



Temp4Cap: Temporally-aligned Automated Audio Captioning

Ho-Young Choi^{1,2,*}, Jae-Heung Cho^{1,3,*}, Pil Moo Byun¹, Won-Gook Choi¹, Joon-Hyuk Chang^{1,†}

¹Electronics Engineering, Hanyang University, Republic of Korea

²Samsung Electronics, Republic of Korea

³Hyundai Motor Company, Republic of Korea

ho0.choi@samsung.com, heung88@hyundai.com, fordream0309@hanyang.ac.kr,
onlyworld94@hanyang.ac.kr, jchang@hanyang.ac.kr

Abstract

Automated audio captioning (AAC) is a crucial task in machine perception within the audio domain. AAC struggles to interpret and incorporate temporal relationships of sound events in captions. However, existing studies often fail to capture the temporal relationship, leading to incorrect captions. Some recent studies leverage sound event detection models to extract temporal relationships but remain limited by their dependence on independent pre-trained models. In this study, we propose Temp4Cap, a novel AAC framework that directly trains temporal alignment via contrastive learning, using the “temporal caption” generated by a large language model. To capture temporal relationships, we apply a temporal negative sampling strategy, which includes event- and order-level shuffle and random substitution when generating negative samples during contrastive learning. Experimental results on Clotho and AudioCaps show that Temp4Cap significantly improves both captioning and temporal metrics.

Index Terms: Temporal alignment, temporal negative sampling, contrastive learning, large language model, automated audio captioning

1. Introduction

Automated audio captioning (AAC) is a cross-modal translation task that converts the comprehensive content of input audio into natural language [1]. An AAC model typically follows an encoder-decoder framework. The encoder extracts features from the input audio, and the decoder generates captions based on them. Primarily, the encoder leverages pre-trained audio models, such as pre-trained audio neural networks (PANNs) [2], hierarchical token-semantic audio transformer [3] or VGGish [4], which have been known to extract rich audio representations. The decoder is designed as either shallow transformers [5] or recurrent neural networks (RNNs) [1, 6, 7] to generate captions that consider the contextual information of these extracted audio features. Audio containing environmental sounds tend to exhibit relatively less structured patterns, such as inconsistent rhythms or random frequency variations unlike speech, which typically adheres to structured patterns including syllables, words, or sentences under linguistic rules. Thus, representing audio necessitates capturing both the spatial-temporal relationships of sound events occurring in specific sections and the background sounds, and AAC aims to offer a comprehensive description of audio encompassing these features.

In vision-based captioning tasks, recent research leveraging contrastive learning for semantic correspondences between cross-modal data has shown promising results [8, 9].

Accordingly, numerous attempts have been made to integrate contrastive learning for audio-text semantic correspondence in AAC tasks [10, 11, 12, 13]. In previous studies, various approaches have been conducted to enhance caption quality through contrastive learning. For example, CL4AC [11] introduced a contrastive learning framework specifically designed for audio captioning, ACTUAL [13] used contrastive learning as a regularization technique for caption consistency, and CLIP-AAC [12] applied contrastive learning to bridge the domain gap by learning the correspondence between audio signals and their paired captions. Consequently, these studies have enhanced the caption quality by improving audio-text representations through contrastive learning. However, these approaches fail to align captions with the temporal structure of audio events, such as event order or background sounds, which are crucial for AAC.

Z. Xie *et al.* [14] proposed a temporal tag-guided captioning system to address this issue, which takes temporal tag guidance inferred from the output of a sound event detection (SED) model that detects the on- and off-sets of each sound event. However, this approach has limitations, as they tend to rely heavily on the performance of a pretrained SED model, and while it may be effective in generating temporal conjunctions simply, they appear to produce captions of lower quality, as indicated by a significant decrease in the CIDEr [15] score.

In this study, we propose **Temp4Cap**, a novel framework that can enhance both the quality and aligning temporal relations of the captions by performing temporal alignment through the contrastive learning between predicted and temporal captions. The temporal alignment progresses in the following ways: 1) A large language model (LLM) generates temporal captions explicitly structured as “event description: event order” representations. 2) For temporal alignment training, a temporal negative sampling strategy is introduced for generating negative samples during the contrastive learning process. To generate negative samples, we first selected random events within temporal captions, substituted by random events from random temporal captions within the same batch, and then shuffled the order of the events.

Different from other studies, we adopt subjective metrics to validate the proposed method shows robust performance in capturing temporal relations. As a result, the proposed method outperforms the other methods on both objective and subjective metrics.

2. Proposed Methods

2.1. Captioning model

Let $\mathbf{A} \in \mathbb{R}^{T \times F}$ denotes the audio input, where T and F refer to the number of time frames and mel filters, respectively.

* These authors contributed equally

† Corresponding author.

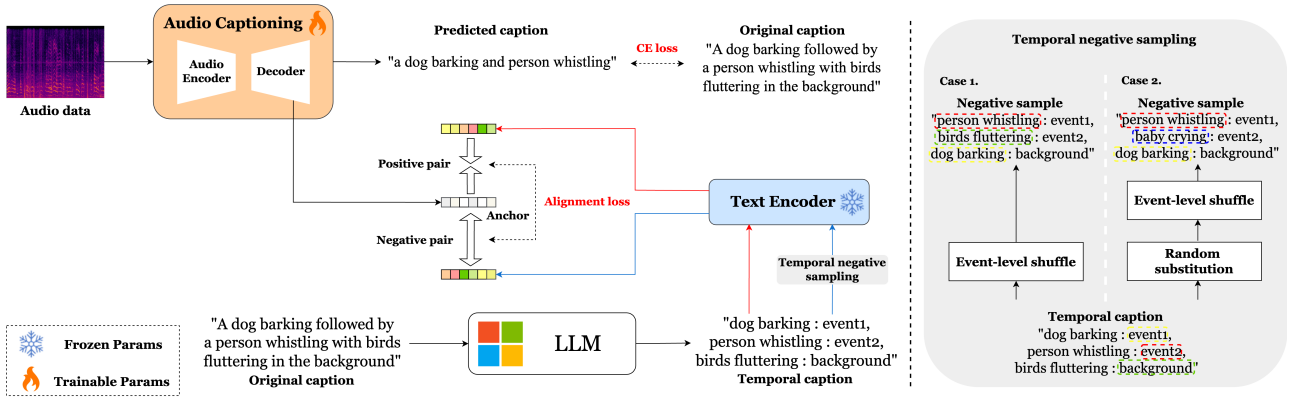


Figure 1: *Left: Overview of the Temp4Cap framework. Right: Temporal negative sampling algorithms. The blue dotted box indicates an event within a randomly-selected temporal caption from the same batch.*

We utilize PANNs [2] as the audio encoder, which is convolutional neural network-14 (CNN-14) to extract the audio representation. PANNs are pretrained models using Audioset [16], a large-scale audio-tagging dataset. CNN-14 comprises six CNN blocks, each of which is composed of a 3×3 CNN, batch normalization, and a rectified linear unit (ReLU) [17]. In this study, the bidirectional and auto-regressive transformer (BART) [18] model is introduced for text generation as the decoder of captioning model in conjunction with CNN-14. The BART model is a combination of bidirectional encoder representations from transformers (BERT), which have a bidirectional structure, and a generative pretrained transformer, which has an autoregressive structure. The bi-directional encoder includes multi-head self-attention, multi-layer preception, and residual connections. Consequently, the BART encoder and decoder, composed of multi-head cross-attention and self-attention, play a role in generating captions. Both the encoder and decoder in the BART model are composed of six transformer layers. Furthermore, the cross-entropy (CE) loss is employed as the objective function for the captioning model:

$$\mathcal{L}_{CE} = -\frac{1}{L} \sum_{l=1}^L \log p(g_l | g_{1:l-1}, \mathbf{A}), \quad (1)$$

where g_l is the l -th ground truth token in a sentence and L is the length of the sentence.

2.2. Temporal alignment training

As shown in Fig. 1, the LLM is used to generate a temporal caption comprising a pair of events and orders from a original caption for temporal alignment training. To enhance the temporal relationship interpretation ability of the captioning model including the encoder and decoder, we perform contrastive learning between the decoder’s hidden representation $\mathbf{x} \in \mathbb{R}^{T_1 \times D}$ and the text embedding $\mathbf{y} \in \mathbb{R}^{T_2 \times D}$ of the temporal caption, where D refers to embedding dimension, T_1 and T_2 denote the sequence lengths of \mathbf{x} and \mathbf{y} .

For equivalence with the audio captioning model, we use the BART model as the text encoder to extract the embeddings. In particular, temporal negative sampling is performed to improve temporal relationships. Temporal negative sampling comprises two processes: **event-level shuffle** and **random substitution** processes, as depicted on the right side

of Fig. 1. Event shuffling is the process of rearranging order of events while maintaining the content of each event in the temporal captions, and random substitution comprises randomly selecting one event and then replacing it with another event within a mini-batch. These processes allow the model to learn the temporal relationships and semantic information of the audio clips.

We train the temporal alignment structure using an InfoNCE loss function [22] to ensure that generated captions align closely with the temporal captions, making them more similar to each other than negative shuffled pairs. We repeat the temporal negative sampling K times to create K negative pairs. In addition, negative sampling is randomly performed to provide diversity. The alignment loss function \mathcal{L}_{al} is thus configured as follows:

$$S(\mathbf{x}, \mathbf{y}) = \frac{1}{T_1 T_2} \sum_{i=1}^{T_1} \sum_{j=1}^{T_2} (\mathbf{x}_i \cdot \mathbf{y}_j), \quad (2)$$

$$\mathcal{L}_{al} = \frac{1}{N} \sum_{i=0}^N -\log \frac{\exp(S(\mathbf{x}_i, \mathbf{y}_i^+)/\tau)}{\sum_{k=1}^K \exp(S(\mathbf{x}_i, \mathbf{y}_k^-)/\tau)}, \quad (3)$$

where $S(\mathbf{x}, \mathbf{y})$, \mathbf{y}^+ and \mathbf{y}^- represent the cosine similarity, the text embeddings of positive temporal captions and negatively sampled temporal captions, respectively. Also, N is the batch size, and τ is the temperature parameter. Finally, during the training process, we utilize objective functions that compute the sum of the CE loss and alignment loss. The overall objective function is configured as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{CE} + \mathcal{L}_{al}. \quad (4)$$

For data comprising only a single temporal order, the temporal negative sampling process is not performed, in which case only CE loss is used.

3. Experiments

3.1. Dataset

For the experiment, we employed the Clotho [23] and AudioCaps [24] datasets for training and evaluation. Clotho is a dataset that includes 5 k audio clips from the FreeSound [25] comprising audio samples having 15 to 30 s duration. AudioCaps is a large audio captioning dataset composed of approximately 50 k audio clips sourced from Audioset. The captions in the training set were a single caption per audio clip, whereas

Table 1: Results comparison with CNN encoder-based models, using Clotho and AudioCaps test splits. For all metrics, higher values mean better performance.

Dataset	Model	BLEU ₄ (↑)	ROUGE _L (↑)	CIDEr (↑)	METEOR (↑)	SPIDEr (↑)	ACC _{temp} (↑)	F1 _{temp} (↑)
Clotho	CNN-GRU [19]	16.8	38.3	40.8	17.5	26.5	69.2	17.4
	CNN-Transformer [20]	15.6	37.2	37.2	16.9	24.4	68.2	13.1
	CNN-BART [21] (Baseline)	16.1	37.8	40.2	17.2	25.9	68.8	15.1
	Temp4Cap (Ours)	16.7	38.2	41.9	17.4	27.0	70.2	23.2
AudioCaps	CNN-GRU	26.3	49.4	72.6	23.9	45.1	37.3	23.4
	CNN-Transformer	24.5	46.2	64.1	22.4	40.4	36.2	21.6
	CNN-BART (Baseline)	27.1	48.6	73.6	23.5	45.4	37.8	23.8
	Temp4Cap (Ours)	27.4	49.1	76.1	23.6	46.7	46.4	36.6

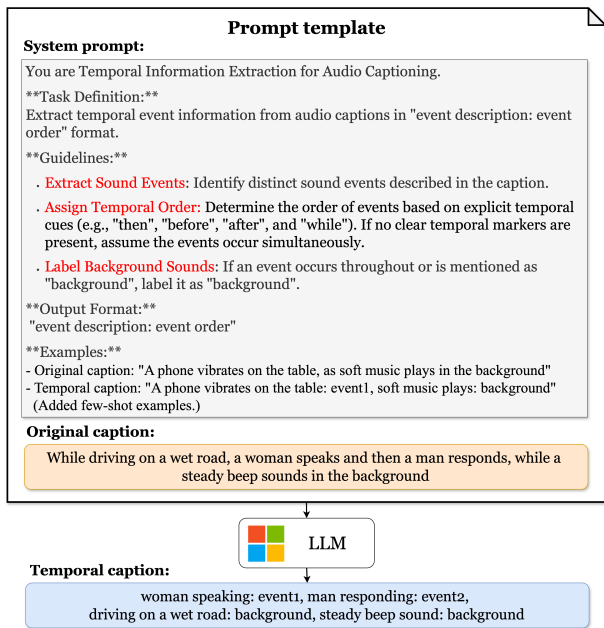


Figure 2: The prompt template for generating temporally structured captions using an LLM.

those in the validation and test sets were five captions per audio clip. AudioCaps is composed of audio clips containing more multiple events compared to Clotho. Analyzing temporal captions, data consisting of multiple event orders account for 79% of the training set in Clotho, while in AudioCaps, it constitutes 89% of the development set.

3.2. Data pre-processing

To obtain a log-mel spectrogram for each audio clip, we applied a Hanning window of size 1024 with a hop size of 320 and a 64-bin mel filterbank. The audio clips were resampled to 32 kHz with a maximum length of 30 s. The captions from Clotho and AudioCaps datasets were converted to lowercase, and special tokens <eos> and <eos> were appended to the beginning and end of each caption. As shown in Fig. 2, we employed Phi-4¹ [26] as an LLM to generate temporal captions from the dataset. To ensure structured event extraction, we designed a system prompt that formatted the output as "event description: event order" while providing explicit guidelines on event extraction, temporal ordering, and background labeling. Additionally, to improve the model's comprehension of the task, we included five few-shot examples illustrating the expected outputs.

¹<https://huggingface.co/microsoft/phi-4>

Table 2: Comparisons for the different systems on the AudioCaps test splits. MOS is provided with 95% confidence interval (CI).

Method	Caption quality score (↑)	Temporal alignment score (↑)
GT	4.15 ± 0.18	4.31 ± 0.16
CNN-Transformer	3.36 ± 0.18	3.18 ± 0.22
CNN-BART	3.59 ± 0.19	3.57 ± 0.28
Temp4Cap	3.99 ± 0.14	4.07 ± 0.13

3.3. Experiment setups

The proposed model was trained for 30 epochs using a batch size of 24 and a learning rate of 1×10^{-5} , which lasted 12 h. The text encoder was frozen during the training process, whereas the audio encoder and decoder were not. We used the Adam optimizer [27] with a weight decay of 1×10^{-6} . The experiments were conducted on a single NVIDIA RTX 3090 Ti. To prevent overfitting, a dropout rate of 0.2 was applied to both the encoder and decoder components, and the spec-augmentation [28] method was applied. In the inference phase, we applied a beam search algorithm with a beam size of 3 to enhance the caption generation quality of the decoder. In addition, to facilitate contrastive learning, we set the size of the negative samples to 24 and set τ to 0.07.

3.4. Evaluation

Objective metrics. We evaluated the overall generation quality of the trained model using widely used captioning metrics, such as BLEU [29], ROUGE_L [30], METEOR [31], CIDEr [15] and SPIDEr [32], in the captioning task. To evaluate the quality of the generated captions considering the temporal relations, we used ACC_{temp} and F1_{temp} [14]. Temporal metrics measured the presence of conjunctions ("after", "then", "followed", and "follow") that indicate temporal relationships in the generated captions.

Subjective metric. These temporal metrics have a limitation in that they simply measure the presence or absence of temporal conjunctions, failing to consider the actual order of events (e.g., "A after B" and "B after A" yielded the same result). To compensate for this limitation, we also conducted the mean opinion score (MOS) test. The test was performed on researchers total of 20 with 30 selected captions, each containing at least two temporal conjunctions. Specifically, 20 captions were sourced from AudioCaps and 10 from Clotho. Specifically, the generated captions were assessed in two aspects: i) **caption quality score** focused on how well they contained the content of the audio and ii) **temporal alignment score** focused on how accurately they captured the actual order of events.

Table 3: Results comparison with and without each processes, using Clotho and AudioCaps test splits. $E \cdot S$ and $R \cdot S$ refer event shuffle and random substitution, respectively.

Dataset	Model	BLEU ₄	ROUGE _L	CIDEr	METEOR	SPIDEr	ACC _{temp}	F1 _{temp}
Clotho	Temp4Cap ($E \cdot S$)	15.6	37.2	40.6	17.2	26.1	69.7	17.7
	Temp4Cap ($R \cdot S$)	16.0	37.1	41.0	16.9	26.3	68.2	20.6
	Temp4Cap ($E \cdot S + R \cdot S$)	16.7	38.2	41.9	17.4	27.0	70.2	23.2
AudioCaps	Temp4Cap ($E \cdot S$)	26.9	49.4	74.1	23.5	45.3	44.3	38.4
	Temp4Cap ($R \cdot S$)	26.9	48.9	76.4	24.1	47.1	41.9	33.0
	Temp4Cap ($E \cdot S + R \cdot S$)	27.4	49.1	76.1	23.6	46.7	46.4	36.6

4. Results and discussion

4.1. Performance comparison

To assess the effectiveness of the proposed method, we compared performance with the CNN encoder-based models [19, 20, 21] widely used in AAC task, and the results are presented in Table 1. The captioning metrics are influenced by semantic information from the audio, and the clarity of the captions. The proposed temporal negative sampling offers two notable advantages in this regard: First, the LLM generates captions in various sentence structures, allowing for diversity in model training, similar to data augmentation. This enhances the ability of the model to generalize effectively across different inputs. Second, owing to the influence of random substitution, the model can acquire richer audio semantic information via contrastive learning. As a result, Temp4Cap outperforms the other models across most captioning metrics. Temporal metrics were also employed to assess how well the generated captions captured the temporal order of audio events. Temp4Cap performed well compared with the other models, which proves that the temporal alignment training helps to interpret the order of the events well. Notably, Table 1 shows that temporal alignment training works better for AudioCaps than for Clotho. This is due to AudioCaps containing a larger number audio clips featuring multiple events, as mentioned in Section 3.1. In particular, event shuffling has less impact as the number of events n is smaller, because the number of possibilities for negative samples is limited to the n factorial. Consequently, Temp4Cap not only improved temporal metrics but also showed enhancement across all captioning metrics.

To illustrate results that are difficult to prove with temporal metrics, we compared the generated captions in Fig. 3. The sample was specifically selected to demonstrate the effect of the temporal alignment training. The CNN-Transformer represented the sequence of events through “*followed by*”, but the order of events was expressed in reverse. The CNN-BART failed to recognize the temporal order of occurrence of “*woman speaking*” and “*men speaking*” and also failed to capture “*digital beeps*”. In contrast, Temp4Cap successfully identified the temporal sequence of events. Consequently, the generated captions show that the proposed framework is excellent at distinguishing events and can accurately understand the temporal relationships between events.

As shown in Table 2, we performed MOS tests for caption quality and temporal alignment. CNN-Transformer and CNN-BART, trained only with CE loss, received lower scores than Temp4Cap in caption quality and temporal alignment. Especially when evaluating captions by listening to samples, we found that Temp4Cap effectively generates captions reflecting event occurrence and sequence.

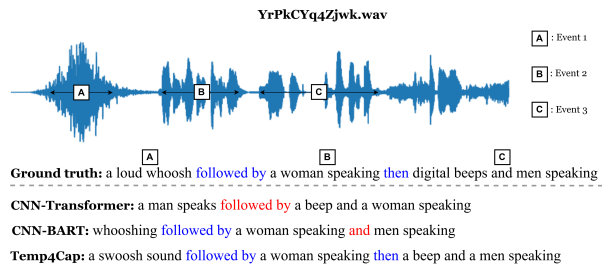


Figure 3: Compared of generated captions by a CNN-Transformer, CNN-BART and Temp4Cap. The blue and red denote correct, incorrect, respectively.

4.2. Ablation studies

Ablation studies were conducted to evaluate the effectiveness of the proposed temporal negative sampling process for temporal alignment using Temp4Cap. We evaluated the performance under three different scenarios: applying only event shuffling, using only random substitution, and combining both methods. The results are shown in Table 3, based on evaluations conducted on the Clotho and AudioCaps test sets. When only the event shuffle was applied, there was an increase in the temporal metrics compared to the baseline. This indicates that through the event shuffle, the model can more effectively capture the temporal relations between events. When only random substitution was applied, the results of the captioning metrics were significantly improved. These results show that the random substitution process helps in training contrastive learning more effectively due to the semantic misalignment caused by replacing events within a caption with different events in the mini-batch.

5. Conclusions

In this study, we introduced Temp4Cap, an innovative AAC framework designed to enhance the temporal alignment of cross-modal data. Rather than relying on transfer learning approaches utilizing pre-trained models, our strategy involved creating a temporal caption dataset using LLM. This allowed us to train the model directly on the sequence and relationships of events. Temp4Cap employs a contrastive learning method based on temporal negative sampling, including event shuffling, and random substitution, allowing the model to effectively capture both temporal relationships and semantic context.

6. Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2020-II201373, Artificial Intelligence Graduate School Program(Hanyang University))

7. References

- [1] K. Drossos, S. Adavenne, and T. Virtanen, "Automated audio captioning with recurrent neural networks," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2017, pp. 374–378.
- [2] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and D. M. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 2880–2894, 2020.
- [3] K. Chen *et al.*, "HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection," in *Proc. IEEE Int Conf. Acoust., Speech Signal Process.*, 2022, pp. 646–650.
- [4] S. Hershey *et al.*, "CNN architectures for large-scale audio classification," in *Proc. IEEE Int Conf. Acoust., Speech Signal Process.*, 2017, pp. 131–135.
- [5] A. Vaswani *et al.*, "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.
- [6] X. Xu, H. Dinkel, M. Wu, and K. Yu, "Audio caption in a car setting with a sentence-level loss," in *Proc. Int. Symp. Chinese Spok. Lang. Process.*, 2021, pp. 1–5.
- [7] M. Wu, H. Dinkel, and K. Yu, "Audio caption: Listen and tell," in *Proc. IEEE Int Conf. Acoust., Speech Signal Process.*, 2019, pp. 830–834.
- [8] R. Mokady, A. Hertz, and A. H. Bermano, "Clipcap: Clip prefix for image captioning," *arXiv:2111.09734*, 2021.
- [9] M. Tang, Z. Wang, Z. Liu, F. Rao, D. Li, and X. Li, "Clip4caption: Clip for video caption," in *Proc. ACM Int. Conf. Multimedia*, 2021, pp. 4858–4862.
- [10] J. H. Cho, Y. A. Park, J. Kim, and J. H. Chang, "Hyu submission for the DCASE 2023 task 6a: Automated audio captioning model using AL-MixGen and synonyms substitution," *DCASE Challenge, Tech. Rep.*, 2023.
- [11] X. Liu *et al.*, "CLAAC: A contrastive loss for audio captioning," in *Proc. Detection Classification Acoust. Scenes Events Workshop*, 2021.
- [12] C. Chen, N. Hou, Y. Hu, H. Zou, X. Qi, and E. S. Chng, "Interactive audio-text representation for automated audio captioning with contrastive learning," in *Proc. Interspeech*, 2022, pp. 2773–2777.
- [13] Y. Zhang *et al.*, "ACTUAL: Audio captioning with caption feature space regularization," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 2643–2657, 2023.
- [14] Z. Xie, X. Xu, M. Wu, and K. Yu, "Enhance temporal relations in audio captioning with sound event detection," in *Proc. Interspeech*, 2023.
- [15] R. Vedantam, Z. C. Lawrence, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4566–4575.
- [16] J. Gemmeke *et al.*, "Audio set: An ontology and human-labeled dataset for audio events," in *Proc. IEEE Int Conf. Acoust., Speech Signal Process.*, 2017, pp. 776–780.
- [17] A. F. Agarap, "Deep learning using rectified linear units (ReLU)," *arXiv:1803.08375*, 2018.
- [18] M. Lewis *et al.*, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proc. 58th Annu. Meet. Assoc. Comput. Linguist.*, 2020, pp. 7871–7880.
- [19] X. Xu, Z. Xie, M. Wu, and K. Yu, "The SJTU system for DCASE2022 challenge task 6: Audio captioning with audiotext retrieval pre-training," *DCASE Challenge, Tech. Rep.*, 2022.
- [20] X. Mei *et al.*, "An encoder-decoder based audio captioning system with transfer and reinforcement learning," in *Proc. Detection Classification Acoust. Scenes Events Workshop*, 2021, pp. 206–210.
- [21] F. Gontier, S. Romain, and C. Christophe, "Automated audio captioning by fine-tuning bart with audioset tags," in *Proc. Detection Classification Acoust. Scenes Events Workshop*, 2021, pp. 170–174.
- [22] A. V. D. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv:1807.03748*, 2018.
- [23] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: An audio captioning dataset," in *Proc. IEEE Int Conf. Acoust., Speech Signal Process.*, 2020, pp. 736–740.
- [24] C. D. Kim, H. L. B. Kim, and G. Kim, "Audiocaps: Generating captions for audios in the wild," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2019, pp. 119–132.
- [25] F. Font *et al.*, "Freesound technical demo," in *Proc. Int Conf. Multimedia*, 2013, pp. 411–412.
- [26] M. Abdin *et al.*, "Phi-4 technical report," *arXiv:2412.08905*, 2024.
- [27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2015.
- [28] D. S. Park *et al.*, "Specaugment: A simple data augmentation method for automatic speech recognition," in *Proc. Interspeech*, 2019, pp. 2613–2617.
- [29] K. Papineni, S. Roukos, T. Ward, and Z. Wei-Jing, "BLEU: a method for automatic evaluation of machine translation," in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2002, pp. 311–318.
- [30] C. Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Proc. Text Summarization Branches Out*.
- [31] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proc. Ninth Workshop Stat. Mach. Transl.*, 2014, pp. 376–380.
- [32] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy, "Improved image captioning via policy gradient optimization of spider," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 873–881.