Table 3 lists the evaluation results for each version in each of the three categories. All versions used Transformer Decoder as decoder and had different encoders. TraEnc. denotes Transformer Encoder, SpaAttn. denotes spatial attention.

Transformer Encoder vs. spatial attention As shown in Table 3, Spatial attention performed best for "stationary sound changes." As shown in Table 1, the number of words for "stationary sound changes" was 214, which is much smaller than the other categories. For steady sound changes, it is considered important to capture the change in the global features between two sounds. Therefore, spatial attention, which has a relatively easy task difficulty and consists of a two-layer CNN, was more effective. Transformer Encoder was more effective for "periodic sound changes" and "non-periodic sound changes" because the vocabulary was large and it is considered important to capture the temporal dependency between the two sounds.

Validity of PANNs as feature extractor In all three categories, there was no performance improvement due to feature extraction with PANNs. This may be due to the fact that PANNs is trained by solving audio tagging, so features are lost in MIMII-Change in which all sounds are classified as machine-operation sounds.

Our experiments showed that different model architectures were suitable for different categories of sounds with distinct characteristics. Specifically, we found that using spatial attention was effective for the "stationary sound changes," while using Transformer Encoder was effective for the "periodic sound changes" and "non-periodic sound changes."

6. CONCLUSION

We defined a new problem, "audio-change captioning," which describes what has changed between two audio samples and proposed a neural-network model for generating sentences that explain how a machine's normal and anomalous sounds changed in relation to each other. We also created the MIMII-Change dataset that is based on MIMII-DG, annotated each type of sound, and investigated the characteristics of audio-change captioning. Our experiments showed that different categories of sounds with distinct characteristics required different model architectures for optimal performance. By utilizing models tailored to each category of sound, we were able to achieve high accuracy by leveraging the specific features of the sound.

7. REFERENCES

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

[2] A. S. Koepke, A.-M. Oncescu, J. Henriques, Z. Akata, and S. Albanie, "Audio retrieval with natural language queries: A benchmark study," *IEEE Transactions on Multimedia*, pp. 1–1, 2022.

[3] X. Mei, X. Liu, J. Sun, M. D. Plumbley, and W. Wang, "On metric learning for audio-text cross-modal retrieval," *arXiv preprint arXiv:2203.15537*, 2022.

[4] X. Mei, X. Liu, M. D. Plumbley, and W. Wang, "Automated audio captioning: an overview of recent progress and new challenges," *EURASIP Journal on Audio, Speech, and Music Processing*, no. 1, Oct 2022.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Proc. of Advances in Neural Information Processing Systems (NIPS)*, vol. 30, 2017.

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

[7] Z. Chen, D. Zhang, J. Wang, and F. Deng, "Audio captioning with meshed-memory transformer," DCASE2021 Challenge, Tech. Rep., 2021.

[8] 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," 2020.

[9] Z. Ye, H. Wang, D. Yang, and Y. Zou, "Improving the performance of automated audio captioning via integrating the acoustic and semantic information," in *DCASE*, 2021.

[10] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4651–4659.

[11] J. Yuan, C. Tian, X. Zhang, Y. Ding, and W. Wei, "Video captioning with semantic guiding," in *IEEE Fourth International Conference on Multimedia Big Data (BigMM)*, 2018, pp. 1–5.

[12] Y. Kawaguchi, K. Imoto, Y. Koizumi, N. Harada, D. Niizumi, K. Dohi, R. Tanabe, H. Purohit, and T. Endo, "Description and discussion on DCASE 2021 challenge task 2: Unsupervised anomalous sound detection for machine condition monitoring under domain shifted conditions," 2021.

[13] H. Jhamtani and T. Berg-Kirkpatrick, "Learning to describe differences between pairs of similar images," in *EMNLP*. Association for Computational Linguistics, Oct.-Nov. 2018, pp. 4024–4034.

[14] D. H. Park, T. Darrell, and A. Rohrbach, "Robust change captioning," in *Proc. of IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 4624–4633.

[15] https://dcase.community/.

[16] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, July 2002, pp. 311–318.

[17] A. Lavie and A. Agarwal, "METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments," in *Proc. of the Second Workshop on Statistical Machine Translation*. Association for Computational Linguistics, June 2007, pp. 228–231.

[18] R. Vedantam, C. L. Zitnick, and D. Parikh, "CIDEr: Consensus-based image description evaluation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2015, pp. 4566–4575.

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

[20] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *EMNLP-IJCNLP*. Association for Computational Linguistics, Nov. 2019, pp. 3982–3992.

[21] Z. Zhou, Z. Zhang, X. Xu, Z. Xie, M. Wu, and K. Q. Zhu, "Can audio captions be evaluated with image caption metrics?" in *ICASSP*, 2022, pp. 981–985.

[22] K. Dohi, T. Nishida, H. Purohit, R. Tanabe, T. Endo, M. Yamamoto, Y. Nikaido, and Y. Kawaguchi, "MIMII DG: Sound dataset for malfunctioning industrial machine investigation and inspection for domain generalization task," *In arXiv e-prints: 2205.13879*, 2022.

[23] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. of European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[24] Y. Okamoto, S. Horiguchi, M. Yamamoto, K. Imoto, and Y. Kawaguchi, "Environmental sound extraction using onomatopoeic words," in *ICASSP*, 2022, pp. 221–225.

[25] Y. Okamoto, K. Imoto, S. Takamichi, R. Yamanishi, T. Fukumori, and Y. Yamashita, "Onoma-to-wave: Environmental sound synthesis from onomatopoeic words," *APSIPA Transactions on Signal and Information Processing*, vol. 11, no. 1, 2022.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[27] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "SPICE: Semantic propositional image caption evaluation," in *Proc. of European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 382–398.

[28] S. Ikawa and K. Kashino, "Generating sound words from audio signals of acoustic events with sequence-to-sequence model," in *ICASSP*, 2018, pp. 346–350.

[29] https://github.com/KeisukeImoto/RWCPSSD_Onomatopoeia/blob/master/katakana2accphrase.csv.

[30] 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," vol. 28. IEEE, 2020, pp. 2880–2894.